Annals of Computer Science and Information Systems
Volume 21

Proceedings of the 2020 Federated Conference on Computer Science and Information Systems

September 6–9, 2020. Sofia, Bulgaria

Maria Ganzha, Leszek Maciaszek, Marcin Paprzycki (eds.)

PTI

IEEE

Annals of Computer Science and Information Systems, Volume 21

Proceedings of the 2020 Federated Conference on Computer Science and Information Systems

Maria Ganzha, Leszek Maciaszek, Marcin Paprzycki (eds.)

PTI
POLSKIE TOWARZYSTWO INFORMATYCZNE
POLISH INFORMATION PROCESSING SOCIETY

2020, Warszawa,
Polskie Towarzystwo
Informatyczne

IEEE

2020, New York City,
Institute of Electrical and
Electronics Engineers

Annals of Computer Science and Information Systems, Volume 21

Proceedings of the 2020 Federated Conference on Computer Science and Information Systems

ART: ISBN 978978-83-955416-9-8, IEEE Catalog Number CFP2085N-ART
USB: ISBN 978-83-955416-8-1, IEEE Catalog Number CFP2085N-USB
WEB: ISBN 978-83-955416-7-4

ISSN 2300-5963
DOI 10.15439/978-83-955416-7-4

Ul. Solec 38/103
00-394 Warsaw, Poland

Contact: secretariat@fedcsis.org
`http://annals-csis.org/`

Cover art: Balans
Monika Brzykca,
Elbląg, Poland

Also in this series:

Volume 23: Communication Papers of the 2020 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-959183-2-2, ISBN USB: 978-83-959183-3-9**
Volume 22: Position Papers of the 2020 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-959183-0-8, ISBN USB: 978-83-959183-1-5**
Volume 20: Communication Papers of the 2019 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-955416-3-6, ISBN USB: 978-83-955416-4-3**
Volume 19: Position Papers of the 2019 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-955416-1-2, ISBN USB: 978-83-955416-2-9**
Volume 18: Proceedings of the 2019 Federated Conference on Computer Science and Information Systems, **ISBN Web 978-83-952357-8-8, ISBN USB 978-83-952357-9-5, ISBN ART 978-83-955416-0-5**
Volume 17: Communication Papers of the 2018 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-952357-0-2, ISBN USB: 978-83-952357-1-9**
Volume 16: Position Papers of the 2018 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-949419-8-7, ISBN USB: 978-83-949419-9-4**
Volume 15: Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-949419-5-6, ISBN USB: 978-83-949419-6-3, ISBN ART: 978-83-949419-7-0**
Volume 14: Proceedings of the First International Conference on Information Technology and Knowledge Management, **ISBN WEB: 978-83-949419-2-5, ISBN USB: 978-83-949419-1-8, ISBN ART: 978-83-949419-0-1**
Volume 13: Communication Papers of the 2017 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-922646-2-0, ISBN USB: 978-83-922646-3-7**
Volume 12: Position Papers of the 2017 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-922646-0-6, ISBN USB: 978-83-922646-1-3**
Volume 11: Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-946253-7-5, ISBN USB: 978-83-946253-8-2, ISBN ART: 978-83-946253-9-9**

DEAR Reader, it is our pleasure to present to you Proceedings of the 15th Conference on Computer Science and Information Systems (FedCSIS'2020), which took place fully remotely, on September 7-9, 2020. Conference was originally planned to take place in Sofia, Bulgaria, but the global COVID-19 pandemics forced us to adapt and organize the conference online.

FedCSIS 2020 was Chaired by prof. Stefka Fidanova, while dr. Nina Dobrinkova acted as the Chair of the Organizing Committee. This year, FedCSIS was organized by the Polish Information Processing Society (Mazovia Chapter), IEEE Poland Section Computer Society Chapter, Systems Research Institute Polish Academy of Sciences, Warsaw University of Technology, Wrocław University of Economics and Business, and Institute of Information and Communication Technologies, Bulgarian Academy of Sciences.

FedCSIS 2020 was technically co-sponsored by: IEEE Poland Section, IEEE Czechoslovakia Section Computer Society Chapter, IEEE Poland Section Systems, Man, and Cybernetics Society Chapter, IEEE Poland Section Computational Intelligence Society Chapter, IEEE Poland Section Control System Society Chapter, Committee of Computer Science of the Polish Academy of Sciences, Mazovia Cluster ICT Poland, Eastern Cluster ICT Poland and Bulgarian Section of SIAM.

During FedCSIS 2020, the keynote lectures were delivered by:

- Christian Blum, Artificial Intelligence Research Institute (IIIA-CSIC), Barcelona, Spain, "*Are you a Hybrid? Yes, of course, everyone is a Hybrid nowadays!"*
- George Boustras, European University Cyprus, "*Critical Infrastructure Protection – on the interface of safety and security"*
- Hans-Georg Fill, University of Fribourg, Switzerland, *„From Digital Transformation to Digital Ubiquity: The Role of Enterprise Modeling"*

FedCSIS 2020 consisted of five Tracks. Within each Track, topical Technical Sessions have been organized. Some of these Technical Sessions have been associated with the FedCSIS conference series for many years, while some of them are relatively new. Their role is to focus and enrich discussions on selected areas pertinent to the general scope of each Track.

- **Track 1: Artificial Intelligence**
  - *Topical technical sessions:*

    - 15th International Symposium on Advanced Artificial Intelligence in Applications (AAIA'20)

    - 13th International Workshop on Computational Optimization (WCO'20)

    - 5th International Workshop on Language Technologies and Applications (LTA'20)
- **Track 2: Computer Science & Systems**
  - *Topical technical sessions:*

    - Advances in Computer Science and Systems (ACS&S'20)

    - 13th Workshop on Computer Aspects of Numerical Algorithms (CANA'20)

    - 11th Workshop on Scalable Computing (WSC'20)
- **Track 3: Network Systems and Applications**
  - *Topical technical sessions:*

    - Advances in Network Systems and Applications (ANSA'20)

    - 4th Workshop on Internet of Things – Enablers, Challenges and Applications (IoT-ECAW'20)

    - International Forum of Cyber Security, Privacy, and Trust (NEMESIS'20)
- **Track 4: Information Systems and Technology**
  - *Topical technical sessions:*

    - Advances in Information Systems and Technologies (AIST)

    - 2nd Special Session on Data Science in Health, Ecology and Commerce (DSH'20)

    - 15th Conference on Information Systems Management (ISM'20)

    - 26th Conference on Knowledge Acquisition and Management (KAM'20)
- **Track 5: Software and System Engineering**
  - *Topical technical sessions:*

    - Advances in Software and System Engineering (ASSE'20)

    - 4th International Conference on Lean and Agile Software Development (LASD'20)

    - 6th Workshop on Model Driven Approaches in System Development (MDASD'20)

    - Joint 40th IEEE Software Engineering Workshop (SEW-40) and 7th International Workshop on Cyber-Physical Systems (IWCPS-7)

The 2020 edition of an AAIA'20 Data Mining Challenge was entitled *"Network Device Workload Prediction"*. The task was related to the monitoring of large IT infrastructures, and the estimation of their resource allocation. The challenge was sponsored by the EMCA Software and the Mazowia Branch of the Polish Information Processing Society (PTI). Papers resulting from the competition are included in the Conference Proceedings (within Track 1: AI).

Each paper, found in this volume, was refereed by at least two referees, and the acceptance rate of regular full papers was ~25.8% (52 regular full papers, out of 201 general submissions).

The program of FedCSIS required a dedicated effort of many people. We would like to express our warmest gratitude to all Committee members, of each Track and each Technical Session, for their hard work in attracting and later refereeing 206 submissions (regular and data mining).

We thank the authors of papers for their great contribution into theory and practice of computing and software systems. We are grateful to the invited speakers, for sharing their knowledge and wisdom with the participants.

Last, but not least, we thank prof. Fidanova and dr. Dobrinkova. It should be stressed that they made all the preparations to organize the conference in Bulgaria. They also worked with us diligently when we were forced to move the conference online. Stefka nad Nina, we are very grateful for all your efforts! As a matter of fact, we hope to organize FedCSIS in Bulgaria as soon as the World returns to normal (even if it will be the "new normal").

We hope that you had an inspiring conference. We also hope to meet you again for the 16th Conference on Computer Science and Intelligence Systems (FedCSIS 2021). Please note an upcoming change in the conference name, from Information Systems to Intelligence Systems. The change is warranted, first, by the changes in the world around us. As can be easily observed, broadly understood, intelligence is permeating all aspects of our reality. Second, this change is already reflected by the kinds of paper submissions that are being received by all FedCSIS Tracks, and our intent to attract even more submissions related to all sorts of Intelligence Systems (including of course Artificial Intelligence, but also Business Intelligence, Management Intelligence, Human Intelligence, Financial Intelligence, Embedded Intelligence, Computational Intelligence, Collective Intelligence, Biomedical Intelligence, Military Intelligence, Network Intelligence…).

Taking into account the level of uncertainty related to COVID-19, we are seriously considering organizing the next edition of the conference online, again. However, the final decision has not been reached, yet.

Co-Chairs of the FedCSIS Conference Series

Maria Ganzha, *Warsaw University of Technology, Poland and Systems Research Institute Polish Academy of Sciences, Warsaw, Poland*
Leszek Maciaszek, *Wrocław University of Economics and Business, Wrocław, Poland and Macquarie University, Sydney, Australia*
Marcin Paprzycki, *Systems Research Institute Polish Academy of Sciences, Warsaw Poland and Management Academy, Warsaw, Poland*

Proceedings of the Federated Conference on Computer Science and Information Systems

September 6–9, 2020. Sofia, Bulgaria

TABLE OF CONTENTS

Conference Keynote Papers

Artificial Intelligence

15TH International Symposium Advances in Artificial Intelligence and Applications

5TH INTERNATIONAL WORKSHOP ON LANGUAGE TECHNOLOGIES AND APPLICATIONS

13TH INTERNATIONAL WORKSHOP ON COMPUTATIONAL OPTIMIZATION

COMPUTER SCIENCE & SYSTEMS

ADVANCES IN COMPUTER SCIENCE & SYSTEMS

13TH WORKSHOP ON COMPUTER ASPECTS OF NUMERICAL ALGORITHMS

11TH WORKSHOP ON SCALABLE COMPUTING

NETWORK SYSTEMS AND APPLICATIONS

ADVANCES IN NETWORK SYSTEMS AND APPLICATIONS

4TH WORKSHOP ON INTERNET OF THINGS - ENABLERS, CHALLENGES AND APPLICATIONS

1ST INTERNATIONAL FORUM ON CYBER SECURITY, PRIVACY AND TRUST

INFORMATION SYSTEMS AND TECHNOLOGY

ADVANCES IN INFORMATION SYSTEMS AND TECHNOLOGY

2ND SPECIAL SESSION ON DATA SCIENCE IN HEALTH, ECOLOGY AND COMMERCE

15TH CONFERENCE ON INFORMATION SYSTEMS MANAGEMENT

26TH CONFERENCE ON KNOWLEDGE ACQUISITION AND MANAGEMENT

Software and System Engineering

Advances in Software and System Engineering

Joint 40th IEEE Software Engineering Workshop and 7th International Workshop on Cyber-Physical Systems

4th International Conference on Lean and Agile Software Development

6TH WORKSHOP ON MODEL DRIVEN APPROACHES IN SYSTEM DEVELOPMENT

Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 1–4

DOI: 10.15439/2020F001
ISSN 2300-5963 ACSIS, Vol. 21

Enterprise Modeling: From Digital Transformation to Digital Ubiquity

Hans-Georg Fill
University of Fribourg
Boulevard de Pérolles 90,
1700 Fribourg Switzerland
Email: hans-georg.fill@unifr.ch

***Abstract*—While digital transformation is still a challenge for many companies when introducting digital technologies in existing processes and business models, digital ubiquity stands for the next step in digitalization. It characterizes the omnipresence of a large range of digital technologies, connectivity, and data as well as entirely digital organizations. This includes for example upcoming technologies such as distributed ledgers, artificial intelligence or augmented reality and according interfaces and data sources as well as decentralized apps and autonomous organizations. The challenge thus becomes to optimally deal with these opportunities and deploy them efficiently in business scenarios. In this paper we will investigate the role of enterprise modeling under this paradigm and how it can contribute to a well-structured, systematic understanding of complex digital phenomena for supporting business and technological decisions.**

I. Introduction and Motivation

ALMOST any existing business is today being confronted with the need to engage in digital transformation [1], [2]. May it be the provision of digital services for physical products, e.g. when a car manufacturer collects maintenance data from its customers' vehicles via remote interfaces [3], the digitalization of government services that companies need to interact with, e.g. for filing tax statements electronically [4], or the entire transformation of value chains such as banks operating without any physical presence [5]. This stems on the one hand from internal demands for gaining efficiency by using digital technologies, e.g. for optimizing throughput and lowering costs. On the other hand, external factors come into play such as the increased demand from customers for digitally-enabled offerings, the potential or already effective advancement of competitors, or the necessity to connect to business partners or the public administration through digital means.

However, digital transformation involves more than just *using* technology. In many cases, the adaptation of products, services, and processes to digitally-enabled versions requires fundamental changes in the overall business model, the organization, and the IT infrastructure [6], [7], [8], as well as the development of radically new software applications [9]. This is turn necessitates according expertise that has either to be built up within an organization or sourced from external specialists. For supporting these endeavors, enterprise modeling has traditionally been a widely used method to structure this transition and integrate the knowledge of all stakeholders [10], [11], [12].

Digital *transformation* however also implies that at some point in time the transition to a new state has been accomplished and a sufficient level of maturity is reached [13]. The question thus becomes what happens after this state has been reached and which challenges lie beyond it. In the following we will denote this state as *digital ubiquity*. Such a state could be characterized as follows: at this point, digital technologies are well integrated into products and services; the IT infrastructure offers unlimited connectivity, storage space, and massive processing power if required; data and according analytics of all business activities are available on different levels of granularity and to all necessary stakeholders; the organization is represented as a digital twin that can be used for simulations and real-time analyses, and there may even be organizations that exist only in the digital space; the organization constantly monitors and adapts to new technologies; know-how on new technologies is dynamically made available within the organization.

In such a scenario, the major challenge will thus not be to become acquainted with digital technologies in the first place and of finding ways for replacing non-digital approaches. Rather, it is necessary to quickly assess the potential of any new technical development, potentially replace existing digital components through updated ones, and adapt to them where necessary. This not only leads to potentially quick and frequent changes of complex organizational and technical environments. It also necessitates a solid and profound understanding of emerging technical concepts and their contribution to an organization's value. In the following we will briefly characterize enterprise modeling as a method of support for decision makers. Subsequently, we will show how enterprise modeling can aid in the context of digital ubiquity.

II. Enterprise Modeling

The modeling of the structure and behavior of enterprises has a long tradition in science and practice for accomplishing diverse tasks. This includes for example the analysis of an organization's capabilities and resources, and comparing them to others, the facilitation of the implementation of changes or for aiding decision makers in identifying possible options for solutions in complex environments [14], [10], [15].

Whereas the creation of models in general focuses on the abstraction from reality for specific purposes and for particular groups of individuals [16], we regard enterprise modeling as a sub-discipline of *conceptual modeling*. At its core, conceptual modeling reverts to specifically created schemas, which define artificial languages for creating valid models [17]. These languages are further composed of a visual or textual notation and an according semantics that defines the meaning of the elements of the language and how the resulting models are to be processed [14], [18]. Such language-based models with a limited set of pre-defined semantic concepts greatly ease the creation and understanding of models due to the reduced cognitive load. They permit an intuitive understanding of the contained concepts and how they are applied to create models.

In an enterprise context, such conceptual models may be used for example to formalize knowledge [14], for designing, engineering and structuring information systems [19], or for the integration of different perspectives [20]. In many cases, so-called domain-specific conceptual modeling languages are created, whose concepts are tailored towards particular application domains [21]. This includes for example modeling languages for supporting business process improvement [22], for integrating semantic technologies in information systems [23], [24], for managing risks [25], [26] or for designing product-service systems [27].

III. Enterprise Modeling for Digital Ubiquity

As we will show in the following, these properties of enterprise modeling are particularly useful in times of digital ubiquity. As outlined above, digital ubiquity is characterized by continuous changes of digital technologies and the constant adaptation of already digitalized business areas. In order to succeed in such an environment, the ability to quickly understand and adapt to new technologies is of primary concern.

Enterprise modeling can support this process through the *abstraction from complex technologies* and by presenting them in a way that facilitates their application by domain experts. An example for such an abstraction that is of high relevance for digital businesses are modeling approaches for data analytics [28], [29], [30], [31]. These permit even users with little technical knowledge to use these technologies for their tasks and thus quickly leverage their potential.

Enterprise models can further act as *interfaces to digital technologies*. Thereby, the content of the models is either processed by according engines or the models provide information for configuring machines [32]. Besides the classical example of workflow engines that execute tasks specified in the form of process models [33], more recent approaches offer interfaces to technologies such as machine learning [34], rule engines [24], blockchains [35], [36], chatbot platforms [37] or cyber-physical systems [38].

Finally, enterprise models can contribute to setting of standards as *reference models* by making best practices and successful patterns for the usage of new technologies explicit, e.g. in telecommunications [39] or for smart cities [40]. They thus contribute to the fast sharing of detailed knowledge within and across organizations. Approaches in this direction have recently been sought after for example for blockchains and distributed ledger technologies [41].

A. Exemplary Application for Distributed Ledger Technologies

For illustrating the application of enterprise models in the context of digital ubiquity, we present in the following two sample models for characterizing so-called *decentralized autonomous organizations* (DAO). DAOs are a recent phenomenon that is based on the broad availability of public blockchains. A DAO is an organization that is entirely governed through algorithms encoded in immutable blockchains so that the paradigm of *code is law* becomes a reality for all processes running in this organization [42]. Ideally, all processes are thus transparent to everyone and there is no central instance governing the organization but rather a community that is open for anyone to join. As the infrastructure of blockchains is decentralized, not even technical systems for running the according algorithms are under the control of one entity.

Although DAOs are not yet widespread, first implementations exist that can be publicly accessed. One such available DAO is Aragon[1] that is a platform for creating your own DAO. To understand how Aragon operates, users can consult the documentation on it's website. However, the information there is spread across several pages and held in technical terms. By using an enterprise modeling language for analyzing business models such as the one shown in the example in Figure 1, the core relationships between the involved partners, customers and value contributions can be investigated more easily.

The modeling language used for these *business transaction models* is based on the entities of the Business Model Canvas [43]. It extends these concepts however by adding explicit relationships between the entities as well as advanced functionalities for guiding the user through the creation and analysis of business models [12]. In this way, the actual behavior of a business model can be depicted and analyzed both visually and with algorithms.

Further, a user may want to investigate the enterprise architecture behind the Aragon DAO for understanding how its business functions are aligned with the underlying technology. Also in this respect enterprise models can be of great value for making these relationships explicit. As shown in Figure 2, the standardized modeling language of ArchiMate can show how business entities such as customers of a DAO based on Aragon interact via specific roles with the offered business functions and processes and how these are realized on the application and technology layer [7]. Further, such models permit conducting algorithmic analyses of this enterprise architecture [44], e.g. to determine which components depend on each other, whether sufficient backup systems have been installed, whether the architecture complies with legal regulations such as data protection, which systems need to be transitioned to updated

[1] See https://aragon.org/

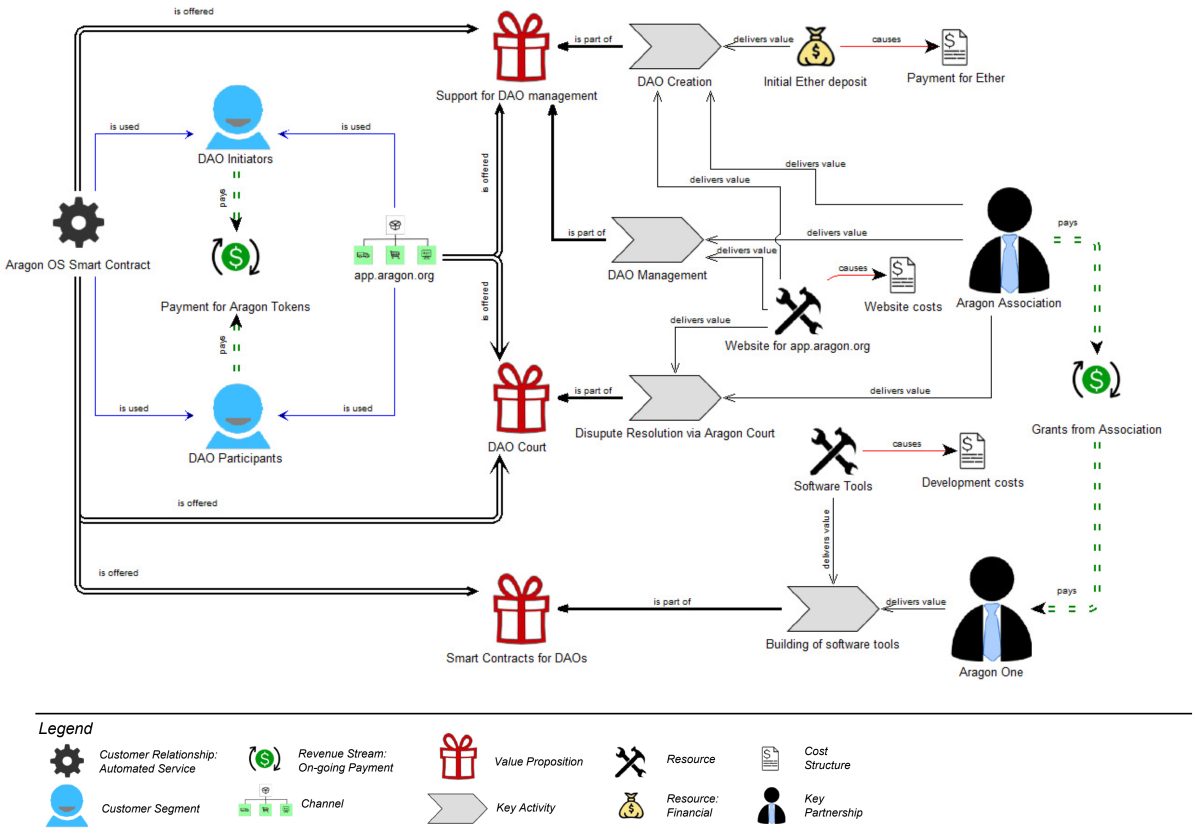


Fig. 1. Business Transaction Model of Aragon for Decentralized Autonomous Organizations

versions due to security issues, whether the systems are oriented towards scalability or to identify affected business processes in case of failures for ensuring business continuity.

IV. Conclusion and Outlook

As we have seen, enterprise modeling can aid in the structuring of complex domains, the abstraction from technologies for easing user interaction and for providing best practices in the form of reference models. These features make it useful in times of digital ubiquity where the application of digital technologies is fast paced and continuously changing. Future challenges will include the combination of enterprise modeling with recent digital technologies such as for example the upcoming distributed ledger technologies or augmented reality environments. The use of domain-specific modeling languages thereby eases the interaction with technologies and permits to quickly integrate them in organizational processes.

References

[1] P. Drews and T. Böhmann, "Riding the digital transformation wave," *Bus. Inf. Syst. Eng.*, vol. 59, no. 4, pp. 302–303, 2017. doi: 10.1007/s12599-017-0484-2

[2] C. Legner, T. Eymann, T. Hess, C. Matt, T. Böhmann, P. Drews, A. Maedche, N. Urbach, and F. Ahlemann, "Digitalization: Opportunity and challenge for the business and information systems engineering community," *Bus. Inf. Syst. Eng.*, vol. 59, no. 4, pp. 301–308, 2017. doi: 10.1007/s12599-017-0484-2

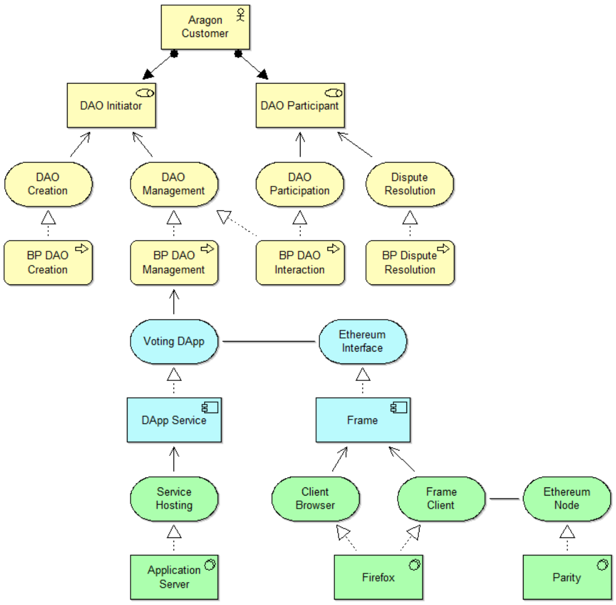


Fig. 2. Enterprise Architecture Excerpt of the Aragon DAO Using ArchiMate

[3] R. Dhall and V. K. Solanki, "An iot based predictive connected car maintenance approach," *IJIMAI*, vol. 4, pp. 16–22, 2017.

[4] C. C. Mattias, "Swiss taxpayers confident about mandatory vat e-filing," *International Tax Review*, Dec 18 2019.

[5] R. M. Stulz, "Fintech, bigtech, and the future of banks," *Journal of Applied Corporate Finance*, vol. 31, no. 4, pp. 86–97, 2019. doi: 10.1111/jacf.12378

[6] A. Caetano, G. Antunes, J. Pombinho, M. Bakhshandeh, J. Granjo, J. Borbinha, and M. M. Da Silva, "Representation and analysis of enterprise models with semantic techniques: an application to archimate, e3value and business model canvas," *Knowledge and Information Systems*, vol. 50, no. 1, pp. 315–346, 2017.

[7] B. Pittl and D. Bork, "Modeling digital enterprise ecosystems with archimate: A mobility provision case study," in *Serviceology for Services - 5th International Conference, ICServ 2017, Vienna, Austria, July 12-14, 2017, Proceedings*, ser. Lecture Notes in Computer Science, Y. Hara and D. Karagiannis, Eds., vol. 10371. Springer, 2017. doi: 10.1007/978-3-319-61240-9_17 pp. 178–189.

[8] R. Winter, *Business Engineering Navigator: Gestaltung und Analyse von Geschäftslösungen" Business-to-IT"*. Springer-Verlag, 2010.

[9] C. Ebert and C. H. C. Duarte, "Digital transformation," *IEEE Software*, no. 4, pp. 16–21, 2018.

[10] K. Sandkuhl, H.-G. Fill, S. Hoppenbrouwers, J. Krogstie, F. Matthes, A. Opdahl, G. Schwabe, Ö. Uludag, and R. Winter, "From expert discipline to common practice: a vision and research agenda for extending the reach of enterprise modeling," *Business & Information Systems Engineering*, vol. 60, no. 1, pp. 69–80, 2018.

[11] K. Pousttchi, "A modeling approach and reference models for the analysis of mobile payment use cases," *Electron. Commer. Res. Appl.*, vol. 7, no. 2, pp. 182–201, 2008. doi: 10.1016/j.elerap.2007.07.001

[12] M. Wieland and H. Fill, "A domain-specific modeling method for supporting the generation of business plans," in *Modellierung 2020*, ser. LNI, D. Bork, D. Karagiannis, and H. C. Mayr, Eds., vol. P-302. Gesellschaft für Informatik e.V., 2020, pp. 45–60. [Online]. Available: https://dl.gi.de/20.500.12116/31846

[13] S. Berghaus and A. Back, "Stages in digital business transformation: Results of an empirical maturity study," in *10th Mediterranean Conference on Information Systems*. University of Nicosia / AISeL, 2016, p. 22.

[14] D. Bork and H. Fill, "Formal aspects of enterprise modeling methods: A comparison framework," in *47th Hawaii International Conference on System Sciences*. IEEE Computer Society, 2014. doi: 10.1109/HICSS.2014.422 pp. 3400–3409.

[15] H. Fill, "Using semantically annotated models for supporting business process benchmarking," in *Perspectives in Business Informatics Research - 10th International Conference, BIR 2011, Riga, Latvia, October 6-8, 2011. Proceedings*, ser. Lecture Notes in Business Information Processing, J. Grabis and M. Kirikova, Eds., vol. 90. Springer, 2011, pp. 29–43.

[16] H. Stachowiak, *Allgemeine Modelltheorie*. Springer, 1973.

[17] H. Fill and D. Karagiannis, "On the Conceptualisation of Modelling Methods Using the ADOxx Meta Modelling Platform," *Enterp. Model. Inf. Syst. Archit. Int. J. Concept. Model.*, vol. 8, no. 1, pp. 4–25, 2013. doi: 10.18417/emisa.8.1.1

[18] H.-G. Fill, *Visualisation for Semantic Information Systems*. Springer, 2009.

[19] Y. Wand and R. Weber, "Research commentary: Information systems and conceptual modeling - A research agenda," *Inf. Syst. Res.*, vol. 13, no. 4, pp. 363–376, 2002. doi: 10.1287/isre.13.4.363.69

[20] D. Karagiannis and P. Höfferer, "Metamodeling as an integration concept," in *International Conference on Software and Data Technologies*. Springer, 2006, pp. 37–50.

[21] N. Visic, H.-G. Fill, R. A. Buchmann, and D. Karagiannis, "A domain-specific language for modeling method definition: From requirements to grammar," in *2015 IEEE 9th International Conference on Research Challenges in Information Science (RCIS)*. IEEE, 2015, pp. 286–297.

[22] F. Johannsen and H.-G. Fill, "Meta modeling for business process improvement," *Business & Information Systems Engineering*, vol. 59, no. 4, pp. 251–275, 2017.

[23] H.-G. Fill, "SeMFIS: A flexible engineering platform for semantic annotations of conceptual models," *Semantic Web*, vol. 8, no. 5, pp. 747–763, 2017.

[24] B. Pittl and H.-G. Fill, "A visual modeling approach for the semantic web rule language," *Semantic Web*, vol. 11, no. 2, pp. 361–389, 2020.

[25] H.-G. Fill, "An approach for analyzing the effects of risks on business processes using semantic annotations," in *European Conference on Information Systems, ECIS*. ESADE / AIS, 2012.

[26] S. Strecker, D. Heise, and U. Frank, "RiskM: A multi-perspective modeling method for IT risk assessment," *Information Systems Frontiers*, vol. 13, no. 4, pp. 595–611, 2011.

[27] X. Boucher, K. Medini, and H.-G. Fill, "Product-service-system modeling method," in *Domain-specific Conceptual Modeling*, D. Karagiannis, H. Mayr, and J. Mylopoulos, Eds. Springer, 2016, pp. 455–482.

[28] W. Grossmann and C. Moser, "Big data—integration and cleansing environment for business analytics with dice," in *Domain-Specific Conceptual Modeling: Concepts, Methods and Tools*, D. Karagiannis, H. C. Mayr, and J. Mylopoulos, Eds. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-39417-6_5 pp. 103–123.

[29] H.-G. Fill and F. Johannsen, "A knowledge perspective on big data by joining enterprise modeling and data analyses," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, 2016, pp. 4052–4061.

[30] H. Khalajzadeh., A. J. Simmons., M. Abdelrazek., J. Grundy., J. Hosking., and Q. He., "Visual languages for supporting big data analytics development," in *15th International Conference on Evaluation of Novel Approaches to Software Engineering - Volume 1: ENASE,*. SciTePress, 2020, pp. 15–26.

[31] S. Nalchigar and E. Yu, "Designing business analytics solutions," *Business & Information Systems Engineering*, vol. 62, no. 1, pp. 61–75, 2020.

[32] H. Demirkan, R. J. Kauffman, J. A. Vayghan, H. Fill, D. Karagiannis, and P. P. Maglio, "Service-oriented technology and management: Perspectives on research and practice for the coming decade," *Electron. Commer. Res. Appl.*, vol. 7, no. 4, pp. 356–376, 2008.

[33] V. Ferme, J. Lenhard, S. Harrer, M. Geiger, and C. Pautasso, "Workflow management systems benchmarking: unfulfilled expectations and lessons learned," in *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*. IEEE, 2017, pp. 379–381.

[34] I. Mierswa, "Rapid miner," *KI*, vol. 23, no. 2, pp. 62–63, 2009.

[35] H.-G. Fill and F. Härer, "Knowledge Blockchains: Applying Blockchain Technologies to Enterprise Modeling," in *Proceedings of the 51st Hawaii International Conference on System Sciences (HICSS-51)*, Waikoloa Village, Hawaii, USA, 2018. doi: 10.24251/HICSS.2018.509. ISBN 978-0-9981331-1-9 pp. 4045–4054.

[36] F. Härer and H.-G. Fill, "Decentralized Attestation of Conceptual Models Using the Ethereum Blockchain," in *21st IEEE International Conference on Business Informatics (CBI 2019)*, Moscow, Russia, 2019.

[37] G. Daniel, J. Cabot, L. Deruelle, and M. Derras, "Xatkit: A multimodal low-code chatbot development framework," *IEEE Access*, vol. 8, pp. 15 332–15 346, 2020.

[38] M. Walch, "Knowledge-driven enrichment of cyber-physical systems for industrial applications using the kbr modelling approach," in *2017 IEEE International Conference on Agents (ICA)*, 2017, pp. 84–89.

[39] C. Czarnecki and C. Dietze, "Domain-specific reference modeling in the telecommunications industry," in *Designing the Digital Transformation*, A. Maedche, J. vom Brocke, and A. Hevner, Eds. Springer, 2017, pp. 313–329.

[40] D. Bork, H. Fill, D. Karagiannis, E. Miron, N. Tantouris, and M. Walch, "Conceptual modelling for smart cities: A teaching case," *IxD&A*, vol. 27, pp. 10–27, 2015.

[41] M. C. Lacity and S. Khan, "Exploring preliminary challenges and emerging best practices in the use of enterprise blockchains applications," in *52nd Hawaii International Conference on System Sciences*, T. Bui, Ed. ScholarSpace, 2019. doi: 10.24251/HICSS.2019.563 pp. 1–10.

[42] S. Wang, W. Ding, J. Li, Y. Yuan, L. Ouyang, and F.-Y. Wang, "Decentralized autonomous organizations: Concept, model, and applications," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 5, pp. 870–878, 2019.

[43] A. Osterwalder and Y. Pigneur, *Business model generation: a handbook for visionaries, game changers, and challengers*. John Wiley & Sons, 2010.

[44] C. Moser and F. Bayer, "IT architecture management: A framework for IT-Services," in *Enterprise Modelling and Information Systems Architectures, Proceedings of the Workshop in Klagenfurt, October 24-25, 2005*, ser. LNI, J. Desel and U. Frank, Eds., vol. P-75. GI, 2005, pp. 137–151.

15th International Symposium Advances in Artificial Intelligence and Applications

AAIA'20 brings together scientists and practitioners to discuss their latest results and ideas in all areas of Artificial Intelligence. We hope that successful applications presented at AAIA'20 will be of interest to researchers who want to know about both theoretical advances and latest applied developments in AI.

Topics

Papers related to theories, methodologies, and applications in science and technology in the field of AI are especially solicited. Topics covering industrial applications and academic research are included, but not limited to:

- Decision Support
- Machine Learning
- Fuzzy Sets and Soft Computing
- Rough Sets and Approximate Reasoning
- Data Mining and Knowledge Discovery
- Data Modeling and Feature Engineering
- Data Integration and Information Fusion
- Hybrid and Hierarchical Intelligent Systems
- Neural Networks and Deep Learning
- Bayesian Networks and Bayesian Reasoning
- Case-based Reasoning and Similarity
- Web Mining and Social Networks
- Business Intelligence and Online Analytics
- Robotics and Cyber-Physical Systems
- AI-centered Systems and Large-Scale Applications

Professor Zdzisław Pawlak Best Paper Awards

We are proud to continue the tradition started at the AAIA'06 and grant two "Professor Zdzisław Pawlak Best Paper Awards" for contributions which are outstanding in their scientific quality. The two award categories are:

- Best Student Paper. Papers qualifying for this award must be marked as "Student full paper" to be eligible.
- Best Paper Award.

Each award carries a prize of 300 EUR funded by the Mazowsze Chapter of the Polish Information Processing Society.

Technical Session Chairs

- **Ślęzak, Dominik,** University of Warsaw, Poland
- **Szczuka, Marcin,** University of Warsaw, Poland

Steering Committee

- **Kacprzyk, Janusz,** Polish Academy of Sciences, Poland
- **Kwaśnicka, Halina,** Wrocław University of Science and Technology, Poland
- **Marek, Victor,** University of Kentucky, United States
- **Matwin, Stan,** Dalhousie University, Canada
- **Michalewicz, Zbigniew,** University of Adelaide, Australia
- **Skowron, Andrzej,** Polish Academy of Sciences, Poland

Program Committee

- **A, Mani,** International Rough Set Society, HBCSE, Tata Institute of Fundamental Research, India
- **Agre, Gennady,** Bulgarian Academy of Sciences, Bulgaria
- **Betliński, Paweł,** Security On-Demand, United States
- **Bianchini, Monica,** University of Siena, Italy
- **Calpe, Javier,** University of Valencia, Spain
- **Chelly Dagdia, Zaineb,** INRIA, France
- **Cyganek, Bogusław,** AGH University of Science and Technology, Poland
- **Düntsch, Ivo,** Fujian Normal University, China & Brock University, Canada
- **Froelich, Wojciech,** University of Silesia, Poland
- **Girardi, Rosario,** UNIRIO, Brazil
- **Grabowski, Adam,** Institute of Informatics, University of Bialystok, Bialystok, Poland
- **Ignatov, Dmitry,** National Research University Higher School of Economics, Russia
- **Janusz, Andrzej,** University of Warsaw, Poland
- **Jaromczyk, Jerzy,** University of Kentucky, United States
- **Jin, Xiaolong,** Institute of Computing Technology, Chinese Academy of Sciences, China
- **Kasprzak, Włodzimierz,** Warsaw University of Technology, Poland
- **Kayakutlu, Gulgun,** Istanbul Technical University, Turkey
- **Kryszkiewicz, Marzena,** Warsaw University of Technology, Poland
- **Lavrov, Eugeniy,** Sumy State University, Ukraine
- **Lingras, Pawan,** Saint Mary's University, Canada
- **Loukanova, Roussanka,** Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Bulgaria
- **Markowska-Kaczmar, Urszula,** Wrocław University of Science and Technology, Poland
- **Matson, Eric T.,** Purdue University, United States
- **Menasalvas, Ernestina,** Universidad Politécnica de Madrid, Spain
- **Meneses, Claudio,** Universidad Católica del Norte, Chile

- **Moshkov, Mikhail,** King Abdullah University of Science and Technology, Saudi Arabia
- **Myszkowski, Paweł B.,** Wrocław University of Science and Technology, Poland
- **Pancerz, Krzysztof,** University of Rzeszów, Poland
- **Pataricza, Andras,** Budapest University of Technology and Economics, Hungary
- **Po, Laura,** Università di Modena e Reggio Emilia, Italy
- **Porta, Marco,** University of Pavia, Italy
- **Przybyła-Kasperek, Małgorzata,** University of Silesia, Poland
- **Raghavan, Vijay,** University of Louisiana at Lafayette, United States
- **Ramanna, Sheela,** University of Winnipeg, Canada
- **Raś, Zbigniew,** University of North Carolina at Charlotte, United States
- **Rauch, Jan,** University of Economics, Prague, Czech Republic
- **Reformat, Marek,** University of Alberta, Canada
- **Salem, Abdel-Badeeh M.,** Ain Shams University, Egypt
- **Schaefer, Gerald,** Loughborough University, United Kingdom
- **Sikora, Marek,** Silesian University of Technology, Poland
- **Sosnowski, Łukasz,** Systems Research Institute, Polish Academy of Sciences and Dituel Sp. z o.o., Poland
- **Stańczyk, Urszula,** Silesian University of Technology, Poland
- **Stell, John,** University of Leeds, United Kingdom
- **Stoean, Catalin,** University of Craiova, Romania
- **Subbotin, Sergey,** Zaporizhzhya National Technical University, Ukraine
- **Świechowski, Maciej,** QED Software, Poland
- **Zdravevski, Eftim,** Ss.Cyril and Methodius University, Faculty of Computer Science and Engineering, Macedonia
- **Zielosko, Beata,** University of Silesia, Poland

DOI: 10.15439/2020F149
ISSN 2300-5963 ACSIS, Vol. 21

Superiority of Simplicity: A Lightweight Model for Network Device Workload Prediction

Alexander Acker*†, Thorsten Wittkopp*†, Sasho Nedelkoski*, Jasmin Bogatinovski*, Odej Kao*
* Technische Universität Berlin, Germany
{alexander.acker, t.wittkopp, sasho.nedelkoski, jasmin.bogatinovski, odej.kao}@tu-berlin.de
† Alphabetic order, equal contribution

Abstract*—The rapid growth and distribution of IT systems increases their complexity and aggravates operation and maintenance. To sustain control over large sets of hosts and the connecting networks, monitoring solutions are employed and constantly enhanced. They collect diverse key performance indicators (KPIs) (e.g. CPU utilization, allocated memory, etc.) and provide detailed information about the system state. Predicting the future progress of those KPIs allows ahead of time optimizations like anomaly detection or predictive maintenance and can be defined as a time series forecasting problem. Although, a variety of time series forecasting methods exist, forecasting the progress of IT system KPIs is very hard. First, KPI types like CPU utilization or allocated memory are very different and hard to be modelled by the same model. Second, system components are interconnected and constantly changing due to soft- or firmware updates and hardware modernization. Thus a frequent model retraining or fine-tuning must be expected. Therefore, we propose a lightweight solution for KPI series forecasting. It consists of a weighted heterogeneous ensemble method composed of two models - a neural network and a mean predictor. As ensemble method a weighted summation is used, whereby a heuristic is employed to set the weights. The modelling approach is evaluated on the available FedCSIS 2020 challenge dataset and achieves an overall R^2 score of 0.10 on the preliminary 10% test data and 0.15 on the complete test data. We publish our code on the following github repository: *https://github.com/citlab/fed_challenge**

I. Introduction

IT systems are rapidly evolving to meet the growing demand for new applications and services in a variety of fields like industry, medicine or autonomous transportation. This entails an increasing number of interconnected devices, large networks and growing data centres to provide the required infrastructure. Although accelerating innovations and business opportunities, this trend increases complexity and thus, aggravates the operation and maintenance of these systems. Operators are in need of assistance to be able to maintain control over this complexity. Therefore, monitoring solutions are implemented. They constantly collect system KPIs like latency, throughput, or system resource utilization and provide detailed information about the monitored IT system. One particularly important aspect of system monitoring is the prediction of future system load. Several efforts where made to enable this ranging from linear regression [1], Bayesian statistics [2] and neural networks [3].

A precise prediction of future system load enables ahead of time decision making. An anomaly detection methods can be employed to compare the difference between the predicted and the actual state and raise alarms in case of unforeseen deviations [4]. Furthermore, scheduling decision [5], network routing and dimensioning [6], data centre cooling control [7] or predictive maintenance [8] all benefit from precise system load predictions.

The task of system load prediction can be formulated as a time series forecasting problem but comes with specific challenges. First, different KPI types are highly non-uniform. CPU utilization is usually very volatile, memory allocation is rarely overlaid by noise and disk read and write operations expose bursty patterns due to buffering resulting in flat sequences with sporadic peaks. The concrete pattern of these series depend of partly unknown external and a variety of internal factors. There are temporal dependencies night- and daytime hours or occasional events like Christmas days influencing the system load. Also, the IT system itself is problematic from modeling perspective due to their dynamic nature and high uncertainty. Frequent soft- and firmware updates or hardware modernization change system properties and usually require model retraining or fine-tuning. This imposes the requirement of frequent and fast model adaption.

Related work on time series forecasting is diverse and ranges from traditional linear or non-linear regression [9], stochastic methods [10], deep learning models [11] and ensemble methods [12], [13]. Traditional regressive or statistical models are often not able to capture the underlying complex processes while neural networks or ensemble methods suffer from high complexity and an accompanying high computational overhead.

Considering this, we present our solution for this years FedCSIS 2020 challenge [14], which is a model for network device workload prediction. It combines the overall average of each KPI series with a prediction from a linear neural network. Furthermore, we employed heuristics to tackle numerical imprecision and enhance overall prediction performance. Our solution achieved an overall R^2 score of 0.105 on the preliminary 10% test data and 0.15 on the complete test data.

The rest of the paper is structured as follows. Section II provides a preliminary analysis of the problem and available training data set. Section III introduces our solution for workload prediction. It includes a formal problem definition and explains each element of our proposed method. An evaluation is performed and results are presented in section IV. Finally section V concludes our paper.

II. Network Device Workload Prediction

This year FedCSIS 2020 challenge [14] was to predict the future workload of network devices based on past workload observations. More specifically, the workload of a set of devices, referred to as hosts, were characterized by KPI series such as CPU utilization, incoming and outgoing network traffic or allocated main memory. The data were collected hourly over a period of 3 months with sporadically missing samples. Overall, 45 different KPIs were recorded from 3,716 hosts, whereby the workload of individual hosts was described by different KPI subsets. Each hourly KPI series sample consists of seven aggregated measurements. These are the number of collected samples, the mean and standard deviation, the first, last, highest and lowest measurement. Out of the seven aggregations only the mean value must be predicted, resulting in a possibly multivariate input but univariate output.

The plots in Fig. 1 show four different KPI mean values from six different hosts. Thereby, the series was split into weekly windows from Monday until Sunday and arranged by the hour of the week resulting in ten aggregated weekly series for each plot. The dark line shows the mean value while the light line visualizes the 0.95 confidence interval. It can be observed that KPI series are highly non-uniform, which indicates the major challenge when faced with forecasting the expected future values of the KPIs.

III. Lightweight Workload Prediction Model

In this section we present our method for lightweight workload prediction. Its concept and architecture were chosen based on the previously described observations and analyses in section II.

A. Preliminaries

We define the task of workload prediction as a time series forecasting problem. A time series is an temporally ordered sequence of values $X = (X_t(\cdot) \in \mathbb{R}^d : t = 1, 2, \dots, T)$, where d is the dimensionality of each point. For $X_b^a(\cdot) = (X_a(\cdot), X_{a+1}(\cdot), \dots, X_b(\cdot))$, we denote indices a and b with $a \leq b$ and $0 \leq a, b \leq T$ as time series boundaries in order to slice a given series $X_T^0(\cdot)$ and acquire a subseries $X_b^a(\cdot)$. The variable T defines the time stamp of the last sample of the past observations. Additionally, we use the notion $X(i)$ to refer to a certain dimension i, with $1 \leq i \leq d$. Furthermore, meta information for each time series value $X_t(\cdot)$ are denoted as M_t.

The problem of workload prediction is modelled as the forecasting of a future univariate value $X_{T+w}(i)$, with $w \geq 1$, conditioned on a sequence of past values $X_T^0(\cdot)$, and known meta information about the future time stamp M_{T+w}. Therefore, the learning objective is to select a function $h : \mathbb{R}^N \mapsto \mathbb{R}$, where N is the dimensionality of the input, that results in a small generalization loss:

$$\mathcal{L} = \frac{1}{|\mathcal{W}|} \sum_{w \in \mathcal{W}} L(h(X_T^0(\cdot), M_{T+w}), X_{T+w}(\cdot)). \qquad (1)$$

Thereby, L is a bounded loss function and $\mathcal{W}$ is the set of offsets defining all future time stamps to predict.

B. Lightweight Workload Prediction Model

The overall architecture of our method is depicted in Fig. 2. A future time series value $X_{T+w}(i)$ should be predicted based on the history $X_T^0(\cdot)$ and its known meta information M_{T+w}. For the task of workload prediction, each time series X represents an KPI. The respective dimensions of samples $X_t(\cdot)$ are aggregated values of that KPI between time $t-1$ and t. Due to their importance, we selectively define the mean and last measurement as $\overline{x}_t$ and $x_t^{(l)}$, where $\overline{x}_t, x_t^{(l)} \in X_t(\cdot)$. The mean value of the sample $\overline{x}_{T+w} \in X_{T+w}(\cdot)$ is the prediction target. Since many workload series are seasonal, we additionally add the encoded day of week and hour of day as meta information M_{T+w}. Subsequently, each model element is described in detail.

Preprocessing. Initially, a rescaling of each value in the KPI series $X_T^0(\cdot)$ to a fixed upper bound d and a respectively linear scale to the lower bound is performed. Furthermore, values in $X_T^0(\cdot)$ are expected to be sampled hourly. If samples are missing, a linear interpolation is employed.

Feature Selection. Due to the additional overhead that is introduced by automated feature selection methods, we choose to select a fixed subset of features manually. The features are selected depending on the model that they are forwarded to. Therefore, we define a filter F_1 for the mean predictor and a filter F_2 for the neural network model (NN). The filter F_1 includes only the mean values of $X_T^0(\cdot)$. Filter F_2 applies two feature selection operations. First, out of the aggregated values in the last available series sample, we pick the mean and last value, i.e. $\overline{x}_T, x_T^{(l)} \in X_T(\cdot)$. Second, motivated by the seasonality of system load, we additionally use the mean value of the same hour of the week as the prediction target of previous k weeks.

The Models. The mean predictor calculates the overall average over the filtered sample series $F_1(X_T^0(\cdot))$. As NN a linear feed-forward neural network is used. It receives the preprocessed and filtered data $F_2(X_T^0(\cdot))$, the meta-information values M_{T+w} and the output of the mean model. These are combined to a flat input vector $\mathbf{x}$. The learning objective is to minimize the squared error loss between the prediction and the mean value of $\overline{x}_{T+w} \in X_{T+w}(\cdot)$:

$$L = (h(\mathbf{x}) - \overline{x}_{T+w})^2. \qquad (2)$$

The proposed NN architecture is a fanning out first hidden layer. The subsequent layers are tampered, which works as regularization. We use a dropout between the first and second hidden layer as an additional regularization. A rectifier linear unit (ReLU) activation is applied to the output value of the network. The output of the mean model and NN model are respectively denoted as $o_{T+w}^{(1)}$ and $o_{T+w}^{(2)}$.

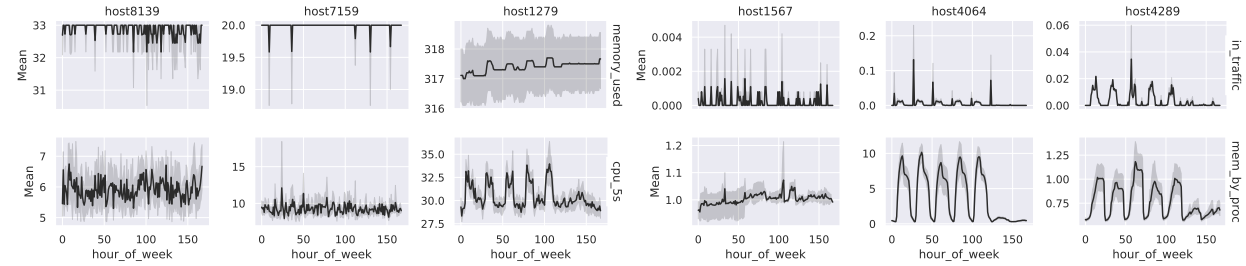


Fig. 1. Example of four KPIs for six hosts. A great in-between and within KPI value diversity for the different hosts can be observed.

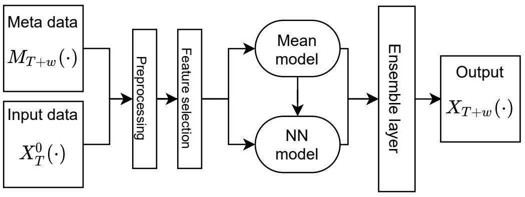


Fig. 2. Overall solution architecture.

Ensemble Layer. To combine the predictions of the mean model and the NN model, a weighted average over the model outputs is calculated:

$$o_{T+w} = \sum_{o^{(i)}_{T+w} \in \{o^{(1)}_{T+w}, o^{(2)}_{T+w}\}} w_i o^{(i)}_{T+w}, \text{where} \sum_i w_i = 1. \quad (3)$$

The usage of two models is motivated by the non-uniformity of KPI series. While the neural network is capable to predict seasonal series fairly well, it fails to accurately predict constant but noisy series. A simple average over all mean metrics of a KPI resulted in good predictions for constant but noisy series but resulted in bad predictions for seasonal series. By combining both, we expect to achieve a generally better result.

IV. Evaluation

Based on the provided dataset, the future progress of 10,000 KPI series must be predicted. Samples are sampled hourly. This results in a sequence of 168 samples that have to be predicted for each series. In this section, we evaluate the proposed method in terms of runtime and prediction performance.

A. Training and Parameterization

KPI series are diverse depending on the type and the host from which they were collected. Therefore, we choose to train individual models for each KPI series. The mean predictor calculates an overall mean over all mean values from the available three months of data.

Training of the NN requires the definition of a training set. Therefore, a set of inputs and prediction targets are defined. The target is always a specific mean value $\overline{x}_{t_p} \in X_{t_p}(\cdot)$ at prediction target time stamp $t_p \leq T$. The hour of day $m_1 \in \{1, 2, \ldots, 24\}$ and day of week $m_2 \in \{1, 2, \ldots, 7\}$ are defined as meta information $M_{t_p} = \{m_1, m_2\}$. This KPI training series slice is defined as $X^s_e(\cdot)$ with $e = t_p - ((m_2 - 1) * 24 + m_1)$ and $s = e - 168 * k$, where 168 are the hours of one week, $s \geq 0$ and $k \geq 1$. Thereof, the mean and last value from the last sample are selected $\overline{x}_e, x^l_e \in X_e(\cdot)$. Further, respecting the seasonality of several KPI series, the mean value of the same hour of the week as the prediction target is added to the input. These can be accessed via $\{\overline{x}_\tau \in X_\tau(\cdot) : \tau = t_p - i * 168, i = 1, 2, \ldots, k\}$

To create the training data we set $k = 2$. For the rescaling, we define $d = 100$. Training of the NN is done via backpropagatuion on the mean square error as optimization criterion and Adam as the optimizer. We set the learning rate to $10e^-3$ and use dropout probability of 0.1.

B. Runtime Analysis

A preliminary runtime analysis is conducted where our neural network is compared to a recurrent version of it. For the recurrent network, we use long short term memory (LSTM) instead of linear cells. We measure the training time per epoch on a bare-metal machine with an Intel(R) Core(TM) i5-9600K CPU @ 3.70GHz, 3x32 GB RAM and two Nvidia GeForce RTX 2080 Titan GPUs whereof one was utilized during the runtime measurement experiments. Ubuntu 18.04.3 LTS with kernel version 5.3.0-51-generic is installed as OS and Python version 3.6.7 and PyTorch version 1.4.0 are used to implement the networks.

The LSTM version requires significantly more time for training than the network with linear cells. In comparison, the runtime increases by a factor of ten. The mean training runtime per epoch of the linear version is 2.37 seconds per epoch with a standard deviation of 0.03 and 0.95 confidence interval of $[2.38, 2.37]$. For the network version with LSTM cells a training time per epoch of 25.72 seconds per epoch is measured with a standard deviation of 0.18 and 0.95 confidence interval of $[25.74, 2.70]$. Having six training epochs per series and a total number of $10,000$ series means a total required training time of 39.5 hours for the linear cells and 17.9 days when using LSTM cells.

Although recurrent neural network architectures especially with LSTM cells are reported to perform well on sequential data prediction tasks [15], our runtime analysis shows that the required training time is very high and considered as infeasible.

TABLE I
R^2 SCORES OF BEST THREE SUBMISSIONS TOGETHER WITH THE BASELINE.

	baseline	1st	2nd	Ours
Preliminary test set (10%)	0.2267	0.1888	0.1841	0.1053
Complete test set (100%)	0.2295	0.163	0.1515	0.1501

C. Prediction Results

The performance of the proposed workload prediction method is evaluated against the withheld test set by submitting the solution via the official FedCSIS 2020 challenge submission system. The submissions are scored by the R^2 score defined as

$$R^2 = 1 - \frac{\sum_t (\overline{x}_t - o_t)^2}{\sum_t (\overline{x}_t - \overline{\overline{x}})^2}, \tag{4}$$

where $\overline{x}_t \in X_t(\cdot)$ and $\overline{\overline{x}}$ as the overall average over all mean samples. Based on our observation several KPI series are mainly constant with sporadic deviations, resulting in a very small normalization value (denominator of in Eq. 4). This results in high division values and thus, low R^2 scores even for small deviations of the predicted values. These values had a negative impact on the overall R^2 score. Furthermore, several KPI series can be described as the noise around a baseline. This motivates us to implement a heuristic to choose an adaptive weighting of the model outputs. First, the neural network is trained. Second, the last available week is used as a prediction target and the data before that week as input. Since this last week was explicitly trained on, we assume precise prediction results, i.e. R^2 score close to 1. If the neural network output resulted in a lower score than the output of the average predictor, we set the weight for the average predictor to 1.0 and the neural network weight to 0.0. Otherwise, the both weights were set to 0.5.

Finally, the prediction of the submission is done based on the k last available weeks in the training data set. The R^2 scores of the best three submissions are listed in TABLE I. None of the submitted results is able to achieve the specified baseline. Two submissions achieved a better R^2 score than our solution with 0.1888 and 0.1841 on the preliminary 10% of test data and 0.163 and 0.1515 on the complete test dataset. With our proposed lightweight model, we achieve an R^2 score of 0.1053 on the preliminary 10% test data and 0.1501 on the complete test dataset. We did not carry out any attempts to optimize for the 10% preliminary test data since it was not clear whether it is a general representation of the complete test dataset. Therefore, it is interesting to observe that our solution is the only one achieving a better score on the complete dataset than on the preliminary 10%.

V. Conclusion

We tackle the given challenge of network device workload prediction based on KPI data with a lightweight model that ensembles the predictions of a neural network and a mean predictor. The ensemble is done by a weighted summation. A heuristic is used to selectively set the weights for each model prediction. The lightweight nature of the method allows training individual models for each KPI series respecting the diverse nature of different KPI types and host. Furthermore, frequent retraining is feasible with the proposed solution.

We provide a runtime analysis between LSTM cells and linear cells showing revealing the usage of LSTM cells as infeasible. We evaluate our solution against the FedCSIS 2020 challenge dataset. The experiment results show that the lightweight approach predicts future KPI values with an overall R^2 score of 0.105 on the preliminary 10% test data and 0.15 on the complete test data.

For future work, we see further experimentation with different network types like convolutional neural networks or attention mechanisms as promising. Furthermore, the learning of the summation weights when aggregating mean predictor and neural network outputs are sources for potential optimization.

References

[1] P. A. Dinda and D. R. O'Hallaron, "Host load prediction using linear models," *Cluster Computing*, vol. 3, no. 4, pp. 265–280, 2000.

[2] S. Di, D. Kondo, and W. Cirne, "Host load prediction in a google compute cloud with a bayesian model," in *SC'12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*. IEEE, 2012, pp. 1–11.

[3] B. Song, Y. Yu, Y. Zhou, Z. Wang, and S. Du, "Host load prediction with long short-term memory in cloud computing," *The Journal of Supercomputing*, vol. 74, no. 12, pp. 6554–6568, 2018.

[4] F. Schmidt, F. Suri-Payer, A. Gulenko, M. Wallschläger, A. Acker, and O. Kao, "Unsupervised anomaly event detection for vnf service monitoring using multivariate online arima," in *2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. IEEE, 2018, pp. 278–283.

[5] J. W. Jiang, T. Lan, S. Ha, M. Chen, and M. Chiang, "Joint vm placement and routing for data center traffic engineering," in *2012 Proceedings IEEE INFOCOM*. IEEE, 2012, pp. 2876–2880.

[6] A. Howard, A. Zhmoginov, L.-C. Chen, M. Sandler, and M. Zhu, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," 2018.

[7] F. Ahmad and T. Vijaykumar, "Joint optimization of idle and cooling power in data centers while maintaining response time," *ACM Sigplan Notices*, vol. 45, no. 3, pp. 243–256, 2010.

[8] M. Yaseen, D. Swathi, and T. A. Kumar, "Iot based condition monitoring of generators and predictive maintenance," in *2017 2nd International Conference on Communication and Electronics Systems (ICCES)*. IEEE, 2017, pp. 725–729.

[9] J. H. Stock and M. W. Watson, "A comparison of linear and nonlinear univariate models for forecasting macroeconomic time series," National Bureau of Economic Research, Tech. Rep., 1998.

[10] M. R. Hassan and B. Nath, "Stock market forecasting using hidden markov model: a new approach," in *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*. IEEE, 2005, pp. 192–196.

[11] B. Lim, S. O. Arik, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *arXiv preprint arXiv:1912.09363*, 2019.

[12] G. P. Zhang, "Time series forecasting using a hybrid arima and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[13] X. Qiu, Y. Ren, P. N. Suganthan, and G. A. Amaratunga, "Empirical mode decomposition based ensemble deep learning for load demand time series forecasting," *Applied Soft Computing*, vol. 54, pp. 246–255, 2017.

[14] A. Janusz, M. Przyborowski, P. Biczyk, and D. Slezak, "Network Device Workload Prediction: A Data Mining Challenge at Knowledge Pit," in *Proceedings of FedCSIS 2020, Sofia, Bulgaria*, 2020.

[15] S. Nedelkoski, J. S. Cardoso, and O. Kao, "Anomaly detection and classification using distributed tracing and deep learning." in *CCGRID*, 2019, pp. 241–250.

DOI: 10.15439/2020F146
ISSN 2300-5963 ACSIS, Vol. 21

A Social Robot-based Platform towards Automated Diet Tracking

Anastasios Alexiadis*, Andreas Triantafyllidis*, Dimosthenis Elmas,
Giorgos Gerovasilis, Konstantinos Votis, Dimitrios Tzovaras
Centre for Research and Technology Hellas
Information Technologies Institute (CERTH/ITI)
6km Charilaou-Thermi, Thessaloniki, Greece
Email: talex, atriand, dimoselmas, ggerovasilis, kvotis, Dimitrios.Tzovaras@iti.gr

***Abstract*—Diet tracking via self-reports or manual taking of meal photos might be difficult, time-consuming, and discouraging, especially for children, which limits the potential of long-term dietary assessment. We present the design and development of a proof of concept of an automated and unobtrusive system for diet tracking integrating: a) a social robot programmed to automatically capture photos of food and motivate children, b) a deep learning model based on Google Inception V3, applied for the use case of image-based fruit recognition, c) a RESTful microservice architecture deployed to deliver the model outcomes to a platform aiming at childhood obesity prevention. We illustrate the feasibility and virtue of this approach, towards the development of the next-generation computer-assisted systems for automated diet tracking.**

I. Introduction

CHILDHOOD obesity is a major public health challenge which is associated with the risk of developing serious life-threatening diseases [1], [2]. In this context, new computer-assisted technologies can provide useful means to monitor and manage childhood obesity, as well as influence health behaviour and lifestyle at early age [3], [4], [5].

Accurate and long-term diet tracking, is of great significance in childhood obesity prevention [6]. In this direction, computerised dietary assessment through food diaries and self-reports is a common approach [7], [8]. However, major problems in developed computerised tools are that they place a significant burden to the user, suffer from recall bias issues, and rely on technological literacy, often resulting to their early abandonment [9], [10]. Therefore, more unobtrusive and automated approaches are intensively required [11], especially in children, which may have difficulties in articulating their eating patterns.

In this work, we present the design and development of a social robot-based platform for automated food recognition, with the capability to further motivate children to adopt healthy diet habits. The platform employs a deep learning approach for fruit detection, based on camera images automatically captured by a commercially available social robot. The outcomes of the detection are delivered to a platform aiming at childhood obesity prevention, developed within the OCARIoT[1] project, via a service-oriented architecture. Overall, this work adopts and uniquely integrates enabling computer-assisted information and communication technologies, such as social robotics, deep learning and interoperable data communication interfaces, towards demonstrating the feasibility, usefulness and virtue of automated dietary assessment for the prevention of childhood obesity.

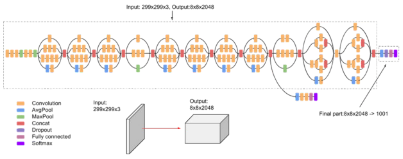


Fig. 1. Inception V3 (Source: https://cloud.google.com/tpu/docs/inception-v3-advanced)

II. Methods

A. Fruit Recognition Model

In a first step, our aim was to train and validate a fruit recognition model, based on the Google Inception V3 model [12], which is pre-trained on the ImageNet database. The Inception V3 model's architecture is shown in Figure 1.

The fruit recognition model classifies images into one of two classes—fruits and non-fruits. We gathered food images from the following sources: ImageNet, Food-101 Data-set[2], UEC Food 256 data-set[3] [13] and a data-set found in Zenodo[4]. The image-sets were split into two classes (fruits and non-fruits) and the pictures in the two classes were balanced. There was a total of 53884 fruit and not-fruit images. The images were cropped into 299x299 pixel chunks and horizontally flipped in a stochastic manner. We split the data-set 80%-20% stochastically.

Inception V3 is a Deep Convolutional Neural Network (ConvNet) designed for classifying images. Google states that the model has been shown to attain greater than 78.1% accuracy on the ImageNet data-set. We extended the model with the addition of the following layers:

- Average Pooling 2D with pool-size 8x8

[1] https://ocariot.eu/

[2] https://data.vision.ee.ethz.ch/cvl/datasets_extra/food-101/

[3] http://foodcam.mobi/dataset256.html

[4] DOI: doi.org/10.5281/zenodo.1310165

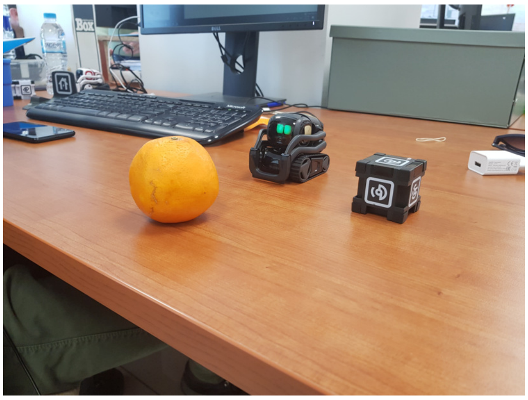
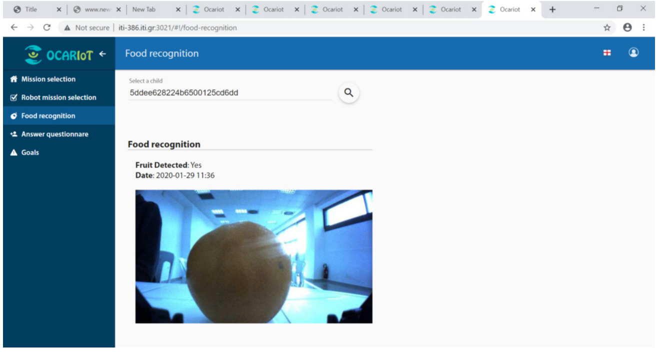


Fig. 2. Social Robot setup



Select a child
5ddee628224b6500125cd6dd

Food recognition

Fruit Detected: Yes
Date: 2020-01-29 11:36

Fig. 3. Platform UI

- Dropout = 0.4
- Flatten
- Dense layer with 1 node, l2 regularization=0.0005, Sigmoid activation function, Xavier uniform initializer

The aim of these additions was to additionally train the model as a fruit classifier. We trained the model using Stochastic Gradient Descent, an optimisation algorithm with good performance over large data-sets. We utilized the following hyper-parameters:

- Initial Learning rate=0.01
- Momentum=0.9
- Decreased learning rate to 0.002 after epoch 15 and to 0.0004 at epoch 28
- Trained to binary cross-entropy loss 0.0018.

We used a multiple-crop (10-crop) strategy for classifying unknown images, where we produced 5 crops for each image to classify (upper left, upper right, lower left, lower right and centre), as well as the flipped versions of these crops. We classified each of these crops, for an image, and kept the one with the maximum value of the dependant variable (thus enabling us to identify a fruit in an image that also includes other unrelated objects).

B. *Integration with a Social Robot*

We further integrated our food recognition model with a commercially available social robot, aiming to apply the model outcomes in real-life and automate the process of food image recognition. The Anki Vector robot was adopted, which is equipped with a 720p (1280X720 pixels) High Definition camera and has additional interesting features, for example, it is easy to carry, it shows an engaging personality, e.g., showing feelings of happiness, sadness, anger, etc., through eye animations and movement with wheels, it can speak, display text/images on its display, and it is programmable via a Software Development Kit (SDK) which we have utilized.

We have developed an application that takes a picture using the Vector's built-in camera and passes it to the model for fruit classification. Upon detection of a fruit, the robot responds with speech, eye animations and movement, providing reward and motivation (e.g., "well done, fruits are good for your health", "congratulations for your choice", etc.). We have used a similar approach in our previous work with social robots targeting childhood obesity [14].

To make the robot automatically move towards the food and capture an image, we implemented a method in which the user is required to place the robot's cube, as a reference object, in front of the food (Figure 2). The software we developed makes the robot to search for the reference cube and positions the robot towards facing the food, resulting overall to an automated food image recognition process with the help of a social robot, without requiring any significant manual effort by the child.

C. *Software Architecture*

Regarding the software architecture, two applications have been developed: An Angular application for the user interface and a Django server that provides an API for the application of the image recognition model and for controlling the robot. Both Angular and Django come equipped with a command line tool that can be used to quickly setup an application. This facilitated the rapid implementation of our prototype.

Regarding the integration of the image recognition model and the social robot, we adopted a REST microservice architecture, a new architectural style that structures an application into a set of small, independently deployable microservices, as opposed to traditional monolithic approaches. The microservice (tagged 'Food Tracking') can store the pictures of the fruit meals in a database in order to create a data-set that could be used to update the image recognition model. When the robot takes a picture of the fruit meal, the image recognition model is applied in order to identify the presence of a healthy food (certain varieties of fruit in this case). When the output of the model is available, the image is sent to the backend software and then the classification result and the image are correlated to the user's id. After the backend has received the image and the result, the platform's "dashboard" application is updated and the user can browse an updated version of their profile.

III. Results

We measured 99.68% accuracy on the validation set comprising a total of 10655 fruit and not-fruit images from the

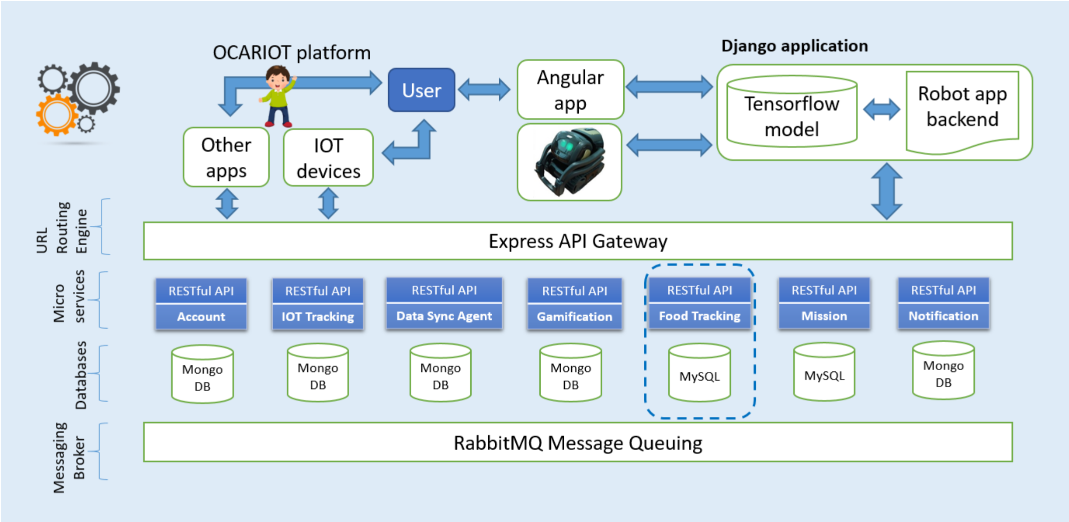


Fig. 4. Fruit Recognizing Social Robot integrated to OCARIoT

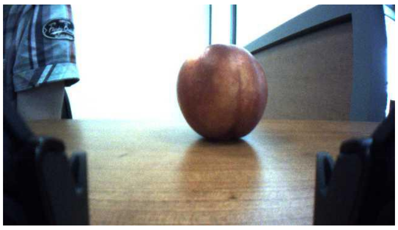


Fig. 5. Recognized as fruit

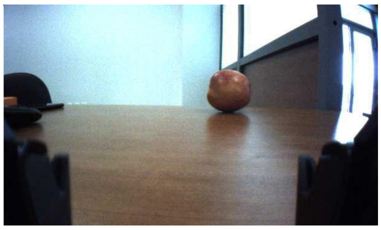


Fig. 6. Not recognized as fruit

combined data-set mentioned in section II-A, which shows that fruit recognition through images is an accurate method, and it could potentially replace tedious self-reports or surveys for fruit consumption.

The Angular and Django applications for controlling the functionality of the social robot and utilising the fruit-recognition model, were integrated within the OCARIoT software platform for childhood obesity prevention. A demonstration of our integrated system is shown in a YouTube video[5]. Figure 4 shows the architecture of the OCARIoT Platform with the social robot application that recognizes fruits integrated. In Figures 5 and 6 we show correct and incorrect classifications on fruits from the integrated fruit-recognition social robot system. We have observed that the distance to the target fruit is the most important indicator for classification accuracy. The robot managed to identify fruits from a distance $\sim$ 20 cm. The apple in Figure 6 was further from the threshold distance.

The Express Gateway, an open source API Gateway which is based on the Express.js framework, is used to redirect the client's requests to the respective microservices through URL routing. The RabbitMQ message broker is adopted in order to manage the message queues maintained between the microservices, thereby facilitating their communication (for example the food tracking microservice can use account information derived from the account microservice through RabbitMQ). RabbitMQ is an open source broker which allows transport-level security, through the use of the Advanced Message Queuing Protocol (AMQP), and fast communication over the Transmission Control Protocol (TCP).

In order for the robot apps to be executed successfully, the robot must be connected (via WiFi) to the same network with the computer executing the robot SDK. Both the robot's application and the Tensorflow model can be accessed by an API built using the Django web framework.

IV. Discussion

We presented the design and development of platform for automated diet tracking based on a programmable social robot, the application of a deep learning model for fruit recognition, and the integration of the model outcomes with a computerised system targeting childhood obesity prevention, through a REST microservice architecture. Our platform constitutes a proof-of-concept, demonstrating the integration of different enabling technologies, towards the development of the next-generation computer-assisted systems for automated diet assessment, which are also capable to motivate individuals to be more engaged with the acquisition of healthy diet habits. Through taking advantage of the programmable robot's in-built capabilities such as a camera, text-to-speech synthesis and

[5] https://www.youtube.com/watch?v=ZSH-WW-rBjY

eye animations, the predictive capabilities of deep learning, as well as an architecture which allows extensibility and interoperability with other software components, our aim was to develop a novel unobtrusive system requiring minimal user interactions.

The system developed could be particularly useful for children, which may face difficulties in self-reporting diet information due to issues related to recall or tedious repetitive user-to-system interactions. Furthermore, child interaction with other computerised systems such as mobile phone devices, would be likely to require parental consent, a good knowledge of using mobile apps, as well as manual taking of photos which may be of low quality. In this context, our system differentiates from previously systems which have been examined [15], [16], [17], and shows the high potential of the application of significantly more engaging and automated systems. In particular, social robot-assisted systems have been demonstrated to be highly attractive to children and useful [18], [19], which has been a motivation for following this approach.

Our future work involves the recognition of different categories of food through social robot-captured images, which would enable a more holistic approach in accurate dietary assessment. Furthermore, the addition of speech recognition in the system would enable dialogue interactions between the robot and the child, which could facilitate motivation of children to acquire healthy diet habits or improve system certainty (e.g., the social robot could ask the child about a meal in the case of robot's low certainty in detecting a specific food in an image, and receive a verbal response). Moreover, the longitudinal collection of all captured diet information would reinforce personalised data analytics, which could indicate behaviours requiring guidance and attention, or revealing potential risks. The deployment of computational models and decision support systems has shown promise in this direction [20]. Finally, a study with children should be conducted, enabling the evaluation of the platform in real-world scenarios, e.g. at home, or within educational sessions at school settings. Furthermore this will allow us to measure the real-world accuracy of the model on a set of images captured by the social robot in practical usage scenarios.

In conclusion, we regard social robots as valuable agents that can support humans in engagement with healthy behaviours. To this end, the work presented in this paper is a step towards automated dietary support of children by social robots.

Acknowledgements

The study was supported by the European Union‘s HORIZON 2020 Programme (2014-2020), under ID no 777082, and from the Brazilian Ministry of Science, Technology and Innovation through Rede Nacional de Ensino e Pesquisa (RNP) under OCARIoT

References

[1] M. Shields, M. S. Tremblay, S. Connor Gorber, and I. Janssen, “Abdominal obesity and cardiovascular disease risk factors within body mass index categories.,” Heal. reports, vol. 23, no. 2, pp. 7–15, Jun. 2012.

[2] I. Vucenik and J. P. Stains, “Obesity and cancer risk: evidence, mechanisms, and recommendations,” Ann. N. Y. Acad. Sci., vol. 1271, no. 1, pp. 37–43, Oct. 2012, doi: 10.1111/j.1749-6632.2012.06750.x.

[3] E. B. Tate et al., “mHealth approaches to child obesity prevention: successes, unique challenges, and next directions.,” Transl. Behav. Med., vol. 3, no. 4, pp. 406–415, Dec. 2013, doi: 10.1007/s13142-013-0222-3.

[4] A. J. Smith, A. Skow, J. Bodurtha, and S. Kinra, “Health Information Technology in Screening and Treatment of Child Obesity: A Systematic Review,” Pediatrics, vol. 131, no. 3, pp. e894–e902, Mar. 2013, doi: 10.1542/peds.2012-2011.

[5] P. W. C. Lau, E. Y. Lau, D. P. Wong, and L. Ransdell, “A Systematic review of information and communication technology-based interventions for promoting physical activity behavior change in children and adolescents,” J. Med. Internet Res., vol. 13, no. 3, 2011, doi: 10.2196/jmir.1533.

[6] E. P. Abril, “Tracking Myself: Assessing the Contribution of Mobile Technologies for Self-Trackers of Weight, Diet, or Exercise,” J. Health Commun., vol. 21, no. 6, pp. 638–646, Jun. 2016, doi: 10.1080/10810730.2016.1153756.

[7] A. G. Arens-Volland, L. Spassova, and T. Bohn, “Promising approaches of computer-supported dietary assessment and management-Current research status and available applications.,” Int. J. Med. Inform., vol. 84, no. 12, pp. 997–1008, Dec. 2015, doi: 10.1016/j.ijmedinf.2015.08.006.

[8] A. H. Andrew, G. Borriello, and J. Fogarty, “Simplifying mobile phone food diaries,” in Proceedings of the 2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops, PervasiveHealth 2013, 2013, pp. 260–263, doi: 10.4108/icst.pervasivehealth.2013.252101.

[9] S. M. Schembre et al., “Mobile Ecological Momentary Diet Assessment Methods for Behavioral Research: Systematic Review.,” JMIR mHealth uHealth, vol. 6, no. 11, p. e11170, Nov. 2018, doi: 10.2196/11170.

[10] D. Lupton, “‘I Just Want It to Be Done, Done, Done!’ Food Tracking Apps, Affects, and Agential Capacities,” Multimodal Technol. Interact., vol. 2, no. 2, p. 29, May 2018, doi: 10.3390/mti2020029.

[11] T. Prioleau, E. Moore Ii, and M. Ghovanloo, “Unobtrusive and Wearable Systems for Automatic Dietary Monitoring.,” IEEE Trans. Biomed. Eng., vol. 64, no. 9, pp. 2075–2089, Sep. 2017, doi: 10.1109/TBME.2016.2631246.

[12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* Las Vegas, NV, 2016, pp. 2818-2826, doi: 10.1109/CVPR.2016.308.

[13] Kawano Y., Yanai K. (2015) Automatic Expansion of a Food Image Dataset Leveraging Existing Categories with Domain Adaptation. In: Agapito L., Bronstein M., Rother C. (eds) Computer Vision - ECCV 2014 Workshops. ECCV 2014. Lecture Notes in Computer Science, vol 8927. Springer, Cham

[14] A. Triantafyllidis, A. Alexiadis, D. Elmas, K. Votis, D. Tzovaras, A social robot-based platform for prevention of childhood obesity, in: Proc. - 2019 IEEE 19th Int. Conf. Bioinforma. Bioeng. BIBE 2019, Institute of Electrical and Electronics Engineers Inc., 2019: pp. 914–917. doi:10.1109/BIBE.2019.00171.

[15] A. Myers et al., “Im2Calories: towards an automated mobile vision food diary,” 2015, 10.1109/ICCV.2015.146.

[16] S. Mezgec and B. Koroušić Seljak, “NutriNet: A Deep Learning Food and Drink Image Recognition System for Dietary Assessment,” Nutrients, vol. 9, no. 7, p. 657, Jun. 2017, doi: 10.3390/nu9070657.

[17] Y. Hswen, V. Murti, A. Vormawor, R. Bhattacharjee, and J. Naslund, “Virtual avatars, gaming, and social media: Designing a mobile health app to help children choose healthier food options,” J. Mob. Technol. Med., vol. 2, no. 2, p. 8, 2013, doi: 10.7309/jmtm.2.2.3.

[18] O. Mubin, C. J. Stevens, S. Shahid, A. Al Mahmud, and J.-J. Dong, “A Review of the Applicability of Robots in Education,” Technol. Educ. Learn., vol. 1, no. 1, pp. 1–7, 2013, doi: 10.2316/Journal.209.2013.1.209-0015.

[19] O. A. Blanson Henkemans et al., “Design and evaluation of a personal robot playing a self-management education game with children with diabetes type 1.” 01-Jan-2017, doi: 10.1016/j.ijhcs.2017.06.001.

[20] A. Triantafyllidis et al., “Computerized decision support and machine learning applications for the prevention and treatment of childhood obesity: A systematic review of the literature,” Artif. Intell. Med., vol. 104, p. 101844, Apr. 2020, doi: 10.1016/j.artmed.2020.101844.

DOI: 10.15439/2020F211
ISSN 2300-5963 ACSIS, Vol. 21

Short-term air pollution forecasting based on environmental factors and deep learning models

Mirche Arsov, Eftim Zdravevski,
Petre Lameski,
Ss Cyril and Methodius University,
Faculty of Computer Science and Engineering,
Skopje, North Macedonia
Email: mirche.arsov@gmail.com,
eftim.zdravevski@finki.ukim.mk,
petre.lameski@finki.ukim.mk

Roberto Corizzo,
Department of Computer Science
American University, Washington DC, USA
Email: rcorizzo@american.edu

Nikola Koteli, Kosta Mitreski,
Vladimir Trajkovik
Ss Cyril and Methodius University,
Faculty of Computer Science and Engineering,
Skopje, North Macedonia
Email: nikola.koteli@hotmail.com,
kosta.mitreski@finki.ukim.mk,
vladimir.trajkovik@finki.ukim.mk

***Abstract*—The effects of air pollution on people, the environment, and the global economy are profound - and often under-recognized. Air pollution is becoming a global problem. Urban areas have dense populations and a high concentration of emission sources: vehicles, buildings, industrial activity, waste, and wastewater. Tackling air pollution is an immediate problem in developing countries, such as North Macedonia, especially in larger urban areas. This paper exploits Recurrent Neural Network (RNN) models with Long Short-Term Memory units to predict the level of PM10 particles in the near future (+3 hours), measured with sensors deployed in different locations in the city of Skopje. Historical air quality measurements data were used to train the models. In order to capture the relation of air pollution and seasonal changes in meteorological conditions, we introduced temperature and humidity data to improve the performance. The accuracy of the models is compared to PM10 concentration forecast using an Autoregressive Integrated Moving Average (ARIMA) model. The obtained results show that specific deep learning models consistently outperform the ARIMA model, particularly when combining meteorological and air pollution historical data. The benefit of the proposed models for reliable predictions of only 0.01 MSE could facilitate preemptive actions to reduce air pollution, such as temporarily shutting main polluters, or issuing warnings so the citizens can go to a safer environment and minimize exposure.**

***Index Terms*—RNN, LSTM, deep learning, air pollution**

I. Introduction

AIR pollutants exert a wide range of impacts on biological and socio-economic systems. Their effects on human health are of great interest. In particular, PM2.5 and PM10 (Particulate Matter) have been proven to have a significant impact on human respiratory efficiency. A number of studies have shown that the increase of respiratory diseases is correlated by a high concentration of air pollutants [1]. [2] represents a useful review of the emerging challenges and requirements for understanding adverse health outcomes from ambient particles. Consequently, it has become an important task to accurately track and analyze ambient air pollution in order to adjust public policies and health protection measures. The ability to predict exceeding of critical air quality thresholds is of particular importance. The potential for alert management systems that will provide warning communication to authorities and the population of health and environmental risks is high. Such systems have already been developed and deployed [3]. Studies such as [4], have shown that the data of ambient air quality can be modelled as stochastic time series, thereby making it possible to make a short-term forecast based on historical data. There are successful approaches relying on successful forecasting models over large multi-sensor data sets, based on sliding-window-based feature extraction and feature subset ensemble selection [5]–[9]. The latter approaches also show that it is feasible to use short-term predictions of dangerous concentrations in coal mines to reduce the workload, so preventing to reach the dangerous thresholds. The air pollution forecast in cities can be employed in a similar manner, such as temporarily shutting main polluters, or issuing warnings so the citizens can go to a safer environment and minimize exposure.

Long short-term memory (LSTM) [10] is an artificial recurrent neural network (RNN) [11] architecture used in the field of deep learning. Due to their chain-like nature, LSTMs are considered to be the typical architecture of neural networks to be used with sequences and lists. LSTM networks have already been used for time series multistep forecasting in multiple studies [12] [13] [14] [15] [16]. In [17] a similar approach is described, where convolutional neural networks are combined with LSTM to classify PM10 levels. In [18], an approach for air pollution forecasting using RNN with LSTM is presented. Alternative studies in the literature exploit feature extraction as a pre-processing step for the predictive task [19] [20] [21] [22] [23].

The widespread adoption of LSTM across different domains shows the effectiveness and reliability of this model in multistep forecasting task. The reason for their effectiveness is the ability to extract time-variant dependencies and correlations that are inherently present in real-life scenarios, and exploit them to predict future time steps. Differently than ARIMA models, which are autoregressive and capable to analyze exclusively univariate time series, LSTM models can exploit multiple time series in a combined manner. Potentially, leveraging the existing correlations between them can lead to obtain more accurate predictions.

In this paper, we performed experiments with air quality measurements data from the area of Skopje, North Macedonia. We used PM10 level measurements of the pollution combined with meteorological parameters to predict the PM10 level at point +3 hours in the future. The main contribution of this paper is that it combines different data sources to perform forecasting and compares the results to predictions when only air quality data is used. Our approach was to train the models with a data set from a single sensor, then gradually increase the number of air quality sensors used. A graphical representation of the workflow is shown in Figure 1. The results were then compared with the performance of the models trained using air quality and meteorological sensor data combined. Additionally, we used the data to examine the performance of different RNN architectures. For this purpose, we used the Keras framework, a high-level neural networks API capable of running on top of Tensorflow.

II. Methods

A. Dataset

The dataset consists of air quality sensor measurements from sensors deployed to different locations in the city of Skopje. Variety of parameters are monitored by the sensors, including PM10 and PM2.5 particles, as well as the presence of NO2, CO, O3, and SO2. Measurements are done in intervals of one hour. This dataset was also enriched with meteorological parameters, namely temperature and atmospheric pressure, measured at the Skopje-Petrovec meteorological station. For the purpose of this research, 12 consecutive measurements from the air pollution measurement points and from the meteorological station are used. In some of the models, we used a set of 24 consecutive measurements, but no considerable improvement was observed.

The dataset in this form has not been studied and published before. However, a subset of the data has been used in a previous study. In [17] a similar approach is described, where subset of the data is used to classify future values using a combination of LSTM and convolutional neural networks. Repository with the source code used, as well as the preprocessed dataset, is available at https://gitlab.com/magix.ai/air-pollution-skopje.

Fig. 2 shows the seasonality and trend in the data set. It is clearly noticeable that disturbances and irregularities are present in the air quality sensor data. Due to these reasons, to train the recurrent neural network models, we used data in the range from December 2011 to December 2019. In order to model possible malfunctions in the sensors, we introduced a Dropout layer in some of the architectures.

The used measurements are listed bellow, grouped by location:

- Municipality of Karposh, North Macedonia
  - PM10 concentration
- Municipality of Centar, North Macedonia
  - Measurement station Centar - PM10 concentration
  - Measurement station Rektorat - PM10 concentration
- Municipality of Miladinovci, North Macedonia
  - PM10 concentration
- Municipality of Petrovec, North Macedonia
  - Temperature (in Degrees Celsius)
  - Atmospheric Pressure at station level

The sampling frequency of the meteorological parameters differs from the one used in the air quality sensors. A pre-processing phase was needed to fit the data set for training and validation purposes. The pre-processing consists of the following steps:

- Missing data interpolation
- Min-Max normalization
- 12 samples data window preparation

The data was divided into train, validation and test data sets. For training, we used data in the time interval 01.12.2011 - 31.12.2019. This dataset consists of 70129 samples. Validation samples were taken dynamically as 1 per cent from the training data points (709 samples). Before the training process, we used a smaller two-year subset of the data for hyperparameter optimization. Data points for optimization were taken from the interval from 01.08.2014 to 01.08.2016 (17534 samples). Hyper-parameter tuning was validated using a small two-months data set in the time frame 01.11.2016 - 31.12.2016 (1430 samples).

For testing the performance of the different architectures, a test data set was used. This data set consists of the data points in January 2020.

B. Baseline model: ARIMA

Among many available methods for time series regression, one of the most popular and broadly used are Autoregressive integrated moving average (ARIMA) model [24]. Results obtained in this study confirmed that the ARIMA has a strong potential for short-term spot prediction. ARIMA form a class of time series models that are widely applicable in the field of time series forecasting. In the ARIMA model, the future value of a variable is a linear combination of past values and errors after removing the trend – by differencing.

C. Deep learning models architecture

In this paper, we wanted to compare several different architectures and see how they perform in comparison to the ARIMA model. We used LSTM, SimpleRNN and GRU layers. In some of the architectures, we added a dropout layer to mitigate temporary failures of some sensors.

RNN mainly deals with the processing of sequence data, such as text, speech, and time series. This type of data exists in an orderly relationship with each other; each piece of data is associated with the previous piece. Another example is climate data, where, for example, the temperature of a day is related to the temperature of the previous day. Therefore, we can form many sets of sequences from the data using time from a set of continuous data, and the correlation between sequences can be observed from multiple sets of sequences.

Our approach was to build a simple model using LSTM and Dense layers and then gradually increase the complexity of the

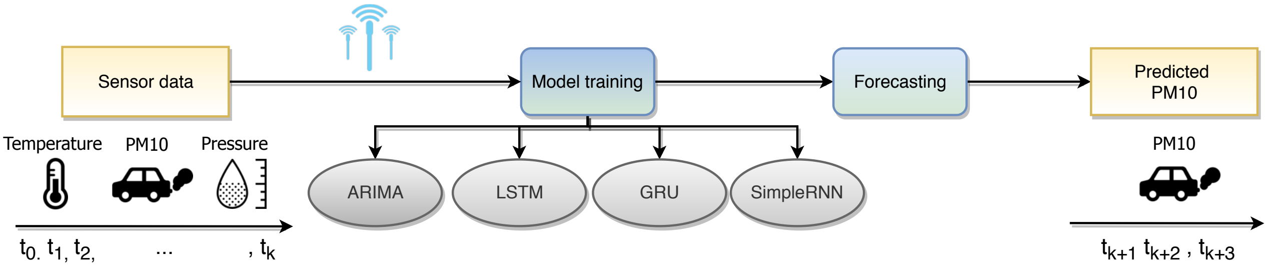


Fig. 1: Graphical representation of the proposed method for pollution forecasting.

architecture. An overview of the models is given in Table I-II. We started with the simplest architecture, one LSTM layers with optimized number of units and one Dense layer. Then, we trained the network using a univariate data set, values from one air quality sensor, and we validated the ability to make short-term predictions, +3 hours in the future.

This simple architecture performed slightly better than the ARIMA model. 3 shows the loss function for the training and validation phases while 4 show the performance of this model compared to the performance of the ARIMA model.

In order to improve the performance, we added data from a second sensor to treat the problem as a multivariate series and compared the performance of the architecture. Then we increased the data set to include data from 4 sensors on different locations (Table I, approach number 3 and 4). This caused a small decrease in the performance of the LSTM model. One of the main ideas of this paper was to investigate the influence of meteorological data. Therefore as a final test, we added temperature and air pressure parameters to the training data set (Table I, approach number 5). The result was an increase in accuracy and performance. Figure 4 shows the performance compared to the ARIMA model in the first week of the test data. This architecture showed the best performance when we compared the models using MSE as a performance metric. The simple LSTM model seems to outperform all other models described in this paper.

As a second approach in this paper, we increased the complexity of the architecture and the number of hidden layers. A second layer of optimized LSTM was added with a Dropout layer in between (Table I-II, approach number 6). This model showed a slightly decreased accuracy.

As a next step, we introduced a SimpleRNN layer in two variants. In [25], it is shown that RNN can be used for time series forecasting. SimpleRNN is a fully-connected RNN layer where the output is to be fed back to the input. As a first experiment, we replaced the first LSTM layer with a SimpleRNN, and in the second the RNN layer was added as an input to the first LSTM followed by a Dropout layer (Table I approach number 8 and 9). The latter approach exhibited a better performance than the first.

As an additional attempt to improve the results, the data set was extended to 24 hours history data window (Table I approach number 10). This did not bring any significant improvement. Gated Recurrent Unit or GRU [26] modifies the LSTM by fusing the forget and input gates into an update gate. Additionally, the cell states and hidden states are merged. The resulting model is simpler than standard LSTM models, and has been growing increasingly popular in the past few years. Due to sensor failures, the data set, as most of the real-life time series data, is characterized by a variety of missing values. It has been noted that missing values and their missing patterns are often correlated with the target labels. There are studies [27] that examine architectures based on GRU to time series data analysis. An experiment with a GRU layer was made. We extended the RNN architecture with a GRU layer followed by a SimpleRNN + LSTM, and a Dense layer as an output. This architecture showed a decreased performance in comparison to ARIMA and the previous architectures (Table I, approach number 7). During training and validation, the model showed improvements, such as faster learning (steeper decline in the loss function in the first couple of epochs). For the test data set, the accuracy decreased, which was an indicator that this architecture was overfitting to the training data set.

Due to the small number of features, we expected that additional layers could only increase the probability of overfitting on the data, although further research is experimentation is necessary to prove this.

For this particular experiment, we use mean squared error loss function, and for the model optimization, we used the Adam optimizer [28]. The implementation is done with Keras [29].

D. Parameter tuning

We used parameter tuning in order to obtain the best predictive model. For hyperparameter optimization, a smaller subset of the training data was used. Data points were taken in the interval from 01.08.2014 to 01.08.2016 (17534 samples). Hyper-parameter tuning was validated using a two-months validation data set in the time frame 01.11.2016 - 31.12.2016 (1430 samples). Table III presents the parameters that are tuned with the ranging values. Optimization was done using the Keras-Tuner library[1].

The following parameters were tuned in order to obtain the best architecture:

- *Dropout* - Deep neural networks with a large number of parameters can be powerful tools. However, overfitting

[1] https://keras-team.github.io/keras-tuner/

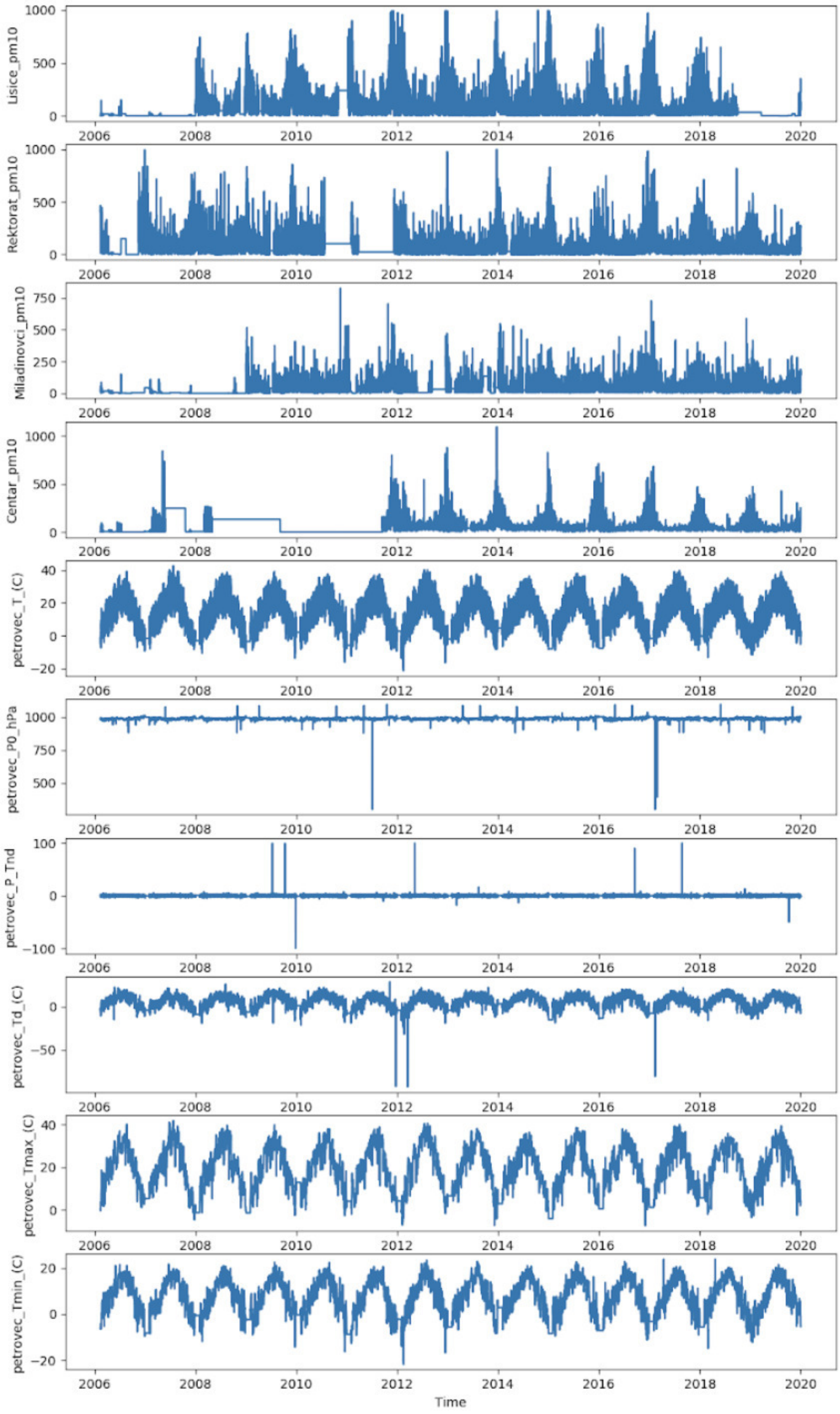


Fig. 2: Exploration of seasonality and trend in the dataset

TABLE I: Summary of evaluated approaches with achieved average forecasting performance in terms of Mean Square Error (MSE), Root Mean Square Error (RMSE), and percentage of improvement in terms of reduction in RMSE with respect to ARIMA.

#	Input	Data	Architecture	MSE	RMSE	% Impr.
1	1 x 12	PM10	LSTM + Dense	0.0140	0.1185	6.02
2	1 x 12	PM10	ARIMA (12)	0.0159	0.1261	/
3	2 x 12	2 x PM10	LSTM + Dense	0.0143	0.1198	4.90
4	4 x 12	4 x PM10	LSTM + Dense	0.0124	0.1114	11.60
5	6 x 12	4 x PM10 + Temp. + Pressure	LSTM + Dense	0.0109	0.1043	17.28
6	6 x 12	4 x PM10 + Temp. + Pressure	LSTM + Dropout + LSTM + Dense	0.0115	0.1072	14.90
7	6 x 12	4 x PM10 + Temp. + Pressure	GRU + SimpleRNN + LSTM + Dense	0.0428	0.2069	-6.40
8	6 x 12	4 x PM10 + Temp. + Pressure	SimpleRNN + LSTM + Dense	0.0150	0.1224	2.93
9	6 x 12	4 x PM10 + Temp. + Pressure	SimpleRNN + LSTM + Dropout + LSTM + Dense	0.0125	0.1118	11.34
10	6 x 24	4 x PM10 + Temp. + Pressure	SimpleRNN + LSTM + Dropout + LSTM + Dense	0.0127	0.1126	10.70

TABLE II: Summary of evaluated approaches with different configurations in terms of number of units in the LSTM layer (U), Learning Rate (LR), Dropout rates (D).

#	Architecture	Parameters optimized
1	LSTM + Dense	U: 2-128 + LR [0.01, 01]
2	ARIMA (12)	None
3	LSTM + Dense	U: 2-124 + Learning rates [0.01, 01]
4	LSTM + Dense	U: 2-124 + LR [0.01, 01]
5	LSTM + Dense	U: 2-124 + LR [0.01, 01]
6	LSTM + Dropout + LSTM + Dense	LSTM (2-24) + D [0.3, 0.2, 0.1] + LSTM (2-124) + LR [0.01, 01]
7	GRU + SimpleRNN + LSTM + Dense	GRU (12-256) + RNN (1-128) + LSTM (2-124) + LR [0.01, 01]
8	SimpleRNN + LSTM + Dense	RNN (1-128) + LSTM (2-124) + LR [0.01, 01]
9	SimpleRNN + LSTM + Dropout + LSTM + Dense	RNN (1-128) + LSTM (2-24) + D [0.3, 0.2, 0.1] + LSTM (2-124) + LR [0.01, 01]
10	SimpleRNN + LSTM + Dropout + LSTM + Dense	RNN (1-128) + LSTM (2-24) + D [0.3, 0.2, 0.1] + LSTM (2-124) + LR [0.01, 01]

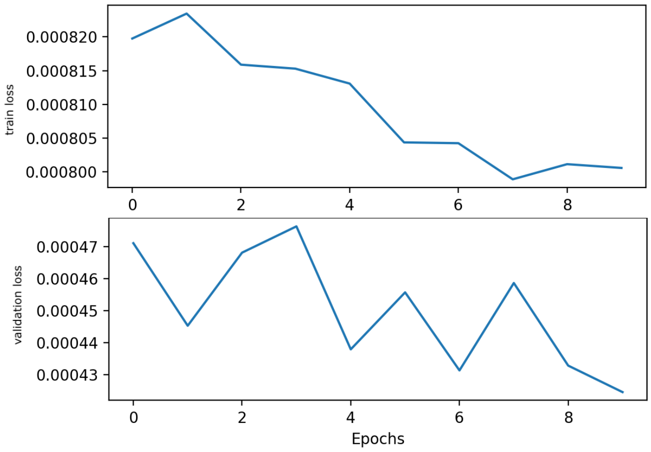


Fig. 3: Train and validation MSE of the single layer LSTM model tested

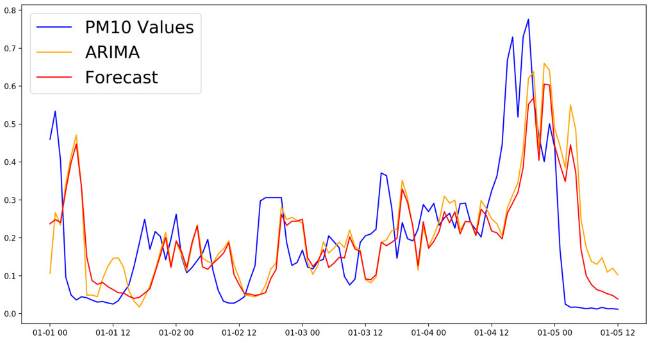


Fig. 4: Performance comparison to ARIMA model in the first week of the test data set

can be a problem in such networks. This often happens when neural nets are trained on relatively small datasets. The lack of control over the learning process often leads to cases where the neural network can not generalize and make forecasts for new data. Dropout is a technique for addressing this problem. The idea is to randomly drop units from the neural network in the training phase in order to prevent units from co-adapting too much.

- *Learning rate* - The learning rate is a hyperparameter that controls how much to change the model in response to the estimated error each time the model weights are updated. Choosing the learning rate is challenging as a value too small may result in a lengthy training process that could get stuck, whereas a value too large may result in learning a sub-optimal set of weights too fast or an unstable training process. The learning rate controls how

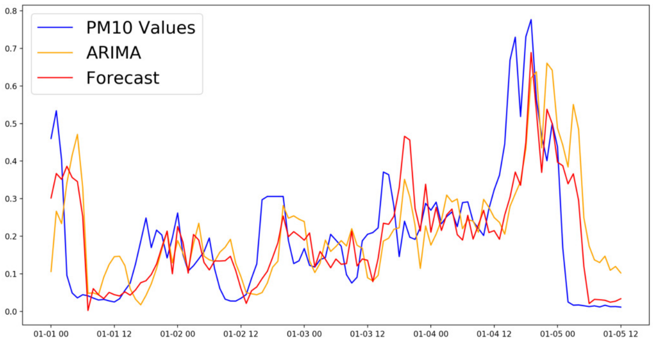


Fig. 5: Performance comparison to ARIMA model in the first week of the test data set. Training data includes meteorological parameters

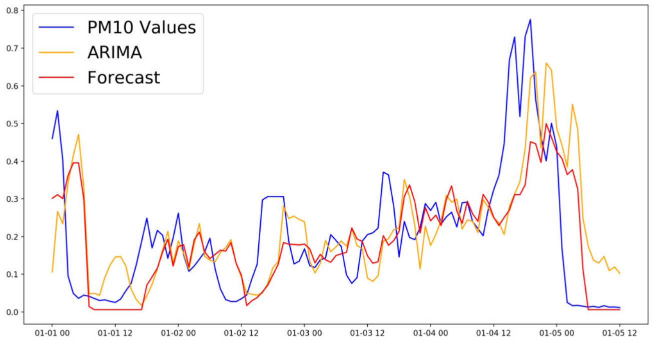


Fig. 6: Performance comparison of the SimpleRNN+LSTM architecture to the ARIMA model in the first week of the test data set. Training data includes meteorological parameters

quickly the model is adapted to the problem.

- *LSTM layer units* - the number of LSTM cells in the layer is a parameter that we used in our model optimization. The number of units determines the dimensionality of the output space.
- *RNN units* - the number of RNN cells in the layer. By default, the output of an RNN layer contains a single vector per sample. This vector is the RNN cell output corresponding to the last timestep, containing information about the entire input sequence. The units parameter determines the shape of this output. A RNN layer can also return the entire sequence of outputs for each sample (one vector per timestep per sample).
- *GRU units* - parameter that determines the dimensionality of the output vector.

TABLE III: Parameters used for tuning the neural network

Parameter name	Min Value	Max Value	Step
Learning rate	32	128	2
Dropout rate	0.1	0.3	0.1
LSTM 1 layer units	2	128	2
LSTM 2 layer units	2	124	4
RNN layer units	1	128	4
GRU layer units	12	256	4

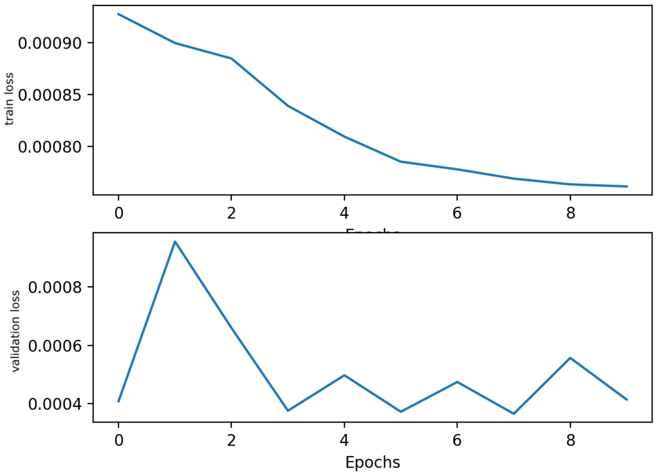


Fig. 7: Train and validation MSE of the RNN+LSTM model with 24 hours training data samples

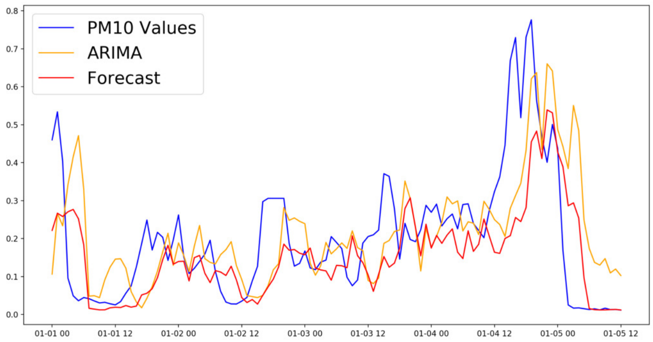


Fig. 8: Performance comparison of the RNN+LSTM architecture trained with extended 24 hours data sequence

We performed a grid search through the parameter space, trying every possible combination of the parameters.

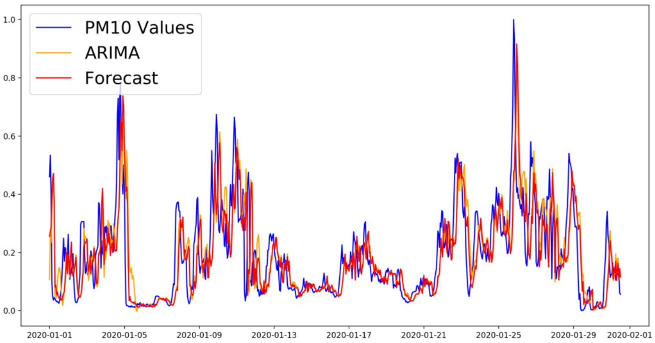


Fig. 9: Performance of the LSTM model with data from two sensors

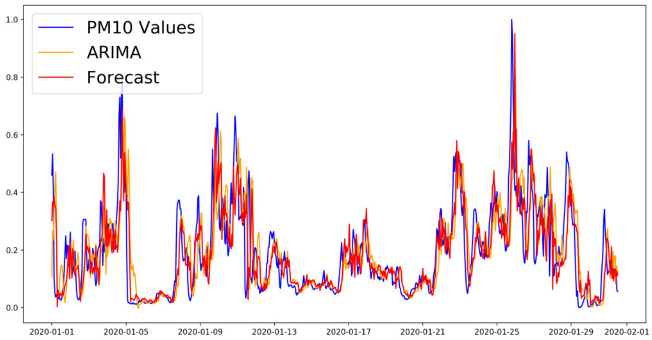


Fig. 10: Performance of the LSTM model with data extended with meteorological parameters

III. Conclusion

All architectures, except for the model with GRU layer, outperformed the ARIMA model in forecasting near-term future values (+3 hours). Models based on simple LSTM architecture exhibited the best results, leading to an improvement of up to 17.28% in terms of RMSE reduction with respect to ARIMA. Table I shows a summary of the results obtained in the experiments. More complex architectures can lead to overfitting. Further research is needed to solve this problem. We concluded that the combination of meteorological and air pollution measurements data improves the performance of LSTM and RNN+LSTM neural networks as a near-term predictive models over the performance of the same architectures used with air quality data alone. Additionally, the results show that the combination of meteorological and air pollution measurements data with LSTM and RNN + LSTM neural networks leads to good short-term predictive models.

It is important to note that our approach can be used to analyze different pollution datasets, as well as time series data in other domains. In fact, the possibility to query weather stations through web services allows to easily complement on-site sensor measurements of pollutants with historical and predicted weather data.

Further experiments are needed to examine the existing models by introducing additional data of air quality as well as of meteorological nature. Other experiments should also be performed to examine the model's accuracy for extended near-term future forecasts, for example, +6 and +9 hours in the future.

Acknowledgment

The work presented in this paper was partially funded by the Ss. Cyril and Methodius University in Skopje, Faculty of Computer Science and Engineering. We also acknowledge the support of NVIDIA through a donation of a Titan V GPU.

References

[1] A. S. Whittemore, "Air pollution and respiratory disease," *Annual review of public health*, vol. 2, no. 1, pp. 397–429, 1981.

[2] M. R. Heal, P. Kumar, and R. M. Harrison, "Particles, air quality, policy and health," *Chemical Society Reviews*, vol. 41, no. 19, pp. 6606–6630, 2012.

[3] R. Arasa, M. Picanyol, and J. Solé, "Analysis of the integrated environmental and meteorological forecasting and alert system (siam) for air quality applications over different regions of the iberian peninsula," in *Proceedings of HARMO15 Congress. Madrid. http://www.harmo.org/Conferences/Proceedings/_Madrid/publishedSections/H15-70.pdf*, 2013.

[4] G. Fronza and P. Melli, *Mathematical Models for Planning and Controlling Air Quality: Proceedings of an October 1979 IIASA Workshop*. Elsevier, 2014.

[5] D. Slezak, M. Grzegorowski, A. Janusz, M. Kozielski, S. H. Nguyen, M. Sikora, S. Stawicki, and L. Wrobel, "A framework for learning and embedding multi-sensor forecasting models into a decision support system: A case study of methane concentration in coal mines," *Information Sciences*, vol. 451-452, pp. 112 – 133, 2018.

[6] A. Janusz, M. Grzegorowski, M. Michalak, L. Wrobel, M. Sikora, and D. Slezak, "Predicting seismic events in coal mines based on underground sensor measurements," *Engineering Applications of Artificial Intelligence*, vol. 64, pp. 83–94, 2017.

[7] E. Zdravevski, P. Lameski, R. Mingov, A. Kulakov, and D. Gjorgjevikj, "Robust histogram-based feature engineering of time series data," in *2015 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2015, pp. 381–388.

[8] A. Janusz, D. Slezak, M. Sikora, and L. Wrobel, "Predicting dangerous seismic events: Aaia'16 data mining challenge," in *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2016, pp. 205–211.

[9] E. Zdravevski, P. Lameski, and A. Kulakov, "Automatic feature engineering for prediction of dangerous seismic activities in coal mines," in *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2016, pp. 245–248.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] A. C. Tsoi and A. Back, "Discrete time recurrent neural network architectures: A unifying review," *Neurocomputing*, vol. 15, no. 3-4, pp. 183–223, 1997.

[12] L. Yunpeng, H. Di, B. Junpeng, and Q. Yong, "Multi-step ahead time series forecasting for different data patterns based on lstm recurrent neural network," in *2017 14th Web Information Systems and Applications Conference (WISA)*. IEEE, 2017, pp. 305–310.

[13] M. Ceci, R. Corizzo, D. Malerba, and A. Rashkovska, "Spatial autocorrelation and entropy for renewable energy forecasting," *Data Mining and Knowledge Discovery*, vol. 33, no. 3, pp. 698–729, 2019.

[14] A. Tokgöz and G. Ünal, "A rnn based time series approach for forecasting turkish electricity load," in *2018 26th Signal Processing and Communications Applications Conference (SIU)*. IEEE, 2018, pp. 1–4.

[15] B. B. Sahoo, R. Jha, A. Singh, and D. Kumar, "Long short-term memory (lstm) recurrent neural network for low-flow hydrological time series forecasting," *Acta Geophysica*, vol. 67, no. 5, pp. 1471–1481, 2019.

[16] R. Corizzo, M. Ceci, H. Fanaee-T, and J. Gama, "Multi-aspect renewable energy forecasting," *Information Sciences*, 2020.

[17] V. Stojov, N. Koteli, P. Lameski, and E. Zdravevski, "Application of machine learning and time-series analysis for air pollution prediction," in *CIIT 2018*, 2018.

[18] Y.-T. Tsai, Y.-R. Zeng, and Y.-S. Chang, "Air pollution forecasting using rnn with lstm," in *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*. IEEE, 2018, pp. 1074–1079.

[19] R. Corizzo, M. Ceci, E. Zdravevski, and N. Japkowicz, "Scalable autoencoders for gravitational waves detection from time series data," *Expert Systems with Applications*, vol. 151, p. 113378, 2020.

[20] A. Zhao, L. Qi, J. Dong, and H. Yu, "Dual channel lstm based multi-feature extraction in gait for diagnosis of neurodegenerative diseases," *Knowledge-Based Systems*, vol. 145, pp. 91–97, 2018.

[21] B. Petrovska, E. Zdravevski, P. Lameski, R. Corizzo, I. Štajduhar, and J. Lerga, "Deep learning for feature extraction in remote sensing: A case-study of aerial scene classification," *Sensors*, vol. 20, no. 14, p. 3906, 2020.

[22] B. Petrovska, T. Atanasova-Pacemska, R. Corizzo, P. Mignone, P. Lameski, and E. Zdravevski, "Aerial scene classification through fine-tuning with adaptive learning rates and label smoothing," *Applied Sciences*, 2020.

[23] S. Ryan, R. Corizzo, I. Kiringa, and N. Japkowicz, "Pattern and anomaly localization in complex and dynamic data," in *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, 2019, pp. 1756–1763.

[24] U. Kumar and V. Jain, "Arima forecasting of ambient air pollutants (o 3, no, no 2 and co)," *Stochastic Environmental Research and Risk Assessment*, vol. 24, no. 5, pp. 751–760, 2010.

[25] J. Zhang and K. Man, "Time series prediction using rnn in multi-dimension embedding phase space," in *SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 98CH36218)*, vol. 2. IEEE, 1998, pp. 1868–1873.

[26] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[27] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, "Recurrent neural networks for multivariate time series with missing values," *Scientific reports*, vol. 8, no. 1, pp. 1–12, 2018.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[29] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

A simple crime hotspot forecasting algorithm

Robert Bogucki
University of Warsaw
ul. Banacha 2, 02-097 Warsaw, Poland
and
deepsense.ai,
Al. Jerozolimskie 162A, 02-342 Warsaw, Poland
Email: robert@deepsense.ai

Jan Kanty Milczek, Patryk Miziuła
deepsense.ai,
Al. Jerozolimskie 162A, 02-342 Warsaw, Poland
Email: {jan.milczek, patryk.miziula}@deepsense.ai

***Abstract*—Crime hotspot forecasting is an important part of crime prevention and reducing the delay between a 911 call and the physical intervention. Current developments in the field focus on enriching the historical data and sophisticated point process analysis methods with a fixed grid. In the paper we present a simple spatio-temporal point process allowing one to perform exhaustive (literal) grid searches. We then show that this approach can compete with more complex methods, as evidenced by the results on data collected by the Portland Bureau of Police. Finally, we discuss the advantages and potential implications of the new method.**

I. Introduction

SPATIO-TEMPORAL crime forecasting is a field that grabs the attention of both scientists and practitioners. Many academic researchers have published results based on time series analysis ([1]), regression methods ([2], [3], [4]), kernel density estimation ([5], [6], [7], [8], [9], [10]) or self-exciting point processes ([11], [12], [13], [14], [15], [16], [17]). Moreover, the US Government appreciates the impact predictive policing has on society (see [18]).

In a typical crime prediction task, the forecast area is fixed and divided into small sub-regions, called cells. The cells are then scored separately over a given future time window. The ones with the highest rate are chosen as the most dangerous areas and called hotspots. In this article we present a point of view for hotspot forecasting that differs from those which can be found in the literature. We emphasise the simplicity and efficiency of our algorithm for a fixed grid to get an opportunity to check as many grids as possible. We place those attributes over sophisticated methods, with state-of-the-art results in practice. Our models won eight categories of the Real-Time Crime Forecasting Challenge conducted by the National Institute of Justice ([19]).

The rest of the paper is organized as follows. In section II we explain our approach in detail. Section III contains a comprehensive description of case study of our method – the Real-Time Crime Forecasting Challenge. Further comments and summary are placed in section IV.

II. The model

A. The choice of grid

There is a vast literature available about crime forecasting for a given grid of cells based on past crimes committed (see references in the Introduction). In such a setup, more or less sophisticated methods are applied to predict which fixed parts of the investigated region will experience the highest future rate of crime. Clearly, changing the grid changes the entire task as well and may lead to completely different predictions with different levels of effectiveness in the real world. However, as far as we know, whenever the cell division is not imposed in advance, searching for a good grid is in practice reduced to *grid search*, *random search* (see [20]) or another primitive method of walking among parametrizations of possible tessellations. The reason there is a lack of 'smarter' grid choosing techniques may be that spatial distributions of crimes committed in urban areas are 'weird': they contain atoms with very high crime rates (related to, for example, large-area stores or shelters for the homeless). Therefore, using the same data-driven algorithm for even very similar grids can cause a huge discrepancy in the qualities of the predictions obtained. Hence, grid optimization cannot be neglected.

Taking into consideration the massive number of grids worth checking we concluded there was a need for a very fast but still well performing supervised model for a fixed grid, one that would simply execute a random search on a rich space of grid parameterizations to find the 'optimal' grid. This would yield a better final result than a more sophisticated, but slower algorithm applied to a random set of grids that would be too small to contain any decent tessellation.

B. Fast algorithm for a given grid

The main idea behind our algorithm for a fixed grid is simple: count the past crimes in every cell and mark the cells with 'the worst past' as hotspots. In other words, we assume that if many crimes occurred somewhere, more are likely to happen. This principle may strike some as naive and outdated, but we believe that it is both accurate enough and fast. Up-to-date crime registries are freely available for several US cities. They form the main dataset in data-driven crime forecasting algorithms. One can search for any external data which could affect future crimes, but have not left a trace on those crimes that have already been committed. We are aware that weather, demographics and even social media information (see [4]) are sometimes used in similar contexts. Unfortunately, they significantly increase the model's complexity, often without a guarantee of noticeably improved accuracy. Keeping computations as simple as possible, by using merely historical crime data, enables us to spend more time on selecting the right grid.

We refine the raw algorithm by taking care of data aging and seasonality. Namely, we assign weights to all the past crimes and then sum up the weights of all the crimes in consecutive cells to find the hotspots. The weight of an event decreases exponentially as a function of age (in days) of a crime. The intensity of the decrease is a hyperparameter. Also, we boost the weights of crimes committed on the same days of the year as those in the forecasted time span. The power of boosting is a hyperparameter as well.

Moreover, we introduce a primitive 'spatial radiation' of past crimes. For each data point, we put eight of its copies with reduced weights in the corners and in the center of the sides of the rhombus around it (see Figure 1). In this way, a 'part' of an event that has occured close to the cell border could fall into a neighboring cell. We chose to use a rhombus because it reflects the Manhattan metric, a reasonable match for North-South-oriented axis grid street plans, of which there are many in US cities. The size of the rhombus and reduction of weights of added copies are hyperparameters.

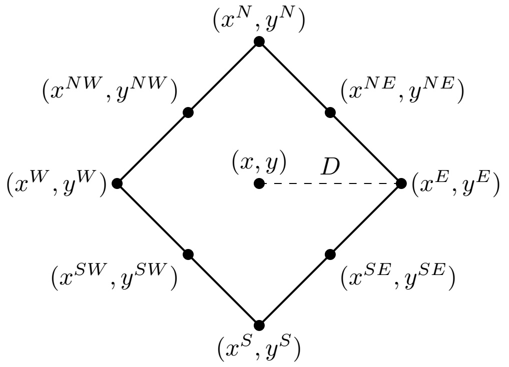


Fig. 1. Points on the rhombus around given point.

The approach presented here can be expressed in the language of spatio-temporal point processes (cf., e.g., [14] and references therein). Consider a counting process $N(t, x, y)$ characterized by its conditional intensity function λ. In our case we define lambda as (1), where

- $\mu_{\text{age}}(t_j, t) = e^{-A(t-t_j)}$,
- $\mu_{\text{seas}}(t_j) = 1 + B \cdot \nu(t_j)$ for $\nu(t_j) = 1$ if t_j is the same day of year as those placed in the forecasted timespan and $\nu(t_j) = 0$ elsewhere,
- $\varphi(x_j, y_j, x, y) = \begin{cases} 1, & \text{if } (x_j, y_j) \text{ and } (x, y) \\ & \text{are in the same cell} \\ 0, & \text{elsewhere,} \end{cases}$
- $\varphi_{\text{blur}}(x_j, y_j, x, y) = C \cdot \sum_{\{i \in \mathcal{D}\}} \varphi(x_j^i, y_j^i, x, y)$ for $\mathcal{D} = \{N, NE, E, SE, S, SW, W, NW\}$ and (x^i, y^i) defined as in Figure 1, i.e., $(x^N, y^N) = (x, y + D)$, $(x^{NE}, y^{NE}) = (x + D/2, y + D/2)$, $(x^E, y^E) = (x + D, y)$, etc.

We sum through all the past events (t_j, x_j, y_j) with $t_j < t$. A, B, C, D are hyperparameters. Our lambda is much simpler than those found in the literature. We need neither smoothing nor symmetry properties. Also, for a fixed t, $\lambda(t, x, y) = \lambda(t, x', y')$ for every (x, y) and (x', y') lying in the same cell. Hence, we can think about λ as of the intensity of the entire cell and simply choose cells with the greatest values of λ as hotspots.

C. Validation

To find the best grid and hyperparameter values, we split the dataset into training, validation and test parts in the following way: the last period becomes the test set, the second-to-last is treated as the validation set and all the earlier events make up the training set. Then we generate hotspots for different grids and hyperparameters using training data and compare them on validation data to choose the best settings. Finally, we compute the hotspots for the best model once more - this time with use of both training and validation data - and obtain the ultimate score using test data.

In classic crime forecasting, the score functions taken from the binary classification – ROC/AUC, sensitivity, etc. – are used (see [6]). There are also two newer functions on the market: predictive accuracy index (PAI, [6]) and prediction efficiency index (PEI, [21]) given by $PAI = \frac{n/N}{a/Ar}$, $PEI = \frac{n}{n^*}$, respectively, where:

- n - the number of future crimes in k proposed hotspots,
- n^* - the number of future crimes in k 'worst' cells,
- N - the number of all future crimes in the entire area,
- a - the total volume of k proposed hotspots,
- Ar - the volume of the entire area,

assuming that k cells were indicated as hotspots. They all have their disadvantages. Binary classification-based functions are inconvenient if the area of the hot-spots to be forecast is a very small fraction of the investigated jurisdiction, which is typical. As for other functions, PAI favors smaller single cell areas while PEI likes as great a single cell area as possible. For this reason it is impossible to maximize both PAI and PEI with the same grid, which casts doubt on the validity of using either of them. Moreover, PEI is bounded by 1 from above whereas the range of PAI is a positive half line, so they are not directly comparable. Nevertheless, our approach is metric-agnostic, therefore any reasonable score function can be applied here.

III. Case study

A. The competition

In September 2016, the National Institute of Justice in the US announced the *Real-Time Crime Forecasting Challenge*. The goal was to predict future crimes in Portland, Oregon. Contestants were asked to divide the area under Portland police jurisdiction (an area roughly 15 by 20 miles) into a grid of small cells (i.e., 250 by 250 feet) and indicate the cells that would have the highest future crime rate - hotspots. Several restrictions on the cells' shape and the total volume of hotspots were imposed.

$$\lambda(t,x,y) = \sum_{\{j\,:\,t_j<t\}} \mu_{\text{age}}(t_j,t)\cdot\mu_{\text{seas}}(t_j)\cdot[\varphi(x_j,y_j,x,y)+\varphi_{\text{blur}}(x_j,y_j,x,y)], \tag{1}$$

Four different categories of crime were considered separately: all crimes, burglaries, car thefts and street crimes (including assaults, robberies, shots fired). Five future time spans (starting in March 2017) were involved: one week, two weeks, a month, two months and three months. Hence, there were 20 type/time categories. In each of them, the predictions were compared against the actual state of affairs in Portland using both PAI and PEI. Thus, the competition consisted of $4 \cdot 5 \cdot 2 = 40$ separate sub-competitions in total. Only the best submission was awarded in each of them. Three independent tracks of the challenge were run simultaneously: intended for large businesses, small businesses and students, respectively. Each track had the same rules and goals, but separate contestants, winners and prizes.

B. Data

The NIJ delivered historical data on all the crimes registered in Portland between March 2012 and February 2017. Almost 1,000,000 records were provided in total. Each of them contained the day the crime was committed, coordinates (with accuracy to one foot) and the type of crime committed. There were no data gaps.

The distribution of data between crime categories was highly imbalanced: burglaries, car thefts and street crimes were only 0.5%, 1%, and 16.5% of records, respectively. Thus, we expected a huge discrepancy in the numbers of crimes committed between particular type/time categories between March and May 2017. That was true, two extreme cases were: all the crimes between March and May 2017 - 65,000 records, and burglaries in the first week of March 2017 - only 20 events.

Distributions of crimes in all the categories with a big enough number of events had similar characteristics: they consisted of the 'dense' part looking like a sample from a continuous distribution and the 'discrete' part made from atoms. It seems that although the accuracy of the coordinates of crimes committed was in general one foot, police officers tended to 'discretize' some areas like stores or shelters to a single spatial point next to the entrance to the building/area.

C. Computations

The first attempts showed that in each of the 20 type/time categories the PAI metric was maximized by a lot of small hotspots whereas PEI behaved best for a small number of large hotspots. Hence it was clear that we should not attempt to satisfy both metrics simultaneously. Since each metric formed an independent sub-competition with a separate prize, it was better to have a good score for one metric than mediocre results for both. So, for each of the 20 type/time categories we had to decide which metric to focus on in our further work. The metrics were incomparable, scores between the categories were incomparable and we did not know other competitors and their results. Thus, we did not have any hitching point that would help us to choose a metric. Moreover, our approach was metric-agnostic. Hence, to choose a metric, we just tossed a coin for each of 20 type/time categories.

During the competition we were examining parallelogram, triangular and hexagonal grids. No shape proved noticeably better than other ones. We ultimately decided to only use unrotated rectangular grids, parameterized by cell height, width, horizontal and vertical shift. The number of predicted hotspots was also a hyperparameter. We optimized the grid and our model hyperparameters for each of 20 type/time categories separately.

D. Results

Table I gathers information about seven categories with the largest numbers of crimes committed during the test periods. Our predictions proved the most accurate in all of them in the contest track for large businesses. Moreover, all of those predictions remained on the top after comparing results from the competition's three tracks (for large business, small businesses and students). This was the best result among all the competitors, while the runner-up achieved four across-track wins.

TABLE I
COMPETITION CATEGORIES WITH THE LARGEST NUMBERS OF CRIMES COMMITTED.

category	number of crimes	metric used	metric value
all, 3 months	55744	PAI	60.53
all, 2 months	35770	PEI	0.989
all, 1 month	17873	PAI	61.37
street, 3 months	8480	PEI	0.967
all, 2 weeks	8021	PEI	0.957
street, 2 months	5352	PEI	0.940
all, 1 week	3876	PAI	62.35

One more category for which our prediction was the most effective in the large business competition (but not in the total rank) was for burglaries between March and May 2017. However, in our opinion the number of crimes committed, 268, was so low that no model would be able to credibly predict them, so our success was just a matter of luck. We would conclude the same about seven more categories: the other time periods for burglaries (175, 93, 41, and 20 crimes) and car thefts in a one-month period and less (273, 135, and 71 incidents).

The results allowed us to conclude that for both the PAI and PEI metrics we were able to find grids and hotspots with quality competing with predictions obtained by authors of more complicated methods described in the literature (cf. [22], [23]). Our approach proved especially effective in categories with the biggest number of crimes committed.

Since different competitors submitted different grids, we are unable to compare algorithms for a fixed grid created by particular contestants. Therefore, we cannot judge whether the good performance of our models was an effect of thoroughly scouring potential grids or the power of simplicity of our algorithm for a fixed grid, or perhaps both.

IV. Discussion

The comparative case study on crime data from Portland, OR, shows that our computation time-oriented approach can compete with more sophisticated crime forecasting methods existing in the literature. This result is somewhat surprising. One may conclude that the spatio-temporal distribution of crimes committed is too complicated to be estimated well enough with the use of parametric methods. Or maybe the choice of the proper grid matters much more than it seems. Moreover, we have no reason to claim that the good performance of our algorithm is a one-shot success valid only for Portland since our model contains no part priorly adapted to any particular city. Unfortunately, we did not have the opportunity to compare the quality of crime forecasts done with use of different methods (including our own) for the same fixed grid. Such research would shed more light on this field.

The advantage of our algorithm for cases with thousands or more crimes to forecast can be attributed to two possible factors: a specific spatial distribution of crimes or computational simplicity. As stated above, for most statistical parametric methods it may be intractable to cover a distribution containing both a continuous and a discrete part. Comparing the performance of different models for a fixed grid would bear this out. On the other hand, sophisticated algorithms can paradoxically struggle to find the optimal grid and hotspots when presented with large volumes of training data. A time-consuming training procedure for a fixed grid does not allow one to check a sufficient number of potential grids. This problem may be addressed by more efficient algorithms' implementations and significantly increasing computing resources. Also, adding more constraints on the admissible grid shapes clearly solves the problem, though it also makes it less universal.

Finally, we note that in the perspective of maintaining and updating the crime forecasting system, using only the historical crime data seems to be a good solution. It is hard to find any non-constant external factor which can both influence future crimes and be easier to predict than crimes themselves. Besides, the impact of any hidden important feature is ultimately reflected in the historical data. Moreover, changes in the spatial crime distribution caused by system-driven preventive police activities may be not easy to manage when external data sources are used for forecasting. At the same time, a forecasting system based on merely historical data is able to simply retune to the current crime distribution.

References

[1] W. Gorr, A. Olligschlaeger, and Y. Thompson, “Short-term forecasting of crime,” *International Journal of Forecasting*, vol. 19, pp. 579–594, 2003.

[2] J. Cohen, W. L. Gorr, and A. M. Olligschlaeger, “Leading indicators and spatial interactions: A crime-forecasting model for proactive police deployment,” *Geographical Analysis*, vol. 39, pp. 105–127, 2007.

[3] L. W. Kennedy, J. M. Caplan, and E. Piza, “Risk clusters, hotspots, and spatial intelligence: Risk terrain modeling as an algorithm for police resource allocation strategies,” *Journal of Quantitative Criminology*, vol. 27, pp. 339–362, 2011.

[4] X. Wang, M. S. Gerber, and D. E. Brown, “Automatic crime prediction using events extracted from twitter posts,” in *Social Computing Behavioral - Cultural Modeling and Prediction*. Springer Berlin Heidelberg, 2012, pp. 231–238.

[5] K. J. Bowers, S. D. Johnson, and K. Pease, “Prospective hot-spotting: The future of crime mapping?” *The British Journal of Criminology*, vol. 44, pp. 641–658, 2004.

[6] S. Chainey, L. Tompson, and S. Uhlig, “The utility of hotspot mapping for predicting spatial patterns of crime,” *Security Journal*, vol. 21, pp. 4–28, 2008.

[7] M. Fielding and V. Jones, “Disrupting the optimal forager: Predictive risk mapping and domestic burglary reduction in trafford, greater manchester,” *International Journal of Police Science & Management*, vol. 14, pp. 30–41, 2012.

[8] W. L. Gorr and Y. Lee, “Early warning system for temporary crime hot spots,” *Journal of Quantitative Criminology*, vol. 31, pp. 25–47, 2015.

[9] M. D. Porter and B. J. Reich, “Evaluating temporally weighted kernel density methods for predicting the next event location in a series,” *Annals of GIS*, vol. 18, pp. 225–240, 2012.

[10] M. A. Boni and M. S. Gerber, “Automatic optimization of localized kernel density estimation for hotspot policing,” in *Proc. 15th IEEE Int. Conf. Machine Learning and Applications (ICMLA)*, Dec. 2016, pp. 32–38.

[11] H. Liu and D. E. Brown, “Criminal incident prediction using a point-pattern-based density model,” *International Journal of Forecasting*, vol. 19, pp. 603–622, 2003.

[12] M. A. Taddy, “Autoregressive mixture models for dynamic spatial poisson processes: Application to tracking intensity of violent crime,” *Journal of the American Statistical Association*, vol. 105, pp. 1403–1417, 2010.

[13] G. Rosser and T. Cheng, “Improving the robustness and accuracy of crime prediction with the self-exciting point process through isotropic triggering,” *Applied Spatial Analysis and Policy*, pp. 1–21, 2016.

[14] G. O. Mohler, M. B. Short, P. J. Brantingham, F. P. Schoenberg, and G. E. Tita, “Self-exciting point process modeling of crime,” *Journal of the American Statistical Association*, vol. 106, pp. 100–108, 2011.

[15] G. Mohler, “Marked point process hotspot maps for homicide and gun crime prediction in chicago,” *International Journal of Forecasting*, vol. 30, pp. 491–497, 2014.

[16] G. O. Mohler, M. B. Short, S. Malinowski, M. Johnson, G. E. Tita, A. L. Bertozzi, and P. J. Brantingham, “Randomized controlled field trials of predictive policing,” *Journal of the American Statistical Association*, vol. 110, pp. 1399–1411, 2015.

[17] C. Loeffler and S. Flaxman, “Is gun violence contagious? A spatiotemporal test,” *Journal of Quantitative Criminology*, 2017.

[18] W. Perry, B. McInnis, C. Price, S. Smith, and J. Hollywood, “Predictive policing: The role of crime forecasting in law enforcement operations,” RAND Corporation, Santa Monica, Tech. Rep., 2013.

[19] National Institute of Justice, “Real-Time Crime Forecasting Challenge,” https://www.nij.gov/funding/Pages/fy16-crime-forecasting-challenge.aspx, 2017. [Online]. Available: https://www.nij.gov/funding/Pages/fy16-crime-forecasting-challenge.aspx

[20] L. A. Rastrigin, “About convergence of random search method in extremal control of multi-parameter systems,” *Automation and Remote Control*, vol. 24, pp. 1467–1473, 1963.

[21] J. M. Hunt, “Do crime hot spots move? exploring the effects of the modifiable areal unit problem and modifiable temporal unit problem on crime hot spot stability,” Ph.D. dissertation, 2016.

[22] G. Mohler and M. D. Porter, “Rotational grid, PAI-maximizing crime forecasts,” 2018, to appear in Statistical Analysis and Data Mining.

[23] S. Flaxman, M. Chirico, P. Pereira, and C. Loeffler, “Scalable high-resolution forecasting of sparse spatiotemporal events with kernel methods: a winning solution to the nij "Real-Time Crime Forecasting Challenge",” 2018.

DOI: 10.15439/2020F140
ISSN 2300-5963 ACSIS, Vol. 21

Data Mining-Based Phishing Detection

Jan Bohacik
Department of Informatics,
University of Zilina, Univerzitna
8215/1, 010 26 Zilina, Slovakia
Email: Jan.Bohacik@fri.uniza.sk

Ivan Skula
Department of Informatics,
University of Zilina, Univerzitna
8215/1, 010 26 Zilina, Slovakia
Email: skula@dobraadresa.sk

Michal Zabovsky
University Science Park, University
of Zilina, Univerzitna 8215/1, 010
26 Zilina, Slovakia
Email: Michal.Zabovsky@uniza.sk

***Abstract*—Webpages can be faked easily nowadays and as there are many internet users, it is not hard to find some becoming victims of them. Simultaneously, it is not uncommon these days that more and more activities such as banking and shopping are being moved to the internet, which may lead to huge financial losses. In this paper, a developed Chrome plugin for data mining-based detection of phishing webpages is described. The plugin is written in JavaScript and it uses a C4.5 decision tree model created on the basis of collected data with eight describing attributes. The usability of the model is validated with 10-fold cross-validation and the computation of sensitivity, specificity and overall accuracy. The achieved results of experiments are promising.**

I. Introduction

PHISHING is understood to be a criminal attack on obtaining personal information, such as passwords and payment card information, through webpages or e-mails [13]. Webpage creators can easily make fake pages which are virtually identical to the original ones, so people can easily fall victim to them. An alarming sign is the availability of guides about how to make fake web pages directly on the internet, e.g. [6]. At the same time, online payments are increasingly being used and many other activities are being moved to the internet. For example, the transaction value of digital payments is expected to show a growth rate of 17.0 percent between 2020 and 2024 [11]. The number of internet users has grown 1,187 percent since 2000 and there are 4,648,228,067 internet users at this moment, which is 59.6 percent of the world population [2]. Therefore, it is very important for internet users to be able to detect phishing webpages. This is recognized by the Anti-Phishing Working Group that reported 165,772 phishing sites detected in the first quarter of 2020 in its Phishing Activity Trends Report published on 11 May 2020 [1]. As it is outlined in this Report, a recent trend has been the use of the COVID-19 pandemic for phishing attacks. For example, a fake site claiming to be an official registration for the immediate withdrawal of money from a compensation fund of the Brazilian government was disseminated in Brazil via WhatsApp in the first quarter of 2020. According to [1], in the first quarter of 2020, the most targeted phishing sectors were SAAS/webmails (33.5 percent), financial institutions (19.4 percent), payments (13.3 percent), social media (8.3 percent), e-commerce/retail (6.2 percent), and others.

According to [8], there are anti-phishing approaches based on: a) heuristic; b) content; c) blacklist; d) knowledge discovery in data; and e) hybrid combination of several previously mentioned approaches. The most complex and potentially most effective is an approach based on knowledge discovery in data and its merger with other approaches in a hybrid combination. This paper is focused especially on the collection of phishing data, the data mining step of knowledge discovery and the creation of a Chrome plugin for data collection and phishing detection. The interest of academics in the data mining step for the purposes of phishing detection has been shown in several papers [5], [9], [14]. One of the most popular algorithms for the data mining step is the C4.5 algorithm creating an easily interpretable decision tree for classification [15], [7]. In the three referenced academic papers regarding the data mining step for the purposes of phishing detection, the results of decision trees were shown to be promising. The C4.5 algorithm uses training data consisting of instances (webpages) described by defined describing attributes and classified into the class attribute with possible values legitimate and phishing. It is a recursive algorithm which associates the available describing attribute with the highest normalized information gain at each node of the decision tree. That eventually leads to the splitting of the instances into subsets enriched in value legitimate or phishing. Each leaf node of the decision tree is associated with a possible value of the class attribute. The Chrome plugin is written in JavaScript and it contains a created decision tree for the performance of the detection. It is used for the collection of instances with a manual assignment of value legitimate or phishing on the basis of an expert inspection of the webpage.

The following organization of the paper is used. The data collected for the creation of the data mining-based phishing detection is described and analyzed in Section II. In Section III, the developed plugin detecting phishing webpages in the Chrome browser and its decision tree model made with the collected data are presented. The results achieved in employed experiments with 10-fold cross-validation are in Section IV. And finally, Section V concludes the paper.

IEEE Catalog Number: CFP2085N-ART ©2020, PTI

II. Collected Data

The data characterized here contains descriptions of 1000 webpages visited through the Chrome browser with a developed plugin described in Section III. The plugin was used for saving data about these webpages and values legitimate or phishing were assigned to them manually on the basis of an expert inspection. Let us have a defined set $\boldsymbol{W}$ with 1000 webpages (instances). Let them be described by a defined set $\boldsymbol{B}$ with eight describing attributes and let them be classified into one class attribute D. The attributes in $\boldsymbol{B}$ and attribute D are presented in Table I. Describing attributes $\boldsymbol{B} = \{B_1; \ldots; B_k; \ldots; B_8\}$. If B_k is a numerical attribute and its value is v for a webpage $\boldsymbol{w} \in \boldsymbol{W}$, mark $B_k(\boldsymbol{w}) = v$ is used. Mark $B_k = \boldsymbol{P}$, B_k is a numerical attribute, $\boldsymbol{P}$ is a set of numerical values, means that $\boldsymbol{P}$ contains possible numerical values of B_k. If B_k is a categorical attribute and its categorical value is $b_{k,l}$ for a webpage $\boldsymbol{w} \in \boldsymbol{W}$, mark $B_k(\boldsymbol{w}) = b_{k,l}$ is used. Mark $B_k = \{b_{k,1}; \ldots; b_{k,l}; \ldots; b_{k,l_k}\}$ where B_k is a categorical attribute and $b_{k,1}, \ldots; b_{k,l}, \ldots, b_{k,l_k}$ are categorical values means $b_{k,1}, \ldots; b_{k,l}, \ldots, b_{k,l_k}$ are possible categorical values of categorical attribute B_k.

Table I.
Defined attributes

Attribute	Type of values	Possible values	Used units
AtInURL (B_1)	Categorical	*absent* ($b_{1,1}$)	N/A
		present ($b_{1,2}$)	
HyphenInURL (B_2)	Categorical	*absent* ($b_{2,1}$)	N/A
		present ($b_{2,2}$)	
SubdomainsInURL (B_3)	Numerical	1, 2, 3, ...	count
IPAddressInURL (B_4)	Categorical	*absent* ($b_{4,1}$)	N/A
		present ($b_{4,2}$)	
URLLength (B_5)	Numerical	4, 5, 6, ...	count
RatioOfLinksToOther Domains (B_6)	Numerical	[0;100]	%
RatioOfObjectsFrom OtherDomains (B_7)	Numerical	[0;100]	%
HTTPSProtocol (B_8)	Categorical	*trusted* ($b_{8,1}$)	N/A
		untrusted ($b_{8,2}$)	
ClassAttribute (D)	Categorical	*legitimate* (d_1)	N/A
		phishing (d_2)	

Attribute $B_1 = AtInURL = \{b_{1,1}; b_{1,2}\} = \{absent; present\}$ indicates if the URL address of some webpage $\boldsymbol{w}$ contains the @ symbol ($B_1(\boldsymbol{w})$ = *present*) or $\boldsymbol{w}$ does not contain it (i.e., $B_1(\boldsymbol{w})$ = *absent*). Normally, anything that is placed prior the @ symbol is ignored by the internet browser and redirection to what is typed after the @ symbol is performed. Attribute B_2 indicates if the URL address of some webpage $\boldsymbol{w}$ contains the '–' symbol ($B_2(\boldsymbol{w})$ = *present*) or $\boldsymbol{w}$ does not contain it ($B_2(\boldsymbol{w})$ = *absent*). This symbol may be used for creating a fake domain similar to the original one. *SubdomainsInURL* (B_3) contains the number of sub-domains in the URL address of the webpage. For example, fri.uniza.sk is a sub-domain of uniza.sk. B_3 = {1, 2, 3, ...}. Adding sub-domains to the URL address is another possible way for creating a fake domain and so a higher number of sub-domains is suspicious. Attribute B_4 indicates if the URL address contains an IP address. IP addresses may be used for hiding the real domains from the user. *URLLength* (B_5) contains the number of characters in the URL address. B_5 = {4, 5, 6, ...}. URL addresses with many characters may be used for hiding some information from the user. *RatioOfLinksToOtherDomains* (B_6) contains the ratio of links to other domains to all links in the webpage. B_6 = [0;100]. Too many links to other domains in the webpage might be indicative of a fake webpage. Attribute B_7 contains the ratio of objects such as images and videos from other domains to all objects in the webpage. B_7 = [0;100]. Similarly, too many objects from other domains might mean it is a fake webpage. *HTTPSProtocol* (B_8) indicates if some webpage $\boldsymbol{w}$ uses a trusted HTTPS protocol. Since HTTPS protocols are used for safe transfer of sensitive data, HTTPS protocols issued by unsound issuers or no HTTPS protocols at all are suspicious. Analysis of the collected data is provided in Table II.

Table II.
Analysis of the data collected with the Chrome plugin

Attribute	Particular value	Frequency of the value	Median	Mode
B_1	*absent* ($b_{1,1}$)	999	*absent*	*absent*
	present ($b_{1,2}$)	1		
B_2	*absent* ($b_{2,1}$)	874	*absent*	*absent*
	present ($b_{2,2}$)	126		
B_3	N/A	N/A	2	2
B_4	*absent* ($b_{4,1}$)	995	*absent*	*absent*
	present ($b_{4,2}$)	5		
B_5	N/A	N/A	46	22
B_6	N/A	N/A	95.24	100
B_7	N/A	N/A	100	100
B_8	*trusted* ($b_{8,1}$)	559	*trusted*	*trusted*
	untrusted ($b_{8,2}$)	441		
D	*legitimate* (d_1)	829	*legitimate*	*legitimate*
	phishing (d_2)	170		

III. Created Plugin and Decision Tree Model

Since no realistic data set for the creation of the decision tree model had been found, data about phishing and legitimate webpages were prepared first. Although there are some available data sets about webpages, they contain a very high percentage of data about phishing webpages, which is not the case in the real world and models created with this type of data might have issues. For example, the Phishing Websites Data Set from the UCI Repository of Machine Learning Databases [3] has only 40.5 percent

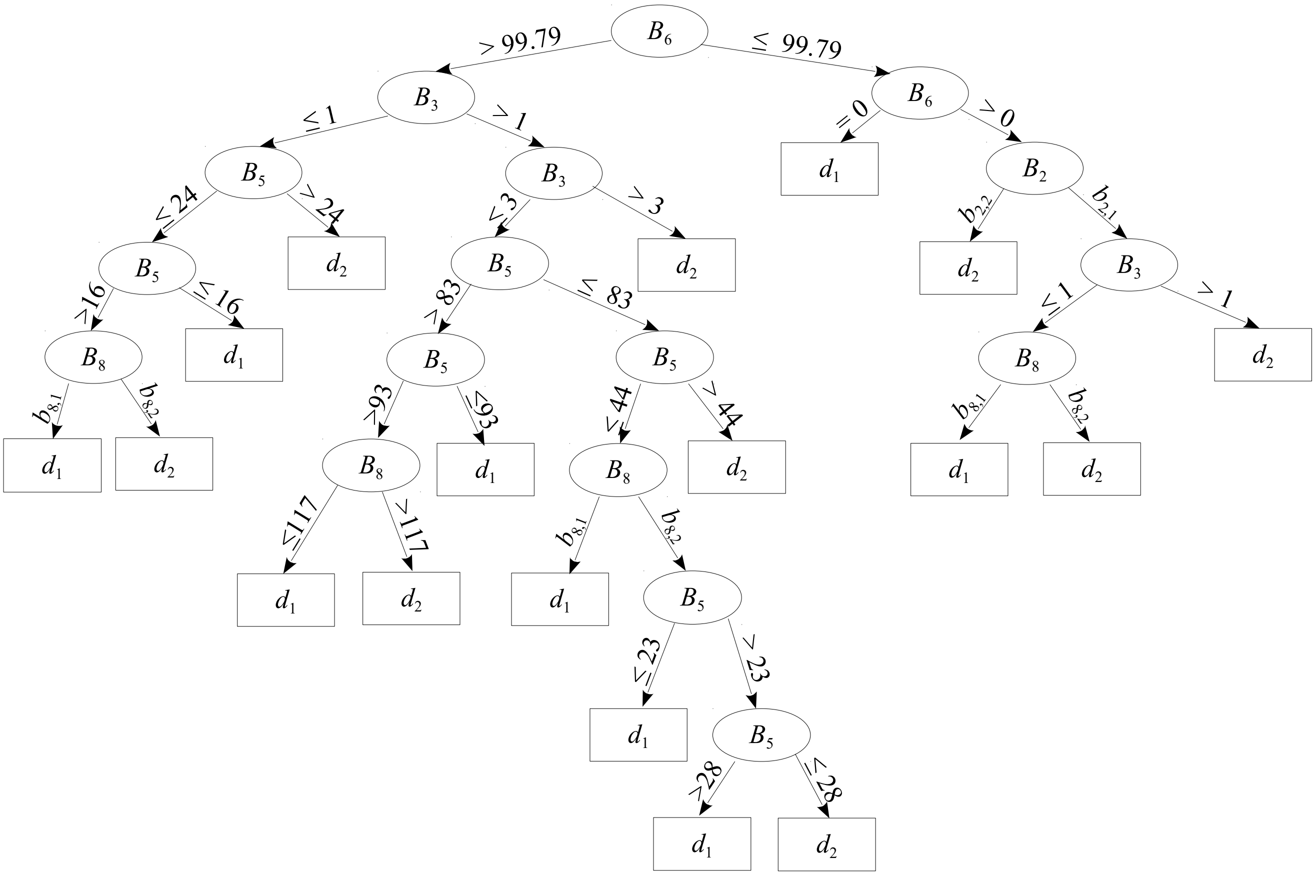


Fig 1. Decision tree model in the Chrome plugin

legitimate webpages. In addition, the functionality for the collection of data about a particular webpage is required in the Chrome plugin even after the decision tree model is created, as the collected data is used as the input of the model. The development of a Chrome plugin is similar to the development of a webpage because it consists of HTML, CSS and JavaScript codes hosted by the Chrome browser with the possibility to access some additional JavaScript APIs [10]. JavaScript APIs and a JavaScript code are used for the determination of the values for particular attributes in $\boldsymbol{B}$. The value for D is is set manually on the basis of an expert inspection. If the expert trusts some webpage $\boldsymbol{w}_1$, $D(\boldsymbol{w}_1)$ = *legitimate*, otherwise $D(\boldsymbol{w}_2)$ is set to *phishing*. Webpages $\boldsymbol{w} \in \boldsymbol{W}$ described by attributes in $\boldsymbol{B}$ and classified into D were given as the input to an implementation of the C4.5 algorithm in the Waikato Environment for Knowledge Analysis (Weka) [15]. The created decision tree shown in Fig. 1 is implemented into the plugin.

IV. Results of Employed Experiments

The results obtained in employed experiments with the C4.5 algorithm and with the collected data from Section II are described here first. It was important to see how the created decision tree model would perform potentially. All 1000 webpages from set $\boldsymbol{W}$ were loaded in Weka and then 10-fold cross-validation [4] was executed. In the validation, the real values for D and the detected values for D were compared and put into a confusion matrix [12] shown in Table III. There are 129 true positives, 27 false positives, 41 false negatives and 803 true negatives. Measures sensitivity, specificity and overall accuracy computed from the values in Table III are presented in Table IV. The achieved sensitivity is 0.7588, which means that 75.88 percent phishing websites were detected in the validation. The achieved specificity is 0.9675, which means that 3.25 percent websites generated false warnings about phishing activities in the validation. Finally, the achieved overall accuracy was 0.9320, which means that 6.80 percent of all websites were classified incorrectly. The results of 10-fold cross-validation show that the use of the decision tree model is promising. Several other data mining models were tried, but none of them gave significantly better results. In addition, it is simple to implement the decision tree with its use for detection in the Chrome plugin and its interpretability is high. Therefore, the decision tree shown in Fig. 1 was created on the basis of all webpages in $\boldsymbol{W}$ and implemented in the plugin. The plugin was tested on phishing and legitimate websites on the internet and the achieved results were similar to those in Table IV. The values of the describing attributes for three sample webpages are presented in Table V. When the correct leaf node for particular values of each website is found in the decision tree in Fig. 1, $\boldsymbol{w}_1$ is *phishing*, $\boldsymbol{w}_2$ is *legitimate* and $\boldsymbol{w}_3$ is *phishing*. Some comprehensive analysis of the decision tree indicates that describing attributes *AtInURL* (B_1), *IPAddressInURL* (B_4), *RatioOfObjectsFromOtherDomains* (B_7) are not predictive when the combination of the other attributes from $\boldsymbol{W}$ is used. It is likely their unique use is not common nowadays.

Table III.
Confusion matrix after 10-fold cross-validation

		Real	
		phishing	*legitimate*
Detected	*phishing*	129	27
	legitimate	41	803

Table IV.
Measures computed from the confusion matrix

Measure/Method	**Decision tree model**
Sensitivity	0.7588
Specificity	0.9675
Overall accuracy	0.9320

Table V.
Sample of data about webpages

Describing attribute	**Sample webpage**		
	$\boldsymbol{w}_1$	$\boldsymbol{w}_2$	$\boldsymbol{w}_3$
B_1	*present*	*absent*	*absent*
B_2	*absent*	*absent*	*absent*
B_3	5	1	2
B_4	*absent*	*absent*	*absent*
B_5	103	19	19
B_6	100	29	71
B_7	100	9	98
B_8	*untrusted*	*trusted*	*untrusted*

V. Conclusions

A Chrome plugin with a decision tree model for the detection of phishing webpages was described in the paper. The decision tree model was created with the C4.5 algorithm on the basis of collected data about 1000 legitimate and phishing webpages. It checks hyphens in URLs of webpages, sub-domains, lengths of URLs, links to other domains and HTTPS protocols. The results of using the C4.5 algorithm on the collected data in 10-fold cross-validation were promising with achieved sensitivity 0.7588, specificity 0.9675 and overall accuracy 0.9320. The use of the decision tree model in the Chrome plugin during browsing the internet led to similar values of the observed measures to the performed 10-fold cross-validation. The decision tree model suggests the use of the @ symbol in the URL address, IP address and objects from other domains in the webpage does not appear to be very predictive when the combination of hyphens, sub-domains, lengths of URLs, links to other domains and HTTPS protocols is checked. In the future, more describing attributes might be included and more webpages might be collected for the model.

Acknowledgment

The authors thank Peter Rovnanik for working on JavaScript implementations of the Chrome plugin, on collecting data and on conducting experiments under the supervision of the first author Jan Bohacik within his studies related to the preparation of the final thesis.

References

[1] Anti-Phishing Working Group, *Phishing Activity Trends Report, 1st Quarter 2020*. USA: Anti-Phishing Working Group, 2020, https://docs.apwg.org/reports/apwg_trends_report_q1_2020.pdf.

[2] E. d. Argaez, *Internet Usage Statistics: The Internet Big Picture*, Bogota, Colombia: Internet World Stats, 2020, https://www.internetworldstats.com/stats.htm.

[3] D. Dua and C. Graff, *UCI Machine Learning Repository*, USA: University of California, School of Information and Computer Science, 2019, http://archive.ics.uci.edu/ml.

[4] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. : Springer-Verlag, 2009, https://dx.doi.org/10.1007/978-0-387-84858-7.

[5] M. Karabatak, T. Mustafa, "Performance comparison of classifiers on reduced phishing website dataset," in *Proc. of the International Symposium on Digital Forensic and Security*, IEEE, Turkey, 2018, pp. 1-5, https://dx.doi.org/10.1109/ISDFS.2018. 8355357.

[6] B. M. Lawrence, *How to Make Fake Web Pages*. : Techwalla, 2020, https://www.techwalla.com/articles/how-to-make-fake-web-pages.

[7] K. Pancerz, V. Levashenko, E. Zaitseva, J. Gomuła, "Experiments with classification of MMPI profiles using fuzzy decision trees," in *Proc. of the Federated Conference on Computer Science and Information Systems*, IEEE, Poland, 2018, pp. 125-128, https://dx.doi.org/10.15439/2018F111.

[8] S. Patil, S. Dhage, "A methodical overview on phishing detection along with an organized way to construct an anti-phishing framework," in *Proc. of the International Conference on Advanced Computing & Communication Systems*, IEEE, India, 2019, pp. 588-593, https://dx.doi.org/10.1109/ICACCS.2019.8728356.

[9] Y. Pristyanto, A. Dahlan, "Hybrid resampling for imbalanced class handling on web phishing classification dataset," in *Proc. of the International Conference on Information Technology, Information Systems and Electrical Engineering*, IEEE, Indonesia, 2019, pp. 401-406, https://dx.doi.org/10.1109/ICITISEE48480.2019.9003803.

[10] J. Sonmez, *How to Create a Chrome Extension in 10 Minutes Flat*. Australia: sitepoint, 2015, https://www.sitepoint.com/create-chrome-extension-10-minutes-flat/.

[11] Statista, *Digital Payments: Worldwide*. Germany: Statista, 2020, https://www.statista.com/outlook/296/100/digital-payments/worldwide.

[12] S. V. Stehman, "Selecting and interpreting measures of thematic classification accuracy", *Remote Sensing of Environment*, vol. 62, no. 1, pp. 77–89, 1997, https://dx.doi.org/10.1016/S0034-4257(97)00083-7.

[13] L. Wenyin, G. Huang, L. Xiaoyue, X. Deng, and Z. Min, "Phishing web page detection," in *Proc. of the International Conference on Document Analysis and Recognition*, IEEE, South Korea, 2005, pp. 560–564, https://dx.doi.org/10.1109/ICDAR. 2005.190.

[14] R. Wahyudi, H. Marcos, U. Hasanah, B. P. Hartato, T. Astuti, R. A. Prasetyo, "Algorithm evaluation for classification 'phishing website' using several classification algorithms", in *Proc. of the International Conference on Information Technology, Information Systems and Electrical Engineering*, IEEE, Indonesia, 2018, pp. 265-270, https://dx.doi.org/10.1109/ICITISEE.2018.8720975.

[15] I. H. Witten, E. Frank, M. A. Hall, C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*. USA: Morgan Kaufmann, 2017, https://dx.doi.org/10.1016/C2015-0-02071-8.

DOI: 10.15439/2020F169
ISSN 2300-5963 ACSIS, Vol. 21

Computing Duals of Finite Gödel Algebras

Pietro Codara, Gabriele Maurina and Diego Valota
Dipartimento di Informatica, Università degli Studi di Milano, Italy
codara@di.unimi.it, gabriele.maurina@studenti.unimi.it, valota@di.unimi.it

***Abstract*—We introduce an algorithm that computes and counts the duals of finite Gödel-Dummett algebras of $k \geq 1$ elements. The computational cost of our algorithm depends on the factorization of k, nevertheless a Python implementation is sufficiently fast to compute the results for very large values of k.**

I. Introduction

Mathematical Fuzzy Logics (MFL) interpret predicates in truth-degrees ranging over the unitary real interval $[0, 1]$. It has been argued that this is a valid approach to deal with the inherent *vagueness* of terms in human languages [1]. In his seminal book [2], Hájek introduced a family of many-valued fuzzy logical systems where conjunction and implication connectives are modeled by continuous *t-norms* [1] and their residua, respectively. One of the three main fuzzy logics in Hájek's framework is Gödel-Dummett logic $\mathcal{G}$, whose conjunction is modeled by the *minimum*. Gödel-Dummett logic is a non-classical logic whose studies date back to Gödel [3] and Dummett [4]. Note that $\mathcal{G}$ can be obtained by adding the prelinearity axiom to *Intuitionistic logic*.

The algebraic counterpart of Gödel-Dummett logic is the variety of Gödel algebras $\mathbb{G}$. In the study of the algebraic semantics of non-classical logics, the notion of free algebra is of particular importance. This is due to the well-known isomorphism between free algebras and *Lindenbaum algebras* of logically equivalent formulas in a given logic. One can find in the literature several methods to obtain the order structures and cardinalities of free Gödel algebras. In 1969, Horn [5] obtained a recurrence formula to compute the cardinalities of free k-generated Gödel algebras, for any $k \in \mathbb{N}^+$. Another solution to this problem can be achieved by restating the Horn's recurrence in terms of finite forests [6].

A related counting problem is the *fine spectrum* problem [7], which aim to find the number of non-isomorphic algebras of cardinality k in a given variety. In [8], the author introduces a method to generate duals of finite Gödel algebras of a given cardinality and a recurrence relation to count the number of such structures, solving in this way the fine spectrum problem for $\mathbb{G}$. Such a result is obtained by exploiting the relation between finite forests and finite Gödel algebras.

In this paper, we build on [8] to obtain an algorithm that accepts a positive integer k as input and returns the number of non-isomorphic k-elements Gödel algebras. Moreover, we propose a Python implementation of such algorithm which is also able to generate the dual structures of the algebras. Finally, we compare the execution times of our algorithm with those obtained using Mace4 [9], a general purpose computer algebra system used to generate finite models. Besides being an interesting theoretical problem, the generation of finite algebras has also important applications in *automated reasoning* [10]. Indeed, such procedures can be used to find countermodels of logical formulas.

II. Gödel Algebras, Forests and Recurrence

We assume that the reader is acquainted with many-valued logics in Hájek's sense and with their algebraic semantics. We refer the reader to [2] for any unexplained notion. Throughout the paper we use the same symbols for logic's connectives and their algebraic interpretations.

Hájek's logic $\mathcal{BL}$ is the logic of all continuous t-norms and their residua, built over the language $\{\odot, \wedge, \vee, \rightarrow, \neg, \bot, \top\}$. The algebraic semantics of $\mathcal{BL}$ is given by the variety $\mathbb{BL}$ of BL algebras, that is, prelinear, divisible, commutative, bounded, integral, residuated lattices [2]. A *BL algebra* is an algebra $\mathbf{A} = \langle A, \wedge, \vee, \odot, \rightarrow, \bot, \top\rangle$ of type $(2, 2, 2, 2, 0, 0)$ such that $\langle A, \wedge, \vee, \bot, \top\rangle$ is a bounded lattice, with top $\top$ and bottom $\bot$, $\langle A, \odot, \top\rangle$ is a commutative monoid, satisfying the *residuation* equivalence, $x \odot y \leq z$ if and only if $x \leq y \rightarrow z$, the *prelinearity* equation $(x \rightarrow y) \vee (y \rightarrow x) = \top$, and *divisibility* $x \odot (x \rightarrow y) = x \wedge y$. Notice that divisibility implies that the lattice structure is distributive. A BL algebra satisfying $x \wedge y = x \odot y$ is called *Gödel algebra*. Hence, the variety of Gödel algebras $\mathbb{G}$ is a subvariety of $\mathbb{BL}$ [2].

Let $\mathbf{A}$ be a Gödel algebra, a *filter* of $\mathbf{A}$ is a non-empty subset p of A such that for all $y \in A$, if there is x in p such that $x \leq y$ then $y \in p$, and $x \wedge y \in p$ for all $x, y \in p$. We call *propers* the filters p such that $p \neq A$. A proper filter p of $\mathbf{A}$ is said to be *prime* when for all $x, y \in A$, either $x \rightarrow y \in p$ or $y \rightarrow x \in p$. The set of all prime filters $\mathsf{Spec}(\mathbf{A})$ of $\mathbf{A}$ ordered by reverse inclusion is called the *prime spectrum* of $\mathbf{A}$. When $\mathbf{A}$ is finite, each prime filter p of $\mathbf{A}$ is generated by a join-irreducible element a as $p = \{b \in \mathbf{A} \mid a \leq b\}$. On the other hand, each join-irreducible element of $\mathbf{A}$ singly generates a prime filter of $\mathbf{A}$. Hence, $\mathsf{Spec}(\mathbf{A})$ is isomorphic with the poset of the join-irreducible elements of $\mathbf{A}$. See Fig. 1 as an example where $\mathbf{A}$ is the free 1-generated Gödel algebra.

A *forest* F is a poset such that the downset $\downarrow q$ of every $q \in F$ is a *chain*, that is $\downarrow q$ is totally ordered. A forest with a bottom element is called a *tree*. Such bottom element is called the *root* of the tree. A *subforest* of a forest F is the downset of some $Q \subseteq F$. Finite forests and open maps form a category FF. Given two forests F, F' we denote $F \sqcup F'$ the disjoint union

[1] A t-norm is an associative, commutative and monotone function $\odot \colon [0, 1]^2 \rightarrow [0, 1]$, where 1 is the neutral element.

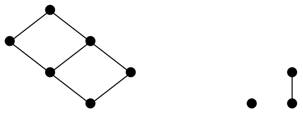


Fig. 1. The free Gödel Algebra on one generator and its prime spectrum.

of F and F'. Since FF is a category, disjoint unions are in fact coproducts in FF. For our purposes, we need to introduce an additional operation, called the *lifting* of a forest F, denoted by $F_\perp$, and obtained by adding a common root to all trees in F. Clearly, for every forest F, $F_\perp$ is a tree. A complete account with proofs on the operations in FF can be found in [11]. The prime spectrum of a finite Gödel algebra forms a forest, as shown by Horn in [12]. We can also obtain a Gödel algebra from a finite forest F in the following way. Let $\mathsf{Sub}(F)$ be the finite set of subforests of F. We equip $\mathsf{Sub}(F)$ with the structure of a Gödel algebra $\langle \mathsf{Sub}(F), \cap, \cup, \rightarrow, \emptyset, F\rangle$, where $F' \rightarrow F'' = F \setminus \uparrow (F' \setminus F'')$, for all $F', F'' \in \mathsf{Sub}(F)$. In this way, we can obtain the following isomorphisms $\mathsf{Spec}(\mathsf{Sub}(F)) \cong F$ and $\mathsf{Sub}(\mathsf{Spec}(\mathbf{A})) \cong \mathbf{A}$, for a given finite Gödel algebra $\mathbf{A}$. When $F \cong \mathsf{Spec}(\mathbf{A})$ and $\mathbf{A} \cong \mathsf{Sub}(F)$, we call such F the *dual* of $\mathbf{A}$, while $\mathbf{A}$ is the *primal* of F. It is also possible to define Sub and Spec over maps, making them functors acting on the category of finite forests and open maps FF and the category of finite Gödel algebras and their homomorphisms, extending in this way the above equivalence to a full categorical duality. These constructions go beyond the scope of the present paper, and we refer the interested reader to [13] for further details. With this machinery in mind we are ready to formally define the fine spectrum problem for $\mathbb{G}$.

Problem: $\mathrm{Fine}_{\mathbb{G}}(k)$
Input: $k \in \mathbb{N}^+$
Output: $m \in \mathbb{N}^+$ such that $m = |\{[\mathbf{A}] \in \mathbb{G} \mid k = |A|\}|$, where $[\mathbf{A}]$ is the class of finite Gödel algebras isomorphic with $\mathbf{A}$.

To solve this problem, we summarize the recurrence relation introduced in [8] to generate duals of finite Gödel algebras. This procedure is based upon the concept of *multiplicative partition* of a positive integer k [14], that is

$$\begin{aligned} &\mathsf{MP}(k) := \{(n_1, \dots, n_t) \mid \\ &k = n_1 \times \dots \times n_t, n_1 \le \dots \le n_t, t > 1\}. \end{aligned}$$

Each t-uple in $\mathsf{MP}(k)$ is composed of natural numbers whose product is equal to k. Note that the usual definition of multiplicative partition of k includes (k), while our definition does not. We define recursively the following sets of forests, which are fundamental for our work:

$$H_1 = \{\emptyset\} \tag{H_1}$$

$$H_k = P_k \cup Z_k \tag{H_k}$$

$$P_k = \{F_\perp | F \in H_{k-1}\} \tag{P_k}$$

$$\begin{aligned} Z_k = \{&F_1 \sqcup \dots \sqcup F_t \mid \\ &F_1 \in P_{n_1}, \dots, F_t \in P_{n_t}, (n_1, \dots, n_t) \in \mathsf{MP}(k)\} \end{aligned} \tag{Z_k}$$

Theorem 1 ([8]): $F \in H_k$ if and only if $|\mathsf{Sub}(F)| = k$. Moreover, $\langle \mathsf{Sub}(F), \cap, \cup, \rightarrow, \emptyset, F\rangle$ is isomorphic to a k-elements Gödel algebra $\mathbf{A}$ such that $\mathsf{Spec}(\mathbf{A}) \cong F$.

Thanks to this recursive definition of the set of duals of k-elements Gödel algebras H_k, we are also able to compute the cardinality of H_k, that is the fine spectrum of $\mathbb{G}$.

Corollary 1 ([8]): $\mathrm{Fine}_{\mathbb{G}}(k) = f(k) + pr(k) \times g(k)$ with,

$$pr(k) = \begin{cases} 0 & \text{if } k \text{ is prime;} \\ 1 & \text{otherwise.} \end{cases} \tag{pr}$$

$$f(1) = 1 \tag{f_1}$$

$$f(k) = \mathrm{Fine}_{\mathbb{G}}(k-1) \tag{f_k}$$

$$g(k) = \sum_{(n_1, \dots, n_t) \in \mathsf{MP}(k)} f(n_1) \times \dots \times f(n_t) \tag{g_k}$$

The sequence of numbers generated by $\mathrm{Fine}_{\mathbb{G}}(k)$ was already contained in the *On-Line Encyclopedia of Integer Sequences* as sequence A130841, that counts the number of ways to express an integer as a sum of so-called *oterms*. In [8], the author shows that oterms are a syntactic description of finite forests. In the next section we show how to obtain a fast algorithm implementing $\mathrm{Fine}_{\mathbb{G}}(k)$, able to compute such values for very large k.

III. Algorithm

A naive implementation of the recurrences of the previous Section leads to recursive procedure that runs at exponential costs by recursively calling $\mathrm{Fine}_{\mathbb{G}}(m)$ for every instance of m occurring in $\mathsf{ML}(i)$ for $1 \le i \le k$. We rewrite the recurrence in Corollary 1 in a more compact way:

$$\begin{aligned} &\mathrm{Fine}(k) = \mathrm{Fine}(k-1) + \\ &+ pr(k) \times \sum_{(n_1, \dots, n_t) \in \mathsf{MP}(k)} \mathrm{Fine}(n_1 - 1) \times \dots \times \mathrm{Fine}(n_t - 1). \end{aligned}$$

We can now obtain a more efficient algorithm just by applying *dynamic programming* [15] to this recurrence. Indeed, for computing $\mathrm{Fine}(k)$ we need (potentially all) the values $\mathrm{Fine}(i)$ for $1 \le i \le k$. So, we compute $\mathrm{Fine}(i)$ for every $i \in (1 \le 2 \le 3 \le \dots \le k)$ following the natural integers order and storing the computed values in a k-element vector $Fine$. The algorithm is outlined in Algorithm 1, and it assumes that there exists a function `multpart` that receives a $m \in \mathbb{N}^+$ and returns the set of multiplicative partitions $\mathsf{ML}(m)$ when m is composite, otherwise when m is prime `multpart` returns the empty set. We show in Section IV how to implement such a function using Python libraries.

The number of multiplicative partitions $\mathsf{MP}(k)$ is less than or equal to $\frac{k}{log\ k}$ for every $k \in \mathbb{N}^+$ such that $k \neq 144$ [16]. Hence, it is straightforward to see that for every $k \in \mathbb{N}^+$ such that $k \neq 144$, the inner `for` cycle (Line 8 in Algorithm 1) makes $O(\frac{k}{log\ k})$ steps to compute G. Then, the computation of $Fine[k]$ need necessarily $(k \times \frac{k}{log\ k})$ steps, but this is not sufficient. This bound cannot be fruitfully used to study the cost of the full algorithm. Indeed to compute the multiplicative partitions $\mathsf{MP}(n)$ of each n in $\{1, \dots, k\}$, we need to factorize

Algorithm 1 A function calculating $\mathrm{Fine}_{\mathbb{G}}(k)$

```
    $Fine[1] \leftarrow 1$;
 2: if $k == 1$ then
        return $Fine[1]$;
 4: end if
    for $n = 2;\ n \leq k;\ n = n + 1$ do
 6:     $M \leftarrow \mathtt{multpart}(n)$
        $G \leftarrow 0$
 8:     for each $(n_1, \dots, n_t) \in M$ do
            $G \leftarrow G + (Fine[n_1 - 1] \times \cdots \times Fine[n_t - 1])$
10:     end for
        $Fine[n] \leftarrow Fine[n-1] + G$;
12: end for
    return $Fine[k]$;
```

n in the function multpart(n) (Line 6 in Algorithm 1). By now, no efficient integer factorization algorithm can be found in literature. In the next Section we see that our strategy relies on the *prime factorization* of n, and this is essentially an exponential procedure.

IV. Implementation

Algorithm 1 has been implemented in Python using SymPy library [17]. The main issue in the implementation is to find an efficient way to obtain the set of t-uples $\mathsf{MP}(k)$ for a given $k \in \mathbb{N}^+$. We have used the sympy.factorint method to obtain the list of prime factors $fact(k)$ of k. Such method is particularly useful to our purpose because it uses different algorithms in the library, selecting the most efficient one according to the size of k. Now it is easy to realize that each t-uple $\mathsf{MP}(k)$ can be obtained from the multiset partitions of $fact(k)$. Hence, we have used the method multiset_partitions in sympy.utilities.iterables to create exactly the list of t-uples corresponding to elements of $\mathsf{MP}(k) \cup (k)$. As mentioned above we don't need the one-block partition (k), so our code just ignore it.

By slightly modifying Algorithm 1, we have also implemented functions to generate the full set of forests H_k by building trees using parenthesis representation, and to produce images of such forests using GraphViz library [18].

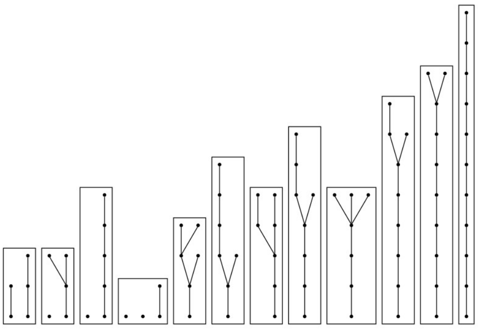


Fig. 2. The set of forests H_{12} generated by our Python code.

Example 4.1: Let $k = 12$. Then, $fact(k) = \{2, 3\}$ and applying multiset_partitions to $fact(12)$ we obtain $[[[2,2,3]], [[2,2],[3]], [[2,3],[2]], [[2],[2],[3]]]$. From this list it is easy to obtain the t-uples in $\mathsf{MP}(12) \cup (12)$ by multiplying elements in the same blocks, that is $[[12],[4,3],[6,2],[2,2,3]]$. Since by definition $(12) \notin \mathsf{MP}(12)$, the code skip the first partition $[12]$. Fig. 2 is the depiction of H_{12} produced by our program. As an instance, the parenthesis representation of the first and second forest on the left in Fig. 2 are $[[]], [[[]]]$ and $[], [[[], []]]$ respectively.

The following results have been obtained on a GNU/Linux Debian 4.9.130-2 system with an Intel Core i7-5500U CPU and 8GB of RAM. The Python implementation, together with the Mace4 input and output files, can be downloaded from https://homes.di.unimi.it/~valota/code/finegodel.zip.

To study the effectiveness of our implementation we have used Mace4 [9] to compute the number of finite Gödel algebras. Mace4 produces the algebraic structures on output files, then we need to run two tools: interpformat to convert the outputs in a readable format, and isofilter to get rid of isomorphic copies of our algebraic structures. Running times, calculated with the Debian GNU/Linux command-line tool time are summarized in Table I. The · symbol indicates that the instance is not meaningful. In fact, Algorithm 1 computes only the structure of forests in H_i for every $i \in \{2, \dots, k\}$.

TABLE I
Running times to compute finite Gödel algebras (or their duals) of cardinality 2 to 11.

k	Running Times	Mace4	interpformat +isofilter	Algorithm 1
2 to 9	real user	0m49.623s 0m49.272s	0m5.427s 0m5.384s	0m0.024s 0m0.024s
10	real user	10m58.814s 10m56.092s	0m44.284s 0m44.148s	· ·
11	real user	191m27.975s 190m48.804s	8m28.547s 8m26.960s	· ·
2 to 11	real user	· ·	· ·	0m0.024s 0m0.020s

The huge increase in computing time when passing from cardinality 10 to cardinality 11 in Mace4 runnings, shows the unsuitability of brute-force approaches. In fact, inspecting (H_k) one realizes that to obtain duals of finite Gödel algebras of cardinality 11, it is sufficient to lift all the forest in H_{10}. To compare, our Python implementation is able to generate the parenthesis representation of forests in H_i from $i = 2$ to $i = 11$, our script takes time real: 0m0.024s and user: 0m0.020s (last line in Table I). However, for counting purposes the script runs very fast, as testified by the performances summarized in Table II.

TABLE II
RUNNING TIMES OF ALGORITHM 1 FOR LARGE VALUES OF k.

k	Running Times	$\mathrm{Fine}_{\mathbb{G}}(k)$
2 to 1000	`real`: 0m0.558s `user`: 0m0.556s	$\mathrm{Fine}_{\mathbb{G}}(1000) =$ 3316527416
2 to 5000	`real`: 0m24.486s `user`: 0m24.476s	$\mathrm{Fine}_{\mathbb{G}}(5000) =$ 772140728313177
2 to 10000	`real`: 2m2.602s `user`: 2m2.580s	$\mathrm{Fine}_{\mathbb{G}}(10000) =$ 416184590541943029

V. CONCLUSIONS AND FURTHER WORKS

We should point out that Mace4 generates full models with tables for each algebraic operation, while our software only generates the order structure of the duals of finite algebras. To obtain the algebraic structures, one can appeal to Theorem 1 and obtain from each forest F produced by the Python script, the structure of the corresponding Gödel algebra by considering every subforest in $\mathsf{Sub}(F)$. Then for instance, the construction of the conjunction operation's table amount to consider set-inclusion among the subforests. Such functionality is not present in our software and is left as future work. Mace4 is a general-purpose Computer Algebra System. Another interesting approach to compare with our work is contained in [19], where authors introduce a brute-force algorithm with a heuristic test to detect isomorphic lattices, that computes tables operation for classes of residuated lattices. The counting of such structures is a byproduct of their work. They count several finite algebras related to many-valued logics until cardinality 12, including Gödel algebras. Our algorithm is essentially based on the duality between finite forests and open maps, and finite Gödel algebras and their homomorphisms. So, it makes sense to generalize this duality-based approach to other classes of algebras for which combinatorial dualities exist. In particular, we need subforest representations for (at least) finite algebras. Such type of representations can be found in the literature for locally finite subvarieties of MTL algebras [20], such as nilpotent minimum algebras [21], and revised drastic product algebras [22] and their subvarieties: drastic product algebras and EMTL algebras [23]. Another interesting duality for finite Gödel$_\triangle$ algebras is introduced in [24], and in [25] is shown that the dual category of this variety is also dual to the variety of drastic product algebras (studied in [26]). All these varieties are subvarieties of an interesting algebraic variety related to weak negation functions over $[0, 1]$, the variety of WNM algebras. In [27] one can find an extensive study of WNM chains that leads to a combinatorial representation of finitely generated free WNM algebras. Free and finite algebras are closely related. Indeed, every k-generated algebra in a variety $\mathbb{V}$ can be obtained as a quotient of the free k-generated algebra in $\mathbb{V}$. However, different congruences may generate isomorphic quotients. Hence, studies on free and fine spectra can be also used to investigate congruences in varieties. Finally, our algorithm can be used to obtain additional insight on the structure of finite Gödel algebras, helping researchers to find structural properties of such algebraic structures or a bound for the fine spectrum of $\mathbb{G}$.

REFERENCES

[1] N. Smith, "Fuzzy logics in theories of vagueness," in *Handbook of Mathematical Fuzzy Logic. Vol. 3*, P. Cintula, C. Fermüller, and C. Noguera, Eds. College Publications, 2016, vol. 58, pp. 1237–1281.

[2] P. Hájek, *Metamathematics of Fuzzy Logic*, ser. Trends in Logic. Kluwer Academic Publishers, 1998, vol. 4.

[3] K. Gödel, "Zum intuitionistischen Aussagenkalkul," *Anzeiger Akademie der Wissenschaften Wien*, vol. 69, pp. 65–66, 1932.

[4] M. Dummett, "A propositional calculus with denumerable matrix," *J. Symb. Log.*, vol. 24, no. 2, pp. 97–106, 1959.

[5] A. Horn, "Free L-Algebras," *J. Symb. Log.*, vol. 34, no. 3, pp. 475–480, 1969.

[6] O. M. D'Antona and V. Marra, "Computing coproducts of finitely presented Gödel algebras," *Ann. Pure Appl. Logic*, vol. 142, no. 1, pp. 202–211, 2006.

[7] W. Taylor, "The fine spectrum of a variety," *Algebra Universalis*, vol. 5, no. 1, pp. 263–303, 1975.

[8] D. Valota, "Spectra of Gödel Algebras," in *Language, Logic, and Computation. TbiLLC 2017*, ser. Lecture Notes in Computer Science, A. Silva, S. Staton, P. Sutton, and C. Umbach, Eds., 2019, vol. 11456.

[9] W. McCune, "Prover9 and mace4," 2005–2010, `http://www.cs.unm.edu/~mccune/prover9/`.

[10] J. A. Robinson and A. Voronkov, Eds., *Handbook of Automated Reasoning (in 2 volumes)*. Elsevier and MIT Press, 2001.

[11] S. Aguzzoli and P. Codara, "Recursive formulas to compute coproducts of finite Gödel algebras and related structures," in *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2016, pp. 201–208.

[12] A. Horn, "Logic with Truth Values in a Linearly Ordered Heyting Algebra," *J. Symb. Log.*, vol. 34, no. 3, pp. pp. 395–408, 1969.

[13] S. Aguzzoli, S. Bova, and B. Gerla, "Free Algebras and Functional Representation for Fuzzy Logics," in *Handbook of Mathematical Fuzzy Logic*, P. Cintula, P. Hájek, and C. Noguera, Eds. College Publications, 2011, vol. 2, pp. 713–791.

[14] A. Knopfmacher and M. E. Mays, "A survey of factorization counting functions," *International Journal of Number Theory*, vol. 01, no. 04, pp. 563–581, 2005.

[15] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction To Algorithms*. McGraw-Hill Publishing Company, 2001.

[16] F. Dodd and L. Mattics, "Estimating the number of multiplicative partitions," *Rocky Mountain Journal of Mathematics*, vol. 17, no. 4, pp. 797–814, 12 1987.

[17] A. Meurer and et al., "Sympy: symbolic computing in python," *PeerJ Computer Science*, vol. 3, p. e103, 2017. [Online]. Available: https://doi.org/10.7717/peerj-cs.103

[18] J. Ellson, E. R. Gansner, E. Koutsofios, S. C. North, and G. Woodhull, "Graphviz and dynagraph - static and dynamic graph drawing tools," in *GRAPH DRAWING SOFTWARE*. Springer-Verlag, 2003, pp. 127–148.

[19] R. Belohlavek and V. Vychodil, "Residuated lattices of size $\leq$ 12," *Order*, vol. 27, no. 2, pp. 147–161, 2010.

[20] F. Esteva and L. Godo, "Monoidal t-norm based logic: Towards a logic for left-continuous t-norms," *Fuzzy Sets and Systems*, vol. 124, no. 3, pp. 271–288, 2001.

[21] S. Aguzzoli, M. Busaniche, and V. Marra, "Spectral Duality for Finitely Generated Nilpotent Minimum Algebras, with Applications," *J. Log. Comput.*, vol. 17, no. 4, pp. 749–765, 2007.

[22] S. Bova and D. Valota, "Finite RDP-algebras: Duality, Coproducts and Logic," *J. Log. Comput.*, vol. 22, no. 3, pp. 417–450, 2012.

[23] D. Valota, "Representations for logics and algebras related to revised drastic product t-norm," *Soft Computing*, vol. 23, pp. 2331–2342, 2019.

[24] S. Aguzzoli, M. Bianchi, B. Gerla, and D. Valota, "Probability Measures in Gödel$_\triangle$ Logic," in *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, A. Antonucci, L. Cholvy, and O. Papini, Eds. Springer, 2017, pp. 353–363.

[25] ——, "Free algebras, states and duality for the propositional Gödel$_\triangle$ and Drastic Product logics," *International Journal of Approximate Reasoning*, vol. 104, pp. 57–74, 2019.

[26] S. Aguzzoli, M. Bianchi, and D. Valota, "A note on Drastic Product logic," in *Information Processing and Management of Uncertainty*, ser. Communications in Computer and Information Science, vol. 443. Springer, 2014, pp. 365–374.

[27] S. Aguzzoli, S. Bova, and D. Valota, "Free weak nilpotent minimum algebras," *Soft Computing*, vol. 21, no. 1, pp. 79–95, 2017.

DOI: 10.15439/2020F92
ISSN 2300-5963 ACSIS, Vol. 21

BoostSole: Design and Realization of a Smart Insole for Automatic Human Gait Classification

Badis Djamaa
Computer Science Department
Ecole Militaire Polytechnique
Algiers, Algeria
badis.djamaa@gmail.com

Med Messaoud Bessa
Computer Science Department
Ecole Militaire Polytechnique
Algiers, Algeria
messaoudbessa@gmail.com

Badreddine Diaf
Computer Science Department
Ecole Militaire Polytechnique
Algiers, Algeria
diafbadreddine1@gmail.com

Abdenebi Rouigueb
Computer Science Department
Ecole Militaire Polytechnique
Algiers, Algeria
rouigueb.abdenebi@gmail.com

Ali Yachir
Computer Science Department
Ecole Militaire Polytechnique
Algiers, Algeria
ali.yachir@gmail.com

***Abstract*—This paper presents BoostSole; a smart insole based system for automatic human gait recognition. It consists of a smart instrumented insole connected to the cloud via the patient's smartphone using low-power wireless communication. First, the design of BoostSole is introduced with discussions of sensors choice, placement, calibration, and data communication. Next, an adaptive multi-boost classification algorithm is deployed to accurately identify different gait patterns. The algorithm is fast and lightweight and can be implemented in ordinary smartphones with a small footprint in terms of computational requirements, energy consumption, and communication usage. Raw and on-device classified data can be securely uploaded to a distant cloud server for continuous monitoring and analysis. Indeed, they can be visualized and exploited by doctors to identify/correct walking habits and assess the risks of chronic pain associated with an abnormal walk. The system has been evaluated on a dataset containing three gait patterns, namely: shuffle walk; toe walking; and normal gait. Obtained results are promising with more than 97% classification accuracy accompanied by low response time and computational demands.**

***Index Terms*—Smart insole, Human Gait Analysis, Force Sensing Resistors, MultiBoost Classification, Internet of Things.**

I. Introduction

WALKING is a fundamental movement of the human body, which has a direct impact on its health. Indeed, a simple abnormality in walking can cause serious health problems that can range from simple pain to the loss of the walking ability. This is why gait analysis is very important for assessing human health. Indeed, such an analysis allows the evaluation and diagnosis of walk abnormalities before medical interventions. It also makes it possible to monitor surgical procedures and rehabilitation of patients from interventions that can affect their ability to stand or walk.

In the past, gait analysis was conducted using subjective methods, which are essentially based on the observations of specialists under clinical conditions. Indeed, the various parameters related to a patient's gait are observed, noted, and evaluated by the specialist while he is walking on a predetermined circuit. Now, advances in new technologies have given rise to devices and techniques allowing an objective, automatic, and fast assessment of different gait parameters. Thus, allowing more effective measurement and providing specialists with a large amount of reliable information on patients' gaits. This reduces the cost and the margin of error caused by subjective techniques.

Such technological devices can be classified into two different approaches: those based on Non-Wearable Sensors (NWS); and those relying on Wearable Sensors (WS). NWS systems, generally based on image processing and ground sensors, require the use of controlled stations where the sensors are located and capture walking data while the subject is moving on a marked walkway. Their main advantage is in liberating the subject from any constraints, but they are too expensive and might not capture real-world gait characteristics. On the other hand, WS systems make it possible to analyze data outside the laboratory and to capture information on human walks during their daily activities. Indeed, such sensors can be placed on different parts of the patient's body, such as the feet, knees, or hips, to measure relevant gait characteristics. However, WS may be constraining as they must be worn by the subject.

From the WS class, footwear systems stand as a non-obstructive method that addresses most of the issues related to wearable systems while preserving their advantages. For instance, footwear systems only require to instrument the insole/shoe with non-obstructive invisible sensors such as flexion sensitive sensors, force-sensitive sensors, and inertial measurement units [1]. The patient will wear the instrumented shoe/insole similarly to an ordinary one. Moreover, footwear systems are generally more accurate and lower cost when compared with other WS and NWS systems. Furthermore, they can serve other needs such as preventing foot ulcers in diabetics and detecting falls in the elderly.

In this work, a prototype of a smart insole based system for human gait recognition and classification, dubbed

BoostSole, is developed. BoostSole comprises a low-cost, low-power instrumented insole that continuously acquires gait data and transfers it, via low-power wireless communication, to the patient's smartphone for on-device analysis before being reported to the physician for decision making. The aim is to provide a low-cost, reliable, and time-efficient decision support system for physicians to identify and classify human gait in real-world scenarios in a non-obstructive way. More particularly, this paper provides the following contributions:

- Design, conception, and realization of BoostSole along with sensors choice, placement, and calibration.
- Development of BoostSole smartphone and desktop software applications as well as data acquisition, processing, and decision support processes.
- Extensive performance evaluations of BoostSole with a multitude of machine learning algorithms to classify three gait types under different performance metrics including accuracy, time efficiency, and lightweight aspects.

The remainder of this paper is organized as follows. Section II presents and discusses related work. The architecture of the BoostSole system is presented in Section III, while the design, choice, placement, and calibration of sensors are the object of Section IV. Section V is devoted to detailing the communication, feature extraction, and software components of the BoostSole system. This is followed by extensive performance evaluations of BoostSole for classifying three gait types (shuffle, toe, and normal) using a multitude of machine learning algorithms in Section VI. The paper ends in Section VII with conclusions and ideas for future directions.

II. Related Work

In the last few years, many research works have used footwear sensors for human gait analysis. [2]–[7] are examples of such research. Overall, there is a big similarity in the type of sensors used in these works, with some exceptions in the number and the placement of the sensors. Differences, mainly, reside in the artificial intelligence algorithms used to classify human gait for identification, activity recognition, and/or injury/fall detection and prevention.

For instance, [8] uses hidden Markov chains to detect the phases of the human's gait, [9] used Support Vector Machine (SVM) techniques to classify three types of walks, and [10] analyzed their data using Principal Component Analysis (PCA). Besides, [2] also used PCA to analyze their data and classify three types of walking: normal, toe, and dragging foot walking. PCA results showed a similarity between dragging foot walking and normal walking.

Recently, the authors of [11] used the AdaBoost tree classifier for gait asymmetry detection with smart insole attending an accuracy of 89.9%. On the other hand, [12] used Deep Convolution Neural Network (DCNN) to classify seven (07) types of gait: walking, fast walking, running, stair climbing, stair descending, hill climbing, and hill descending with an accuracy of more than 90%. Finally, in [13], the authors used a commercial "FootLogger" smart insole to classify seven (07) types of gait with Null-Space Linear Discriminant Analysis (NLDA), and they found that the larger the number of steps of a sample, the higher the classification performance becomes. Table I summarizes some of those work with a focus on the classification method used and walking phases detected. Besides, the table also presents the type of used sensors along with the hardware and software costs of the considered works.

Different from the above, the presented system is designed to be low-cost, lightweight, resource-lean, and time-efficient while providing reasonable accuracy in abnormal walk identification and classification. In the sequel, we will discuss the system architecture, the sensors used, and their placement.

III. System Architecture

The architecture of our system is depicted in Fig. 1, which is made of five main components, namely: (1) the low-cost instrumented sole; (2) patient's smartphone; (3) physician's working station; along with (4) local and (5) remote communication bridges.

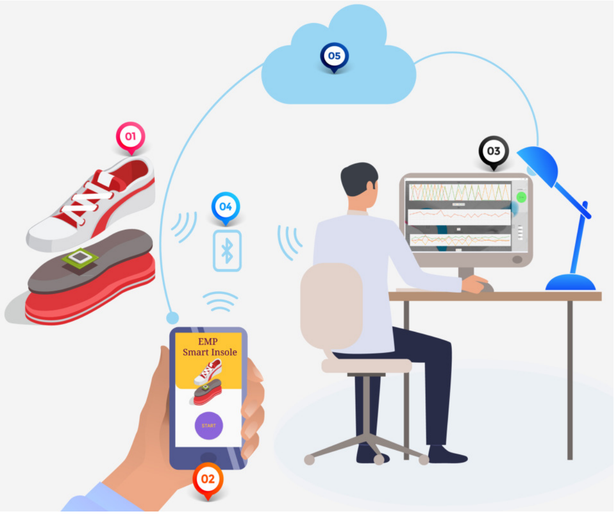


Fig. 1: System architecture

The first component is the main element of the BoostSole system. It consists of a smart insole equipped with a multitude of miniaturized sensors of force, flexion, and IMUs continuously collecting gait characteristics. Indeed, such sensors can measure many parameters that characterize walking, such as the timing of the heel strike and detachment of the foot, dorsi/plantar flexion, step length, and walking speed.

This data will be transmitted to patients' smartphones via low-power wireless communications where it will be processed and analyzed with lightweight on-device machine learning and visualized via mobile applications which can be used anywhere and at any time. Raw and on-device processed data can be then transmitted to treating physicians via secure remote communications as can be seen from Fig. 1. The remote application (component 3 in Fig. 1) can visualize and analyze the gait further so to help the physician decide by comparing the obtained results with a reference.

TABLE I: Related work summary and paper contributions

Ref.	Walking phases/types	Classification method	FSR	IMU	Flex	Hardware cost	Software cost	Observation
[12]	- Walking - Fast walking. - Running. - Stairs Ascending/Descending. - Hill Climbing/Descending	DCNN	Yes	Yes	No	High	High	Used ”FootLogger” smart insole with a classification accuracy of more than 90%.
[14]	- Walking. - Sidestepping. - Jumping. - Kicking. - Squatting.	DNN	Yes	Yes	No	High	Medium	A system able to predict the movement of the lower body.
[15]	- Walking. - Running. - Stairs Ascending/Descending.	SVM	Yes	Yes	No	High	High	An accuracy of 99.8% in the recognition of daily living activities.
[13]	- Heel strike. - Foot flat. - Mid stance. - Heel off - Toe off. - Mid swing. - Late swing.	NLDA	Yes	Yes	No	High	High	Used ”Footlogger” smart insole to classify seven types of the gait cycle.
[16]	- Heel strike. - Stance. - Heel off. - Swing.	None	Yes	Yes	No	High	Low	”FootMov” system detects gait phases using a developed algorithm.
[2]	- Normal walking. - Tip-toe walking. - Dragging foot walking.	PCA	Yes	No	Yes	High	Low	Can classify three types of walks using ZigBee for communication and PCA for recognition.
This work	- Normal walking. - Tip-toe walking. - Shuffle walk.	MultiBoostAB with Random Forest	Yes	Yes	Yes	Low	Low	Can classify three types of walks with a lightweight algorithm that can be implemented in a smartphone.

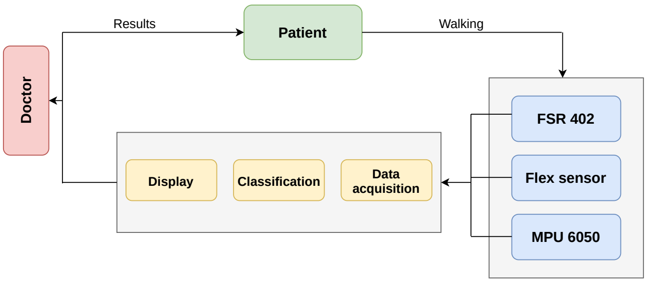


Fig. 2: System’s components and processes.

The processes and functionalities realized by the main components of our architecture are depicted in Fig. 2. Thus, the smart insole acquires and transmits data to local patient’s gadgets, while the smartphone and/or the desktop application is/are responsible for pre-processing, classification, and visualization by the patient and/or the physician.

IV. Design and Realisation of BoostSole

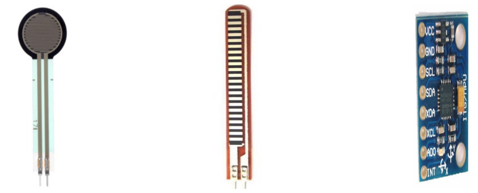
The first step in designing BoostSole was to choose the appropriate sensors, to create a highly instrumented low-cost sole capable of reliability detecting gait parameters. The following subsections detail the choice, role, and number of sensors along with their locations and calibrations.

A. *Choice, role, and number of sensors*

BoostSole relies on several sensors to capture fundamental human gait parameters. In our design, the focus is on using the minimum number of sensors that allows identifying human gait. The main used sensors are described below.

1) Force sensor: Force sensor or Force Sensing Resistor (FSR), is a robust device made of thick polymer film, which exhibits a decrease in resistance with the increase in the force applied to the sensor surface. This force sensitivity is optimized for use in human touch control of electronic devices such as automotive electronics, medical systems, and in industrial and robotic applications [4]–[7], [17].

We chose to use FSR 402, shown in Fig. 3a), because of its very miniature size (0.45mm), its simplicity, and ease of integration. Its robustness is up to 10 million actuation with a low activation force of 0.1N and a sensitivity of up to 10N. Besides, it is low-cost, ergonomic, and fits well to measure the pressure applied to the bottom of the foot. Furthermore, the combination of several FSRs can be applied to find the center of force beneath the foot. The number of FSRs depends on the accuracy required by the application and the cost of the developed prototype.



(a) FSR 402 (b) Bend Sensor (c) MPU 6050

Fig. 3: Used sensors

2) Bend sensor: A bend or flex sensor, shown in Fig. 3b, is a sensor that measures the bending angles. The resistance of the elements of the sensor increases by bending. The more the sensor is bent, the more it tends towards an infinite resistance (open circuit). Since the resistance is directly proportional to the curvature, it is used as a goniometer and is often called a flexible potentiometer. It is also used in a multitude of other domains including, rehabilitation, physical activities, machines, measuring tools.

The substrate of the bending sensor, which is produced from ink, carbon, or graphite [18] plays an important role in its performance. In our system, the bend sensor is used to measure the flexion angles below the foot depending on gait type and phases. We have used a flex sensor of 2.2” height, which can give values between 45 and 15 KOhms depending on the curvature radius, which is enough for BoostSole.

3) Inertial measurement unit: Inertial measurement units, generally, comprise an accelerometer and a gyroscope and can be attached to a mobile or any other object. The accelerometer can measure the linear acceleration along one or 3 orthogonal axes. On the other hand, when one seeks to detect a rotation or angular speed, the gyroscope is used. These sensors are pervasively used in a multitude of applications including games, gesture recognition, location-based services, movement-based game controllers, 3D remote controls for digital TVs, and portable sensors for health, fitness, and sports.

In this project, we used an MPU 6050 module (Fig. 3c), which combines a 3-axis gyroscope and a high-precision 3-axis accelerometer to form an inertial unit calculating acceleration and angular speeds of a human gait. Table II summarises the main characteristics of each sensor, their number, and their unit prices in the market. It can be observed from this table that the realization of a gait analysis support system can be low-cost compared to its usefulness and its reliability.

B. Sensors’ placements and BoostSole prototype

This section details the sensors’ placements and presents the realized prototype.

1) Sensors’ placements: Once the choice of sensors is made, the emphasis is on choosing the right locations to place them beneath the foot. In this prototype, presented in Fig. 4, the 03 force sensors are placed under the toe, between the toe and the middle of the foot, and under the heel to capture the movements made by the patient. This is justified by the fact that a human being when walks, his weight is generally distributed on three essential points on the foot. These points are the toe, the heel, and the place between the middle and the toe [19]. The flex sensor is placed in the middle of the foot to calculate angles. Remains, the last sensor, which is MPU 6050. This latter is fixed behind the foot to capture the translations with the angular velocities during feet movements. It is put in the microcontroller unit detailed below.

2) Wiring BoostSole: Based on these locations and the sensors seen in the previous sections, we created the first prototype of BoostSole, illustrated in Fig. 5, by wiring them to a microcontroller unit. The microcontroller brings together the essential elements necessary for wiring and reading sensor data such as micro-controller, memory, peripheral units, and input interfaces. The realization of the prototype was made using Arduino UNO; a well-known low-cost system-on-chip.

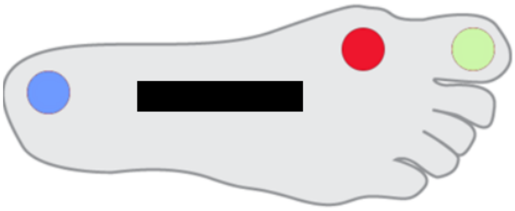


Fig. 4: Sensors’ locations

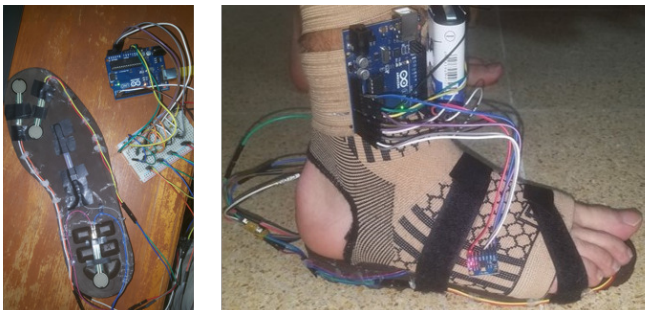


Fig. 5: The first prototype.

C. Sensors’ calibrations

To be correctly used, the sensors must be calibrated. This section explains how we calibrated the flex sensor, FSRs, and MPU 6050 to obtain correct values.

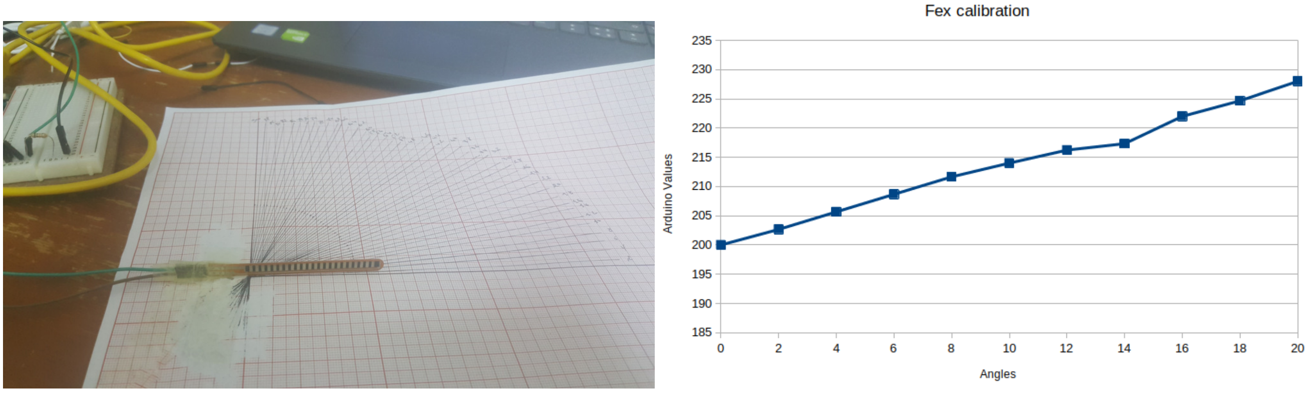
For the flex sensor, we proceed as follows. First, we draw a semi-circle on a paper and draw angles from 0° to 180° by a step of 2° as can be seen from Fig. 6a. Next, we have interfaced the flex sensor with Arduino and fixed it in the prepared paper. Then, we bend the sensor at each angle and note the value given in the Arduino IDE. This experiment has been repeated multiple times. Finally, we draw a graph representing the table containing values obtained from Arduino and the values of the real angles. Fig. 6a presents the flex sensor calibration process, and Fig. 6b presents a portion of the calibration data plotted in a graph. The values represented in this figure are for angles from 0° to 20°. As can be seen from this figure, flex values show a linear relationship with measured angles that can reliably be exploited for gait analysis.

For FSR calibration, we proceeded similarly to the flex sensor, but in this case, we applied different weights and read the FSR values on Arduino IDE. After that, we drew a table similar to that of the flex. The details of this calibration process are alike those of [20]. Obtained results are in a concordance with the conclusions of [20] and show a linear relationship between FSR data and weights up to a value of 80Kgs.

Finally, for the calibration of the MPU 6050, we calculated the average of the first 1000 values and then subtract this value from the values read by Arduino. these relative values were used. It should be noted, however, that the values of the MPU 6050 diverge quickly, so we need to re-calibrate the sensor periodically to get correct measurements.

TABLE II: Sensors' characteristics

	IMU	FSR	Flex sensor
Trade name	MPU 6050	FSR 402	FS
Number	01	03	01
Price	\$ 5.98	\$ 8.67	\$ 12.30
Uses	- Angles between the two legs.	- To calculate pressure under the insole.	- To calculate angle between the insole and the ground.
Dimensions (mm)	15.6*20.3*2.5	18.28*56.33*1.25	6.35*112.24*0.43
Life cycle	12 months	10 millions values	>1 million values
Temp range	-40°C to +105°C	-30 - +70 ℃	-35°C to +80°C



(a) Process

(b) Results

Fig. 6: Flex sensor calibration.

V. Data communication, classification and, visualization

Once the prototype is completed, we focused on the other parts of the architecture, namely: sending data to a processing station, classifying movements, and displaying the results.

A. Data communication

Communication plays an important role in the design of a smart insole. Indeed, besides being the key component in ensuring reliable transmissions of gait data from the embedded microcontroller to the processing station, it is crucial to the system's energy consumption and hence on its lifetime.

Today, a multitude of wireless communication technologies exist in the market. Each has its applications, advantages, and drawbacks as can be seen from Table III. Thus, while WiFi-based solutions are very pervasive, they consume much energy making their lifetime in hours, which does not fulfill the requirements of boostSole. On the other hand, IEEE 802.15.4 solutions provide better energy consumption that fulfills the requirements of BoostSole, but they are not pervasive and are not available in ordinary smartphones/PCs, which limit their applicability. Bluetooth Low-Energy (BLE) has the advantages of both, making it an important candidate for the BoostSole prototype. To do so, we have chosen the HM-10 BLE module, which implements Bluetooth 4.1 specification. Indeed, it provides reliable communication by channel hopping to avoid interference with co-existing networks, along with high throughput for capturing sensor data. Furthermore, the module goes into sleep automatically when no data activity is detected. Besides, it can be integrated into Arduino via a serial link. Finally, it should be noted that the pairing between the processing station and BoostSole is initiated by the station allowing the insole to start sending gait data just after pairing.

TABLE III: Low-power wireless communication technologies

	Bluetooth	WiFi	ZigBee	BLE	Z-Wave
Standard	802.15.1	802.11n	802.15.4	802.15.1	G.9959
Frequency	2.4 GHz	2.4/5 GHz	2.4 GHz	2.4 GHz	868 MHz
Topology	star	star	star, mesh	star, mesh	mesh
Data rate	2 mbps	100 mbps	250 kbps	1 mbps	40 Kbps
Range	15-30 m	10-100 m	10-100 m	15-30 m	30-100 m
Battery	Months	Days	Years	Years	years
Pervasive.	Yes	Yes	No	Yes	No

B. Feature extraction and classification

The raw data captured by the sole are transmitted via BLE to the patient's Smartphone for on-device gait analysis. Before being processed, the acquired data will be segmented. Fig. 7 details the feature extraction process. As can be seen from this figure, the input signals $S1, ...Sn$ are respectively discretized into $Y1, ...Yn$ sequences, where each $Yi(j)$ represents the mean of the j^{th} interval of Si (Fig. 7). Such sequences are, then, segmented with a sliding window procedure, where a fixed-length window W is shifted along the signal sequence for frame extraction. Consecutive frames usually overlap to some degree (less than 50%). In the end, a set of vectors of size $n * |W|$ are generated.

For classification, we deploy a supervised learning approach. Thus, the generated vectors along with the labels provided by experts are fed to a supervised machine learning algorithm for training as can be seen from Fig. 8. In order to

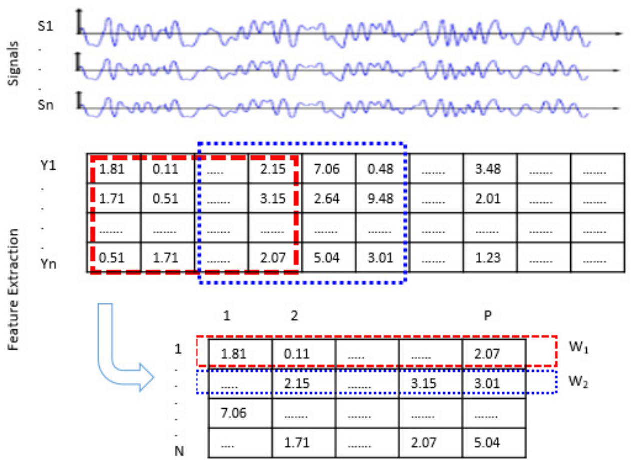


Fig. 7: Feature extraction

select the best classifiers, a number of well-known algorithms including SVM, kNN, decision trees, and ensemble classifiers will be evaluated in terms of accuracy, time, and complexity.

The chosen classifiers are known to be powerful with a high capacity of generalization. For instance, SVM belongs to the kernel-based family which aims to fit an optimal hyperplane to accurately classify both linearly separable and linearly inseparable data [21]. kNN is a non-parametric method used for classification and regression. Boosting classifiers such as Bagging, Boosting, AdaBoost, and MultiBoostAB are a type of meta-algorithms that use decision trees and discriminant analysis learners to improve the classification. Their main idea is to boost weak classifiers. Multi-boosting [22] is a representative sophisticated algorithm of this class. It is an extension to the AdaBoost with Wagging.

The trained models will be used to classify feature vectors extracted from a given test signal. Then, a majority vote can be performed to predict the gait class of the signal (Fig. 8).

C. *BoostSole software application*

We developed both a desktop and an android application.

1) Desktop application: The desktop application is developed using JavaFX. Before it shows up, the application must first connect with the insole. After that, the user can see a graphical user interface that contains three (03) empty charts, one is for the three FSR sensors, the second is for the flex sensor, and the last one (at the bottom) is for MPU 6050. At the right, there is a start button to choose the walking period (30 seconds, 1, 2, 5, and 10 minutes). When the user clicks on that button, the signals acquired from each sensor are visualized in their corresponding places and stored in a specified path as can be seen in Fig. 9. At the end of the walking period, the application stops plotting the data and the classification results can be displayed.

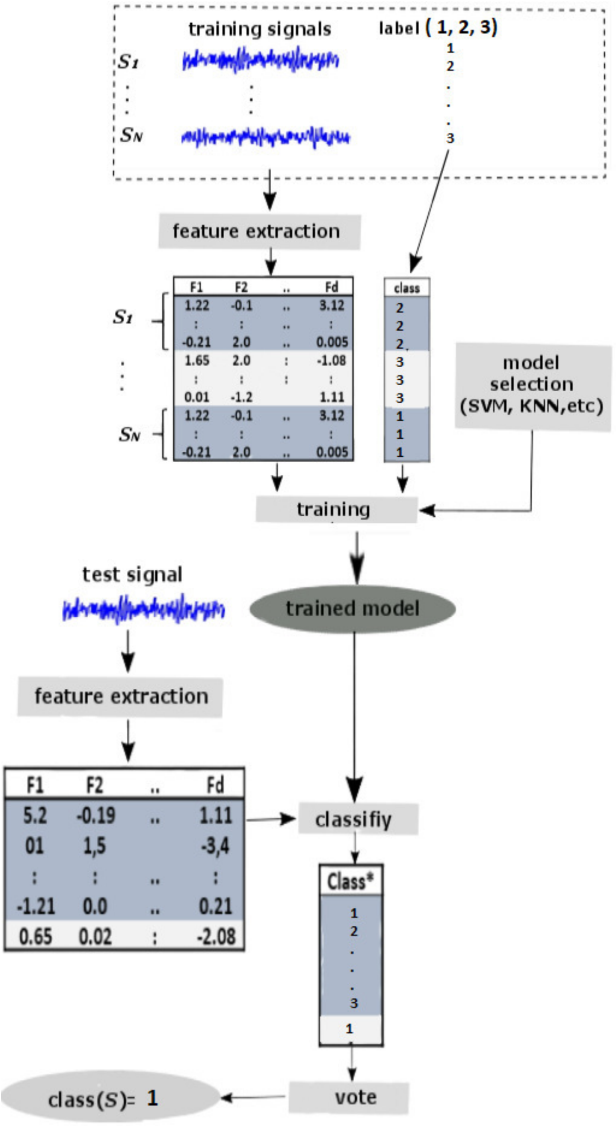


Fig. 8: Classification process

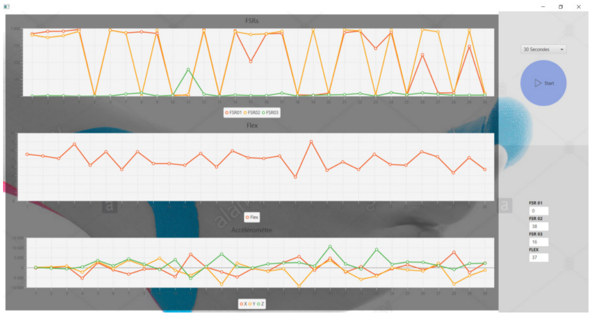


Fig. 9: The desktop application

2) Android application: The Android application is developed using Android Studio. It allows a user to connect with the BootSole using BLE, and visualize the pressure sensors data in a Heat-Map. We used three Android activities: the first starts the communication, the second visualizes the pressure map, and the third analyzes and displays gait recognition results.

VI. Performance Evaluation

This section evaluates the performance of the proposed system. We start by describing the dataset, methodology, and metrics before discussing the obtained results.

A. Experimental dataset

To access the system, we collected data from 5 healthy volunteers (age [y]: 23.5 ± 1.3; height [m]: 1.77 ± 0.08, weight [kg]: 78 ± 5). All walking sequences were tracked using BoostSole. A single sensing unit was captured with a $1Hz$ sampling rate in order to save energy. All the recorded data was sent via BLE to a laptop placed in close proximity to the participant. The volunteers performed a continuous sequence of three walking types, namely: shuffle walking (class 1); normal walking (class 2); and toe walking (class 3). For each one, the volunteers walked for 30s, which make it a 90s total. Fig. 10 presents a screenshot for a representative gait data collected for each type by the desktop application.

B. Evaluation methodology and metrics

In our evaluations, six (06) classifiers were considered, namely SVM, kNN (k = 5), Stacked, Random Forest (RF), MultiBoostAB with RF (MB-RF), and MultiBoostAB with Logistic Model Tree (MB-LMT). The 5-fold cross-validation method is followed to evaluate the accuracy of the aforementioned classifiers under different window lengths (from 1s to 10s). Also, in classification, we have only used the data collected from FSRs and bend sensors to assess their ability to distinguish walk patterns. All main results were obtained using an i7-8750H @2.20GHz, 16 Go RAM, and a GTX-1050 4Go GPU. Average accuracy, precision, recall, F-measures, and Receiver Operating Characteristic Curve (ROC) Areas were used to evaluate the effectiveness of the involved classifiers. They are measured as follows:

- $Accuracy$: the ratio of number of correct predictions to the total number of input samples.

$$Accuracy = \sum_{c} \frac{TP_c+TN_c}{TP_c+TN_c+FP_c+FN_c}, c \in classes \quad (1)$$

- $Precision$: An average per-class agreement of the data class labels with those of a classifiers.

$$Precision = \sum_{c} \frac{TP_c}{TP_c + FP_c}, c \in classes \quad (2)$$

- $Recall$: Average per-class effectiveness of a classifier to identify class labels.

$$Recall = \sum_{c} \frac{TP_c}{TP_c + FN_c}, c \in classes \quad (3)$$

- F-$Measure$: The harmonic mean of the macro-average precision and recall.

$$F\text{-}Measure = \frac{2 * Precision * Recall}{Precision + Recall} \quad (4)$$

- $Receiver\ Operating\ Characteristic\ Curve\ Areas$ (ROC areas): It represents the area below the plot of the true positive rate against the false positive rate. It shows the trade-off between sensitivity and specificity.

Furthermore, the best models were deployed on an android device (Samsung S7-Edge) in order to evaluate their average response time, memory, CPU, and battery usage.

C. Results and discussions

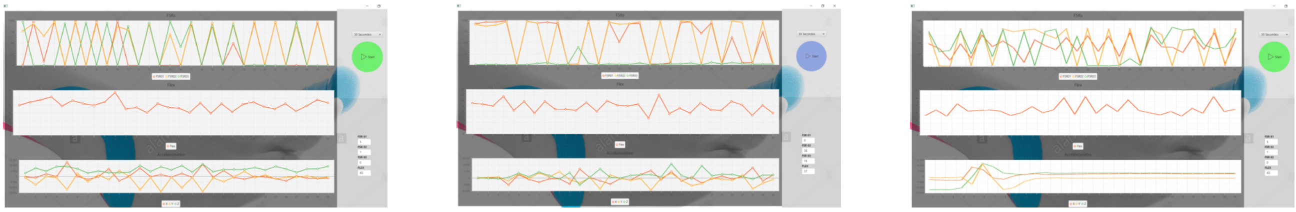
1) Comparison of classifiers: This experiment aims to find a suitable classifier and window length for the BootSole system. Table IV shows the average accuracy for different classifiers under different window lengths. Overall, the accuracy of all algorithms increases with increasing window length up to a window of 8s. When it comes to individual classifiers, RF and MB-RF have achieved better accuracy compared to SVM, kNN, MB-LMT, and stacked classifiers. Indeed, average accuracies of above 95% were observed as soon as a window of 3s for both RF and MB-RF. Besides, MB-RF achieved an accuracy of about 97% in a 4s window length. Furthermore, this classifier reached almost 100% accuracy for 7s window length. For the sake of time, energy, and computational resources, a 4s window is used.

In addition to the accuracy results, MB-RF has shown the smallest test response time, which makes it very promising for classifying walking types using BootSole. Before embedding it in the Android application, we will get a closer look at MB-RF in the following section.

2) A detailed evaluation of MB-RF: Table V shows different evaluation metrics obtained for the MB-RF classifier. On average, a precision value of around 0.969 has been recorded, which allows MB-RF to predict correctly the positive observations to the total predicted positive observations. A similar value has also been registered for recall, allowing MB-RF to classify positive observations w.r.t. the observations in the actual class with 96.8%. Besides, the confusion matrix, given in Table VI, shows that the toe and shuffle classes are well discriminated, whereas, signals from the normal walk are slightly hard to be correctly classified. To confirm such results, we have drawn ROC curves and assessed the area under ROC. All the curves start on the left-hand border and then follow the top border of the ROC space, which justifies the results presented in Table V. Indeed, MB-RF recorded a 0.989 average ROC area (0.988 for the shuffle, 0.982 for normal, and 0.995 for toe walking) in a 4s window length.

3) MB-RF resource consumption on Android: By giving the best results, MB-RF is a promising classifier for BoostSole. However, before embedding it in handheld devices, its resource consumption needs to be examined. To do so, we have conducted a new battery of tests on a smartphone. To put results into context, we have compared MB-RF with the two following best classifiers that shown accuracy around 95% in the 4s window length (SVM and RF). The three models (trained with the Weka software) were deployed on an android device (Samsung S7-Edge). The average response time along with memory, CPU, and battery usages are reported.

Table VII presents the Average Response Time (ART) and the memory usage of the three algorithms. It is clear from this

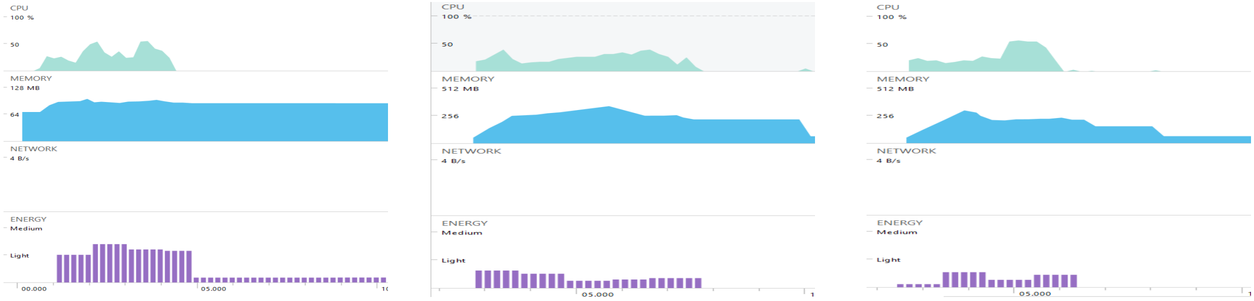


(a) Normal Walk (b) Toe Walk (c) Shuffle Walk

Fig. 10: Data visualisation

TABLE IV: Average accuracy of different classification algorithms using different window lengths

Win. (s)\classifiers	SVM	RF	MB-RF	MB-LMT	Stacked	kNN
1	72.16%	79.06%	78.62%	75.28%	64.14%	69.47%
2	90.40%	91.74%	91.74%	90.63%	83.89%	87.45%
3	93.30%	**95.09%**	**95.54%**	91.07%	91.51%	87.91%
4	94.62%	**96.41%**	**96.86%**	93.27%	89.26%	87.84%
5	93.96%	91.28%	91.28%	89.93%	89.91%	85.82%
6	93.92%	94.59%	95.27%	93.24%	88.39%	85.57%
7	**97.30%**	**100%**	**100%**	93.69%	91.21%	84.62%
8	**98.20%**	**100%**	99.10%	95.50%	89.67%	87.45%
9	97.75%	94.38%	98.88%	94.38%	87.39%	86.27%
10	97.73%	97.73%	96.59%	96.59%	67.71%	80.86%



(a) SVM (b) Random Forest (c) MultiBoostAB-RandomForest

Fig. 11: Classifiers resource consumption in Android

TABLE V: Metrics of MultiBoostAB-RF for 4s window length

Class	Precision	Recall	F-Measure	ROC Area
Shuffle Walking	0.961	0.987	0.974	0.996
Normal Walking	0.986	0.920	0.952	0.988
Toe Walking	0.961	1.000	0.980	0.998
Weighted Avg.	0.969	0.969	0.968	0.994

TABLE VI: Confusion matrix

Class	Shuffle Walking	Normal Walking	Toe Walking
Shuffle Walking	0.9867	0.0133	0.0000
Normal Walking	0.0400	0.9200	0.0400
Toe Walking	0.0000	0.0000	1.0000

table that MB-RF has a better response time and less memory storage. With regards to the CPU, memory, and battery usage, Fig. 11 shows the results obtained with the android profiler tool in the onStart method. It is clear from this figure that SVM consumes a lot of resources (high battery and memory consumption with almost 50% extra CPU usage). For RF, we can see a bit more memory and energy usage than MB-RF.

TABLE VII: Average response time and memory usage

	MultiBoostAB-RF	RandomForest	SVM
ART (ms)	0.458893563	1.611349833	0.727204667
RAM (MB)	1.3	1.8	1.9

VII. Conclusion and Future Work

In this paper, a smart insole based system for automatic human gait analysis, dubbed BoostSole, was proposed. The aim was to develop a low-cost, objective, and reliable system to help physicians in continuous analysis of walk patterns. Obtained results demonstrated the capacity of BoostSole to provide accurate rates while consuming fewer resources. Nevertheless, BoostSole can be enriched by adding more sensors to be able to distinguish a wider array of gait types across heterogeneous populations. Furthermore, using flexible and miniaturized chips can make BoostSole more practical and ergonomic. Moreover, while the local communication might not require strong security, remote communications with the server need to be investigated for proper security and less bandwidth consumption. Finally, we are planning to expertise BoostSole with health specialists. Investigating other on-device learning algorithms such as deep federated learning is also planned.

References

[1] A. Muro-de-la Herran, B. Garcia-Zapirain, and A. Mendez-Zorrilla, "Gait Analysis Methods: An Overview of Wearable and Non-Wearable Systems, Highlighting Clinical Applications," *Sensors (Basel, Switzerland)*, vol. 14, pp. 3362–3394, Feb. 2014.

[2] W. Donkrajang, N. Watthanawisuth, J. P. Mensing, and T. Kerdcharoen, "Development of a wireless electronic shoe for walking abnormalities detection," in *The 5th 2012 Biomedical Engineering International Conference*, pp. 1–5, Dec. 2012.

[3] S. J. Morris, *A shoe-integrated sensor system for wireless gait analysis and real-time therapeutic feedback*. PhD thesis, Massachusetts Institute of Technology, 2004.

[4] S. Bamberg, A. Benbasat, D. Scarborough, D. Krebs, and J. Paradiso, "Gait Analysis Using a Shoe-Integrated Wireless Sensor System," *IEEE Transactions on Information Technology in Biomedicine*, vol. 12, pp. 413–423, July 2008.

[5] Hyejeong Nam, Jin-Hyun Kim, and Jee-In Kim, "Smart Belt : A wearable device for managing abdominal obesity," in *2016 International Conference on Big Data and Smart Computing (BigComp)*, (Hong Kong, China), pp. 430–434, IEEE, Jan. 2016.

[6] A. De Santis, E. Gambi, L. Montanini, L. Raffaeli, S. Spinsante, and G. Rascioni, "A simple object for elderly vitality monitoring: The smart insole," in *2014 IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, (Senigallia, Italy), pp. 1–6, IEEE, Sept. 2014.

[7] W. Donkrajang, N. Watthanawisuth, J. P. Mensing, and T. Kerdcharoen, "A wireless networked smart-shoe system for monitoring human locomotion," in *The 4th 2011 Biomedical Engineering International Conference*, (Chiang Mai, Thailand), pp. 54–58, IEEE, Jan. 2012.

[8] J. Bae and M. Tomizuka, "Gait Phase Analysis based on a Hidden Markov Model," *IFAC Proceedings Volumes*, vol. 43, no. 18, pp. 746–751, 2010.

[9] G.-M. Jeong, P. Truong, and S.-I. Choi, "Activity classification of three types of walking regarding stairs using plantar pressure sensors," *IEEE Sensors Journal*, vol. PP, pp. 1–1, 03 2017.

[10] T. Nilpanapan and T. Kerdcharoen, "Social data shoes for gait monitoring of elderly people in smart home," in *2016 9th Biomedical Engineering International Conference (BMEiCON)*, (Laung Prabang, Laos), pp. 1–5, IEEE, Dec. 2016.

[11] S. Marquez J, R. Atri, M. R. Siddiquee, C. Leung, and O. Bai, "A Mobile, Smart Gait Assessment System for Asymmetry Detection Using Machine Learning-Based Classification," *Journal of Biomedical Engineering and Medical Devices*, vol. 03, no. 02, 2018.

[12] S.-S. Lee, S. T. Choi, and S.-I. Choi, "Classification of Gait Type Based on Deep Learning Using Various Sensors with Smart Insole," *Sensors*, vol. 19, p. 1757, Apr. 2019.

[13] S.-I. Choi, S.-S. Lee, H.-C. Park, and H. Kim, "Gait Type Classification Using Smart Insole Sensors," in *TENCON 2018 - 2018 IEEE Region 10 Conference*, (Jeju, Korea (South)), pp. 1903–1906, IEEE, Oct. 2018.

[14] W.-k. Tam, A. Wang, B. Wang, and Z. Yang, "Lower-body posture estimation with a wireless smart insole," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, (Berlin, Germany), pp. 3348–3351, IEEE, July 2019.

[15] D. Chen, Y. Cai, X. Qian, R. Ansari, W. Xu, K.-C. Chu, and M.-C. Huang, "Bring Gait Lab to Everyday Life: Gait Analysis in Terms of Activities of Daily Living," *IEEE Internet of Things Journal*, vol. 7, pp. 1298–1312, Feb. 2020.

[16] N. Carbonaro, F. Lorussi, and A. Tognetti, "Assessment of a Smart Sensing Shoe for Gait Phase Detection in Level Walking," *Electronics*, vol. 5, p. 78, Nov. 2016.

[17] R. Das and N. Kumar, "Investigations on postural stability and spatiotemporal parameters of human gait using developed wearable smart insole," *Journal of Medical Engineering & Technology*, vol. 39, pp. 75–78, Jan. 2015.

[18] G. Saggio, F. Riillo, L. Sbernini, and L. R. Quitadamo, "Resistive flex sensors: a survey," *Smart Materials and Structures*, vol. 25, no. 1, p. 013001, 2015.

[19] J. Pineda-Gutierrez, L. Miro-Amarante, M. Hernandez-Velazquez, F. Sivianes-Castillo, and M. Dominguez-Morales, "Designing a Wearable Device for Step Analyzing," in *2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)*, (Cordoba, Spain), pp. 259–262, IEEE, June 2019.

[20] A. Jor, S. Das, A. S. Bappy, and A. Rahman, "Foot Plantar Pressure Measurement Using Low Cost Force Sensitive Resistor (FSR): Feasibility Study," *Journal of Scientific Research*, vol. 11, pp. 311–319, Sept. 2019.

[21] A. Jović, K. Brkić, and N. Bogunović, "Decision Tree Ensembles in Biomedical Time-Series Classification," in *Pattern Recognition*, vol. 7476, pp. 408–417, Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. Series Title: Lecture Notes in Computer Science.

[22] G. I. Webb, "Multiboosting: A technique for combining boosting and wagging," *Machine Learning*, vol. 40, no. 2, pp. 159–196, 2000.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 45–48
DOI: 10.15439/2020F113
ISSN 2300-5963 ACSIS, Vol. 21

Concept Blueprints Serving More Focused User Queries

Kurt Englmeier
Schmalkalden University of
Applied Science, Blechhammer,
98574 Schmalkalden, Germany
Email: k.englmeier@hs-sm.de

***Abstract*—Information Retrieval is about user queries and strategies executed by machines to find the documents that best suit the user's information need. However, this need reduced to a couple of words gives the retrieval system (IRS) a lot room for interpretation. In order to zero in on the user's need many a IRS expands the user query by implicitly adding or explicitly recommending the users further useful terms that help to specify their information need.**

Queries often do not comprise more than a handful of terms, which, in turn, do not sufficiently represent the user's need. In this paper, we propose and demonstrate an approach that enables users to resort to implicitly more complex query expressions. We call these semantic structures concept blueprints. Furthermore, users have the possibility to define the blueprints on their own. The purpose of the blueprints is to spot more precisely the text passage that fits the user's information need.

I Introduction

INFORMATION Retrieval (IR) is the process of looking up documents that suit the information need of the user or, in other words, that are relevant to the query terms expressed by the user. The more detailed the search query, the better the retrieval results. Therefore, IRS usually encourage users to add further query terms from a list of recommended terms that may also address the context of their query. The recommended terms happen to appear together in texts in close proximity or have been selected together previously by other users presumably having the same information need.

By looking up documents whose content is best summarized by terms that match the query terms the IRS supposedly provides the user with the required information. The terms summarizing the document's content and the ones representing the query must be somehow similar.

A query "long-term consequences covid-19 infection" will quite likely lead us to the information we are looking for, because the query terms appear in one form or another (e.g. as synonyms) in the retrieved texts. We probably will be satisfied with the documents provided.

Things are slightly different with a user query "covid-19 infections Paris yesterday". We may get statistics about Covid-19 infections including detailed figures for Paris. If we are lucky, we find yesterday's figures for the French capital in one of the retrieved documents. However, many retrieval results may not mention this particular figure we are looking for. One may think, it's a bit strange to use the term "yesterday" in query. Our retrieval experiences tell us that this term may not be quite useful for a successful search.

In other situations, things are not so obvious. Querying Google about the "global average runtime of nuclear power plants" provides mainly statistical information that enables you to calculate the answer yourself. Your query results in useful data around the information you need, but it takes you a lot of time and effort to scan through all the documents provided and to produce the answer you require.

There is a useful document available (also on the web) answering exactly your question *in one of its paragraphs* (see figure 1). However, you won't find the corresponding document among the first thirty something retrieval results.

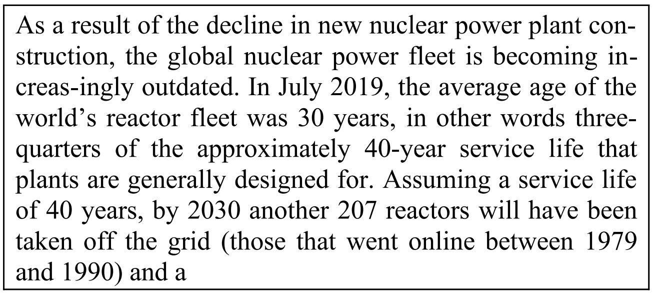

As a result of the decline in new nuclear power plant construction, the global nuclear power fleet is becoming increas-ingly outdated. In July 2019, the average age of the world's reactor fleet was 30 years, in other words three-quarters of the approximately 40-year service life that plants are generally designed for. Assuming a service life of 40 years, by 2030 another 207 reactors will have been taken off the grid (those that went online between 1979 and 1990) and a

Fig. 1. Section of a text and its representation after basic text patterns have been identified and accordingly annotated.

The problem results from the typical design of information retrieval processes. In short, all documents of the data source are indexed using the weighted index terms according to their relevance for the content of the *entire* document. User queries are matched against these index terms, and the documents with highest relevance values rank top in the result list provided to the user. The relevance value depends on the content of the entire document. The relevance of a single chapter in a document is blurred by the overall relevance value and term list of the document.

So far, the problem is well-known and barely spectacular. Search engines just work this way. In principle, text classification and text mining adopt the fundamental methods of information retrieval.

The work presented in this article reflects the current state-of-work of the research group of the Schmalkalden University of Applied Science. The prototype applies supervised learning for a semi-automatic approach to extract, distill, and standardize data from text. Even though the prototype shown here still represents work in progress, it demonstrates its potential in the detection of fake news and misinformation.

II. Related Work

Our approach is designed around the paradigm of fact retrieval emphasizing natural language [1, 2, 3, 4] and the support of users in constructing more complex search queries [5, 6]. It is based on a combination of Named Entity Recognition (NER), Bag of Words (BoW), and Word N-Grams [7, 8]. We assume that a specific combination of keywords and annotated numeric expressions uniquely reflects a particular fact.

We can imagine a variety of theme-specific BoWs (for locations, names, expressions of aggression etc.) applicable in our context together with Named Entities for common patterns in text reflecting time, amounts, distances, and the like. This process usually combines key words and common text patterns. Finally, each pattern is annotated by an appropriate term that summarizes the meaning of the pattern.

Generic named entities help to standardize factual information and to abstract away the different forms of expressions for essentially the same thing. However, it does not suffice just to annotate generic patterns. We can also easily imagine that Named Entities may relate to ontologies that serve specific interpretation or calculation purposes.

NER in the context described here operates with BoWs addressing locations, persons, organizations, or institutions (Wall Street, Dow Jones, White House, Bangladesh, for instance). Furthermore, we use key words such as “Mr.” or “Health Senator” that hint to names of persons. The system takes these names and feeds them into the respective bag of words.

There are further interesting key terms pointing to names. For example, the term “by” following the title of an article leads the list of names authoring that article. The identification of proper names benefits from the analysis of sequential dependencies when bags of words can be produced automatically instead of manually. There are promising approaches to automatically identify names (and other important key expressions) in texts using conditional random fields (CFR) [9] or hidden Markov Models (HMMs) [10]. Inclined to CFR, we integrated a feature that proposes, for example, all names starting with capital letters and followed by an abbreviation as organization names, such as National Institute of Health (NIH) or Korean Electric Power Corporation (KEPCO).

The identification of facts starts with information extraction [11] and the annotation of the extracted text pieces according to the meaning they express [12]. Annotation has two roles: first, it adds a meaningful term to the extracted text, in particular the numeric data. Such patterns, for example, represent dates, percentages, numerical data, distances, and the like. Second, the annotations (and keywords) from the first annotation are further annotated. This process (if iteratively performed) produces an increasingly more abstract representation of the text and numeric data in the text piece under consideration. Semantic markers [13] are the smallest fraction of a text covering a certain meaning discernable from the other fractions. Together they mark the meaning of a particular piece of text.

III. The Paradigm of Concept Blueprints

For a more fine-grained retrieval that spots only the most relevant text sections in all documents, the classic IR approach needs to be adjusted. Application areas of such a type of retrieval are finding and extracting particular facts from texts of a collection or locating text sections that are pertinent for a particular situation manifested in a balance sheet, service request, or claim. This form of information retrieval has a prominent place in Legal Technology (LT), for instance.

To serve such a request, we have to modify the classical information retrieval process and integrate additional functionalities adopted from fact retrieval:

- Retrieval on chapter or sentence level
- Extraction of relevant text sections
- Standardized and contextualized representation of facts
- Special consideration of numerical information
- Inclusion of basic inference mechanisms

The central element in our approach for a combination of text and fact retrieval is the blueprint of facts or concept blueprint. In its basic form the blueprint is a structure of terms where each slot may hold a single term (with or without its corresponding synonyms), an N-Gram, or a Named Entity. The meaning of a particular concept of a slot can be expressed by different terms, much like the type of numerical information (date, price, or growth rate, for instance) can be expressed in different syntactic forms. Each slot is represented by a title. The titles, in turn, represent the content of the slot on a more abstract level. A blueprint, thus, consists of a hierarchy of iteratively integrated slots. Each blueprint stands for a particular concept that is further detailed by its sub-components, that is, the slots on the different levels of abstraction. Each blueprint represents not only the semantic architecture of its concept or its meaning, but also the different syntactic facets its concept may take in texts.

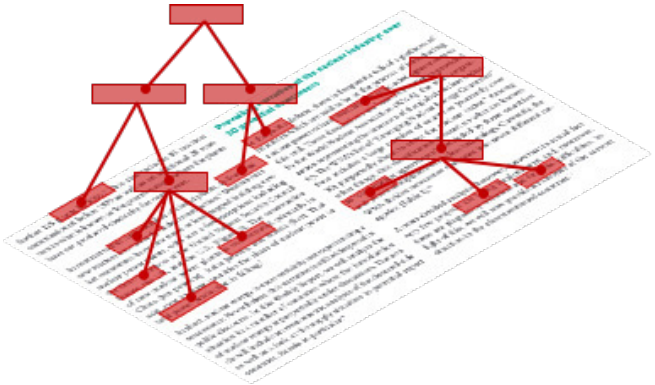


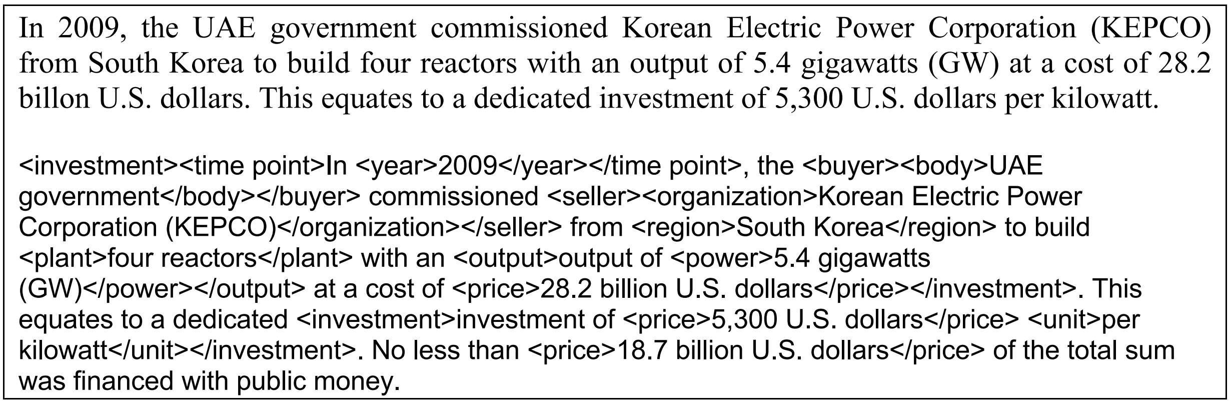
Fig. 2. Schema of a blueprint with its slots.

IV. Defining Concept Blueprints

Text Mining, in our approach, starts with seeds covering annotated definitions of basic text patterns. They include things like dates, distances, or prices. The next group of the seeds addresses proper names for locations, countries, persons, and the like. Our system design includes helper functions to detect proper names which, in general, pose a certain challenge for automatic text analysis. By applying CRF methods, we can determine these expressions. For instance, words starting with uppercase letters and immediately following special terms like "Premier ministre", "Mr.", or "the author" usually indicate that the following terms may be proper names of persons. Some BoWs (for countries, for instance) can also be imported from external sources.

Whenever a slot of a blueprint refers a specific term, all of its applicable synonyms need to be taken into consideration. However, not all possible synonyms are also applicable in every context. In an expression describing a certain amount of money like "to the tune of 12.65 billion U.S. dollars", none of the synonyms of the term "tune" is applicable in this context. In some occasions, it is thus recommendable to consider the applicability of synonyms on the level of N-Grams. Thorough N-Gram analysis reveals, that expressions like the one shown in the example have synonyms like "to the amount of" or "add up to".

Figure 3 shows the representation of a section of a text after the basic text patterns have been identified and annotated accordingly. All instances that meet the qualities of an expression representing a price are identified and marked by the blueprint `price=?"price".money.currency`. These instances are expected to be composed of an instance matching the slot (or subcomponent of the blueprint) (amount of) "money", a further one addressing the currency and an occasional (leading or trailing) word "price" (or a synonym expression such as "at a cost of"). An optional slot is indicated by a leading question mark. Key words are stated in quotes. Internally they are mapped to their standardized (stemmed) form. Terms without quotes thus refer to the blueprint slots. The dots stand for "close proximity" which can range from "immediately adjacent" to "neighboring blueprints spread over a phrase or paragraph".



In 2009, the UAE government commissioned Korean Electric Power Corporation (KEPCO) from South Korea to build four reactors with an output of 5.4 gigawatts (GW) at a cost of 28.2 billon U.S. dollars. This equates to a dedicated investment of 5,300 U.S. dollars per kilowatt.

<investment><time point>In <year>2009</year></time point>, the <buyer><body>UAE government</body></buyer> commissioned <seller><organization>Korean Electric Power Corporation (KEPCO)</organization></seller> from <region>South Korea</region> to build <plant>four reactors</plant> with an <output>output of <power>5.4 gigawatts (GW)</power></output> at a cost of <price>28.2 billion U.S. dollars</price></investment>. This equates to a dedicated <investment>investment of <price>5,300 U.S. dollars</price> <unit>per kilowatt</unit></investment>. No less than <price>18.7 billion U.S. dollars</price> of the total sum was financed with public money.

Fig. 3. Section of a text and its representation after basic text patterns have been identified and accordingly annotated.

Each set of slots is annotated by a title reflecting the concept or the overarching meaning of the slots. This title summarizes the content of all blueprint components on its underlying layer. It thus abstracts away the content details of the slot layer it stands for. Each such blueprint can be a slot in the next layer of abstraction.

By repeatedly applying this process the blueprint gets more layers and covers a growing text area. The blueprint then resembles a hierarchy with a general representation of covered text on its top and growing specialized representations towards its bottom.

Each single blueprint thus consists of a set of slots and its title. It forms an inseparable unit. The repeated pattern identification operates on the blueprint titles, the text sections that are so far not part of any instance of a blueprint.

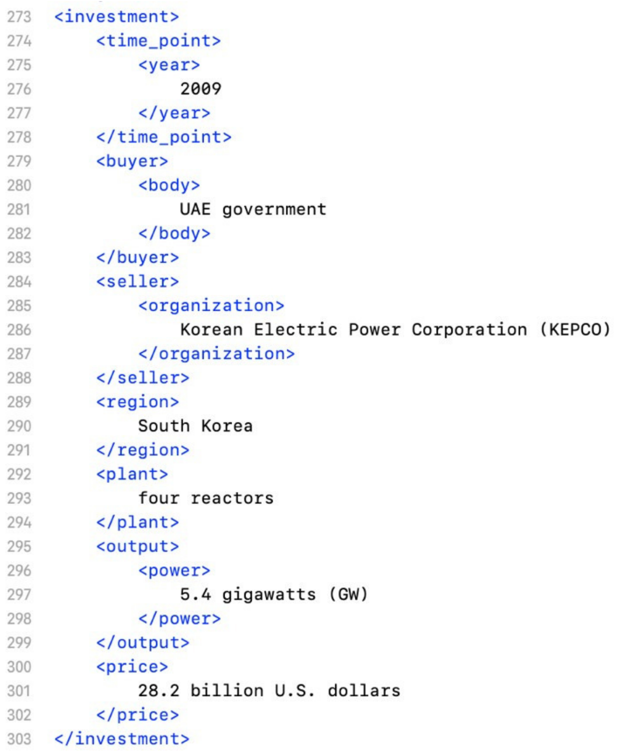
V. Conclusions and Outlook

This paper presents the state of work of the design and prototypical implementation of a fact retrieval system operating on concept blueprints that can be defined by the users. It uses Named Entity Recognition and theme-specific Bag of Words to identify semantic markers in text that point to the specific meaning of a text passage.

The application areas of the content schemas are manifold. The main purpose is identifying facts in texts and representing them in a distilled and standardized way in their respective context. This facilitates the comparison of representations of facts in different sources and, thus, supports the detection of fake news and misinformation.

Named entities and terms from BoWs identify the meaning words as they appear in a phrase or fragment of text. However, they also explicitly include numerical data that are very important for the correct reflection of meaning in text. Iteratively applying standardization to already extracted and annotated pieces of text creates semantic hierarchies which, in turn, reflect the meaning of terms in a more general or more detailed (or specified) context. This, in turn, makes text comparisons more precise and versatile.

Our approach and our prototype are still work in progress, but we already noticed that our content schemas have a certain proximity to ontologies. We use the schemas for text interpretation on a basic level and gradually produce concept hierarchies. However, we clearly see the necessity to add more functionality to schemas, in particular, when parts of the schema address factual (i.e. numerical) information. Quite often, calculations can be helpful to check the plausibility of statements based on numerical information. The standardized representation of facts enables the opportunity to include (at least some decent) inference mechanisms. The representation of extracted instances can be used to link data processing features to the slots of the blueprints.



```
<investment>
    <time_point>
        <year>
            2009
        </year>
    </time_point>
    <buyer>
        <body>
            UAE government
        </body>
    </buyer>
    <seller>
        <organization>
            Korean Electric Power Corporation (KEPCO)
        </organization>
    </seller>
    <region>
        South Korea
    </region>
    <plant>
        four reactors
    </plant>
    <output>
        <power>
            5.4 gigawatts (GW)
        </power>
    </output>
    <price>
        28.2 billion U.S. dollars
    </price>
</investment>
```

Fig. 4. Section of a representation of a fact as annotated and extracted by the blueprint “investment” (see also fig. 3).

These features can, for instance, deduce a specific date that needs to be assigned with the slot containing the word “yesterday”. Representing a text passage in machine-processable form like the one shown in figure 4 offers the opportunity to extract all information concerning investments in nuclear energy power in order to prepare it for automatic reporting.

A further objective of our approach is a stronger involvement of humans in the development and management of text mining tools, in general, to enhance the adoption of this technology on a broader scale. This involvement results in a more active role of the users in designing, controlling, and adapting the learning process that feeds, in this case here, the automatic detection of facts in text. The syntax for the definition of a blueprint is easy to learn. Even users without technical background are in the position to write definitions for concept blueprints. In the next phase of the development of our prototype, the users will be involved more closely in the training of the semi-automatic processes to detect blueprints that come semantically close to existing definitions.

References

[1] J. L. Kolodner, “Requirements for natural language fact retrieval”,

[2] *Proceedings of the ACM '82 conference*, 1982, pp. 192–198.

[3] N. Fuhr, “Integration of probabilistic fact and text retrieval”, *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval,* 1992, pp 211–222.

[4] M. Keikha, J. H. Park, W. B. Croft, and M. Sanderson, “Retrieving Passages and Finding Answers”, *Proceedings of the 2014 Australasian Document Computing Symposium,* 2014, 81–84.

[5] N. Balasubramanian, J. Allan, and W. B. Croft, “A comparison of sentence retrieval techniques”, *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007, pp. 813–814.

[6] R. W. White, S. M. Drucker, G. Marchionini, M. Hearst, and M. C. Schraefel, “Exploratory search and HCI: designing and evaluating interfaces to support exploratory search interaction”, *CHI '07 Extended Abstracts on Human Factors in Computing Systems*, 2007, pp. 2877–2880.

[7] A. T. Nguyen, A. Kharosekar, S. Krishnan, and S. Krishnan, “Believe it or not: Designing a Human-AI Partnership for Mixed-Initiative Fact-Checking”, *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology,* 2018, pp. 189–199.

[8] H. E. Wynne and Z. Z. Wint, “Content Based Fake News Detection Using N-Gram Models”. *Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services (iiWAS2019)*, 2019, pp. 669–673.

[9] W. A. Woods, “Context-Sensitive Parsing”. *Communications of the ACM 13(7),* 1996, pp. 413–445.

[10] F. Sha, F. and F. Pereira, F., “Shallow Parsing with Conditional Random Fields”, *Proceedings of the HLT-NAACL conference*, 2003, pp. 134-141.

[11] D. Freitag and A. McCallum, “Information Extraction with HMM Structures Learned by Stochastic Optimization”, *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, 2000, pp. 584-589.

[12] J. Cowie and W. Lehnert, “Information Extraction”. Communications of the ACM 39(1): 80–91.

[13] G. Salton, J. Allan, C. Buckely, A. Singhal, “Automatic Analysis, Theme Generation, and Summarization of Machine-Readable Texts”, in: Karen Sparck Jones and Peter Willett, *Readings in Information Retrieval*, San Francisco, 1997, pp. 478–483.

[14] J. Jancsary, F. Neubarth, S. Schreitter, and H. Trost, “Towards a context-sensitive online newspaper”. *Proceedings of the 2011 Workshop on Context-awareness in Retrieval and Recommendation*, 2011, pp. 2–9.

DOI: 10.15439/2020F166
ISSN 2300-5963 ACSIS, Vol. 21

Students Group Formation Based on Case-based Reasoning to Support Collaborative Learning

Taís Borges Ferreira, Márcia Aparecida Fernandes
Faculty of Computer Science
Federal University of Uberlândia
João Naves de Ávila 2121, Uberlândia, Brazil
taisbferreira@ufu.br, marcia@ufu.br

***Abstract*—The group formation has been widely investigated since it is a crucial aspect to perform collaborative work. However, there is no consensus about the best set of metrics or how to combine student's characteristics to improve group interactions, so it has been considered a challenge. Aiming to cope with that, this work proposes the use of case-based reasoning to suggest groups for collaboration based on the metrics and previous groups' performances stored in a case base. We gathered data from students working on collaborative tasks to build a case base and ran a grouping experiment in a class of undergraduates to verify the effectiveness of the proposal. The results evidenced that grouping based on the Big Five improved students' interactions.**

***Index Terms*—Group formation, Big-Five Personality Traits, Case-Based Reasoning, Collaborative Learning**

I. Introduction

THREE attributes are linked to effective learning according to cognitive theory: active learning and knowledge building, cooperation and teamwork, and the use of learning based on problem-solving [1]. Collaborative learning meets these three attributes since it involves social processes through which a small group of students interact and work together to reach a shared goal [1] [2].

Some theories also emphasize learning as a social process that occurs more effectively through interpersonal interactions in a cooperative context [3]. That happens because cooperating with other people allows people to question their initial understanding of a topic and observe different points of view, which motivates them to learn. On the other hand, if a group is not able to interact and work together, learning via collaboration is not possible.

As a group composition influences how people work together to achieve a goal, this is one of the most important aspects to be considered before starting a collaborative activity [4] [5]. Silva et al. [6], for example, also point out that automatically formed groups achieve better performance than those randomly generated. Therefore, it is necessary to employ a strategy of group formation that can support collaboration, as grouping students careless can trigger undesirable situations, such as, social isolation.

In computer-supported collaborative learning, the educational benefits are strongly related to strategies that motivate students to interact with the group [7]. Moreover, the group composition is crucial to trigger productive interactions between the peers [4] and eliminate conflicts that hinder the collaboration.

Grouping students to work collaboratively is not a task to be addressed by just employing a clustering technique to form homogeneous or heterogeneous groups according to given criteria [8]. Grouping strategy must also combine students' characteristics with the other grouping criteria and, also, allow forming good groups, whether the groups are homogeneous, heterogeneous, or hybrid.

The Big Five personality traits, one of the most used ways to obtain students' characteristics, are correlated with learning gains [9]. Groups systematically formed based on students' Big Five personality traits can improve significantly group outcomes [10]. Thus, they can support group formation, the understanding of how each trait affects the learning process and can provide holder to students [9], [11].

Each Big Five personality trait represents a set of behavioral tendencies that can influence group outcomes [12], [13]. Aiming to take advantage of Big Five personality traits to form groups to support collaborative learning, this work presents the development of a Case-Based Reasoning (CBR) system to cope with the processes of forming groups in which students will be able to interact and work together.

By querying the case base (CB) to form groups, one can expect to have new groups replicating the previous groups' compositions perceived as good. Through the CBR cyclic operation, each new group is converted into a case that will update the CB with their performance in a collaborative task, so it can be used later to influence new groupings, making the whole grouping process more precise and efficient.

Although all the effects of personality traits on the group's performance may not be known, by searching a new solution and trying to adapt previous solutions (CBR approach), one can expect the new groups to present good performance too. Intending to verify this hypothesis, we did a group formation experiment by querying the CB.

The paper is divided into 7 sections. The second section presents a review of related work. Section III explains how the proposed CBR works. Section IV explains how the CB was built. The results of changing the similarity metrics to query the CB are in Section V, and the results obtained by grouping students based on the CB are in Section VI. Section VII is the conclusion and future perspectives.

II. Literature review

The personality trait inventory used in this work was the well known Big Five [14]. The Big Five is a hierarchical organization composed of five basic dimensions of personality that comprehend a large number of human behavior [15]. Although Big Five was usually employed to support psychology studies, it is also a tool used to detect the student's affective state in Computer-Supported Collaborative Learning (CSCL) [13].

The behavior tendency associated with each trait comes up according to the score obtained in that trait. Those people that score high in Neuroticism will tend to be anxious, wary, concerned about social rules. A high level of Extraversion characterizes people that are gregarious, talkative, and usually show positive emotions. High levels of Openness characterizes the tendency of being curious, inquisitive, interested in new ideas. High scores in Agreeableness are related to being cooperative, warm-hearted, and agreeable. Conscientiousness relates to people that are goal-oriented and well-organized. However, as the traits are bipolar, a person who scores low in a trait will tend to show opposite characteristics.

Spoelstra et al. [16] and Altapoulou et al. [9] suggested that conscientiousness influences how productive groups tend to be. Spoelstra et al. [16] proposal considers as good to form productive and learning groups only individuals high scored in conscientiousness. On the other hand, Altapoulou et al. [9] suggest that those with a high degree of conscientiousness should be distributed among the groups, so they could positively influence the group's ability to meet the deadline effectively and efficiently.

Regarding extraversion, Roberts et al. [17] pointed out extraversion as linked to the intensity of activity in an individual's social network and Neuroticism, associated with social isolation in new groups. Extroverted people tend to act as the link between other people, improving interaction. However, as Altapoulou et al. [9] suggested in their study, a group composed only by highly extroverted individuals may negatively influence student's learning gain because they tend to distract from social interaction.

According to Bozionelos [18], extremes of conscientiousness may be considered inadequate to form social ties. Low conscientiousness leads to irresponsibility and too much conscientiousness can induce excessive preoccupation with activities and neglect social relations. Despite the theoretically poor profile for forming social ties, those who scored high in neuroticism had the same success as those with low scores in the network resource sharing. Although high neuroticism relates to very anxious people that are likely to avoid approaching other people since they are concerned about what other think, they tend to act according to the established norms and make efforts to maintain the social ties that have already been established.

The results obtained in the cited studies suggest ways to combine individuals by considering their traits. For instance, since extroverted students are known to be more likely to neglect the shared goal due to social interaction, groups with all extroverted individuals should be avoided. However, the combination of traits changes the degree of influence of a particular trait in the group [19]. So, forming groups using a base of good groups seem to better than selecting members by evaluating their scores in some specific trait [11].

As for the distribution of student characteristics in a group, Santos et al. [20] showed that heterogeneous groups work better than those where there are similar students. They realized that homogeneous groups take time to collaborate effectively. Ruterfoord's case study [21] also indicated heterogeneous groups in personality as better. In a homogeneous group in terms of personality traits, all members will have the same social skills and weaknesses, without any member to balance these characteristics. If the group is heterogeneous, there are distinct characteristics and greater variability of strengths and weaknesses, making the group able to manage possible issues.

In a recent review of group formation [22], they found evidence that spontaneous groups might take advantage of self-motivation. However, since they are made of participants that share similar interests and points of view, it can lead to unsatisfactory results. Although grouping for collaborations has been widely researched, there is not a consensus about what leads to better results: homogeneous or heterogeneous groups based on student's personality traits[7].

Although Mujkanovic and Bollin [10] concluded that systematically constructed groups show significant improvements of the group outcomes and the composition of the groups are directly related to those outcomes, it is still hard to determine precisely how much personality influences group results. All those results were motivation to build a case base (CB) and use a Case-Based Reasoning (CBR) algorithm to support the new group's suggestion based on the experience of previous groups. So we do not need to program certain types of group configuration. CBR will find them.

In summary, the CBR process can be seen as a cycle involving the tasks of retrieve, reuse, review, and retention of cases. Retrieve seeks to similar cases and in the reuse task, the retrieved cases are adapted to solve a new case. Review task evaluates new cases and, if the solution found is effective, in the retention, it is included as a new case or used to update an existing case [23]. This behavior will be used to update the case base and provide more accurate information to form new groups.

Regarding the similarity metrics used to retrieve similar cases to be used as solutions, Stahl [24] divides the calculation of similarity between case and query, in two steps: the local similarity and global similarity. The global similarity is the function that aggregates the calculated local similarities and can be, for example, a weighted sum of the results. The local similarity concerns the calculation of the distance between the pairs of elements that make up the case and the consultation, to define how close the pairs are.

Besides the functions, to be able to calculate the similarity between a case and a query, it needs to provide a strategy to deal with unknown values. There are two well-known strategies in the literature. The first one considers the distance between an unknown value and any other value as always 0.5 [25], hence the similarity will also be 0.5. The second strategy [26] is to assign distance 0 when both query and case values are unknown, and distance 1, when one of them is known. Thus, if both are unknown, similarity will be 1. When one of them is known, the similarity will be 0.

Approaches related to similarity also include techniques to automatically learn weights of attributes and the distance function, for example [27]. However, even though we ran some experiments to select the similarity metrics to work with the multi-object case (discussed in the next section), learn the similarity automatically was not the focus of this study. Thus, related to similarity, we adopted a well-known strategy to deal with some peculiarities of the case, such as the unsorted list of students that represent the case.

While talking about the implication for Education, Kolodner [28] mentions CBR as a model that can provide, for instance, suggestions about how a student will be able to have richer learning experiences by giving them the chance of applying what they have already learned. This statement reinforces the idea of using CBR to explore data related to previous group cases to extract information that will help students to take advantage of collaboration.

CBR has been successfully applied in many other areas. For instance, recommendation systems to recommend new items or products to a client ([29], [30]), stress monitoring [31], systems to decide the best assembly sequence ([32], [33]), algorithms to make decisions and find solutions in an environmental emergency scenario ([34]), etc. Although that, there are just a few applications of CBR in educational contexts and for collaboration [35].

Aiming to explore this gap, Cocea and Magoulas [35] modeled the student behavior using a CBR that was also a source of information to feed a clustering approach to form groups. Similarly, in our proposal, CBR models and updates information of students and groups. In a literature review conducted by Costaguta [36], among all the approaches of group formation researched, they found and presented only one work that applies CBR to form groups: the Cocea and Magoulas proposal. That evidences the lack of approaches involving CBR to form groups.

III. The proposed Case-Based Reasoning

The use of Case-Based Reasoning (CBR) for supporting group formation improves the ability to build good groups. In addition, it allows identifying groups that need to be undone to avoid poor group performance. The cyclic operation of CBR contributes to the evolution of the case base (CB), due to adding and updating cases, therefore it is expected that the solution quality will be more accurate, as the CBR cycle runs. To meet this goal, a case structure and a CBR operation are specified in the following subsections.

A. Case Structure

A case is an object composed of two types of information: group member characteristics and group metrics. The case structure in the CB can be seen in Figure 1, where the component Group corresponds to group member characteristics and contains the Big Five personality traits (Openness (O), Conscientiousness (C), Extraversion (E), Agreeableness (A) and Neuroticism (N)), for each group member.The group is an unordered list of students and each student an array of personality traits.

The component Group metrics contain the number of group members (size), deadline, and group metrics. The deadline corresponds to a string indicating how much time the group can spend to solve the task and submit a solution. Its values can be C (class time), which means the task should be done during the class, or W (week) when the task is to be completed in extra-class meetings. The group performance metrics are a list of attributes to classify the performance of the groups. Group performance metrics contains:

- Everybody Contributes (EC) measures group members' contributions to solving a task and if it was a significant contribution to the solution.
- Task Completed (TC) indicates whether the group was able to complete the task within the given deadline or not.
- Grade (G) assigned to the group by the teacher.
- Interactions (I) among group members, by using communication tools or face-to-face.

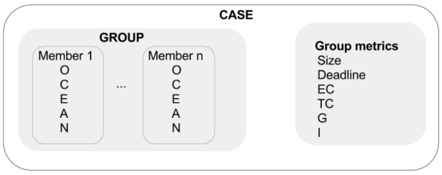


Fig. 1. The structure of a case.

The case structure was designed to represent three different problems related to grouping.The first one, type 1 problem, is to determine the most likely performance of a known group. Then, the case description contains student characteristics of the group and the solution will be the group performance metrics and the group quality. Therefore, the similarity will rely on the similarity between the case and the query groups. Case solution must fit the specified group size and, consequently, solutions containing fewer students are discarded. The deadline can be also used as a filter.

Type 2 problem is to discover the best set of students that are more likely to perform well, which means to find the best partners to form a group given the personality traits of the students. The case description of type 2 is composed of the personality traits of a student. The case solution will be the list of students that could be grouped with a specific student. Task deadline and group size are filters. By performing that kind of query, the group performance is not part of the case description or solution, but thresholds to guarantee the solution is based only on cases that represent good groups.

The type 3 problem is related to find the characteristics of a task that could help students to solve a collaborative task and perform well. The case description is given by the group performance metrics and group composition, and the solution is the deadline. So, the similarity for type 3 must consider both, performance metrics and group composition. The solution is intended to support collaborative tasks. However, these tasks involve other information than their deadline, which is not currently implemented.

B. Similarity Metrics Between Query and Case

To help teachers group their students and take advantage of collaborative tasks by querying to case base, one needs to define suitable similarity metrics to compare the query with the description of the possible solutions in the case. Possible solutions are the most similar cases in the case base, according to a similarity metric. So, the similarity metrics must be defined based on the case structure.

As the case in our proposed CBR is an object that contains two components, the case-query similarity (S) depends on group metrics similarity (GMS) and group similarity (GS) that corresponds to the average of the students' similarity (SS). Similarity calculus also depends on those problem representation types defined in the previous section. Thus, the query type is a parameter considered for querying the CBR.

The final similarity value is $S = GS$ for type 1 and 2, and $S = (GS + GMS)/2$, if the type is the third one. However, since a case (or query) is an object and the similarity between two objects is given by the aggregation of attributes' similarities in a single value, to calculate GMS, SS and GS, it is necessary to calculate the attribute similarity.

Therefore, let a_c and a_q be a pair of corresponding attributes, the first one from the case and the second one from query, the attribute similarity, based on the distance value, can be calculated using one of following functions: Threshold (Eq. (1)), Linear ((2)), Exponential (Eq. (3)) and Sigmoid (Eq. (4)). Then, with the pair distance, given by the linear distance $d_a = |a_c - a_q|$, one of these functions is used to convert it into a similarity. The similarity value belongs to interval $[0, 1]$, where the value 0 means the attributes are completely differents and 1 means they are 100% equal.

$$\text{Threshold:} \quad \text{sim}(a_c,a_q) = \begin{cases} 1, & d_a <= \text{t} \\ 0, & \text{otherwise} \end{cases} , \tag{1}$$

where t is the threshold of d_a below which similarity will be 1 (100%). The value of t was defined as 0. So, if $d_a = 0$, the similarity will be 100%,and if $d_a >= 0$, will be 0%.

$$\text{Linear:} \quad \text{sim}(a_c,a_q) = \frac{\text{max - } d_a}{\text{max - min}} \tag{2}$$

where max is the maximum possible value of the attribute a and min, is the minimum. If a is a student's personality trait, $max = 1$ and $min = 0$.

$$\text{Exponential:} \quad \text{sim}(a_c,a_q) = e^{d_a * \alpha} \tag{3}$$

where $\alpha = -1$ and $d_a \in [0, 1]$. As the similarity function needs to return a value between 0 and 1, due to the characteristics of Exponential function, if $\alpha = 1$, for instance, the result will be something between 1 and 2.7. In case of $d_a \in \{0, 0.5, 1\}$, by using $\alpha = -1$, it will return similarity into the desired interval.

$$\textit{Sigmoid:} \quad \text{sim}(a_c,a_q) = \frac{1}{1+e^{\frac{dif(c_i,q_i)-\theta}{\alpha}}} , \tag{4}$$

where $\alpha = 0.01$ and $\theta = 0.5$. The θ is the value of the central point, making the curve turning point. As the difference is normalizes between $[0, 1]$, the central value was defined as 0.5. If the curve amplitude were to big, the values of similarities will be also out of the interval. Using $\alpha = 0.01$ and $\theta = 0.5$, it will return values into the interval of $[0, 1]$.

Finally, to perform the aggregation of attribute similarities and obtain the value of GSM or SS, two aggregation functions were implemented, Minkowski (Eq. (5)) and Simple Matching (Eq. (6)). The GMS and SS calculus are quite similar, since group metrics and students are represented by arrays that contain the respective attributes. According to [24], GMS and SS are calculated by using a local and global similarities, that correspond to our attribute and object similarities, respectively.

$$\text{Minkowski:} \quad \left(\frac{\sum_{i=1}^{n} sim(a_c,a_q)^p}{n} \right)^{\frac{1}{p}} , \tag{5}$$

where n is the number of attributes. The numeric values of Eq. (5) are aggregated by using exponentiation and radication according to the value of p. When $p = 1$, the Minkowski equation corresponds to the formula of Manhattan distance and, for $p = 2$, Euclidean distance.

$$\text{Simple Matching:} \quad \frac{\#equal}{n} \tag{6}$$

where n is the number of attributes and $\#equal$ is the number of attributes having equal value. Thus, the similarity calculated with Simple Matching is the rate of equal attributes of the compared objects. This is a type of aggregation function that works better with binary or categorical values because values can be only equal or different. Working with numerical values, it would be better to consider the similarity as the rate that indicates how close their values are, even though they are not equal.

As previously, GS is the average of student's similarity (SS). On the other hand, a group is an unsorted list of students, which means that the first student in the list of a query can be similar to the second (or any other) in the case. Moreover, if the case and the query are equal, GS must be 100% even though the students are presented in different orders. The strategy adopted to associate each student in the query with the most similar in the case, in order to obtain the correct pair of students to calculate the SS, is described by the following steps.

1) Calculate the similarity between all pairs of students and make a list.

2) Select the pair $P(s_c, s_q)$ with the highest similarity in the list.
3) Discard every other value calculated with s_c or s_q
4) If the list still has items, return to step 2.

Step 1 does not really require a significant computational cost. The groups are formed by up to 5 members and building the list takes constant time, $\theta(1)$, in the worst-case. To avoid the combinatorial search, a greedy strategy selects the pair with the highest similarity.

C. CBR Operation

Having the case structure and similarity metric defined, which is given by the similarity S, defined in the previous section, the four tasks of the proposed CBR are summarized in Figure 2. When a new problem arrives, it is converted to case structure, becoming a new case to be used for querying the CB and retrieving the most similar cases. Once a retrieved case is taken as a solution to the new problem, this solution is revised, and, finally, the group feedback is evaluated to decide if this solution (case) should be added or not to CB. Each task in Figure 2 is described as follows.

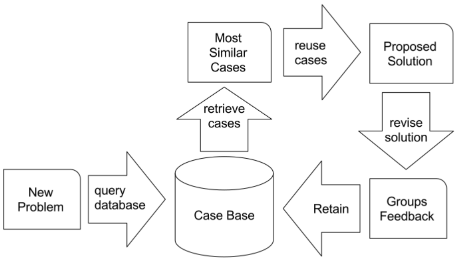


Fig. 2. Case-Based Reasoning operation.

Retrieve Task: To retrieve cases as solutions, one should provide data on the case structure elements and, optionally, minima thresholds for each group metric. These thresholds will work as filters for querying the CB. For example, by providing $G = 0.6$ and $I = 0.8$ as thresholds in the query, the query result will be only those cases where $G \geq 0.6$ and $I \geq 0.8$. The retrieving process will then bring all the cases that fit the given thresholds and order them according to their similarity with query. Those showing high similarity with the query are the most suitable cases to solve it.

Reuse Cases: A query retrieves from CB all suitable cases, given thresholds. The similarity (S) between each retrieved case and query is calculated. The most similar cases are selected and applied to solve the new problem. For example, if the query was built taking into account only students' characteristics, without any group performance information, similar cases will be those where the group formation approximates to the group in the query. Then, the most similar cases returned can be applied to predict group performance. This approach allows predicting if the group is more likely to succeed or fail before students are grouped, due to the considerations of their characteristics provided in the query.

Revise Solution: The revision of a solution occurs after the groups worked in collaborative tasks. Based on group's evaluation performing those tasks, we use their performance to evaluate the solution and to check how well it worked. We can calculate how many of the new groups worked well calculating each group performance, using the four group performance metrics: everybody contributes to the group solution (EC), group's grade (G), members' communication level using communication tools or face-to-face (I) and task completed (TC). The groups were classified as a good or poor, using the weighted average $(WA) = (0,4 * EC) + (0,1 * G) + (0,3 * I) + (0,2 * TC)$.

Once the weighted average is calculated, the group performance (GP) is classified as GOOD, if $WA > 0.5$, and POOR, otherwise. The higher weights were applied to EC and I because they are the metrics that indicate how many group members are interacting to solve the proposed task. They may also point, for instance, when there is some student isolated and not working with the group. On the other hand, TC and G are also metrics that measure group effectiveness. Thus, in addition to interaction metrics, a good group should be able to reach a satisfactory grade and finish the proposed task.

Retain Task: The common CBR operation usually cut off solutions that are not classified as good ones in the revision step. In our proposal, poor solutions are as useful as good solutions. The last ones help to form new good groups and the former ones help to identify group compositions that may not succeed. For this reason, both good and poor groups are inserted in the CB in Retain Task.

The way a revised case is inserted in the CB depends on the combination of students in the revised case. If it is a combination that is already represented in a case of the CB, it will be used to update an existing case. If there is no other case with such a combination of students' characteristics in CB, the case will be inserted as a new case.

IV. Building the Case Base

The CB was formed from real cases. Data were collected in 4 classes composed of students enrolled in Computer Science, Information Management, and Business Management courses. The evaluation activity proposed by the teacher and data collection were different in each class, but all groups were evaluated according to the metrics (EC, TC, G, and I) in the previous section. In two classes, the collaborative task involved a shared writing tool. The tool records all user activities in its logs. The logs, recorded during task solution, were applied to assess the metrics EC, TC, and I. In the third class, the teacher used Moodle's chats and forums. Thus, group metrics were assessed by analyzing students' activity in Moodle. In the fourth class, no collaborative tool was used and the metrics EC, TC, and I were evaluated based on the teacher's report.

Students' personality traits were calculated using the 44-item Big Five inventory [37], translated to Brazilian Portuguese by [38]. Although the collaborative activity was compulsory and part of their evaluative activities, answering the Big Five inventory was not. Thereat, in some groups, some students have not answered the Big Five inventory and their personality traits were unknown. Students were allowed to choose their groups, limited to 5 members per group. As a result, 24 groups were formed: 2 groups of 2, 13 groups of 3, 2 groups of 4, and 7 groups of 5 students. One student decided to work alone. A total of 87 students was involved in group activities.

The formed CB was mainly composed of individuals that have received medium or high scores in openness, conscientiousness, and agreeableness. It has been at a certain level expected since the activities related to Management and Computing involves being organized, goal-oriented, and good group workers. Undergraduate students are also expected to be more open since it is less likely that a person not interested in new knowledge to join a graduation course. Extraversion, however, was mostly medium or high and just a small number of students low on extraversion, a feature that is likely to happen among students. So, the CB may not be as comprehensive as it must be to cover the possibilities of group combination.

Regarding the characteristics of the groups and the performance observed, the combination of individuals in groups leads to good results. On the other hand, the results regarding the collaboration to solve the task were not good when the time for task resolution was very long (W). Of all of which students had a one-week deadline to solve the work, only the 3-member group interacted and worked collaboratively. So, if the task deadline is long, the group must have 3 or 2 students.

Among the groups with a shorter deadline, some characteristics of the individuals seemed to intervene in group interactions. In the class were groups were mainly composed of students with high and medium scores in all traits except Neuroticism, the only poor group, regarding member interactions and students grades, was the one composed by students having a medium degree of Neuroticism. This suggests that having individuals with high and low Neuroticism is positive for collaboration.

In the class where students had a week to solve the task, and almost all groups failed, although the best group was the one composed by students medium scored in Neuroticism, all students were also high for Openness. The degree of Openness can increase the degree of engagement of individuals, especially if a task is perceived as interesting and lead to the gain of new knowledge. In the class where there was no middle score student in Neuroticism, the combination of a high, middle, and low level of Conscientiousness make them show a high level of interaction.

V. Effect of Changing the Similarity Metrics

The CBR proposed supports the change of the functions involved in the similarity metric calculation. To test the effect of each function we used the same query to retrieve cases from the CB, built as in Section IV, but changing the combination of the function involved in similarity calculus. A different function can increase (or decrease) the distance between opposites (H and L) and the similarity between close values, such as M and L. So, the results of each function used to calculate the similarity and its effects according to the case structure defined could be evaluated.

Table I shows the effects on similarity value (column Similarity) obtained using one of the functions available to calculate attribute similarity (AS) together with one of the functions to calculate object similarity (OS). For example, selecting the Threshold function as attribute similarity and the 3-degree Minkowski function (Minkowski p=3) as object similarity, the similarity obtained was 71.42%. The presented values (column Similarity) correspond to the similarity between a case, composed of a group of 3 known students, and a query composed of only 1 known student.

A known student is the one that we know the personality traits. The unknown are those that we do not know the personality traits. As a result, by querying the CB, the known student will be compared to a known student of the case, using one of the available AS functions, and the AS of the unknown students will be calculated using the defined strategy to deal with unknown values. As a result, the similarity will be the mean of the OS calculated for each student (known or unknown).

According to the results, the Simple Matching function did not affect the similarity value, even if different functions are used to calculate AS. One can notice by comparing the results in Table I. Even if the function to calculate AS was changed to Threshold, Linear, Exponential, or Sigmoid, the similarity obtained will be the same: 40%.

It is important to notice that the representation of the students' characteristics is numeric. The levels L, M, and H, used to represent their level in each personality trait, in the database corresponds to 0, 0.5, and 1, respectively. As a result, 0.5 is closer to 1 than 0. The Simple Matching only counts attributes with identical values and ignores the degree of proximity between numerical values, no matter how close the values are.

If the values being compared are slightly different the similarity will always be 0 and will not affect the similarity value calculated. So, Simple Matching is not a good choice to deal with numeric values like those stored in our CB. On the other hand, the Minkowski function works by changing the similarity smoothly as according to how close or distant are the values of each attribute in the case and the query. That implies that the Minkowski function will work better than the Simple Matching.

According to the results, an increase in the Minkowski degree (p) also increases the similarity. However, p=3 might not be a good choice, because it also increases the number of cases retrieved with higher similarity value, even though they are not too similar. The use of the function degree equals 2 or 3 to compare the members of a group caused similarity between medium score (M) and extreme scores (H or L) to increase. Although the medium scored individuals tend to be a bit similar to those that have a high or low score in personality traits, the proximity caused by p=2 can be suitable to retrieve similar cases that will be used as alternative solutions, when the solution is nonexistent in the CB.

The results in Table I corresponds to the similarity of the same pair of query and case retrieved according to the functions selected to calculate AS and OS. Despite that, we also performed other queries and compared them with each case on the CB. The effect of changing the functions to calculate similarity was the same observed and described in above.

TABLE I
THE SIMILARITY BETWEEN A CASE AND A QUERY OBTAINED BY CHANGING THE ATTRIBUTE SIMILARITY (AS) AND OBJECT SIMILARITY (OS)

AS	OS	Similarity
Threshold	Minkowski p=1	65.00%
	Minkowski p=2	69.72%
	Minkowski p=3	71.42%
	Simple Matching	40.00%
Linear	Minkowski p=1	70.00%
	Minkowski p=2	71.10%
	Minkowski p=3	71.89%
	Simple Matching	40.00%
Exponencial	Minkowski p=1	71.07%
	Minkowski p=2	71.73%
	Minkowski p=3	72.26%
	Simple Matching	40.00%
Sigmoid	Minkowski p=1	70.00%
	Minkowski p=2	71.10%
	Minkowski p=3	71.89%
	Simple Matching	40.00%

VI. Group Suggestion Based on CBR queries

In the previous experiments, we did not set up query thresholds, since the goals were to observe the CBR behavior and the similarity changes when the functions involved were alternated. This time, the goal was to suggest good groups. By grouping students based on querying the built CB, it was expected to form new good groups reflecting the CB.

The grouping experiment was conducted in a class of Data and Business Information. To determine student's personality traits, we asked them to answer the 44-item Big Five Inventory, a questionnaire translated to Portuguese and validate in Brazil by Andrade [38]. Using the questionnaire answers, we calculate the value of each personality trait and fill in the array of characteristics that represent each student.

After a previous group activity, in which students could choose their partners, the teacher proposed 3 more activities, but now, using the suggestions based on CBR queries. As it was intended to form groups, the traits of each student in the class were used to build queries of type 2. The result of every type 2 query is a list of students that could work well with the student given in the query. So, type 2 was set up before performing the queries.

All the queries were also set up with the following parameters: grade higher than 60% of the total grade (G > 0.6), the interaction between students greater than 50% (I > 0.5), and everybody contributes to solving a task (EC = 1). To calculate the similarity, the function Linear was chosen to calculate attribute similarity and Minkowski with $p = 2$ to calculate object similarity. The strategy adopted to deal with unknown values in all queries performed was to consider unknown values as 50% similar.

The cases retrieved meeting all the restrictions were ordered according to the similarity between the cases and the query. Then, groups were formed by grouping together those students whose characteristics brought the same case as a solution. That is, together they will be similar to the case used to group them. There was no automation of the grouping based on the query solution when the experiments were conducted. The list of suggestions was manually done.

The eight resulting groups were suggested to the teacher. In Table II, the column "Group" identifies the suggested groups. For each student, their score in each Big Five personality trait is shown in columns O, C, E, A, and N. Students' characteristics in three of the suggested groups were quite similar to the cases used to group them.

The other groups were also similar but, for some traits, the differences were bigger. However, they were formed preserving the patterns observed as good while building the CB. For example, except for a group, there is no suggested group composed only by a middle-scored individual in Neuroticism. Almost every group has more one member with high in Openness, considered good for collaboration.

Groups with all members high scored in Consciousness and Extroversion were also avoided. The exception is the group H. Each of these traits isolated can increase the probability of certain undesirable situations. High Extroversion, for example, may lead to distraction with social interactions and high Conscientiousness to isolation due to excessive focus on goals. However, both simultaneously can reduce the possible negative effects.

The day the collaborative activities were applied, some students were absent or arrived late. Thus, the teacher changed the suggested groups a bit. For the first activity, the teacher decided to remove the absent students without making any other changes. Thus, students in groups D, G, and H worked with fewer members. The group adopted by the teacher in the first activity and the performance observed according to the level interaction (I $\in$ [0, 5]) are shown in Table III.

All groups in the first activity showed good interaction (I = 5). The exception was the group G that lost the student 32, the only member with L score for Extroversion. In the following activities, the groups were also changed because of some students not present in the first activity, showed up for the second one, for example. The group that shown poor interaction in activity 1 was modified by removing the student 31. Most of the groups remained the same formation in the three activities. In the second and third activities, all groups had a good level of interaction (I=5).

According to the report on collaborative activities sent by the teacher, all groups were able to complete the proposed activities. Moreover, the groups formed by means the CBR recommendation were more efficient and better regarding the iteration, when compared to the groups formed by students themselves. Despite the questioning about the group forma-

TABLE II
GROUPS SUGGESTIONS BASED ON THE QUERIES.

Group	Student	O	C	E	A	N
A	1	H	H	L	M	M
	2	M	H	M	M	M
	3	H	H	M	M	H
B	4	H	H	M	H	L
	5	H	H	L	M	H
	6	M	M	L	M	L
	7	M	M	M	M	L
	8	H	M	M	M	H
C	9	H	M	M	H	L
	10	M	H	L	M	M
	11	M	M	M	H	L
	12	M	M	M	H	L
	13	H	M	M	H	L
D	14	H	M	M	H	L
	15	H	M	M	M	H
	16	H	L	M	M	M
	17	H	M	M	H	L
	18	M	M	M	H	M
E	19	H	M	M	M	M
	20	H	M	M	M	M
	21	M	M	H	H	H
	22	H	H	M	H	H
	23	H	M	M	M	M
F	24	M	H	H	H	M
	25	M	M	L	H	M
	26	M	M	H	H	M
	27	M	M	H	H	M
G	28	M	H	M	H	M
	29	M	H	M	H	M
	30	M	M	M	M	L
	31	M	M	M	M	M
	32	M	M	L	M	M
H	33	L	H	H	H	M
	34	H	H	H	H	L
	35	M	H	H	H	L
	36	H	H	H	H	L
	37	H	H	H	H	L

TABLE III
FIRST ACTIVITY GROUPS AND INTERACTION.

Group	Student	O	C	E	A	N	I
A	1	H	H	L	M	M	5
	2	M	H	M	M	M	
	3	H	H	M	M	H	
B	4	H	H	M	H	L	5
	5	H	H	L	M	H	
	6	M	M	L	M	L	
	7	M	M	M	M	L	
	8	H	M	M	M	H	
C	9	H	M	M	H	L	5
	10	M	H	L	M	M	
	11	M	M	M	H	L	
	12	M	M	M	H	L	
	13	H	M	M	H	L	
D	14	H	M	M	H	L	5
	15	H	M	M	M	H	
	16	H	L	M	M	M	
E	19	H	M	M	M	M	5
	20	H	M	M	M	M	
	21	M	M	H	H	H	
	22	H	H	M	H	H	
	23	H	M	M	M	M	
F	24	M	H	H	H	M	5
	25	M	M	L	H	M	
	26	M	M	H	H	M	
	27	M	M	H	H	M	
G	28	M	H	M	H	M	1
	29	M	H	M	H	M	
	30	M	M	M	M	L	
	31	M	M	M	M	M	
H	33	L	H	H	H	M	5
	34	H	H	H	H	L	
	35	M	H	H	H	L	
	36	H	H	H	H	L	

tions, the students' perception of their performance was positive considering the new grouping. Also, groups that worked together on previous activity, in general, continued to show good interaction in the following activities.

Groups B, D, E, and H remained unchanged in the three activities and showed positive results, which points as positive such combinations of characteristics. They were groups of students with extreme scores (H or L) or groups of students with scores M, combined with H or L for the trait of Neuroticism. Group G, the group that had poor interaction in the first activity, was mainly formed by students with score M in most of the traits. By removing a member, it reduced the number of members having M score for Consciousness, Ability, and Neuroticism, which seems to improve the interaction.

Even after the change to deal with students' presence/absence on the day activities 1, 2, and 3 were applied, the groups also suffered alterations that, however, did not negatively influence the interaction and did not result in bad groups. The changes made by the teacher bear some resemblance to previous activity groups and base cases. However, they also showed differences in some traits and therefore could also be used to popular the case base as new examples of good groups.

A student is represented by an array of 5 personality traits that can assume 3 different values (L, M, and H), so they can be represented in 243 different ways. That means we have about 59,000 possibilities for 2-member groups. As the inclusion policy adopted in this work is to include as a new case every new group that has a configuration not found in the case base (there is no case matching under 100% of similarity), which means that, in the worst case, the base will reach around 59,000 2-member cases.

It can get worst if we think about groups formed by 3, 4, and 5 students. Therefore, as the case base grows bigger the strategies deal with such a huge amount of registers are a necessity. The data we worked on until now is not too big so we can search for all the cases to find a suitable one. Despite that, it is an issue that should be addressed in a future version of the CBR presented in this paper.

VII. CONCLUSION

In this work, we proposed the use of case-based reasoning to support the creation of groups to work in collaborative tasks. It is a crucial aspect when it comes to performing collaborative work. The way individuals are grouped to work together can influence their interactions or lead to an undesirable situation, such as isolation in a group. Due to that relevance, group formation has been widely studied in the context of Collaborative Learning. However, there is no consensus on what are the best set of metrics that improve group quality and how to combine students' characteristics to improve group interactions. Therefore, this issue is still considered a challenge.

Many known works suggested clustering algorithms to form groups, but it is not just a problem of clustering students together according to their similarities or differences. It needs to consider the combination of the different attributes of the groups to reach good results. Aiming to cope with this issue, we proposed the case-based reasoning to recommend suitable groups for collaborative work. The basic four operations of case-based reasoning allow the teacher to use previous knowledge on group performance, according to the characteristics of their students, as well to form new groups.

In this work, we used students' characteristics (Big-Five personality traits) and group attributes to represent a case. To populate the case base with real cases, we collect data from undergraduate students' groups working on collaborative tasks. Next, with this case base, we ran experiments to form groups in a different class, aiming to verify the effectiveness of the proposal. Based on the group's performance and results reported by the class teacher, it evidenced that personality traits influence the interaction level in a group. Furthermore, the results demonstrated grouping based on the Big-Five personality traits improved students' interactions in that class.

We also tested the effect of changing similarity metrics employed to retrieve a solution from the case base. Considering the current configuration of our case base, with all attributes represented as numeric values, the Minkowski (with p=2) fits well the role of object similarity metric. As well, the linear function as the similarity between attributes. Despite that, a broader set of functions to calculate similarity metrics supports the inclusion of new group attributes and new types of case representation.

As future work, we plan to include new metrics known as influencing groups' outcomes, for instance, student role and test its effectiveness in interactions, student's motivation, and academic performance of the students in a group. Furthermore, we plan to update and enlarge the case base with new data from groups working collaboratively. That might cause the base to grow big and make the search in all case base registers computationally expensive. Thus, to define an indexing technique or a strategy for selecting relevant cases and removing the unnecessary ones is another further work that must be done. Consequently, retrieving cases to suggest a group formation will keep reasonable computing time, even if the case base grows bigger.

REFERENCES

[1] M. Alavi, "Computer-mediated collaborative learning: An empirical evaluation," *Journal MIS Quarterly*, vol. 18, pp. 159–174, 1994. doi: https://doi.org/10.2307/249763

[2] A. Hron and H. F. Friedrich, "A review of web-based collaborative learning: factors beyond technology," *Journal of Computer Assisted Learning*, vol. 19, pp. 70–79, 2003. doi: https://doi.org/10.1046/j.0266-4909.2002.00007.x

[3] L. Vygotsky, *Mind in Society: The Developmant of Higher Psychological Processes*. Cambridge, MA: Harvard University Press, 1978.

[4] I. Magnisalis, S. Demetriadis, and A. Karakostas, "Adaptive and intelligent systems for collaboration learning support: A review of the field," *IEEE Transactions on Learning Technologies*, vol. 4, pp. 5–20, 2011. doi: https://doi.org/10.1109/TLT.2011.2

[5] S. Manske, T. Hecking, I. Chounta, and H. Hoppe, "Using differences to make a difference: a study on heterogeneity of learning groups," in *Proceedings of International Conference on Computer Supported Collaborative Learning*, 2015. doi: https://doi.dx.org/10.22318/cscl2015.191. ISSN 1573-4552 pp. 182–189.

[6] F. E. O. Silva, C. L. R. Motta, and F. M. Santoro, "Team composer: Assembling groups through social matching," in *Proceedings of the International Conference on Computer Supported Cooperative Work in Design*, 2010. doi: https://doi.org/10.1109/CSCWD.2010.5471990 pp. 128–133.

[7] R. C. D. Reis, C. L. Rodriguez, G. C. Challco, P. A. Jaques, I. I. Bittencourt, and S. Isotani, "Relação entre os estados afetivos e as teorias de aprendizagem na formação de grupos em ambientes cscl," in *Anais do XXVI Simpósio Brasileiro de Informática na Educação*, 2015. doi: http://dx.doi.org/10.5753/cbie.sbie.2015.1012 pp. 1012–1021.

[8] J. V. M. Chaves, "Automatic group formation," Ph.D. dissertation, Faculdade de Engenharia da Universidade do Porto, 7 2019.

[9] P. Altanopoulou and N. Tselios, "How does personality affect wiki-mediated learning?" in *Proceedings of International Conference on Interactive Mobile and Communication Technologies and Learning*, 2015. doi: https://doi.org/10.1109/IMCTL.2015.7359546 pp. 16–18.

[10] A. Mujkanovic and A. Bollin, "Personality-based group formation: A large-scale study on the role of skills and personality in software engineering education," in *OCCE 2018: Empowering Learners for Life in the Digital Age*, 2019. doi: https://doi.org/10.1007/978-3-030-23513-0_21 pp. 207–217.

[11] T. Ferreira, J. Buiar, M. Fernandes, A. Pimentel, and O. Luiz, "Detecção automática de traços de personalidade e recomendação de agrupamento com o modelo big five," in *XXIX Simpósio Brasileiro de Informática na Educação (Brazilian Symposium on Computers in Education)*, 2018. doi: http://dx.doi.org/10.5753/cbie.sbie.2018.1643 pp. 1643–1652.

[12] M. A. G. Peeters, C. G. Rutte, H. F. J. M. van Tuijl, and I. M. M. J. Reymen, "The big five personality traits and individual satisfaction with the team," *Small Group Research*, vol. 37(2), p. 187–211, 2006. doi: https://doi.org/10.1177/1046496405285458

[13] R. C. D. Reis, S. Isotani, C. L. Rodriguezac, K. T. Lyraa, P. A. Jaques, and I. I. Bittencourte, "Affective states in computer-supported collaborative learning: Studying the past to drive the future," *Computers and Education*, vol. 120, pp. 29–50, 2018. doi: https://doi.org/10.1016/j.compedu.2018.01.015

[14] L. R. Goldberg, "Language and individual differences: The search for universal in personality lexicons," *Review of personality and social psychology*, vol. 2, pp. 141–166, 1981.

[15] R. R. McCrae and O. John, "An introduction to the five-factor model and its applications," *Journal of Personality*, vol. 60, pp. 175–215, 1992. doi: https://doi.org/10.1111/j.1467-6494.1992.tb00970.x

[16] H. Spoelstra, P. Van Rosmalen, T. Houtmans, and P. Sloep, "Team formation instruments to enhance learner interactions in open learning environments," *Computers in Human Behavior*, vol. 45, pp. 11–20, 2015. doi: https://doi.org/10.1016/j.chb.2014.11.038

[17] S. G. B. Roberts, R. Wilson, P. Fedurek, and R. I. M. Dunbar, "Individual differences and personal social network size and structure," *Personality and Individual Differences*, vol. 44, pp. 954–964, 2008. doi: https://doi.org/10.1016/j.paid.2007.10.033

[18] G. Bozionelos, "The relationship of the big-five with workplace network resources: More quadratic than linear," *Personality and Individual Differences*, vol. 104, pp. 374–378, 2017. doi: https://doi.org/10.1016/j.paid.2016.08.036

[19] T. Ferreira and M. Fernandes, "Detecção de traços de personalidade em textos para apoiar a formação de grupos para colaboração," in *Proceedings of Brazilian Symposium on Computers in Education*, 2017. doi: http://dx.doi.org/10.5753/cbie.sbie.2017.1627 pp. 1627–1636.

[20] O. C. Santos, A. Rodriguez, E. Gaudioso, and J. G. Boticario, "Helping the tutor to manage a collaborative task in a web-based learning environment," in *Supplementary Proceedings of International Conference on Artificial Intelligence in Education*, vol. 4, Sidney, Austrália, 2003, pp. 153–162.

[21] R. H. Rutherfoord, "Using personality inventories to form teams for class projects – a case study," in *Proceedings of SIGITE'06 Proceedings of the 7th conference on Information Technology Education*. Canterbury, United Kingdom: ACM, 2006. doi: https://doi.org/10.1145/1168812.1168817 pp. 73–76.

[22] S. Borges, R. Mizoguchi, I. I. Bittencourt, and S. Isotani, “Group formation in cscl: A review of the state of the art,” *Higher Education for All. From Challenges to Novel Technology-Enhanced Solutions*, vol. 832, pp. 71–88, 2018. doi: https://doi.org/10.1007/978-3-319-97934-2_5

[23] J. L. Kolodner, “An introduction to case-based reasoning,” *Artificial Intelligence Review*, vol. 6, pp. 3–34, 1992. doi: https://doi.org/10.1007/BF00155578

[24] A. Stahl, “Learning of knowledge-intensive similarity measures in case-based reasoning,” Ph.D. dissertation, Departamento de Ciência da Computaçã da Universidade de Kaiserslautern, 10 2003.

[25] F. Ricci and P. Avesani, “Learning a local similarity metric for case-based reasoning,” in *International Conference on Case-Based Reasoning (ICCBR): Case-Based Reasoning Research and Development*, vol. 1010. Sesimbra, Portugal: Springer, 1995. doi: https://doi.org/10.1007/3-540-60598-3_27 pp. 301–312.

[26] J. Surma and K. Vanhoof, “Integrating rules and cases for the classification task,” in *International Conference on Case-Based Reasoning (ICCBR): Case-Based Reasoning Research and Development*, vol. 1010. Berlin, Heidelberg: Springer, 1995. doi: https://doi.org/10.1007/3-540-60598-3_29 pp. 325–334.

[27] P. Perner, “Case-based reasoning - methods, techniques, and applications.” in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications (CIARP 2019)*, vol. 11896, 2019. doi: https://doi.org/10.1007/978-3-030-33904-3_2 pp. 16–30.

[28] J. L. Kolodner, “Educational implications of analogy a view from case-based reasoning,” *American Psychologist*, vol. 52, pp. 57–66, 1997. doi: https://doi.org/10.1037/0003-066X.52.1.57

[29] V. Gupta and S. K. Sahana, “Nudge-based hybrid intelligent system for influencing buying decision,” *Advances in Computational Intelligence. Advances in Intelligent Systems and Computing*, vol. 988, no. 1, pp. 165–174, 2020. doi: https://doi.org/10.1007/978-981-13-8222-2_14

[30] J. W. Chang, M. C. Lee, and T. I. Wang, “Integrating a semantic-based retrieval agent into case-based reasoning systems: A case study of an online bookstore,” *Computers in Industry*, vol. 78, pp. 29 – 42, 2016. doi: https://doi.org/10.1016/j.compind.2015.10.007 Natural Language Processing and Text Analytics in Industry.

[31] S. Begum, M. U. Ahmed, P. Funk, and R. Filla, “Mental state monitoring system for the professional drivers based on heart rate variability analysis and case-based reasoning,” in *2012 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2012, pp. 35–42.

[32] S. Chen, J. Yi, H. Jiang, and X. Zhu, “Ontology and cbr based automated decision-making method for the disassembly of mechanical products,” *Advanced Engineering Informatics*, vol. 30, no. 3, pp. 564 – 584, 2016. doi: https://doi.org/10.1016/j.aei.2016.06.005

[33] Y. Qin, W. Lu, Q. Qi, X. Liu, M. Huang, P. J. Scott, and X. Jiang, “Towards an ontology-supported case-based reasoning approach for computer-aided tolerance specification,” *Knowledge-Based Systems*, vol. 141, pp. 129 – 147, 2018. doi: https://doi.org/10.1016/j.knosys.2017.11.013

[34] D. Wang, K. Wan, and W. Ma, “Emergency decision-making model of environmental emergencies based on case-based reasoning method,” *Journal of Environmental Management*, vol. 262, p. 110382, 2020. doi: https://doi.org/10.1016/j.jenvman.2020.110382

[35] M. Cocea and G. D. Magoulas, “User behaviour-driven group formation through case-based reasoning and clustering,” *Expert Systems with Applications*, vol. 39, p. 8756–8768, 2012. doi: https://doi.org/10.1016/j.eswa.2012.01.205

[36] R. Costaguta, “Algorithms and machine learning techniques in collaborative group formation,” in *MICAI 2015: Advances in Artificial Intelligence and Its Applications*, 2015. doi: https://doi.org/10.1007/978-3-319-27101-9_18 pp. 249–258.

[37] O. John and S. Srivastava, “The Big Five trait taxonomy: History, measurement, and theoretical perspectives,” *Handbook of personality: Theory and research*, vol. 2, pp. 102–138, 1999.

[38] J. M. Andrade, “Evidências de validade do inventário dos cinco grandes fatores de personalidade para o brasil,” Ph.D. dissertation, Instituto de Psicologia - Universidade de Brasília, 7 2008.

DOI: 10.15439/2020F118
ISSN 2300-5963 ACSIS, Vol. 21

Gradient Boosting Application in Forecasting of Performance Indicators Values for Measuring the Efficiency of Promotions in FMCG Retail

Joanna Henzel*, Marek Sikora‖
Department of Computer Networks and System
Faculty of Automatic Control, Electronics and Computer Science
Silesian University of Technology
ul. Akademicka 16, 44-100 Gliwice, Poland
Email: *joanna.henzel@polsl.pl, ‖marek.sikora@polsl.pl

***Abstract*—In the paper, a problem of forecasting promotion efficiency is raised. The authors propose a new approach, using the gradient boosting method for this task. Six performance indicators are introduced to capture the promotion effect. For each of them, within predefined groups of products, a model was trained. A description of using these models for forecasting and optimising promotion efficiency is provided. Data preparation and hyperparameters tuning processes are also described. The experiments were performed for three groups of products from a large grocery company.**

I. Introduction

FOOD retailing is an industry that most people have contact with. It provides products which are necessary for everyday life. Mostly, food is bought on an ongoing basis and, because of this, precise planning of logistics, chain supplies and sales is very important. Because of the characteristics of sale of these products, they are often called *fast-moving consumer goods* (FMCG).

On the market, many retailers offering FMCG products are available, therefore it is crucial to remain competitive. One way to do this is to offer products in promotion. The importance of creating promotions in the FMCG sector can be proven by seeing the amount of money that are spent on this purpose – in 2014 it was $1 trillion every year as it was mentioned in [1]. Therefore, it is necessary to forecast the promotion effect and plan them with equal importance as a regular sale.

In some cases promotions are planned based on *judgmental forecasting* or using simple baseline statistical forecast with a judgmental adjustment [2]. It means that the promotion planning process is often done manually. However, studies have shown that using only these kinds of forecasting methods may bring bias [3]. A better idea may be to use more advanced methods that rely mostly on knowledge that comes from historic data. Very little has been written about using Machine Learning (ML) methods for the problem of promotion optimisation and forecasting promotion effect.

The objective of this paper is to propose a new way of forecasting promotion effect using the gradient boosting method. Six different indicators are presented in order to capture the efficiency of promotions. The paper describes an advanced data preparation process. Among three groups of products, a model for each indicator was trained, examined and the optimisation of hyperparameters was conducted. The paper also describes how to use the created models in order to perform optimisation of promotions to get better outcome of the forecast. The paper is organised as follows: the next section provides the review of literature and related works, section III describes problem statement and presents proposed indicators. Afterwards, the data preparation process is presented, followed by the experiments explanation. The paper ends with some conclusions and discussion of the results.

II. Related Works

Sales forecasting plays an important part in planning and managing many commercial enterprises, including those connected with the retail sector.

Traditionally, forecasting was made using statistical methods, for example: exponential smoothing [4], moving average and the Auto-regressive Integrated Moving Average (ARIMA) model. Well known and widely used is SARIMA – seasonal auto-regressive integrated moving average. Some improvements of this method were proposed regarding the problem of sales forecasting in the papers [5] and [6].

Over time, more complex methods were used and evaluated in the field of sales forecasting. In [7] a comparison of various linear and non-linear models for this task was conducted. The best obtained model was the neural network built on deseasonalized time series data. The results suggested that non-linear models should be highly considered when dealing with modelling retail sales. Another neural network algorithm regarding forecasting retail sales which was used for this task was back-propagation neural network (BPNN) [8]. Evolutionary neural networks (ENN) were also considered in [9]. The use of the extreme learning machine (ELM) algorithm was also investigated in this area, for example in the papers [10], [11] and [12]. Also, a successful proposition of adding linguistic knowledge in the forecasting process using linear regression has been proposed in [13]. In the paper [14] an interesting forecast technique was presented. The authors combined pre-purchase online search data with economic variables to predict monthly car sales.

An important part of retail forecasting is making sales forecasts for short shelf-life food products, which are very often referred to as Fast-Moving Consumer Goods (FMCG). It is an even more complex task, because the additional products, whose sales may be overestimated, cannot be stored for a very long time in the shop. In the paper [15] a radial basis function (RBF) neural network and a designed genetic algorithm were successfully used for forecasting the sales of fresh milk. In the aspect of FMCG, the authors of [16] showed benefits of applying Machine Learning methods in creating demand forecasting models. The use of the Autoregressive Distributed Lag model was presented in the paper [17]. The authors of [18] proposed using the Dynamic Artificial Neural Network for food sales forecasting for one of multiplexes in India. In the paper [19], different classifiers were analysed and a proposition of combining various forecasting models using neural network was presented in order to improve results for forecasting demands of warehouses. Experiments were performed on real sales data of a national dried fruits and nuts company from Turkey.

Decision and regression tree-based methods were also taken into consideration regarding the sales forecasting. A hybrid method of k-means algorithm and C4.5 algorithm (decision tree classifier) was shown in [20]. In the paper [21] a comparison of different Machine Learning Techniques was conducted regarding sales-forecasting of retail stores. The authors concluded that boosting algorithms gave better results than the regular regression ones. For them, the best results were obtained for the GradientBoost algorithm and the XGBoost implementation has been used in order to increase the accuracy.

Forecasting sales during promotions is a very challenging task as it was mentioned in [2]. In this paper authors pointed out that usually the promotional effect was estimated by combining simple statistical forecasting methods and adding judgmental adjustment, which could lead to miscalculations.

The research about effectiveness of promotions has been conducted for a long time, mostly in the marketing research area and it is described in the practitioner literature. This problem was raised in [22] and [23]. The authors of [1] proposed a new formula for the promotion optimisation problem in the FMCG industry. Although these works concerned estimating the effectiveness of promotions, all of them focus on domain knowledge and do not use machine learning techniques for this task.

Multiple models for forecasting the demand during promotion periods were tested in the paper [24]. The use of PCA and pooled regression was presented in the paper [25] in order to predict sales in the presence of promotions. In the case of direct marketing, machine learning methods were compared and tested in the paper [26]. Interesting findings are presented in [27]. The authors showed that simple statistical methods performed very well for data without promotions. For periods with promotions more advanced methods had to be used. In this paper, regression trees were used for grocery sales forecasting.

To the best of our knowledge, the tree boosting algorithm, especially the extreme gradient boosting (XGBoost) algorithm, has not been used to forecast the effect of promotions and to optimise the promotion itself. XGBoost was introduced in [28]. It is a well known fact that XGBoost is highly effective for a vast range of classification and regression problems. It was, for example, used in the following areas: medicine [29], fault detection [30], finances [31], accident detection [32], and many others.

XGBoost implementation has a wide array of hyper-parameters. In order to obtain the best results, optimisation of those parameters can be performed. The most commonly used methods are random search (RS) and Bayesian Tree Parzen Estimator (TPE). These methods were used in [33] and [34]. Hyper-parameters optimisation was done using Bayesian optimisation, random search, grid search, and manual search in the paper [35].

III. Problem statement

In different industries, promotions may have various characteristics. For example, in fashion retail it is noticeable that promotions take place mostly in specific periods during the year – at the end of the fashion seasons. The situation is different in grocery retail business. Multiple promotions can be observed at the same time and they are changing very rapidly. Also, alongside the regular promotions, we can distinguish promotions related to holidays and special days (e.g. Christmas, Easter or St. Valentine's Day) and discounts that are caused by upcoming expiration date.

The purpose of the promotions may be not so obvious. They should give a company bigger profit, but it is not equivalent to the willingness to sell as much as possible of a promoted product. Of course, selling is one of the components of a successful promotion but not the only one. For example, a grocery retail company that set up a promotion does not want customers to buy only the promoted product but wants clients to buy also multiple different products alongside that may be in their regular prices.

In order to capture the effectiveness of each promotion, six different indicators are proposed:

- Average number of sold units or kilograms each day (shortcut: Avg. Amount) – This indicator shows how many units or kilograms of the promoted product, on average, were sold during the promotion each day.
- Average number of receipts with the promoted product (shortcut: Avg. Nb. Receipts) – The indicator explains in how many baskets the promoted product appeared, on average, each day during the promotion. It can be treated as an indicator of how many customers bought the product each day.

- AVERAGE VALUE OF A BASKET CONTAINING THE PROMOTED PRODUCT (shortcut: AVG. BASKET) – This indicator says what an average value of a basket was where the promoted product appeared. Assuming that customers went for shopping with the will to buy the specific product in promotion, the indicator says how much money they spent in total. The higher the indicator, the more products were bought or the more expensive products were chosen.
- AVERAGE VALUE OF A BASKET CONTAINING THE PROMOTED PRODUCT BUT DISREGARDING THE VALUE OF THE PROMOTED PRODUCT (shortcut: AVG. BASKET WITHOUT ITEM) – This indicator is very similar to the previous one. It shows what an average value of a basket was where the promoted product appeared but the value of the promoted product was not taken into account. It means that this indicator is equal to 0 if the customer buys only the promoted product.
- AVERAGE NUMBER OF UNIQUE PRODUCTS IN THE BASKET (shortcut: AVG. NB. UNIQUE ITEMS) – It says how varied the basket is. The higher the value of the indicator, the better – it means that the customer not only bought a specific product but also many others.
- AVERAGE NUMBER OF THE BASKETS (shortcut: AVG. NB. CLIENTS) – The indicator shows how many, on average, transactions were performed each day during the promotion. It does not matter if the customer bought a promoted product or not.

The reason for choosing the following indicators is that the information they carry is of interest to a company operating a large international retail shop chain and with which we collaborated during the research process.

The values of indicators are calculated per promotion. It means that each promotion can be described by the 6 proposed indicators.

These indicators may seem very similar, because the differences between them are very subtle. In order to show their utility, some examples are introduced:

1) 100 kg of apples were sold during the promotion. The indicator AVERAGE NUMBER OF SOLD UNITS OR KILOGRAMS EACH DAY tells us about it, but it does not give an information if this amount was bought by one person or by 50 people who bought 2 kg on average. This information will be provided by the AVERAGE NUMBER OF RECEIPTS WITH THE PROMOTED PRODUCT.
2) The average value of the basket, with a product that was in promotion, was 50$. It is the value of the indicator AVERAGE VALUE OF A BASKET CONTAINING PROMOTED PRODUCT. Now we may want to know if the rest of the products were a big part of the basket (e.g. 80 %) or only an addition to the promoted product (e.g. 10 % of the total value). The AVERAGE VALUE OF A BASKET CONTAINING THE PROMOTED PRODUCT BUT DISREGARDING THE VALUE OF THE PROMOTED PRODUCT gives this information. We also might want to know if the customers, on average, bought 2 unique products, that gave the value of 50 $, or they bought 25 unique products – the indicator AVERAGE NUMBER OF UNIQUE PRODUCTS IN THE BASKET is proposed in order to capture this.

Each of the proposed indicators are *gain measures*. It means that the higher the value, the better is the promotion. They can be inversely correlated – for example, if the price is very low, clients may buy a lot of the specific product but the diversity of products inside the basket may be very poor.

The proposed indicators describe each promotion very precisely. Knowing the value of each of them, the evaluation of the promotions can be performed. What is even more interesting, is the evaluation of future promotions so it is connected with the promotions planning. By setting up the features of the future promotion, it is possible to determine whether the predicted effect will be satisfying.

The forecasting of the promotion effect can be done for every product separately. Having the history of the promotions and their effects, we can model the characteristics of the promotion for the specific product and it is possible to predict what the effect in the future will be. Unfortunately, a number of past promotions for many products is small, so there are not many examples for training a model. Additionally, a question has been raised how to predict the promotion effect for a new product or an item that has never been in promotion. One solution may be to find similar products that have similar characteristic of sales. The problem is that it is difficult to assure that this will translate to similar characteristics of promotion effect. Another idea would be to create, based on domain knowledge, groups of products that act the same during the promotions. Then a model would be built for each of these groups. This issue, however, is out of scope of our paper.

The problem of forecasting indicators for unknown and rarely promoted products was solved by the authors – the products were grouped by the predefined categories, e.g. vegetables, fruits, dairy products or meat. It is assumed that the products within the group will act similarly during the promotion because they are akin to each other. Therefore, it is expected that the characteristics of the indicators describing the promotion effect will be similar for products within the group.

To summarize: a new approach to the problem of forecasting the promotion effect is to calculate a model for each of the 6 proposed indicators for each predefined category (group) of products.

IV. DATA PREPARATION

In developing models for promotions indicators and in experiments, data from a large grocery retail company were used (more than 500 stores). The data from groups: vegetables, fruits and dairy products were taken into account. Only regular promotions were investigated, therefore the promotions that happened before or during holidays were not included. Additionally, promotions that applied only when:

TABLE I
EXAMPLE OF RECORD DESCRIBING PROMOTION BEFORE PREPARATION

store ID	product	start date	end date	conditional attributes	value of indicator
10	pears	2018-01-22	2018-01-25	...	123.56

- multiple units were bought (type “buy 2 pay for 1"),
- minimum weight condition was met (type “buy minimum 5 kg and get 15 % off"),
- when combination of products was bought

were not taken into consideration. The same goes for products that had reduced prices because of the approaching best-before date. The reason for choosing only regular promotions was that they were the majority of all promotions and we were advised that non-regular promotions have a different characteristic that may bring a bias to the model. Also, in the examined data there were no promotions longer than 7 days. Promotions from the years 2015 to 2018 were used. Data for 2015 and part of 2016 were not completed, so there was a visibly smaller number of promotions at that period.

One record of data described one promotion in one store. Therefore, for example, if there would be a promotion on pears in the store with ID 10 from 2018-01-22 to 2018-01-25, the record, before preparation, would look like in table I.

A. Attributes

In the research, extended numbers of conditional attributes were taken into consideration when preparing data sets. A few main categories of the attributes can be distinguished:

- connected with price,
- connected with the time and duration of the promotion,
- describing the advertisement media (promotion channels),
- describing the store and its surroundings,
- describing the impact of other promotions.

In the first category, only 2 attributes were included: the price of a product and a change of the price.

Time attributes connected with the promotion were:

- number of days of the promotion,
- weekday of the first day of the promotion,
- attributes created based on the date of the first day of the promotion: year number, month number, day number, week number, number of a day in the year, and the season.

Considering information about promotion channels, binary attributes were added. They described if the promotion was advertised on TV, on the radio, on the Internet or in a different way.

Additionally, new variables describing combinations of the promotion channels were added to the data sets. For each combination, new attributes were created as a result of binary operations AND, OR and XOR (only when combination consisted of 2 elements). For example, if the undermentioned statements, were true, then a new variable got value 1, otherwise – 0.

- Promotion was on TV or on the radio. (OR operation)
- Promotion was on TV or on the radio or on the Internet. (OR operation)
- Promotion was on the TV and on the radio. (AND operation)
- Promotion was either on the Internet or on the radio. (XOR operation)

We can assume that promotions in similar stores (for example in small villages or in big cities) can have similar characteristics. For example, the customers in a rich city buy more expensive products in general, therefore the value of the basket is automatically higher than in other stores. The exemplary attributes that were used in order to capture these characteristics were:

- number of inhabitants within 1 km,
- number of inhabitants per 1 square km,
- number of inhabitants within a 5-minute driving range,
- unemployment rate,
- number of cars per 1,000 inhabitants,
- average monthly salary,
- tourism ratio, etc.

The last but not least, attributes connected with the impact of other promotions were added. As it was mentioned in the section III, promotions rarely ever take place one at a time. It is a possible situation, that a client that bought the considered product came to the store because of another promotion. It is impossible to capture clients’ intentions fully, but it can be assumed that the more promotions in the shop, the more clients will come. Because of this, the following attributes were added to the data:

- Number of all promotions in a store.
- Number of all promotions that were advertised on TV, radio or internet.
- Number of all promotions that were advertised on TV, radio, internet or in a different way.

B. Matching periods without promotions

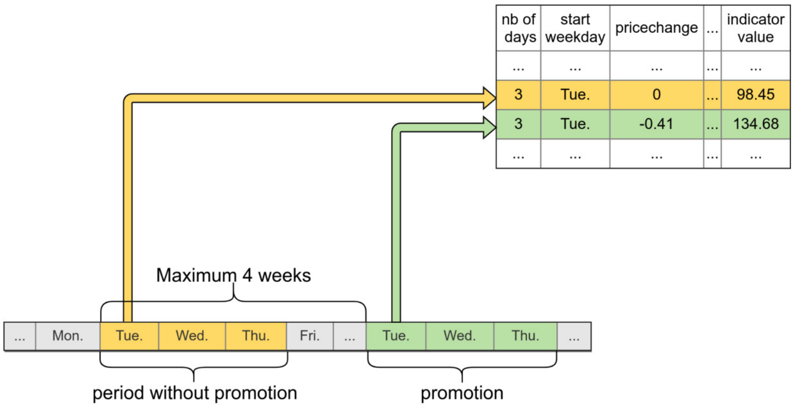
In order to capture the characteristics of products in the group, matching records without promotions were found for most of the records in the data set. The matching period had to meet the following conditions:

- It considered the same product as the promotion.
- It considered the same store.
- It had to last as many days as the considered promotion.
- It had to start on the same weekday as the promotion.
- The considered product was not in promotion on any given day.
- The period without promotion could occur maximum 4 weeks and minimum 1 week before the promotion.

The matching period was not found for all promotions because of the lack of meeting the requirements.

The illustration of finding the matching periods was shown in figure 1.

In the final data sets, records connected with periods without promotions were distinguished from promotions by having 0 value in an attribute describing the change of a price.



nb of days	start weekday	pricechange	...	indicator value
...	...	...	...	...
3	Tue.	0	...	98.45
3	Tue.	-0.41	...	134.68
...	...	...	...	...

Fig. 1. Finding matching record without promotion

C. Standardisation of the indicators

The standardisation of two proposed indicators was performed. These were:

- AVERAGE NUMBER OF SOLD UNITS OR KILOGRAMS EACH DAY and
- AVERAGE NUMBER OF RECEIPTS WITH THE PROMOTED PRODUCT.

The z-score standardisation was used, but for each product and each store separately. The reason for using standardisation for those indicators was that they were referring to the specific values connected with the sale characteristics of a considered product. For example, it is predictable that during promotions with 20% reduction, apples will be sold more than pomelos, because apples are cheaper and they are bought more often in general. The values of the indicator AVERAGE NUMBER OF SOLD UNITS OR KILOGRAMS EACH DAY will be from a different range for those products. This does not mean, however, that the impact of the 20% reduction does not affect in the same way the increase of sold units of apples and pomelos. In order to capture the general characteristics of products in a group, the standardisation of those indicators was performed.

V. EXPERIMENTS

The experiments of the proposed solution for problem of forecasting the promotion effect were conducted for the following categories of products: fruits, vegetables and dairy products. For each category and each proposed indicator, a forecasting model was constructed. In training data sets, records from 2015-2017 describing promotions and matching periods without promotions were included. In test data sets, records with promotions from 2018 were used. For all indicators within one group of products, conditional attributes in data were the same (described in subsection IV-A). The decision attributes were the values of the considered indicators.

When testing models, cross-validation was not performed. The reason for this is the fact that although the data sets were not typical time-series data, the records could be set in chronological order. Using cross-validation, the testing of a model might be performed on records preceding the training data.

XGBoost (eXtreme Gradient Boosting) [28] from the R package `xgboost` [36] implementation was used for training forecasting models. This gradient boosting framework was chosen because it is a well-known method, which get very good results when working with table-structured data. For example, among the 29 challenges winning solutions posted on a machine learning competition site named Kaggle in 2015, 17 solutions used XGBoost [28]. The experiments described in this paper were also based on tabular data, therefore using XGBoost was a justified idea. Additionally, the paper [21] showed that this algorithm has given the best results for sales-forecasting of retail stores in their experiments, so it was very likely to give good results also for the problem of forecasting the promotion effect in retail sector. In order to evaluate the models efficiency, the following error measures were used:

- *Mean Absolute Error (MAE)*:

$$MAE = \frac{\sum_{i=1}^{n} |F_i - A_i|}{n}$$

- *Root Mean Square Error (RMSE)*:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (F_i - A_i)^2}{n}}$$

- *Mean Absolute Percentage Error (MAPE)*:

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{A_i - F_i}{A_i} \right|$$

- *Weighted Mean Absolute Percentage Error (WMAPE)*:

$$WMAPE = \frac{\sum_{i=1}^{n} \mid A_i - F_i \mid}{\sum_{i=1}^{n} A_i}$$

where A_i is the actual value and F_i is the forecast value.

The mean absolute percentage error (MAPE) is very intuitive and easy to interpret, however it is meaningful only when the values are large. If the actual value is close to 0, the value of MAPE is approaching infinity and it gives uninterpreted results. In order to bypass these disadvantages,

TABLE II
RESULTS OF MODELS EFFECTIVENESS USING DEFAULT HYPERPARAMETERS

category	indicator	MAE	RMSE	MAPE	WMAPE
dairy products	AVG. AMOUNT	12.35	19.49	0.51	0.38
dairy products	AVG. NB. RECEIPTS	5.75	8.15	0.44	0.33
dairy products	AVG. BASKET	14.95	21.53	0.19	0.18
dairy products	AVG. BASKET WITHOUT ITEM	14.41	20.97	0.18	0.19
dairy products	AVG. NB. UNIQUE ITEMS	2.12	2.86	0.14	0.14
dairy products	AVG. NB. CLIENTS	165.67	247.54	0.10	0.10
fruits	AVG. AMOUNT	44.57	84.33	1.18	0.51
fruits	AVG. NB. RECEIPTS	27.92	45.22	0.87	0.39
fruits	AVG. BASKET	18.79	26.64	0.19	0.20
fruits	AVG. BASKET WITHOUT ITEM	17.40	25.09	0.19	0.20
fruits	AVG. NB. UNIQUE ITEMS	2.26	3.17	0.13	0.14
fruits	AVG. NB. CLIENTS	135.50	178.62	0.08	0.08
vegetables	AVG. AMOUNT	24.37	44.89	0.48	0.35
vegetables	AVG. NB. RECEIPTS	21.04	37.49	0.42	0.33
vegetables	AVG. BASKET	18.31	26.29	0.18	0.19
vegetables	AVG. BASKET WITHOUT ITEM	18.10	25.53	0.19	0.20
vegetables	AVG. NB. UNIQUE ITEMS	2.34	3.24	0.13	0.14
vegetables	AVG. NB. CLIENTS	171.61	229.00	0.10	0.10

TABLE III
RESULTS OF MODELS EFFECTIVENESS AFTER HYPERPARAMETERS OPTIMISATION

category	indicator	MAE	RMSE	MAPE	WMAPE
dairy products	AVG. AMOUNT	12.31	18.71	0.53	0.38
dairy products	AVG. NB. RECEIPTS	5.75	8.08	0.45	0.33
dairy products	AVG. BASKET	13.93	20.14	0.17	0.17
dairy products	AVG. BASKET WITHOUT ITEM	14.26	20.32	0.19	0.18
dairy products	AVG. NB. UNIQUE ITEMS	2.04	2.72	0.14	0.13
dairy products	AVG. NB. CLIENTS	129.75	177.15	0.08	0.08
fruits	AVG. AMOUNT	39.72	74.62	1.11	0.45
fruits	AVG. NB. RECEIPTS	24.87	39.39	0.85	0.35
fruits	AVG. BASKET	15.29	22.44	0.16	0.16
fruits	AVG. BASKET WITHOUT ITEM	14.73	21.78	0.17	0.17
fruits	AVG. NB. UNIQUE ITEMS	1.84	2.60	0.12	0.11
fruits	AVG. NB. CLIENTS	125.15	164.49	0.07	0.07
vegetables	AVG. AMOUNT	22.97	42.42	0.47	0.33
vegetables	AVG. NB. RECEIPTS	19.52	34.78	0.41	0.31
vegetables	AVG. BASKET	14.39	21.56	0.15	0.15
vegetables	AVG. BASKET WITHOUT ITEM	14.63	21.65	0.16	0.16
vegetables	AVG. NB. UNIQUE ITEMS	1.89	2.71	0.11	0.11
vegetables	AVG. NB. CLIENTS	135.57	178.51	0.08	0.08

a similar measure – WMAPE – was used. It is the sum of absolute errors divided by the sum of the actual values and it works well with smaller numbers. It is widely used in the retail sector.

Firstly, the XGBoost method was used with default hyperparameters. The results, obtained for test data sets, are presented in table II. For two indicators that were standardised (see subsection IV-C), error measures were calculated after changing forecasted, standardised values to the real values.

A. Optimisation

The optimisation of hyperparameters was performed for each created model. A *grid search* method was used. Six hyperparameters were optimised:

- *nrounds* – maximum number of boosting iterations; range: $[1, \infty)$.
- *base_score* – the initial prediction score of all instances; range: $(-\infty, \infty)$.
- *eta* – boosting learning rate; range: $[0, 1]$.
- *gamma* – minimum loss reduction required to make a further partition on a leaf node of the tree; range: $[0, \infty)$.
- *max_depth* – maximum depth of a tree; range: $[1, \infty)$.
- *subsample* – subsample ratio of the training instance; range: $(0, 1]$.

TABLE IV
VALUES OF HYPERPARAMETERS USED IN OPTIMISATION PROCESS

hyperparameter	tested values
nrounds	1, 21, 41, 61, 81, 101, 121, 141, 161, 181, 201
base_score	Depending on indicator values. Calculated as 11 quantiles from indicator values with the following probabilities: 0.0, 0.1, 0.2, ..., 0.9, 1.0.
eta	0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0
gamma	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
max_depth	1, 4, 7, 10, 13
subsample	0.0001, 0.1001, 0.2001, ..., 0.9001

TABLE V
RMSE IMPROVEMENT AFTER HYPERPARAMETER OPTIMISATION. $RMSE_{diff}$ IS A DIFFERENCE OF RMSE BEFORE OPTIMISATION (TABLE II) AND RMSE AFTER OPTIMISATION (TABLE III).

	$RMSE_{diff}$		
indicator	dairy products	fruits	vegetables
AVG. AMOUNT	0.78	9.72	2.47
AVG. NB. RECEIPTS	0.07	5.83	2.71
AVG. BASKET	1.38	4.19	4.74
AVG. BASKET WITHOUT ITEM	0.65	3.31	3.88
AVG. NB. UNIQUE ITEMS	0.13	0.56	0.53
AVG. NB. CLIENTS	70.39	14.13	50.49

A detailed description of the above parameters can be found in [36].

In the beginning, all possible sequences in which the hyperparameters could be optimised were determined. Six parameters were used, so 720 permutations were obtained. For example, the first permutation was *eta, base_score, gamma, max_depth, nrounds, subsample* – it means that at first the *eta* hyperparameter was optimised, then *base_score*, afterwards *gamma* and so on. In each permutation, each hyperparameter was changed several times in order to find the best value. The table IV shows values that were used in this process. After iterating through each hyperparameter, the best set of the hyperparameters values of the specific permutation was obtained. Having results for 720 permutations, the best among them was chosen. After this step, the best order of optimising the parameters and the best values for them were determined. In the end, the neighbourhood of the examined hyperparameters values were searched. It was performed in the order determined in the previous step (the order of the best permutation). The optimisation was performed using the validation set that was extracted from the training data set. The flowchart of the described optimisation process is shown in figure 2. The RMSE measure was used as the optimisation criterion.

The results of models efficiency, calculated for the test data sets after hyperparameters optimisation, were shown in table III. It can be observed that for most of the models metrics, the optimisation has given better results than for default models. The details can be seen by comparing table II and table III. The optimisation was carried out based on the RMSE measure. The improvement of this metric was observed for every examined case. The table V shows the exact results.

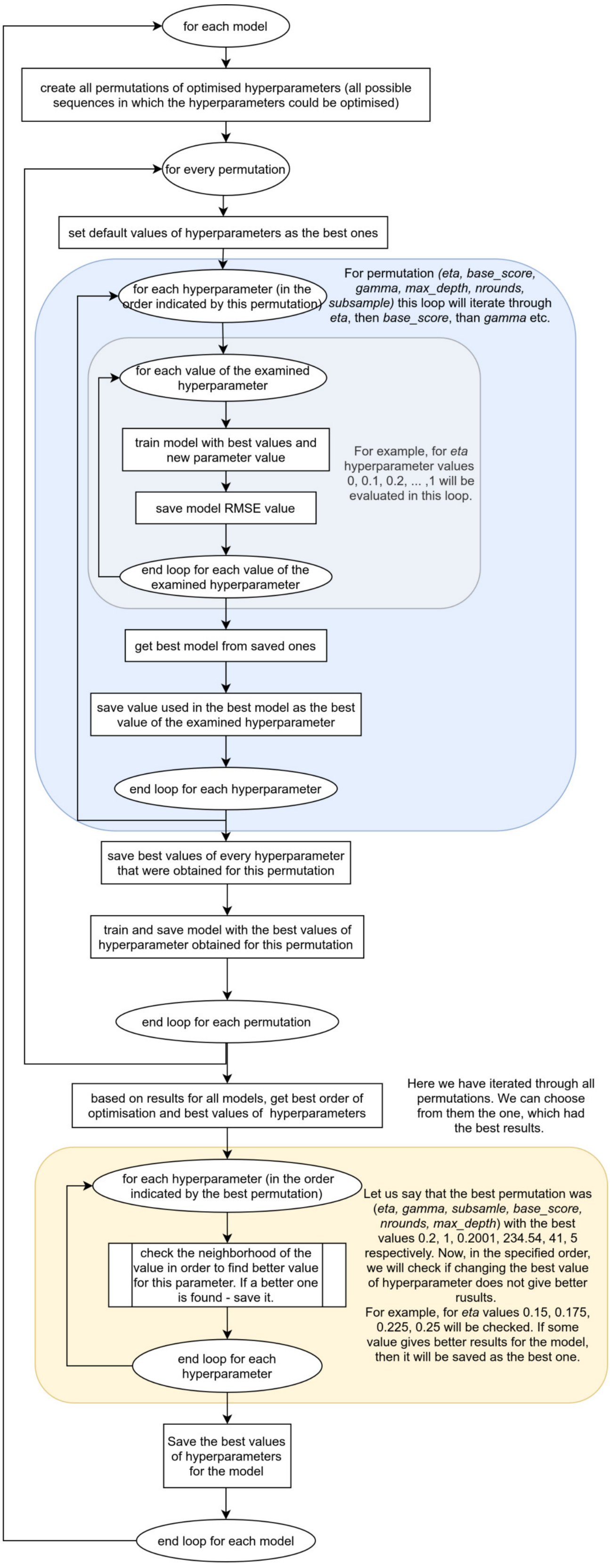

for each model
create all permutations of optimised hyperparameters (all possible sequences in which the hyperparameters could be optimised)
for every permutation
set default values of hyperparameters as the best ones
for each hyperparameter (in the order indicated by this permutation)
For permutation (eta, base_score, gamma, max_depth, nrounds, subsample) this loop will iterate through eta, then base_score, than gamma etc.
for each value of the examined hyperparameter
train model with best values and new parameter value
For example, for eta hyperparameter values 0, 0.1, 0.2, ... ,1 will be evaluated in this loop.
save model RMSE value
end loop for each value of the examined hyperparameter
get best model from saved ones
save value used in the best model as the best value of the examined hyperparameter
end loop for each hyperparameter
save best values of every hyperparameter that were obtained for this permutation
train and save model with the best values of hyperparameter obtained for this permutation
end loop for each permutation
based on results for all models, get best order of optimisation and best values of hyperparameters
Here we have iterated through all permutations. We can choose from them the one, which had the best results.
for each hyperparameter (in the order indicated by the best permutation)
Let us say that the best permutation was (eta, gamma, subsamle, base_score, nrounds, max_depth) with the best values 0.2, 1, 0.2001, 234.54, 41, 5 respectively. Now, in the specified order, we will check if changing the best value of hyperparameter does not give better rusults.
For example, for eta values 0.15, 0.175, 0.225, 0.25 will be checked. If some value gives better results for the model, then it will be saved as the best one.
check the neighborhood of the value in order to find better value for this parameter. If a better one is found - save it.
end loop for each hyperparameter
Save the best values of hyperparameters for the model
end loop for each model

Fig. 2. Flowchart of the hyperparameter optimisation process.

VI. Conclusion and Discussion

Promotions play an important role in the retail sector. When performed suitably, they can give a company bigger profit and bring in more clients to the store.

This study has attempted to introduce a new method of planning and forecasting future promotions using the XGBoost algorithm. Six unique indicators that measure the promotion efficiency were proposed in this paper. These indicators not only describe the sale of a specific product, but characterise the promotions in a much more profound way. Being able to forecast the value of each of them, promotions can be better planned. Indicators forecasts give information if the future promotion, with the given characteristics, like change of price or the weekday when it should start, is likely to be performed satisfactorily. If not, better attributes can be chosen.

In the paper the authors described the data sets preparation process with the use of extended and precisely chosen attributes that could be not so obvious to use. The authors also proposed a solution for forecasting the promotion effect for new, unknown products or products with a small number of past promotions. The models were developed for groups of products and not for each product separately. The experiments were performed for 3 groups: vegetables, dairy products and fruits. A model using XGBoost was developed for each indicator and each group of products. Additionally, the hypermarameters optimisation was performed in order to obtain better models accuracy. It is worth emphasizing that such optimisation can be carried out for any error measure.

The created models provide also a description of the features importance. Figure 3 shows a plot of 10 most important attributes of the model trained for indicator AVG. AMOUNT and dairy products. It can be observed that the change of a price and the price itself are the most important features that influence the amount of sold units during the promotions for this model. In the process of planning promotions, when the results of forecast are not satisfactory, one can tune, starting from these 2 attributes, the promotions characteristics in order to get better results. After making changes in the planned promotions, the predictions can be performed again. If the results are still not satisfying, the previous steps can be repeated. This way the process of optimising future promotions can be performed.

Five most important features for each indicator are presented below. The order, in which the attributes are listed below, was obtained by calculating average importance score of each feature taking into account the results of each group of products:

- AVG. AMOUNT: change of a price; day number (in the year); price; number of all promotions that are happening in the store and are advertised on TV, radio or Internet; day number (in the month).
- AVG. NB. RECEIPTS: change of a price; number of competitors; number of inhabitants within a 10-minute driving range; number of inhabitants within 1 km; number of inhabitants within 500 m.

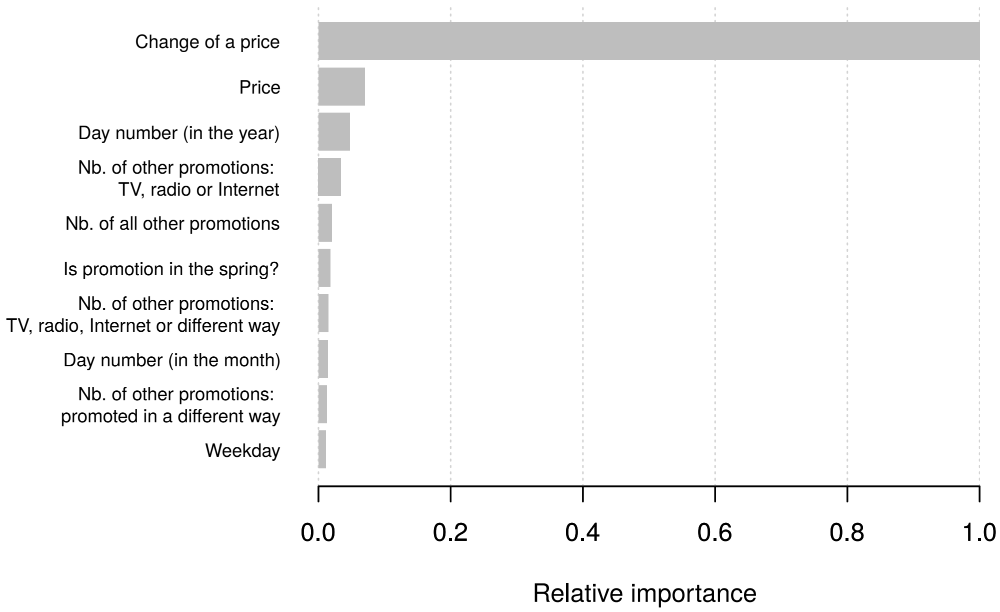


Fig. 3. Plot of feature importance for the model of the indicator AVG. AMOUNT for dairy products. The most important features are shown and the important values are represented as relative to the highest ranked feature.

- AVG. BASKET: price; number of inhabitants within 500 m; change of a price; day number (in the year); weekday.
- AVG. BASKET WITHOUT ITEM: price; number of inhabitants within 500 m; change of a price; day number (in the year); weekday.
- AVG. NB. UNIQUE ITEMS: number of inhabitants within 500 m; price; change of a price; weekday; distance from a competitor.
- AVG. NB. CLIENTS: number of inhabitants within 500 m; number of inhabitants within 1 km; number of inhabitants within a 5-minute driving range; purchasing rate; tourism ratio.

As it can be observed, not all features are possible to change in the process of the promotions planning. However, the ranking may suggest the order in which attribute values should be tuned to get better forecasting results. The most important features for AVG. NB. CLIENTS are not connected with promotions, so the conclusion can be drawn that this indicator is little affected by them.

Summarising the practical aspect of the research: using the presented methodology it is possible to train models for forecasting promotion efficiency. At the input of the models, the features of the future promotion are placed, including change of a price, promotion channels, store attributes and a number of days of the promotion. At the output of the models, the values of the indicators are obtained. They give information on whether the promotion will be successful.

The challenge for future research will be to investigate the efficiency of multi-target prediction methods for the problem of forecasting all six proposed indicators.

Also, there are a few possible additional applications that could benefit from a proposed method. Firstly, the models could be created not for predefined groups of products but for the products that have similar characteristic of a regular sale. For example, products that are bought in general much more often on Saturdays than during different weekdays could be in one group, the products bought steadily through all year long could be in a second group and another group would be products very popular during summer. It is possible that the character of a group might be not so obvious to define for a human. The clustering algorithms for time-series of historical sales could help in finding new groups of products. Another idea is to make models for each special kind of promotions, for example for promotions where multiple units of a product had to be bought. Some modifications of a presented method would need to be proposed because the definition of a matching period without promotion would need to be changed. Lastly, the forecast of an indicator could be obtained as a combination of forecasts from multiple models created for different groups in which this product belongs. The models would be trained in this same manner as described in this paper, only the definition of a group would change. These topics, however, require a great deal of further research.

In conclusion, this paper has shown a new way of planning and forecasting promotions using Machine Learning techniques. This, to our knowledge, is the first study to examine the utility of the Gradient Boosting method in the problem of forecasting the future promotion effect.

Acknowledgment

This work was partially supported by the European Union through the European Social Fund (grant POWR.03.05.00-00-Z305). The work was carried out in part within the project cofinanced by European Funds entitled "Decision Support and Knowledge Management System for the Retail Trade Industry (SensAI)" (POIR.01.01.01-00-0871/17-00).

References

[1] M. C. Cohen, N. H. Z. Leung, K. Panchamgam, G. Perakis, and A. Smith, "The impact of linear optimization on promotion planning," *Operations Research*, vol. 65, no. 2, pp. 446–468, 2017. doi: 10.1287/opre.2016.1573

[2] R. Fildes, P. Goodwin, and D. Önkal, "Use and misuse of information in supply chain forecasting of promotion effects," *International Journal of Forecasting*, vol. 35, no. 1, pp. 144–156, jan 2019. doi: 10.1016/j.ijforecast.2017.12.006

[3] S. Makridakis, "The art and science of forecasting An assessment and future directions," *International Journal of Forecasting*, vol. 2, no. 1, pp. 15–39, 1986. doi: 10.1016/0169-2070(86)90028-2

[4] E. S. Gardner Jr., "Exponential smoothing: The state of the art," vol. 4, no. October 1983, pp. 1–28, 1985.

[5] T.-M. Choi, Y. Yu, and K.-F. Au, "A hybrid SARIMA wavelet transform method for sales forecasting," *Decision Support Systems*, vol. 51, no. 1, pp. 130–140, apr 2011. doi: 10.1016/j.dss.2010.12.002

[6] N. S. Arunraj and D. Ahrens, "A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting," *International Journal of Production Economics*, vol. 170, pp. 321–335, dec 2015. doi: 10.1016/j.ijpe.2015.09.039

[7] C. W. Chu and G. P. Zhang, "A comparative study of linear and nonlinear models for aggregate retail sales forecasting," *International Journal of Production Economics*, vol. 86, no. 3, pp. 217–231, dec 2003. doi: 10.1016/S0925-5273(03)00068-9

[8] C. Y. Chen, W. I. Lee, H. M. Kuo, C. W. Chen, and K. H. Chen, "The study of a forecasting sales model for fresh food," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7696–7702, dec 2010. doi: 10.1016/j.eswa.2010.04.072

[9] K.-F. Au, T.-M. Choi, and Y. Yu, "Fashion retail forecasting by evolutionary neural networks," *International Journal of Production Economics*, vol. 114, no. 2, pp. 615 – 630, 2008. doi: 10.1016/j.ijpe.2007.06.013

[10] Z.-L. Sun, T.-M. Choi, K.-F. Au, and Y. Yu, "Sales forecasting using extreme learning machine with applications in fashion retailing," *Decision Support Systems*, vol. 46, no. 1, pp. 411–419, 2008. doi: 10.1016/j.dss.2008.07.009

[11] M. Xia, Y. Zhang, L. Weng, and X. Ye, "Fashion retailing forecasting based on extreme learning machine with adaptive metrics of inputs," *Knowledge-Based Systems*, vol. 36, pp. 253–259, dec 2012. doi: 10.1016/j.knosys.2012.07.002

[12] Y. Yu, T.-M. Choi, and C.-L. Hui, "An intelligent fast sales forecasting model for fashion products," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7373–7379, jun 2011. doi: 10.1016/j.eswa.2010.12.089

[13] K. Kaczmarek and O. Hryniewicz, "Linguistic knowledge about temporal data in bayesian linear regression model to support forecasting of time series," in *2013 Federated Conference on Computer Science and Information Systems, FedCSIS 2013*, 2013. ISBN 9781467344715 pp. 651–654.

[14] P. Wachter, T. Widmer, and A. Klein, "Predicting automotive sales using pre-purchase online search data," in *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems*, 2019. doi: 10.15439/2019F239 pp. 569–577.

[15] P. Doganis, A. Alexandridis, P. Patrinos, and H. Sarimveis, "Time series sales forecasting for short shelf-life food products based on artificial neural networks and evolutionary computing," *Journal of Food Engineering*, vol. 75, no. 2, pp. 196–204, jul 2006. doi: 10.1016/j.jfoodeng.2005.03.056

[16] E. Tarallo, G. K. Akabane, C. I. Shimabukuro, J. Mello, and D. Amancio, "Machine learning in predicting demand for fast-moving consumer goods: An exploratory research," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 737–742, 2019. doi: 10.1016/j.ifacol.2019.11.203

[17] T. Huang, R. Fildes, and D. Soopramanien, "The value of competitive information in forecasting FMCG retail product sales and the variable selection problem," *European Journal of Operational Research*, vol. 237, no. 2, pp. 738–748, sep 2014. doi: 10.1016/j.ejor.2014.02.022

[18] V. Adithya Ganesan, S. Divi, N. B. Moudhgalya, U. Sriharsha, and V. Vijayaraghavan, "Forecasting food sales in a multiplex using dynamic artificial neural networks," in *Advances in Intelligent Systems and Computing*, vol. 944. Springer Verlag, 2020. doi: 10.1007/978-3-030-17798-0_8. ISBN 9783030177973. ISSN 21945365 pp. 69–80.

[19] I. Islek and S. Gunduz Oguducu, "A decision support system for demand forecasting based on classifier ensemble," in *Communication Papers of the 2017 Federated Conference on Computer Science and Information Systems*, 2017. doi: 10.15439/2017F224 pp. 35–41.

[20] S. Thomassey and A. Fiordaliso, "A hybrid sales forecasting system based on clustering and decision trees," *Decision Support Systems*, vol. 42, no. 1, pp. 408–421, oct 2006. doi: 10.1016/j.dss.2005.01.008

[21] A. Krishna, V. Akhilesh, A. Aich, and C. Hegde, "Sales-forecasting of retail stores using machine learning techniques," in *Sales-forecasting of Retail Stores using Machine Learning Techniques*. IEEE, 2018. doi: 10.1109/CSITSS.2018.8768765. ISBN 9781538660782 pp. 160–166.

[22] R. C. Blattberg and A. Levin, "Modelling the effectiveness and profitability of trade promotions," *Marketing Science*, vol. 6, no. 2, pp. 124–146, 1987. doi: 10.1287/mksc.6.2.124

[23] J. Zhang and M. Wedel, "The effectiveness of customized promotions in online and offline stores," *Journal of Marketing Research*, vol. 46, no. 2, pp. 190–206, 2009. doi: 10.1509/jmkr.46.2.190

[24] K. H. Van Donselaar, J. Peters, A. De Jong, and R. Broekmeulen, "Analysis and forecasting of demand during promotions for perishable items," *International Journal of Production Economics*, vol. 172, pp. 65–75, feb 2016. doi: 10.1016/j.ijpe.2015.10.022

[25] J. R. Trapero, N. Kourentzes, and R. Fildes, "On the identification of sales forecasting models in the presence of promotions," *Journal of the Operational Research Society*, vol. 66, no. 2, pp. 299–307, 2015. doi: 10.1057/jors.2013.174

[26] G. Cui, M. L. Wong, and H. K. Lui, "Machine learning for direct marketing response models: Bayesian networks with evolutionary programming," *Management Science*, vol. 52, no. 4, pp. 597–612, 2006. doi: 10.1287/mnsc.1060.0514

[27] Ö. G. Ali, S. Sayin, T. van Woensel, and J. Fransoo, "SKU demand forecasting in the presence of promotions," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12 340–12 348, dec 2009. doi: 10.1016/j.eswa.2009.04.052

[28] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, vol. KDD '16. ACM, 2016. doi: 10.1145/2939672.2939785. ISBN 9781450342322 pp. 785–794. [Online]. Available: http://doi.acm.org/10.1145/2939672.2939785

[29] L. Torlay, M. Perrone-Bertolotti, E. Thomas, and M. Baciu, "Machine learning–XGBoost analysis of language networks to classify patients with epilepsy," *Brain Informatics*, vol. 4, no. 3, pp. 159–169, sep 2017. doi: 10.1007/s40708-017-0065-7

[30] D. Zhang, L. Qian, B. Mao, C. Huang, B. Huang, and Y. Si, "A data-driven design for fault detection of wind turbines using random forests and XGboost," *IEEE Access*, vol. 6, pp. 21 020–21 031, mar 2018. doi: 10.1109/ACCESS.2018.2818678

[31] J. Nobre and R. F. Neves, "Combining principal component analysis, discrete wavelet transform and XGBoost to trade in the financial markets," *Expert Systems with Applications*, vol. 125, pp. 181–194, jul 2019. doi: 10.1016/j.eswa.2019.01.083

[32] A. B. Parsa, A. Movahedi, H. Taghipour, S. Derrible, and A. K. Mohammadian, "Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis," *Accident Analysis and Prevention*, vol. 136, p. 105405, mar 2020. doi: 10.1016/j.aap.2019.105405

[33] Y. Wang and X. S. Ni, “A XGBoost risk model via feature selection and Bayesian hyper-parameter optimization,” *International Journal of Database Management Systems*, vol. 11, no. 1, pp. 1–17, jan 2019. [Online]. Available: http://arxiv.org/abs/1901.08433

[34] M. Nishio, M. Nishizawa, O. Sugiyama, R. Kojima, M. Yakami, T. Kuroda, and K. Togashi, “Computer-aided diagnosis of lung nodule using gradient tree boosting and Bayesian optimization,” *PLoS ONE*, vol. 13, no. 4, apr 2018. doi: 10.1371/journal.pone.0195875

[35] Y. Xia, C. Liu, Y. Y. Li, and N. Liu, “A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring,” *Expert Systems with Applications*, vol. 78, pp. 225–241, jul 2017. doi: 10.1016/j.eswa.2017.02.017

[36] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou, M. Li, J. Xie, M. Lin, Y. Geng, and Y. Li, *xgboost: Extreme Gradient Boosting*, 2019, r package version 0.90.0.2. [Online]. Available: https://CRAN.R-project.org/package=xgboost

Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 69–75
DOI: 10.15439/2020F55
ISSN 2300-5963 ACSIS, Vol. 21

Extended Distributive Contact Lattices and Extended Contact Algebras

Tatyana Ivanova
Bulgarian Academy of Sciences
Institute of Mathematics and Informatics
in Sofia
1113, Sofia, Bulgaria, Acad. Georgi Bonchev Str., Block 8
Email: tatyana.ivanova@math.bas.bg

***Abstract*—The notion of contact algebra is one of the main tools in mereotopology. This paper considers a generalisation of contact algebra (called extended distributive contact lattice) and the so called extended contact algebras which extend the language of contact algebras by the predicates covering and internal connectedness.**

I. Introduction

IN CLASSICAL Euclidean geometry the notion of point is taken as one of the basic primitive notions. In contrast, region-based theory of space (RBTS) has as primitives the more realistic notion of region (abstraction of physical body) together with some basic relations and operations on regions. Some of these relations are mereological - part–of, overlap and its dual underlap. Other relations are topological - contact, nontangential part-of, dual contact and some others definable by means of the contact and part-of relations. This is one of the reasons that the extension of mereology with these new relations is commonly called mereotopology. There is no clear difference in literature between RBTS and mereotopology. The origin of RBTS goes back to Whitehead and de Laguna ([30], [21]). According to Whitehead points, as well as the other primitive notions in Euclidean geometry such as lines and planes, do not have separate existence in reality and because of this are not appropriate for primitive notions. Survey papers on RBTS are [26], [7], [16], [22] (also the handbook [1] and [5], containing some logics of space).

RBTS has applications in computer science because of its simpler way of representing of qualitative spatial information. Mereotopology is used in the field of Artificial Intelligence, called Knowledge Representation (KR). RBTS initiated a special field in KR, called Qualitative Spatial Representation and Reasoning (QSRR) which is appropriate for automatization [6], [24]. RBTS is applied in geographic information systems, robot navigation. Surveys concerning various applications are for example [8], [9] and the book [17] (also special issues of Fundamenta Informaticae [11] and the Journal of Applied Nonclassical Logics [3]). One of the most popular systems in Qualitative Spatial Rrepresentation and Reasoning is the Region Connection Calculus (RCC) [23].

The notion of contact algebra is one of the main tools in RBTS. This notion appears in the literature under different names and formulations as an extension of Boolean algebra with some mereotopological relations [29], [25], [28], [27], [7], [15], [10], [14]. The simplest system, called just a contact algebra was introduced in [10] as an extension of Boolean algebra $B = (B, 0, 1, \cdot, +, *)$ with a binary relation C called *contact* and satisfying five simple axioms:

(C1) If aCb, then $a \neq 0$,
(C2) If aCb and $a \leq c$ and $b \leq d$, then cCd,
(C3) If $aC(b + c)$, then aCb or aCc,
(C4) If aCb, then bCa,
(C5) If $a \cdot b \neq 0$, then aCb.

The elements of the Boolean algebra are called regions and are considered as analogs of physical bodies. Boolean operations are considered as operations for constructing new regions from given ones. The unit element 1 symbolizes the region containing as its parts all regions, and the zero element 0 symbolizes the empty region.

Topological spaces are among the first mathematical models of space, applied in practice. Standard models of contact algebras are topological. Let X be a topological space and a be its subset. We say that a is regular closed if a is the closure of the interior of a. It is a well known fact that the set $RC(X)$ of all regular closed subsets of X is a Boolean algebra with respect to the following definitions: $a \leq b$ iff $a \subseteq b$, 0 is the empty set, 1 is the set X, $a + b = a \cup b$, $a \cdot b = Cl\,Int\,(a \cap b)$, $a^* = Cl(X \setminus a)$. If we define a contact by taking aCb iff $a \cap b$ is nonempty, then we obtain a contact algebra related to X, namely $\underline{RC(X)} = (RC(X), \leq, 0, 1, \cdot, +, *, C)$ ([10], Example 2.1).

This paper is mostly a summary of the work, contained in [20], [19], [18], [4]. The results, concerning quantifier-free first-order logics for extended contact algebras, are novel and will be submitted as a paper with title “Quantifier-free first-order logics for extended contact algebras”.

II. Extended distributive contact lattices (EDCL)

Sometimes there is a problem in the motivation of the operation Boolean complement $(*)$ of contact algebra. A question arises - if a represents some region, what region does a^* represent - it depends on the universe in which we consider a. Moreover if a represents a physical body, then a^* is unnatural - such a physical body does not exist. Because of this we can drop the operation of complement and replace the Boolean part of a contact algebra with distributive lattice. First steps in this direction were made in [12], [13], introducing the notion of distributive contact lattice. In a distributive contact lattice the only mereotopological relation is the contact relation. Non-tangential inclusion and dual contact are not included in the language. In [20], the language of distributive contact lattices is extended by considering these two relations as nondefinable primitives. An axiomatization is obtained of the theory consisting of the universal formulas in this more expressive language, true in all contact algebras. The structures, satisfying the axioms in question, are called extended distributive contact lattices (EDCL). The well known RCC-8 system of mereotopological relations is definable in the language of EDCL and is not definable in the language of distributive contact lattices.

EDCL is a generalization of contact algebra, defined in the following way:

Definition 2.1: [20] **Extended distributive contact lattice.** Let $\underline{D} = (D, \leq, 0, 1, \cdot, +, C, \widehat{C}, \ll)$ be a bounded distributive lattice with three additional relations $C, \widehat{C}, \ll$, called respectively *contact*, *dual contact* and *nontangential part-of*. The obtained system, denoted shortly by $\underline{D} = (D, C, \widehat{C}, \ll)$, is called *extended distributive contact lattice* (EDCL, for short) if it satisfies the axioms listed below.

Notations: if R is one of the relations $\leq, C, \widehat{C}, \ll$, then its complement is denoted by $\overline{R}$.

Axioms for C alone: The axioms (C1)-(C5) mentioned above.

Axioms for $\widehat{C}$ alone:

$(\widehat{C}1)$ If $a\widehat{C}b$, then $a, b \neq 1$,
$(\widehat{C}2)$ If $a\widehat{C}b$ and $a' \leq a$ and $b' \leq b$, then $a'\widehat{C}b'$,
$(\widehat{C}3)$ If $a\widehat{C}(b \cdot c)$, then $a\widehat{C}b$ or $a\widehat{C}c$,
$(\widehat{C}4)$ If $a\widehat{C}b$, then $b\widehat{C}a$,
$(\widehat{C}5)$ If $a + b \neq 1$, then $a\widehat{C}b$.

Axioms for $\ll$ alone:

$(\ll 1)$ $0 \ll 0$,
$(\ll 2)$ $1 \ll 1$,
$(\ll 3)$ If $a \ll b$, then $a \leq b$,
$(\ll 4)$ If $a' \leq a \ll b \leq b'$, then $a' \ll b'$,
$(\ll 5)$ If $a \ll c$ and $b \ll c$, then $(a + b) \ll c$,
$(\ll 6)$ If $c \ll a$ and $c \ll b$, then $c \ll (a \cdot b)$,
$(\ll 7)$ If $a \ll b$ and $(b \cdot c) \ll d$ and $c \ll (a + d)$, then $c \ll d$.

Mixed axioms:

$(MC1)$ If aCb and $a \ll c$, then $aC(b \cdot c)$,
$(MC2)$ If $a\overline{C}(b \cdot c)$ and aCb and $(a \cdot d)\overline{C}b$, then $d\widehat{C}c$,
$(M\widehat{C}1)$ If $a\widehat{C}b$ and $c \ll a$, then $a\widehat{C}(b + c)$,
$(M\widehat{C}2)$ If $a\overline{\widehat{C}}(b + c)$ and $a\widehat{C}b$ and $(a + d)\overline{\widehat{C}}b$, then dCc,
$(M \ll 1)$ If $a\overline{\widehat{C}}b$ and $(a \cdot c) \ll b$, then $c \ll b$,
$(M \ll 2)$ If $a\overline{C}b$ and $b \ll (a + c)$, then $b \ll c$.

Lemma 2.2: [20] Let (W, R) be a relational system with reflexive and symmetric relation R and let $\underline{D}$ be any collection of subsets of W which is a bounded distributive set-lattice with relations $C, \widehat{C}$ and $\ll$ defined as follows:
(Def C_R) aC_Rb iff $\exists x \in a$ and $\exists y \in b$ such that xRy;
(Def $\widehat{C}_R$) $a\widehat{C}_Rb$ iff $\exists x \notin a$ and $\exists y \notin b$ such that xRy;
(Def $\not\ll_R$) $a \not\ll_R b$ iff $\exists x \in a$ and $\exists y \notin b$ such that xRy.
Then $(\underline{D}, C_R, \widehat{C}_R, \ll_R)$ is an EDCL.

EDCL $\underline{D} = (D, C_R, \widehat{C}_R, \ll_R)$ over a relational system (W, R) is called *discrete EDCL*. If D is a set of all subsets of W then $\underline{D}$ is called a *full discrete EDCL*.

Corollary 2.3: [20] The axioms of the relations $C, \widehat{C}$ and $\ll$ are true in contact algebras.

Generalizing the Stone representation theorem for distributive lattices it is proved the following theorem.

Theorem 2.4: [20] **Relational representation theorem of EDCL.** Let $\underline{D} = (D, C, \widehat{C}, \ll)$ be an EDCL. Then there is a relational system $\underline{W} = (W, R)$ with reflexive and symmetric R and an embedding h into the EDCL of all subsets of W.

Corollary 2.5: [20] Every EDCL can be isomorphically embedded into a contact algebra.

In [20], it is obtained a new stronger form of the well-known in the theory of distributive lattices Filter-extension lemma. This stronger form is equivalent to the Axiom of Choice. This stronger form is used in the proof of the relational representation theorem for EDCL.

Lemma 2.6: [20] **Strong filter-extension Lemma.** Let F_0 be a filter, I_0 be an ideal and $F_0 \cap I_0 = \emptyset$. Then there exists a prime filter F such that $F_0 \subseteq F$, $(\forall x \in F)(x \notin I_0)$ and $(\forall x \notin F)(\exists y \in F)(x \cdot y \in I_0)$.

III. Topological representation theory of EDCL

In [20], are considered also some axiomatic extensions of EDCL yielding representations in T_1 and T_2 topological spaces.

Several additional axioms for EDCL are formulated which are adaptations for the language of EDCL of some known axioms considered in the context of contact algebras. The first new axioms for EDCL are the so called extensionality axioms for the definable predicates of overlap - $aOb \leftrightarrow_{def} a \cdot b \neq 0$ and underlap - $a\widehat{O}b \leftrightarrow_{def} a + b \neq 1$.
(Ext O) $a \nleq b \rightarrow (\exists c)(a \cdot c \neq 0$ and $b \cdot c = 0)$ - *extensionality of overlap*,
(Ext $\widehat{O}$) $a \nleq b \rightarrow (\exists c)(a{+}c = 1$ and $b{+}c \neq 1)$ - *extensionality of underlap*.

We say that a lattice is *O-extensional* if it satisfies (Ext O) and *U-extensional* if it satisfies (Ext $\widehat{O}$). Note that the conditions (Ext O) and (Ext $\widehat{O}$) are true in Boolean algebras but not always are true in distributive lattices.

The following additional axioms are considered too:

(Ext C) $a \neq 1 \to (\exists b \neq 0)(a\overline{C}b)$ - *C-extensionality*,

(Ext $\widehat{C}$) $a \neq 0 \to (\exists b \neq 1)(a\overline{\widehat{C}}b)$ - *$\widehat{C}$-extensionality*,

(Con C) $a \neq 0, b \neq 0$ and $a+b=1 \to aCb$ - *C-connectedness axiom* ,

(Con $\widehat{C}$) $a \neq 1, b \neq 1$ and $a \cdot b = 0 \to a\widehat{C}b$ - *$\widehat{C}$-connectedness axiom* ,

(Nor 1) $a\overline{C}b \to (\exists c,d)(c+d=1, a\overline{C}c$ and $b\overline{C}d)$,

(Nor 2) $a\overline{\widehat{C}}b \to (\exists c,d)(c \cdot b = 0, a\overline{\widehat{C}}c$ and $b\overline{\widehat{C}}d)$,

(Nor 3) $a \ll b \to (\exists c)(a \ll c \ll b)$.

(U-rich $\ll$) $a \ll b \to (\exists c)(b+c=1$ and $a\overline{C}c)$,

(U-rich $\widehat{C}$) $a\overline{\widehat{C}}b \to (\exists c,d)(a+c=1, b+d=1$ and $c\overline{C}d)$,

(O-rich $\ll$) $a \ll b \to (\exists c)(a \cdot c = 0$ and $c\overline{\widehat{C}}b)$,

(O-rich C) $a\overline{C}b \to (\exists c,d)(a \cdot c = 0, b \cdot d = 0$ and $c\overline{\widehat{C}}d)$.

Let $(D_1, C_1, \widehat{C}_1, \ll_1)$ and $(D_2, C_2, \widehat{C}_2, \ll_2)$ be two EDCL and D_1 is a substructure of D_2. It is valuable to know under what conditions we have equivalences of the form:

D_1 satisfies some additional axiom iff D_2 satisfies the same axiom.

Remark 3.1: [20] The importance of such conditions is related to the representation theory of EDCL satisfying some additional axioms. In general, if we have some embedding theorem for EDCL D satisfying a given additional axiom A, it is not known in advance that the lattice in which D is embedded also satisfies A. That is why it is good to have such conditions which automatically guarantee this. Below several such "good conditions" are formulated: dense and dual dense sublattice, C-separable sublattice.

Definition 3.1: [20] **Dense and dual dense sublattice.** Let D_1 be a distributive sublattice of D_2. D_1 is called a *dense* sublattice of D_2 if the following condition is satisfied:

(Dense) $(\forall a_2 \in D_2)(a_2 \neq 0 \Rightarrow (\exists a_1 \in D_1)(a_1 \leq a_2$ and $a_1 \neq 0))$.

If h is an embedding of the lattice D_1 into the lattice D_2 then we say that h is a *dense* embedding if the sublattice $h(D_1)$ is a dense sublattice of D_2.

Dually, D_1 is called a *dual dense* sublattice of D_2 if the following condition is satisfied:

(Dual dense) $(\forall a_2 \in D_2)(a_2 \neq 1 \Rightarrow (\exists a_1 \in D_1)(a_2 \leq a_1$ and $a_1 \neq 1))$.

If h is an embedding of the lattice D_1 into the lattice D_2 then we say that h is a *dual dense* embedding if the sublattice $h(D_1)$ is a dually dense sublattice of D_2.

(See [13] for some known characterizations of density and dual density in distributive lattices.)

For the case of contact algebras [26] and distributive contact lattices [13] the notion of C-separability is introduced as follows. Let D_1 be a substructure of D_2; we say that D_1 is a C-separable sublattice of D_2 if the following condition is satisfied:

(C-separable) $(\forall a_2, b_2 \in D_2)(a_2\overline{C}b_2 \Rightarrow (\exists a_1, b_1 \in D_1)(a_2 \leq a_1, b_2 \leq b_1, a_1\overline{C}b_1))$.

For the case of EDCL this notion is modified, adding two additional clauses corresponding to the relations $\widehat{C}$ and $\ll$ just having in mind the definitions of these relations in contact algebras. Namely

Definition 3.2: [20] **C-separability.** Let D_1 be a substructure of D_2; we say that D_1 is a *C-separable EDC-sublattice of* D_2 if the following conditions are satisfied:

(C-separability for C) -
$(\forall a_2, b_2 \in D_2)(a_2\overline{C}b_2 \Rightarrow (\exists a_1, b_1 \in D_1)(a_2 \leq a_1, b_2 \leq b_1, a_1\overline{C}b_1))$.

(C-separability for $\widehat{C}$) -
$(\forall a_2, b_2 \in D_2)(a_2\overline{\widehat{C}}b_2 \Rightarrow (\exists a_1, b_1 \in D_1)(a_2 + a_1 = 1, b_2 + b_1 = 1, a_1\overline{C}b_1))$.

(C-separability for $\ll$) -
$(\forall a_2, b_2 \in D_2)(a_2 \ll b_2 \Rightarrow (\exists a_1, b_1 \in D_1)(a_2 \leq a_1, b_2 + b_1 = 1, a_1\overline{C}b_1))$.

If h is an embedding of the lattice D_1 into the lattice D_2 then we say that h is a *C-separable embedding* if the sublattice $h(D_1)$ is a C-separable sublattice of D_2.

Theorem 3.3: [20] **Topological representation theorem for EDCL.** Let $\underline{D} = (D, C, \widehat{C}, \ll)$ be an EDCL. Then there exists a topological space X and an embedding of $\underline{D}$ into the contact algebra $RC(X)$ of regular closed subsets of X.

Definition 3.4: [20] **U-rich and O-rich EDCL.** Let $\underline{D} = (D, C, \widehat{C}, \ll)$ be an EDCL. Then:

(i) $\underline{D}$ is called U-rich EDCL if it satisfies the axioms (Ext $\widehat{O}$), (U-rich $\ll$) and (U-rich $\widehat{C}$).

(ii) $\underline{D}$ is called O-rich EDCL if it satisfies the axioms (Ext O), (O-rich $\ll$) and (O-rich $\widehat{C}$).

In [20], is developed the topological representation theory of U-rich EDCL. In a dual way can be developed the topological representation theory of O-rich EDCL.

Theorem 3.5: [20] **Topological representation theorem for U-rich EDCL.**
Let $\underline{D} = (D, C, \widehat{C}, \ll)$ be an U-rich EDCL. Then there exists a compact semiregular T_0-space X and a dually dense and C-separable embedding h of $\underline{D}$ into the Boolean contact algebra $RC(X)$ of the regular closed sets of X. Moreover:

(i) $\underline{D}$ satisfies (Ext C) iff $RC(X)$ satisfies (Ext C); in this case X is weakly regular.

(ii) $\underline{D}$ satisfies (Con C) iff $RC(X)$ satisfies (Con C); in this case X is connected.

(iii) $\underline{D}$ satisfies (Nor 1) iff $RC(X)$ satisfies (Nor 1); in this case X is κ-normal.

There is also a topological representation theorem of U-rich EDCL, satisfying (Ext C), in T_1-spaces.

Adding the axiom (Nor 1), it is obtained representability in compact T_2-spaces.

IV. Logics for EDCL

In [19], are considered a logic for EDCL and several extending it logics, corresponding to topological spaces possessing various additional properties. Completeness theorems are given with respect to both algebraic and topological semantics for these logics. It turns out that they are decidable.

It is considered the quantifier-free first-order language $\mathcal{L}$ which includes:

- constants: 0, 1;
- function symbols: $+$, $\cdot$;
- predicate symbols: $\leq$, C, $\widehat{C}$, $\ll$.

Every EDCL is a structure for $\mathcal{L}$.

It is considered the logic L with rule MP and the following axioms:

- the axioms of the classical propositional logic;
- the axiom schemes of distributive lattice;
- the axioms for C, $\widehat{C}$, $\ll$ and the mixed axioms of EDCL - considered as axiom schemes.

The following additional rules and an axiom scheme are considered:

(R Ext $\widehat{O}$) $\frac{\alpha\rightarrow(a+p\neq 1\vee b+p=1)\text{ for all variables }p}{\alpha\rightarrow(a\leq b)}$, where α is a formula, a, b are terms

(R U-rich $\ll$) $\frac{\alpha\rightarrow(b+p\neq 1\vee aCp)\text{ for all variables }p}{\alpha\rightarrow(a\overline{\ll}b)}$, where α is a formula, a, b are terms

(R U-rich $\widehat{C}$) $\frac{\alpha\rightarrow(a+p\neq 1\vee b+q\neq 1\vee pCq)\text{ for all variables }p,\ q}{\alpha\rightarrow a\widehat{C}b}$, where α is a formula, a, b are terms

(R Ext C) $\frac{\alpha\rightarrow(p\neq 0\rightarrow aCp)\text{ for all variables }p}{\alpha\rightarrow(a=1)}$, where α is a formula, a is a term

(R Nor1) $\frac{\alpha\rightarrow(p+q\neq 1\vee aCp\vee bCq)\text{ for all variables }p,\ q}{\alpha\rightarrow aCb}$, where α is a formula, a, b are terms

(Con C) $p\neq 0\wedge q\neq 0\wedge p+q=1\rightarrow pCq$

The additional axioms for EDCL (the axioms (Ext $\widehat{O}$), (U-rich $\ll$), (U-rich $\widehat{C}$), (Ext C), (Nor 1)) correspond to these rules.

Let L' be for example the extension of L with the rule (R Ext $\widehat{O}$) and the axiom scheme (Con C). Then we denote L' by $L_{ConC,Ext\widehat{O}}$ and call the axioms (Con C) and (Ext $\widehat{O}$) additional axioms, corresponding to L'. In a similar way we denote any extension of L with some of the considered additional rules and axiom scheme and in a similar way we define its corresponding additional axioms.

The following theorem is true

Theorem 4.1: [19] **Completeness theorem with respect to algebraic semantics.** Let L' be some extension of L with zero or more of the considered additional rules and axiom scheme. The following conditions are equivalent for any formula α:
(i) α is a theorem of L';
(ii) α is true in all EDCL, satisfying the additional axioms, corresponding to L'.

To every of the logics
1) L;
2) $L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C}}$;
3) $L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ExtC}$;
4) $L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ConC}$;
5) $L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},Nor1}$;
6) $L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ExtC,ConC}$;
7) $L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},Nor1,ConC}$;
8) $L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ExtC,Nor1}$;
9) $L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ExtC,ConC,Nor1}$.

is juxtaposed a class of topological spaces:
1) the class of all T_0, semiregular, compact topological spaces;
2) the class of all T_0, semiregular, compact topological spaces;
3) the class of all T_0, compact, weakly regular topological spaces;
4) the class of all T_0, semiregular, compact, connected topological spaces;
5) the class of all T_0, semiregular, compact, κ - normal topological spaces;
6) the class of all T_0, compact, weakly regular, connected topological spaces;
7) the class of all T_0, semiregular, compact, κ - normal, connected topological spaces;
8) the class of all T_0, compact, weakly regular, κ - normal topological spaces;
9) the class of all T_0, compact, weakly regular, connected, κ - normal topological spaces.

We have the following theorems

Theorem 4.2: [19] **Completeness theorem with respect to topological semantics.** Let L' be any of the considered above logics. The following conditions are equivalent for any formula α:
(i) α is a theorem of L';
(ii) α is true in all contact algebras over a topological space from the class, corresponding to L'.

Theorem 4.3: [19] (i) The logics
L,
$L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C}}$,
$L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ExtC}$,
$L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},Nor1}$,
$L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ExtC,Nor1}$
have the same theorems and are decidable;
(ii) The logics
$L_{ConC,U-rich\ll}$,
$L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ConC}$,
$L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ConC,Nor1}$,
$L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ExtC,ConC}$,
$L_{Ext\widehat{O},U-rich\ll,U-rich\widehat{C},ExtC,ConC,Nor1}$
have the same theorems and are decidable.

V. Extended contact algebras

The predicate internal connectedness (intuitively meaning that the interior is connected) cannot be defined in the language of contact algebras ([18], Proposition 2.1). So we consider *extended contact algebras:*

Definition 5.1: [18] **Extended contact algebra (ExtCA, for short)** is a system $\underline{B} = (B, \leq, 0, 1, \cdot, +, *, \vdash, C, c^o)$, where $(B, \leq, 0, 1, \cdot, +, *)$ is a nondegenerate Boolean algebra, $\vdash$ (*covering* or *extended contact*) is a ternary relation in B such that the following axioms are true:
(1) $a, b \vdash c \to b, a \vdash c$,
(2) $a \leq c \to a, b \vdash c$,
(3) $a, b \vdash x,\ a, b \vdash y,\ x, y \vdash c \to a, b \vdash c$,
(4) $a, b \vdash c \to a \cdot b \leq c$,
(5) $a, b \vdash c \to a + x, b \vdash c + x$,
C is a binary relation in B such that
(6) $aCb \leftrightarrow a, b \not\vdash 0$,
c^o (internal connectedness) is a unary predicate in B such that
(7) $c^o(a) \leftrightarrow \forall b \forall c(b \neq 0 \wedge c \neq 0 \wedge a = b + c \to b, c \not\vdash a^*)$.

ExtCAs extend the language of contact algebras by the predicate covering and the predicate internal connectedness. The internal connectedness is defined by the relation of covering ($c^o(a)$ iff $\forall b \forall c(b \neq 0 \wedge c \neq 0 \wedge a = b + c \to b, c \not\vdash a^*)$ ([18], Proposition 3.1)). Another motivation for considering the relation of covering is that by it we can define the property of two regions their intersection to be a region. Extended contact gives also the possibility to define the relation of contact. One of the motivations for adding the predicate internal connectedness is that by its help the property "existing of cavities in a physical body" can be defined: we have "a has cavities" if and only if "a^* is not internally connected". We cannot define "a has cavities" if and only if "the complement of a is not connected", using the predicate connectedness because the complement of a is not necessarily regular closed set i.e. element of the topological model of ExtCA. If we define "a has cavities" if and only if "a^* is not connected", this is wrong - if the cavity in the ball a touches its boundary, a^* is connected (and at the same time is not internally connected). Because of these reasons we need the predicate "internal connectedness" instead of "connectedness" for defining the property "existing of cavities in a physical body".

Primary semantics for ExtCAs is topological. Let X be a topological space. A topological ExtCA over X is the structure with universe the set $RC(X)$ of all regular closed subsets together with the following interpretations: $a \leq b$ iff $a \subseteq b$, $0 = \emptyset$, $1 = X$, $a \cdot b = Cl\, Int\, (a \cap b)$, $a + b = a \cup b$, $a^* = Cl\, (X \setminus a)$, $a, b \vdash c$ iff $a \cap b \subseteq c$, aCb iff $a, b \not\vdash \emptyset$, $c^o(a)$ iff $Int\, a$ is a connected subspace of X.

We have the following

Theorem 5.2: [18] **Topological representation theorem.** Let $\underline{B} = (B, \leq, 0, 1, \cdot, +, *, \vdash, C, c^o)$ be an ExtCA. Then there is a compact, semiregular, T_0 topological space X and an embedding of $\underline{B}$ into the topological ExtCA over X.

It is interesting also to consider a relational semantics for ExtCAs. This is done in [4].

Definition 5.3: [4] An *equivalence frame of type 2* is a relational structure of the form (W, R_1, R_2), where W is a nonempty set and R_1 and R_2 are equivalence relations on W.

Definition 5.4: [4] Let (W, R_1, R_2) be an equivalence frame of type 2. A *relational ExtCA over* (W, R_1, R_2) is the structure: $\underline{B} = (2^W, \subseteq, \emptyset, W, \cap, \cup, *, \vdash, C, c^o)$, where $*$ denotes the set theoretical complement and for any subsets of W a, b, and c:

- $a, b \vdash c$ iff $\forall A, A_1, B, B_1 \Big(AR_1A_1 \in a, BR_1B_1 \in b,$ $AR_2B \to (\exists C, C_1)(CR_1C_1 \in c, AR_2C)\Big)$ and $a \cap b \subseteq c$,
- aCb iff $a, b \not\vdash \emptyset$,
- $c^o(a)$ iff $(\forall b, c \subseteq W)(b \neq \emptyset, c \neq \emptyset, a = b \cup c \to$ $b, c \not\vdash (W \setminus a))$.

We say that a formula is true in (W, R_1, R_2) if it is true in the ExtCA over (W, R_1, R_2).

It turns out that the internal connectedness in a relational ExtCA means the following (see Figure 1):
$c^o(a)$ if and only if $(\forall b, c \subseteq W)(b, c \neq \emptyset$ and $a = b \cup c \to b \cap c \neq \emptyset$ or
$(\exists A, A_1, B, B_1)(AR_1A_1 \in b,\ BR_1B_1 \in c,\ AR_2B,$ $(\forall C, C_1)(AR_2C, BR_2C, CR_1C_1 \to C_1 \in a)))$

We have the following

Theorem 5.5: [4] **Relational representation theorem.** Let $\underline{B}$ be a finite ExtCA. Then $\underline{B}$ is isomorphically embedded in the relational ExtCA over some equivalence frame of type 2 (W, R_1, R_2).

We consider a quantifier-free first-order logic $\mathbb{L}$ for ExtCAs which has the following:

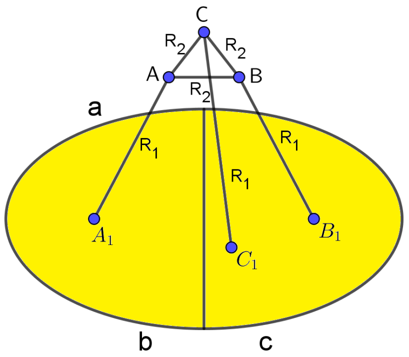


Fig. 1. **Internal connectedness in a relational ExtCA**

- axioms:

- the axioms of the classical propositional logic;
- the axioms of Boolean algebra;
- the axioms of ExtCA concerning the relations extended contact and contact;
- the axiom schemes:

(Ax c^o) $c^o(p) \wedge q \neq 0 \wedge r \neq 0 \wedge p = q + r \rightarrow q, r \not\vdash p^*$
(Ax c^o 1) $c^o(0)$
(Ax c^o 2) $\neg c^o(p+q) \rightarrow \neg c^o(p) \vee \neg c^o(q)$
(Ax c^o 3) $c^o(p+q) \rightarrow c^o(p) \wedge c^o(q)$

- rules:

- MP

This logic is decidable and we have the following

Theorem 5.6: **Completeness theorem with respect to relational semantics.** For every quantifier-free formula α the following conditions are equivalent:
i) α is a theorem of $\mathbb{L}$;
ii) α is true in all equivalence frames of type 2.

Theorem 5.7: **Completeness theorem with respect to topological and algebraic semantics.** For every quantifier-free formula α the following conditions are equivalent:
i) α is a theorem of $\mathbb{L}$;
ii) α is true in all ExtCAs;
iii) α is true in all topological ExtCAs over a compact, T_0, semiregular topological space.

Extended contact gives also the possibility to define the relation of contact (aCb iff $a, b \not\vdash 0$) and the binary relation $RC_\cap$ meaning that the intersection of two regular closed sets is a regular closed set ($RC_\cap(a,b)$ iff $a, b \vdash a \cdot b$). It is worth to consider also a quantifier-free first-order language without the predicate of internal connectedness i.e. $\mathcal{L}(0,1;\cdot,+,*;\leq,\vdash,C)$. In this weaker language one equivalence relation is enough - we consider *equivalence frames of type 1*:

Definition 5.8: [4] An *equivalence frame of type 1* is a relational structure of the form (W, R), where W is a nonempty set and R is an equivalence relation on W.

Definition 5.9: [4] Let (W, R) be an equivalence frame of type 1. A *relational ExtCA over* (W, R) *in* $\mathcal{L}$ is the structure $\underline{B} = (2^W, \subseteq, \emptyset, W, \cap, \cup, *, \vdash, C)$, where $*$ denotes the set theoretical complement and for any subsets of W a, b, and c:

- $a, b \vdash c$ iff $\Big((\exists A \in a)(\exists B \in b)ARB \rightarrow (\exists C \in c)ARC\Big)$ and $a \cap b \subseteq c$,
- aCb iff $a, b \not\vdash \emptyset$

Theorem 5.10: [4] **Relational representation theorem.** Let $\underline{B}$ be a finite ExtCA. Then in $\mathcal{L}$ $\underline{B}$ is isomorphically embedded in the relational ExtCA over some equivalence frame of type 1 (W, R).

This representation theorem is only for finite ExtCA. Trying to overcome this drawback, we define:

Definition 5.11: [4] A *weak extended contact algebra* is a structure of the form $\underline{B} = (B, \leq, 0, 1, \cdot, +, *, \vdash)$, where $(B, \leq, 0, 1, \cdot, +, *)$ is a non-degenerate Boolean algebra and $\vdash$ is a ternary relation on B such that for all $a, b, d, e, f \in B$,
(1) if $a \leq d$, $b \leq e$ and $d, e \vdash f$, then $a, b \vdash f$,
(2) if $a = 0$ or $b = 0$, then $a, b \vdash f$,
(3) if $a, b \vdash f$ and $d, e \vdash f$, then $a \cdot d, b + e \vdash f$ and $a + d, b \cdot e \vdash f$,
(4) if $a, b \vdash d$ and $d \leq f$, then $a, b \vdash f$.

Obviously, every extended contact algebra is also a weak extended contact algebra. The converse is not true.

Definition 5.12: [4] A *parametrized frame* is a structure of the form (W, R), where W is a nonempty set and R is a function associating to each subset of W a binary relation on W.

Definition 5.13: [4] Let (W, R) be a parametrized frame. A *relational weak ExtCA over* (W, R) is the structure $\underline{B} = (2^W, \subseteq, \emptyset, W, \cap, \cup, *, \vdash)$, where $*$ denotes the set theoretical complement and $\vdash$ is the ternary relation on W's powerset defined by

- $a, b \vdash d$ iff for all $S \in a$, $T \in b$ and $u \subseteq W$, if $d \subseteq u$, then $(S, T) \notin R(u)$.

Theorem 5.14: [4] **Relational representation theorem.** Let $\underline{B} = (B, \leq, 0, 1, \cdot, +, *, \vdash)$ be a weak ExtCA. Then $\underline{B}$ is isomorphically embedded in the relational weak ExtCA over some parametrized frame (W, R).

Thus we obtain in $\mathcal{L}$ a relational representation theorem for all ExtCA, not only finite (because every ExtCA is a weak ExtCA), but the structure in which we embed is not an

ExtCA and the parametrized frame it is based on is a relatively complex relational structure.

Let $\mathbb{L}_1$ be the logic obtained from $\mathbb{L}$ by removing axioms (Ax c^o), (Ax c^o 1), (Ax c^o 2) and (Ax c^o 3). This logic is called *extended contact logic*. It is decidable and we have the following

Theorem 5.15: **Completeness theorem with respect to relational semantics.** For every formula α in $\mathcal{L}$ the following conditions are equivalent:
i) α is a theorem of $\mathbb{L}_1$;
ii) α is true in all equivalence frames of type 1.

VI. Conclusion

Possible future research directions are for example:

- the complexity of the considered logics;
- to be obtained representation theorems in Euclidean spaces;
- generalization of Theorems 5.5 and 5.10 for all ExtCAs, not only for finite;
- to be obtained a stronger form of Theorem 5.14, where we embed in an ExtCA and the relational structure is simpler.
- in reference to temporal reasoning, if we add to the language of EDCL the binary relation $P(X, Y)$, meaning that the start of time interval X is before the start of time interval Y, then we obtain a language rich enough to define all possible relations between two intervals of Allen's interval algebra ([2]).

Acknowledgements. I thank the anonymous referees for their valuable remarks which improved the readability of the paper. This paper is supported by contract DN02/15/19.12.2016 "Space, Time and Modality: Relational, Algebraic and Topological Models" with Bulgarian NSF.

References

[1] M. Aiello, I. Pratt-Hartmann and J. van Benthem (Eds.), *Handbook of spatial logics.* Springer, 2007.

[2] J. F. Allen, "Maintaining knowledge about temporal intervals," *Communications of the ACM,* vol. 26, (11), 1983, pp. 832–843.

[3] P. Balbiani (Ed.), *Special Issue on Spatial Reasoning, J. Appl. Non-Classical Logics,* vol. 12, (3-4), 2002.

[4] P. Balbiani and T. Ivanova, "Relational representation theorems for extended contact algebras," *Stud Logica,* to appear, also available online with different title: https://arxiv.org/abs/1901.10367

[5] P. Balbiani, T. Tinchev and D. Vakarelov, "Modal logics for region-based theory of space," *Fundamenta Informaticae, Special Issue: Topics in Logic, Philosophy and Foundation of Mathematics and Computer Science in Recognition of Professor Andrzej Grzegorczyk,* vol. 81, (1-3), 2007, pp. 29–82.

[6] B. Bennett, "Determining consistency of topological relations," *Constraints,* vol. 3, 1998, pp. 213–225.

[7] B. Bennett and I. Düntsch, "Axioms, algebras and topology," *in Handbook of Spatial Logics,* M. Aiello, I. Pratt, and J. van Benthem (Eds.), Springer, 2007, pp. 99–160.

[8] A. Cohn and S. Hazarika, "Qualitative spatial representation and reasoning: An overview," *Fuandamenta informaticae*, vol. 46, 2001, pp. 1–20.

[9] A. Cohn and J. Renz, "Qualitative spatial representation and reasoning," *in F. van Hermelen, V. Lifschitz and B. Porter (Eds.) Handbook of Knowledge Representation,* Elsevier, 2008, pp. 551–596.

[10] G. Dimov and D. Vakarelov, "Contact algebras and region–based theory of space: A proximity approach I," *Fundamenta Informaticae,* vol. 74, (2-3), 2006, pp. 209–249.

[11] I. Düntsch (Ed.), *Special issue on Qualitative Spatial Reasoning, Fundam. Inform.,* vol. 46, 2001.

[12] I. Düntsch, W. MacCaull, D. Vakarelov and M. Winter, "Topological representation of contact lattices," *Lecture Notes in Computer Science,* vol. 4136, 2006, pp. 135–147.

[13] I. Düntsch, W. MacCaull, D. Vakarelov and M. Winter, "Distributive contact lattices: Topological representation," *Journal of logic and Algebraic Programming*, vol. 76, 2008, pp. 18–34.

[14] I. Düntsch and D. Vakarelov, "Region-based theory of discrete spaces: A proximity approach," *in M. Nadif, A. Napoli, E. SanJuan and A. Sigayret (Eds.) Proceedings of Fourth International Conference Journées de l'informatique Messine,* Metz, France, 2003, pp. 123–129, *Journal version in Annals of Mathematics and Artificial Intelligence,* vol. 49, (1-4), 2007, pp. 5–14.

[15] I. Düntsch and M. Winter, "A representation theorem for Boolean contact algebras," *Theoretical Computer Science (B),* vol. 347, 2005, pp. 498–512.

[16] T. Hahmann and M. Gruninger, "Region-based theories of space: Mereotopology and beyond," *S. Hazarika (ed.): Qualitative Spatio-Temporal Representation and Reasoning: Trends and Future Directions,* 2012, pp. 1–62, IGI Publishing.

[17] *Qualitative spatio-temporal representation and reasoning: Trends and future directions.* S. M. Hazarika (Ed.), IGI Global, 1st ed., 2012.

[18] T. Ivanova, "Extended contact algebras and internal connectedness," *Stud Logica,* vol. 108, 2020, pp. 239–254.

[19] T. Ivanova, "Logics for extended distributive contact lattices," *Journal of Applied Non-Classical Logics,* vol. 28(1), 2018, pp. 140–162.

[20] T. Ivanova and D. Vakarelov, "Distributive mereotopology: extended distributive contact lattices," *Annals of Mathematics and Artificial Intelligence,* vol. 77(1), 2016, pp. 3–41.

[21] T. de Laguna, "Point, line and surface as sets of solids," *J. Philos,* vol. 19, 1922, pp. 449–461.

[22] I. Pratt-Hartmann, "First-order region-based theories of space," *in Logic of Space, M. Aiello, I. Pratt-Hartmann and J. van Benthem (Eds.),* Springer, 2007.

[23] D. A. Randell, Z. Cui, and A. G. Cohn., "A spatial logic based on regions and connection," *in B. Nebel, W. Swartout, C. Rich (Eds.) Proceedings of the 3rd International Conference Knowledge Representation and Reasoning*, Morgan Kaufmann, Los Allos, CA, 1992, pp. 165–176.

[24] J. Renz and B. Nebel, "On the complexity of qualitative spatial reasoning: a maximal tractable fragment of the region connection calculus," *Artificial Intelligence,* vol. 108, 1999, pp. 69–123.

[25] J. Stell, "Boolean connection algebras: A new approach to the Region Connection Calculus," *Artif. Intell.,* vol. 122, 2000, pp. 111–136.

[26] D. Vakarelov, "Region-based theory of space: Algebras of regions, representation theory and logics," *in D. Gabbay, S. Goncharov and M. Zakharyaschev (Eds.) Mathematical Problems from Applied Logic II. Logics for the XXIst Century,* Springer, 2007, pp. 267–348.

[27] D. Vakarelov, G. Dimov, I. Düntsch, and B. Bennett, "A proximity approach to some region based theory of space," *Journal of applied non-classical logics,* vol. 12, (3-4), 2002, pp. 527–559.

[28] D. Vakarelov, I. Düntsch and B. Bennett, "A note on proximity spaces and connection based mereology," *in C. Welty and B. Smith (Eds.) Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS'01)*, ACM, 2001, pp. 139–150.

[29] H. de Vries, "Compact spaces and compactifications," Van Gorcum, 1962.

[30] A. N. Whitehead, "Process and Reality," New York, MacMillan, 1929.

DOI: 10.15439/2020F159
ISSN 2300-5963 ACSIS, Vol. 21

Network Device Workload Prediction: A Data Mining Challenge at Knowledge Pit

Andrzej Janusz*†, Mateusz Przyborowski*†, Piotr Biczyk†, Dominik Ślęzak*†
*Institute of Informatics, University of Warsaw, Warsaw, Poland
†QED Software, Warsaw, Poland

***Abstract*—We describe the 7th edition of the international data mining competition held at Knowledge Pit in association with the FedCSIS conference series. The goal was to predict workload-related characteristics of monitored network devices. We analyze solutions uploaded by the most successful participants. We investigate prediction errors which had the greatest influence on their results. We also present our own baseline solution which turned out to be the most reliable in the final evaluation.**

I. Introduction

The topic of the *FedCSIS 2020 Data Mining Challenge* falls into a category of implementing data analysis techniques in operational processes that employ either mechanical or electrical units. Part of the field, known as predictive maintenance, focuses on minimizing downtime and associated costs related to such units. Various techniques are applied to gather relevant data. It can be done using existing in-process sensors and system logs, or sensors introduced purely for predictive maintenance purposes. Data can also be acquired in an active way by injecting a test signal into a system [2].

Such data can be explored using various methods ranging from condition-based qualifiers to machine learning. A common approach relies on prediction of quantitative indicators, their association with given maintenance issues, and determination of their relationship to operational costs and failure risks [8]. The specific direction of analysis depends on individual processes under scrutiny, e.g. variability of their parameters, or importance of anomaly detection versus long-term trend detection. It often requires an ensemble of methods in order to tackle operational issues in a reliable way.

Due to their complexity, the predictive maintenance tasks are premium example of problems that could be solved using crowdsourcing, e.g. via online machine learning competitions. Rise of this approach has been accelerated thanks to popularity of sites like *Kaggle* or *Knowledge Pit*. In its core, a competition method can be drilled down to: formulation of research problem, data preparation by an unbiased team, creation of competition baseline, data analysis and solution preparation by competition participants, and finally evaluation of submissions on a platform that supports fair environment. If all these steps are provided, then the owner of data (usually the main stakeholder interested in the competition outcomes) can expect various benefits, ranging from proof of feasibility, through obtaining insights on how to resolve the problem in real-world application, up to using competition results as a guideline for assembling a dedicated R&D team.

In this paper, we investigate the outcomes of the considered challenge with a particular focus on the most substantial errors in predictions sent by participants. In Section II, we describe the challenge objectives, data sets that we prepared, the selected evaluation function, as well as our baseline model which turned out to be the most robust among all submitted solutions. In Section III, we provide an overview of the competition results and we present conclusions drawn from the analysis conducted on the set of over 700 solutions. We summarize the challenge and the paper in Section IV.

II. Competition Outline

The challenge took place at Knowledge Pit[1]. The data was provided by *EMCA Software*, the company specializing in log analytics. The goal was to predict long-term workload-related characteristics of devices, based on their history. Thus, competition results relate to EMCA's business model. Moreover, we wanted to foster deeper understanding of predictive maintenance nuances in the data science community.

A. Data preparation

The competition was held on real, previously unpublished data gathered from 3728 network devices monitored by EMCA as part of their operations. An additional, quite inspiring difficulty arose from the fact that those devices were not uniform. Logs covered readings from various types of hardware. There were also cases of different hostnames being a part of the same network, thus making their workload states correlated.

The data was collected over a period of December 2019 – February 2020. The raw data was provided in batches corresponding to individual days. Each batch contained ≈ 2.45 GB of data extracted from network device logs (≈ 220 GB in total), in form of a collection of JSON entries. Every entry consisted of device identifier, timestamp, and a list of one or more tuples indicating one of 45 so-called metrics.

Figure 1 shows the first preprocessing step. Each batch was streamed due to its large size. Individual JSON entries were parsed. The extracted hostnames were anonymized using a dynamically extended dictionary. The rest of information was transformed into EAV format (metric-timestamp-value) and aggregated for every metric in consecutive one-hour-long windows. Each such period for a given metric/hostname combination was characterized by some aggregate measures and written down to far smaller files. Similar window-based summaries are widely used in data analytics, e.g. to improve representation [7] or decrease data footprint [1].

Research co-funded by the Polish National Centre for Research and Development in frame of project MAZOWSZE/0198/19 and by QED Software.

[1] https://knowledgepit.ai/fedcsis20-challenge/

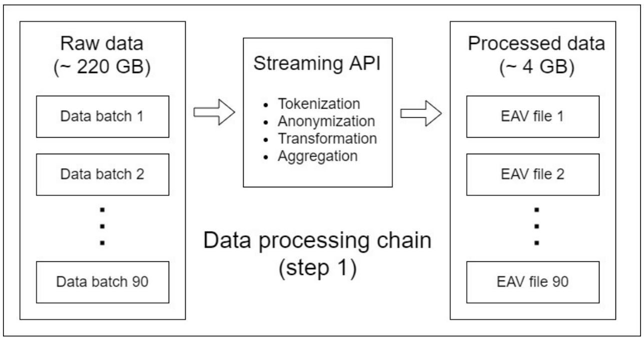


Fig. 1. The initial data preprocessing schema. The input files were processed in a streaming fashion to limit the required computational resources.

In the second step, the data from local files was merged and time series with too low number of entries were filtered out. At this point the total data size was relatively small ($\approx 4GB$). At the end, the data set was divided into training and test parts based on time. The last seven days were used as the test period. Therein, we included 10000 time series (hostname/metric combinations) which did not have any missing values in the test period. From this set, we randomly selected 1000 series to be used for preliminary evaluation of solutions.

B. Task description and evaluation procedure

The training data was made available to teams in form of a CSV file containing 10 columns. The first three of them create a joint identifier, followed by seven window-based aggregations forming in particular a *candlestick* representation of time series. The detailed column meanings are:

1) *hostname*: anonymized ID of device
2) *series*: name of the considered metric
3) *time_window*: timestamp indicating window start
4) *Mean*: the mean of the considered metric values[2]
5) *SD*: standard deviation of the considered metric
6) *Open*: the first reading in the given time window
7) *High*: the maximum reading in the given window
8) *Low*: the minimum reading in the given window
9) *Close*: the last reading in the given window
10) *Volume*: total number of corresponding readings

For each hostname in the training data, values could be arranged into series spanning for over 80 days. However, in many series, some values (time windows) were entirely missing, which typically means that a device was not accessible.

Participants were asked to predict 168 future values (i.e. hourly mean values of a given metric in one full week) of a number of devices. They were submitting their predictions to Knowledge Pit via the online evaluation system. IDs of devices and metrics were indicated in an exemplary solution file.

During the challenge (i.e. before closing it), submissions were evaluated on the already-mentioned subset of 1000 (out of 10000) test time series. We used R^2 measure, i.e. for each time series, the forecasts were compared to ground truth values, and their quality was assessed by the following formula:

TABLE I. SELECTED FINAL AND PRELIMINARY RESULTS. THE SCORES OF TOP 3 TEAMS WITH REGARD TO THE PRELIMINARY AND FINAL SCORES ARE SHOWN. THE LAST COLUMN INDICATES THE NUMBER OF SUBMITTED SOLUTIONS. NOTICEABLE IS THE NEGATIVE CORRELATION BETWEEN THE NUMBER OF SUBMISSIONS AND FINAL SCORE OF PARTICULAR TEAMS.

Rank	Team name	Preliminary	Final score	#subs
1	baseline solution	0.2267	0.229530	3
2	Les Trois Mousquetaires	0.1888	0.162979	19
3	papiez69	0.1841	0.151499	13
4	Wrong Team Name	0.1836	0.143708	6
...	...	...	...	...
13	cdata	0.2766	-0.059837	90
14	amy	0.3130	-0.138349	100
...	...	...	...	...
17	Dymitr	0.3223	-0.779576	146
...	...	...	...	...

$$R^2(f,y) = 1 - \frac{RSS(f,y)}{TSS(y)} \qquad (1)$$

where f is a vector of forecasts, y is the target ground truth, $RSS(f,y) = \sum_i (y_i - f_i)^2$ is the residual sum of forecast squares, and $TSS(y) = \sum_i (y_i - \hat{y})^2$ is the total sum of squares, where $\hat{y}$ is the mean value of time series y estimated using the available training data. The submission score is the average R^2 over all time series from the test set.

The best evaluation results of participating teams were visible on the public Leaderboard. Each team could submit up to 100 solutions, but teams could merge during the competition. Thus in the end, the total number of submissions for a given team could be greater than this limit (in such a case, the merged team could not submit any new solution files).

The final evaluation was performed after completion of the competition using the remaining part of the test data (90%). Those results were published online too. Only those teams who submitted a report describing their approach before the end of the challenge were qualified for final evaluation.

C. Our baseline solution

At the beginning of the challenge, we prepared a relatively simple baseline solution in order to provide to participants a reference score at the Leaderboard. We did also for the purpose of equipping EMCA with a light-weight tool for detecting anomalies in device usage patterns in real-time.

First, we cleaned the training data out of outlying metric readings. Such readings could be a result of some unexpected device malfunction, monitoring software error, or some unusual actions performed by device users. Since we were mostly interested in detecting typical device usage patterns, outliers in the training data could distort our forecasts.

There are many approaches to time series anomaly detection. In our solution, we used the well-known *3-sigma* method. We assumed weekly periodicity of the considered time series, which was confirmed on a random sample using the Fisher's test. We divided the training data into disjoint, one-week long time windows. For each window, we estimated the mean and standard deviation. We clipped the time series values which were identified as outliers by means of 3-sigma. We treated the clipped windows as time series motifs [5].

Then, for each time series in the test data, we extracted the latest weekly time window from the training data as a tempo-

[2] Values of all columns 4-10 are calculated for the considered metric (*series*) of the given device (*hostname*) within the considered *time_window*.

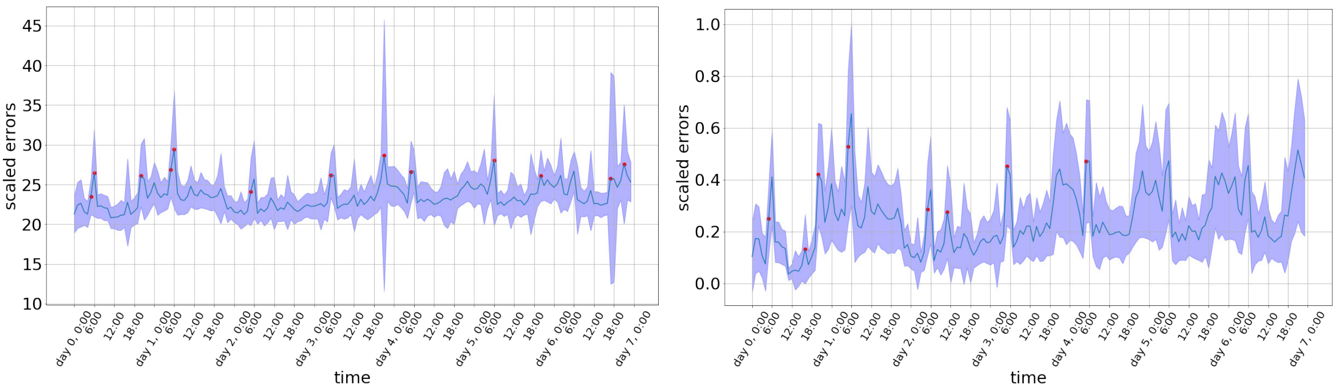


Fig. 2. Squared differences between forecasts and reference values. On the left, scaled by TSS of each series, aggregated by the metric/hostname combination, and averaged for each hour. The shaded area represents a distance of one standard deviation from the mean. Red dots indicate an increase in the error rate above one standard deviation from the mean of all errors. The plot on the right shows the resulting values with additional scaling of each series to [0,1] interval.

rary validation set. We averaged the last three motifs from the remaining time windows to create the series templates. It means that – denoting the last i-th motif for the k-th series by $\vec{x}_{k,i}$ – the corresponding template is specified as:

$$T_k = \frac{1}{3}\sum_{i=1}^{3}\vec{x}_{k,i} \qquad (2)$$

We compared such templates with the corresponding validation windows by means of R^2 score. Then we updated templates with the data from validation period. When making forecasts, we applied templates with $R^2 > 0$ to predict the corresponding series in the test period. Otherwise, our forecast was simply the global mean for the given series, estimated on the whole of training data. Table I indicates that this method achieves the best final score among all submitted solutions.

III. Competition Results

The competition attracted 151 teams from over 30 countries. The highest number of participants had IP addresses from Poland (52), India (19), Russia (15), China (7), and the United States (7). Over 700 correctly formatted solutions were submitted. Besides the above-discussed baseline, Table I reports the final ranks, scores, and the number of submissions for some of the best performing teams. More details about some of those teams can be found in [3], [6], [9], [10].

The best submitted solution, i.e. Rank 2 in Table I, relied on an ensemble of XGBoost, Prophet and linear regression models. The final model took into account their individual performance for each host, applying the mean value if neither of them proved to give satisfactory results, or if there were less than 300 of data points for the host. Rank 3 utilized ensemble of two different XGBoost models – one using only hours and holiday days as features, and the other using also days of the week – together with the mean value for the cases for which the considered models delivered R^2 less than 0.04.

A. Error distribution over time

As device logs vary greatly, we run our comparisons using scaled and averaged values. Figure 2 suggests that reasoning about further-future values does not deteriorate over time. This may indicate that submitted methods can successfully (on average) deal with data seasonality, grasping periodicity within the series that may be observed in the training data set and, based on that, producing future forecasts. On the other hand, the errors tend to happen around the same hours every day. Such events may be unpredictable, although the highest final score solutions are more robust with this respect.

B. Error distribution among the best solutions

At Knowledge Pit, competition participants can verify their solutions using a small test data portion. This functionality usually helped in fine-tuning meta-parameters of developed methods. However this time, the solution with the highest preliminary score did not want to generalize to the entire test set at all. According to Figure 3, the preliminary-best solutions tend to neglect the importance of some series which turn out to be a significant part of the test data. This may be because of under-representation of some patterns and overall difficulty in forecasting those series in the training data set.

C. Errors according to the metrics

Figure 4 shows that metrics vary in terms of forecasting difficulty. This fact could be a reason for some of participating teams to give up some series and focus on increasing R^2 based on a few simpler ones. Accordingly, let us note that our way of splitting test data onto preliminary and final subsets preserved most of desired statistical properties of the metrics aggregations. Out of 27 distinct metrics in the test set, 25 occur in preliminary part, with sufficient cardinality. This yields a corollary that it was not a design of data sets that caused problems, but the very structure of explored data.

IV. Conclusions

We reported our international challenge aimed at predicting long-term network device workload characteristics. Our baseline model achieved the score $R^2 = 0.23$ and we assess that this result can be further improved. The analysis of errors of other submitted solutions revealed no significant correlation with a time horizon of forecasts. It suggests that models can often correctly take into account periodicity, but any deviations are hard to predict. It not only means that modeling of typical workload patterns is feasible, but it may also allow us to design a simple real-time anomaly detection algorithm.

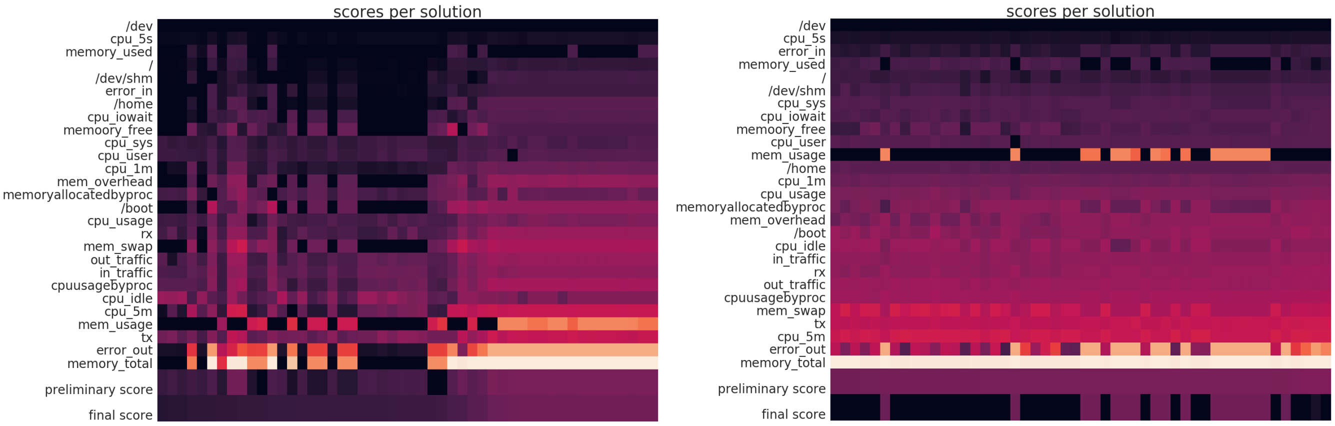


Fig. 3. R^2 scores per submission (the darker the color, the lower the value of R^2 for particular submitted solution on data corresponding to a particular metric), for individual metrics (*series*). Vertically, rows are sorted by the mean score (increasing towards bottom). Horizontally, solutions are sorted by the final score. The plots on the left and right sides show the values for the top 75 submissions with regard to the final and preliminary scores, respectively.

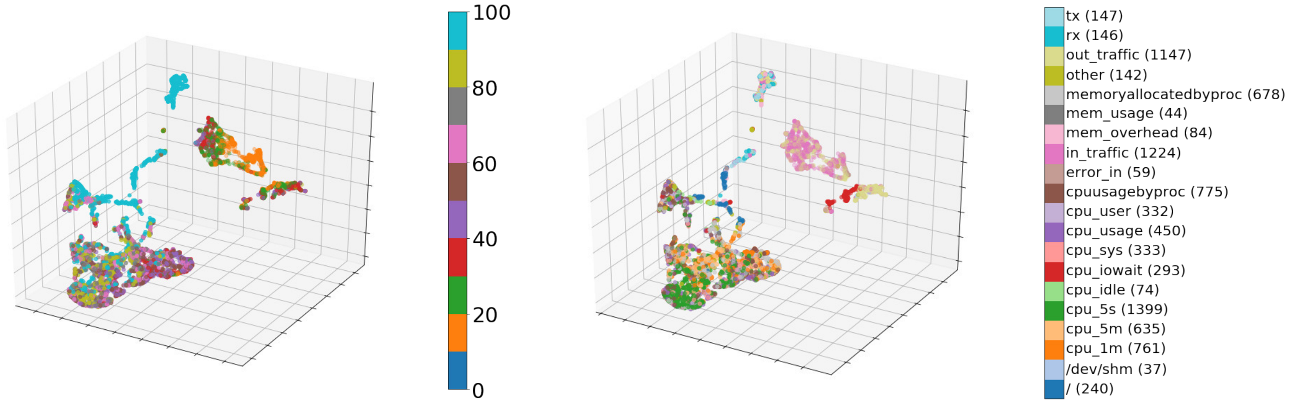


Fig. 4. Errors (9000 series, each 704 dimensions) reduced with UMAP [4] to three dimensions. The plots color the points according to the metric (on the right) and to percentiles of the mean values (on the left). The points form clusters by means of a similar scale of values and metrics.

It is worth discussing why competition participants – representing truly diverse background and data science experience levels – could not come up with any solution yielding better prognostic power than our baseline. One reason is that we focused on typical work patterns of the system, not attempting to forecast outliers. From the perspective of R^2 evaluation score, any model that did try to predict anomalies was highly punished for missed forecasts. It may mean that although – as mentioned above – it seems to be easy to detect anomalies, it is incomparably harder to predict their values.

Future works can include better adjustment of the applied error function to end-user expectations. Moreover, one can develop an ensemble approach combining models tuned to typical workloads (e.g. our baseline) with those reflecting outliers (some other submissions). One can also consider anomalies of various kinds: power-law anomalies (*black swans*), phase transition anomalies (*dragon knights*), and unique events not exceeding operational parameters (*unicorns*).

Finally, as for our general competition-based approach to crowdsourcing of data mining problems, one can see that the current setup at Knowledge Pit fits both, the needs of data science community and expectations of real-life data owners. Accordingly, new challenges will be organized.

References

[1] A. Chądzyńska-Krasowska and M. Kowalski. Quality of Histograms as Indicator of Approximate Query Quality. In *Proc. of FedCSIS 2016*, pages 9–15.

[2] H. M. Hashemian. State-of-the-Art Predictive Maintenance Techniques. *IEEE Trans. Instrum. Meas.*, 60(1):226–236, 2011.

[3] C. Liu. Shallow, Deep, Ensemble Models for Network Device Workload Forecasting. In *Proc. of FedCSIS 2020*.

[4] L. McInnes, J. Healy, N. Saul, and L. Großberger. UMAP: Uniform Manifold Approximation and Projection. *J. Open Source Softw.*, 3(29):861, 2018.

[5] A. Mueen and E. Keogh. Online Discovery and Maintenance of Time Series Motifs. In *Proc. of KDD 2010*, pages 1089–1098.

[6] D. Ruta, L. Cen, and Q. H. Vu. Deep Bi-Directional LSTM Networks for Device Workload Forecasting. In *Proc. of FedCSIS 2020*.

[7] Ł. Sosnowski and T. Penza. Generating Fuzzy Linguistic Summaries for Menstrual Cycles. In *Proc. of FedCSIS 2020*.

[8] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi. Machine Learning for Predictive Maintenance: A Multiple Classifier Approach. *IEEE Trans. Ind. Informatics*, 11(3):812–820, 2015.

[9] T. Wittkopp, A. Acker, S. Nedelkoski, J. Bogatinovski, and O. Kao. Superiority of Simplicity: A Lightweight Model for Network Device Workload Prediction. In *Proc. of FedCSIS 2020*.

[10] M. Zuefle and S. Kounev. A Framework for Time Series Preprocessing and History-based Forecasting Method Recommendation. In *Proc. of FedCSIS 2020*.

DOI: 10.15439/2020F179
ISSN 2300-5963 ACSIS, Vol. 21

An extensive analysis of online restaurant reviews: a case study of the Amazonian Culinary Tourism

Luiz Carlos Fernandes Junior*, Jorge Silva Junior*, Antonio Jacob Junior† and Fábio Lobato*†
*Federal University of Western Pará, Santarém, Brazil
†State University of Maranhão, São Luís, Brazil
Email: {luizcarlossfjr, jorgeluizfigueira, antonio.jacob}@gmail.com, fabio.lobato@ufopa.edu.br

***Abstract*—Analyzing User-Generated Content present in social media has become mandatory for companies looking for maintaining competitiveness. These data contain information such as consumer opinions, and recommendations that are seen as rich sources of information for the development of decision support systems. When observing the state of the art, it was found that there is a lack of antecedents that address the analysis of online reviews of Brazilian restaurants. In this sense, the focus of this work is to fill this gap through a case study of Santarém city. The results show that professionals in this segment can use these analyzes in order to improve the user's experiences and increase their profits.**

I. Introduction

IN 2018, the tourism sector contributed US$ 152.5 billion to the Brazilian Gross Domestic Product (GDP)[1]. In the city of Santarém (Pará, Brazil), located in the very heart of the Amazon rainforest, the collaboration of this sector is significant. According to the municipal secretary of tourism [1], this activity injects about US$ 32 million in the local economy, driving segments like restaurants, hotels, travel agencies, bars etc.

The internet has completely changed the way the information related to tourism are distributed and consumed [2]. The User-Generated Content (UGC) growth has a significant impact on the tourism sector, influencing travelers in the decision-making process [3]. According to [4], UGC is all forms of content created, disseminated, and consumed by users.

Restaurant reviews are useful for the known segment as culinary tourism. In summary, this kind of tourism enables the recognition of values related to a certain territory's culture, so gastronomy is transformed into tourist products. In this panorama, [5] points out that online restaurant reviews influence consumers' decision-making, which is vital to the companies' analysis of this information to improve their services [6]. In the last years, with the data volume available on the internet and diversity growth, many challenges regarding data collection and analysis in this sector have emerged [7], [8]. One of these is to analyze the immense volume of textual data, a task practically impossible to be performed manually [9]. To tackle this obstacle, computational techniques such as Text Mining can be employed in order to identify patterns and generate insights that can support the decision making process [10], [11], [6].

Through a literature review, it was realized a lack of antecedents that explore the knowledge extraction from UGC on social media in Brazilian restaurants. In addition, the related works do not address the correlation of the authors' gender with relevant topics considered by them. In this context, the present work aims to analyze patterns extracted from restaurant reviews on the TripAdvisor platform, carrying out a case study of the city of Santarém. For this purpose, Text Mining techniques were applied to answer the following Research Questions (RQ):

- **RQ-1** What is the predominant sentiment expressed in the TripAdvisor reviews of restaurants in the Santarém, and which genre of customers has the most negative reviews?
- **RQ-2** What are the patterns of positive and negative comments?
- **RQ-3** What are the main topics covered in TripAdvisor reviews of restaurants in the Santarém - Is there a distinction of themes between the male and female gender?
- **RQ-4** How do the identified topics relate to each other?

The remainder of this paper is structured as follows. In Section II the related works are presented. The Experimental Framework used is described in Section III. The results and insights are discussed in Section IV. The conclusions and future works directions are given in Section V.

II. Related Work

Analysis in UGC has been widely addressed in several application domains due to the potential in the process of improving services and products [12]. This information is even more important for the hospitality sector, whose audience considers it to be a very reliable method of decision-making [13], [3]. In this scenario, a large part of this information is made up of textual data, so an appropriate approach to analyze this massive data is the use of text mining techniques [6].

Among the use of these techniques, are highlighted the Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA) for the topic modeling task, as described in the studies by [14], [15]. By analyzing data collected from TripAdvisor, Airbnb, Couchsurfing, and Booking platforms, the authors obtained the segmentation of the type of review (for instance: *comfort, location, and experience*) as well as the identification of main service problems (for instance: *levels of cleaning service*).

[1] Data from Brazil's Ministry of Tourism

In [16], the authors used a classifier based on Naïve Bayes and in [17] the python libraries NLTK and TextBlob were used to identify the polarity of reviews collected from the TripAdvisor and Yelp platform. As a result, both papers present the distribution of consumers' opinions regarding the service. In [16], they combine topic modeling with sentiment analysis to obtain the main topics segmented by polarity. Besides, in [17] used the python library Guess-Gender to identify the gender of the authors' reviews in order to determine differences in assessment methods according to gender.

III. Experimental Framework

In this Section, the experimental framework used in this study will be described.

A. Data Acquisition

The TripAdvisor [2] platform was used as a data source, given its prominent position in the tourism field. All comments from Santarém restaurants that had at least one review on the site by April 2020 were collected, summing up 3,881 reviews from 186 restaurants. The data extracted include: i) restaurant name, ii) restaurant score, iii) review title, iv) comment score, v) review content and vi) username. A web crawler written in Python was developed to extract the data and the informations extracted were stored in a file in the Comma-Separated Values (CSV) format.

B. Data pre-processing

The pre-processing step was divided into two parts. The first one performs the filtering of assessments based on readability. Using the Flesch Kincaid index adapted to Portuguese[18], the most readable comments (scores between 75 and 100) were selected. Given that these are more influential in other customers' decision-making[19], the analyzes on this new dataset tend to reflect more accurate results for business managers.

The second one, consists of handling the information to remove inconsistencies and improve the results reliability [20]. This process was conducted using the NLTK library because it has support for the Portuguese. The following steps were performed: (i) characters conversion to lower case; (ii) accentuation substitution; (iii) punctuation and special characters removal; (iv) numbers deletion; (v) stopwords removal; (vi) emojis elimination.

C. Sentiment Analysis

Sentiment analysis can be defined as a technique for handling opinions, feelings and subjectivity in texts [21]. The Polyglot [22] library was used to perform this task, as it had good results in previous works for Portuguese language [23]. Since this library produces a numerical result (P) that varies from -1 to 1, the values obtained in this analysis were categorized as Neutral when $P = 0$; Positive when $0 < P \leq 1$ and Negative when $-1 \leq P < 0$.

D. Topic Modeling

This task was divided into two stages: i) topics extraction; ii) the correlation analysis of the topics identified.

Regarding the first stage, the Non-Negative Matrix Factorization (NMF) technique was used, given its efficiency for text mining tasks in short documents [24]. The weight used to represent the values of these words in the term matrix was the Term Frequency-Inverse Document Frequency (TF-IDF) because good results were achieved with it in previous text mining tasks [25]. After performing the extraction, it is necessary to generate a results annotation, this step being performed manually from a subjective analysis by the authors [26].

To find the best coherence between the number of topics and the number of words, an analysis of the coherence of this relationship was performed using the metric Pointwise Mutual Information (PMI). Regarding the second stage, the topics correlation was conducted to verify which topics are most related to each other. Each topic represents a set of terms and each term is associated with one or more comments. Thus, the following elements were considered: the nodes represent the topics; the edges represent the relationship between the topics, and the greater the thickness of the edge, the more intense the correlation between the topics.

IV. Results

Initially 3,881 reviews were obtained, and from these, the readability analysis resulted in of 794 reviews helpfulness to conduct the other analyzes. From an analysis of the database, it was noted trend for users to give a high rating score, so that the lowest scores (10 and 20) together have only 38 occurrences.

After the pre-processing step, the sentiment analysis algorithm was applied. There were 62.5% positive comments, 22.3% neutral and 15.2% negative. Given this scenario and the first part of the **RQ-1**, it is possible to conclude that the main polarity expressed is positive. Examples of positive and negative comments can be seen, respectively, in items 1 and 2, 3 in Table I.

TABLE I: Examples of pre-processed comments.

Item	Comment after pre-processing stage
1	has wonderful view location amazing food served great price is worth
2	price pasties expensive size quantity filling are offered quality ingredients good problem cost x benefit pastel meat has wind can catch cold care meat cheese served satiate hunger great
3	location waterfront santarem think unique self service kilo city serves barbecue prepared food rolls prawns bad price simple environment kinda tight food more

The gender identification was performed manually and of a total of 794 usernames, 357 (45.0%) were recognized as being male, 312 (39.2%) female and 125 (15.8%) undefined. The comments classified as undefined are justified by the presence of pseudonyms, such for example, *Dream508624* and *Y4979PGalinem*, being therefore impossible to label them.

In this panorama, considering only the comments in which it was possible to identify the author as belonging to one of the

[2] https://tripadvisor.com/

genders, and correlating these data with the sentiment analysis results, the Figure 1 is presented. Thus, it is possible to answer the second part of the **RQ-1**, in which the male gender obtained a slightly higher occurrence of negative comments. So, there are no significant differences between gender in the negative comments.

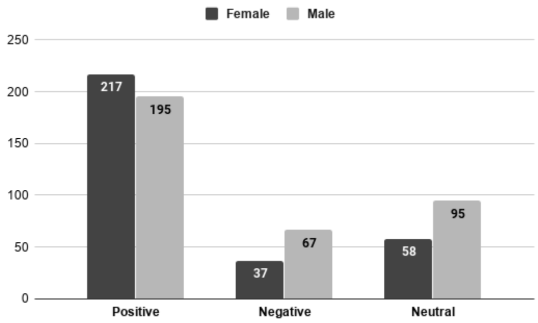


Fig. 1: Correlation between sentiment analysis and gender.

In order to understand how the pattern of a positive or negative comment is characterized, the rule extraction was conducted considering the sentiment polarity as labels. Similar to the analysis conducted in [25], the Decision Tree algorithm was used and, as data entry a BoW representation of the comments with the binary weight scheme. The result obtained was a set of descriptive rules presented in Table II.

TABLE II: Rules extracted *per* polarity.

Class	Rules	Coverage
Positive	Absence of: eat, delay, fried, hunger, leave, high, rotten, waiting, stairs	82,66%
Negative	Occurrence of: good, food, variety. beach, road, liked,	16,52%

When analyzing the extracted rules, it is possible to answer the second research question (**RQ-2**), in which the comments whose sentiment is considered positive tend to have the absence of terms such as "*delay*", "*fried*", "*hunger*", "*leave*" and "*rotten*". From these terms, it is possible to infer that the majority of customers take into consideration the waiting time and food quality. Regarding the negative comments, the occurrence of terms such as "*good*", "*food*", "*variety*", "*beach*", and "*road*" does mention the menu quality, location, and accessibility.

The the topics coherence analysis was conducted based on three viewpoints: i) all 794 comments; ii) only female comments; and, iii) only male comments. The best found for this viewpoints are respectively: combination of the 5 topics of greatest coherence with 100 topics and their 20 main words; combination of the 5 most coherent topics with 80 topics and their 10 main words; combination of the 5 most coherent topics with 100 topics and their top 5 words.

Analyzing Table III, it is possible to answer the first part of the **RQ-3**, in which it is noted that the main topics present in the comments are customer service, location, menu, space/infrastructure and appetizers.

TABLE III: Most prevalent topics.

Mode	Topics	Coherence average
All dataset	Menu, Space/Infrastructure, Customer service, Location, Appetizers	4.38
Female reviews	Environment, Menu, Location, Infrastructure, Customer Service	4.12
Male reviews	Location, Menu, Pricing, Options/Varieties, Customer Service	4.38

Analyzing also the Table III, it is possible to notice some distinct topics addressed between male and female reviews. For example, the topic pricing and options/varieties only appears in males, while infrastructure and environment were more consistent in females. Thus, it is possible to answer the second part of **RQ-3**: there are differences between the aspects addressed by people of different genders in online reviews.

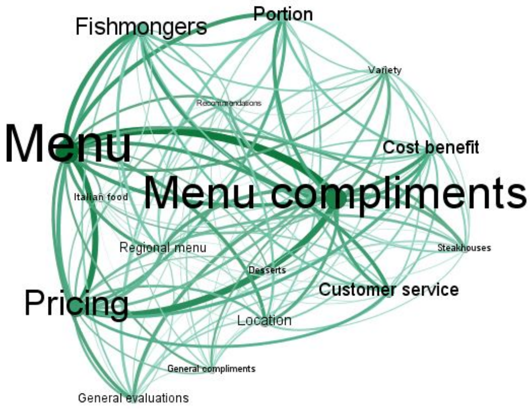


Fig. 2: Relationship between reviews topics.

The correlation between the topics can be verified through the analysis of the Figure 2. The size of the words represents the weight of the node in the network. The variation in tones and the thickness of the edges represent the intensity of the relationship. In this context and considering the **RQ-4**, is possible to highlight that comments that compliment the restaurant's menu usually contain evaluations referring to the price, showing the strong relationship between these attributes. In addition, the consumption of fish by tourists is evident, a fact justified by being a characteristic dish of the region.

The results obtained have practical implications:

- Most of consumer's are using the platform to give positive feedback or only to describe the restaurant facilities and services;
- Service, location, menu, space and appetizers are the most relevant aspects reported in the restaurants reviews;
- Based on identifying the most relevant topics by gender, restaurant managers could offer gender group discounts,

considering that these have different specific needs.

V. Conclusion

In this work, four tasks related to text mining were performed in order to extract relevant knowledge from online restaurant reviews: sentiment analysis, identification of the author's gender, extraction of rules, and topic modeling. In a brief contact with practitioners working directly with restaurants sector, we could validate the knowledge extracted. In this context, we conclude that the UGC is a rich source for the extraction of relevant knowledge from online restaurant reviews, taking the author's gender as a basis.

Our results can contribute to the management of companies related to culinary tourism, helping them to develop better products and services, centered on consumer expectations. Our work has some limitations, for instance, the emojis were removed in the pre-processing phase, which can impact on the sentiment analysis results. In future work, we would like to resolve these limitations, delve further into the modes of assessment by gender and extend the scope of the research, considering other locations and business domains.

Acknowledgment

The authors would like to thank the Maranhão Foundation for the Protection of Research and Scientific (FAPEMA) and the Tutorial Education Program (PET-IEG-UFOPA) for supporting the development of this work.

References

[1] G1, "Turismo em Santarém cresce em 2018 e injeta R$ 176 milhões na economia, aponta estudo," https://g1.globo.com/pa/santarem-regiao/noticia/2019/02/11/turismo-em-santarem-cresce-em-2018-e-injeta-r-176-milhoes-na-economia-aponta-estudo.ghtml. Accessed 21 April 2020., 2019.

[2] J. Navío-Marco, L. M. Ruiz-Gómez, and C. Sevilla-Sevilla, "Progress in information technology and tourism management: 30 years on and 20 years after the internet-revisiting buhalis & law's landmark study about etourism," *Tourism Management*, vol. 69, 2018. doi: https://doi.org/10.1016/j.tourman.2018.06.002

[3] Y. Narangajavana Kaosiri, L. J. Callarisa Fiol, M. A. Moliner Tena, R. M. Rodriguez Artola, and J. Sanchez Garcia, "User-generated content sources in social media: A new approach to explore tourist satisfaction," *Journal of Travel Research*, vol. 58, no. 2, 2019. doi: https://doi.org/10.1177/0047287517746014

[4] A. J. Kim and K. K. Johnson, "Power of consumers using social media: Examining the influences of brand-related user-generated content on facebook," *Computers in Human Behavior*, vol. 58, 2016. doi: https://doi.org/10.1016/j.chb.2015.12.047

[5] S. Lee, H. Ro *et al.*, "The impact of online reviews on attitude changes: the differential effects of review attributes and consumer knowledge." *International Journal of Hospitality Management*, vol. 56, 2016. doi: https://doi.org/10.1016/j.ijhm.2016.04.004

[6] S. Schmunk, W. Höpken, M. Fuchs, and M. Lexhagen, "Sentiment analysis: Extracting decision-relevant knowledge from ugc," in *Information and Communication Technologies in Tourism 2014*. Springer, 2013, doi: https://doi.org/10.1007/978-3-319-03973-2_19.

[7] B. G. Nistoreanu, L. Nicodim, and D. M. Diaconescu, "Gastronomic tourism-stages and evolution," in *Proceedings of the International Conference on Business Excellence*, vol. 12, no. 1. Sciendo, 2018. doi: https://doi.org/10.2478/picbe-2018-0063

[8] G. J. Miller, "Comparative analysis of big data analytics and bi projects," in *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2018. doi: http://dx.doi.org/10.15439/2018F125

[9] A. Klein, M. Riekert, and V. Dinev, "Accurate retrieval of corporate reputation from online media using machine learning," in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019. doi: http://dx.doi.org/10.15439/2019F169

[10] R. Talib, M. K. Hanif, S. Ayesha, and F. Fatima, "Text mining: techniques, applications and issues," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 11, 2016. doi: https://doi.org/10.14569/IJACSA.2016.071153

[11] Y. Zhao, *R and Data Mining: Examples and Case Studies*, 12 2012. ISBN 978-0-12-396963-7

[12] F. Lobato, M. Pinheiro, A. Jacob, O. Reinhold, and Á. Santana, "Social crm: Biggest challenges to make it work in the real world," in *International Conference on Business Information Systems*. Springer, 2016. doi: https://doi.org/10.1007/978-3-319-52464-1_20

[13] L. Yan, N. Cha, H. Cho, and J. Hwang, "Video diffusion in user-generated content website: An empirical analysis of bilibili," in *2019 21st International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2019. doi: https://doi.org/10.23919/ICACT.2019.8701897

[14] C. Marcolin, J. L. Becker, F. Wild, G. Schiavi, and A. Behr, "Business analytics in tourism: Uncovering knowledge from crowds," *BAR-Brazilian Administration Review*, vol. 16, no. 2, 2019. doi: https://doi.org/10.5748/9788599693148-15CONTECSI/PS-5707

[15] G. Santos, M. Santos, V. F. Mota, F. Benevenuto, and T. H. Silva, "Neutral or negative? sentiment evaluation in reviews of hosting services," in *Proceedings of the 24th Brazilian Symposium on Multimedia and the Web*, 2018. doi: https://doi.org/10.1145/3243082.3243091

[16] V. Taecharungroj and B. Mathayomchan, "Analysing tripadvisor reviews of tourist attractions in phuket, thailand," *Tourism Management*, vol. 75, 2019. doi: https://doi.org/10.1016/j.tourman.2019.06.020

[17] M. P. Silveira, W. Z. Xavier, and H. T. Marques-Neto, "Análises de dados de sistemas crowdsourcing: estudo de caso de avaliações de estabelecimentos realizadas no yelp," in *Anais do VII Brazilian Workshop on Social Network Analysis and Mining*. SBC, 2018. doi: https://doi.org/10.5753/brasnam.2018.3593

[18] T. B. F. Martins, C. M. Ghiraldelo, M. d. G. V. Nunes, and O. N. de Oliveira Junior, "Readability formulas applied to textbooks in brazilian portuguese," 1996.

[19] B. Fang, Q. Ye, D. Kucukusta, and R. Law, "Analysis of the perceived value of online tourism reviews: Influence of readability and reviewer characteristics," *Tourism Management*, vol. 52, 2016. doi: https://doi.org/10.1016/j.tourman.2015.07.018

[20] D. Cirqueira, M. F. Pinheiro, A. Jacob, F. Lobato, and Á. Santana, "A literature review in preprocessing for sentiment analysis for brazilian portuguese social media," in *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE, 2018. doi: https://doi.org/10.1109/WI.2018.00008

[21] N. Rodríguez-Barroso, A. R. Moya, J. A. Fernández, E. Romero, E. Martínez-Cámara, and F. Herrera, "Deep learning hyper-parameter tuning for sentiment analysis in twitter based on evolutionary algorithms," in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019. doi: http://dx.doi.org/10.15439/2019F183

[22] Y. Chen and S. Skiena, "Building sentiment lexicons for all major languages," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2014, pp. 383–389.

[23] L. Rodrigues, A. Junior, and F. Lobato, "Disability-Related News: An Analysis of User-Generated Content on Social Media Posts," in *In Proceedings of the 16th National Meeting on Artificial and Computational Intelligence*. SBC, 2020. doi: https://doi.org/10.5753/eniac.2019.9336

[24] Y. Chen, H. Zhang, R. Liu, Z. Ye, and J. Lin, "Experimental explorations on short text topic mining between lda and nmf based schemes," *Knowledge-Based Systems*, vol. 163, 2019. doi: https://doi.org/10.1016/j.knosys.2018.08.011

[25] J. Silva Junior, R. Rossi, and F. Lobato, "A Lyric-Based Approach for Brazilian Music Knowledge Discovery: Brazilian Country Music as a Case Study," in *In Proceedings of the 16th National Meeting on Artificial and Computational Intelligence*. SBC, 2020. doi: https://doi.org/10.5753/eniac.2019.9348

[26] Z. Chen, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh, "Leveraging multi-domain prior knowledge in topic models," in *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

DOI: 10.15439/2020F54
ISSN 2300-5963 ACSIS, Vol. 21

Context Clustering-based Recommender Systems

Eyad Kannout
Institute of Informatics, University of Warsaw
Banacha 2, Warsaw, Poland
eyad.kannout@mimuw.edu.pl

***Abstract*—Recommender systems have gained lots of attention due to the rapid increase in the amount of data on the internet. Therefore, the demand for finding more advanced techniques to generate more useful recommendations becomes an urgent. The increasing need for generating more relevant recommendations led to the emergence of many novel recommendation systems, such as Context-aware Recommender System (CARS), which is based on incorporating the contextual information in recommendation systems. The goal of this paper is to propose new recommender systems that utilize the contextual information to find more relevant recommendations.**

In this paper, we propose CoCl, a novel Context Clustering-based recommender system. We introduce two approaches which utilize the contextual information and KMeans clustering algorithm to generate new forms of user-item matrices. We show that the accuracy of CoCl which uses the new user-item matrices has been improved comparing with the accuracy of classical recommender system which uses the original user-item matrix.

Keywords: collaborative filtering, context-aware recommender system, contextual information, clustering, accuracy of predictions, quality of recommendations.

I. Introduction

THROUGHOUT the past decade, along with the rapid expansion of the online services, many e-commerce, e-tourism, e-resource services, social media, and retail companies started leveraging the power of data in order to boost their profits by assisting the customers in discovering interesting items/products in a huge amount of online data. For the sake of achieving this goal, a recommender system (RS) must be implemented, and therefore, the demand for recommender systems have been increased more than ever before.

In theory, traditional, or two-dimensional (2D), recommender systems tend to estimate user preferences or user's ratings based on the ratings given by the users to other items, and possibly on some other information, such as user demographics and item characteristics. However, early recommender systems emerged without taking into consideration any contextual information, such as time, location, and the company of other people when providing recommendations. During the past decade, the increasing need for generating more relevant recommendations led to the emergence of many novel recommendation systems, such as context-aware recommender systems (CARS), social recommender systems, and group recommender systems.

Recently, the field of context-aware recommender systems (CARS) has attracted a lot of attention due to its importance in many recommendation applications. Although an increasing number of papers on context-aware recommender systems have been appeared recently, this field is still considered as relatively new and several challenges that need more attention by the current researchers still exist. Therefore, in this paper, we work on finding new methods that incorporate the contextual information in recommendation systems to generate more useful and user-related recommendations.

The main contributions of this paper are as follows: 1) proposing two methods which cluster the ratings and the users in user-item matrix using the contextual information; 2) producing new aggregated forms of user-item matrix based on previous grouping of ratings and users; 3) employing collaborative filtering model to predict missing preference of a user for an item using new aggregated user-item matrices.

The remainder of this paper is organized as follows. In Section 2, we provide background information for collaborative filtering and context-aware recommender systems. Section 3 describes the problem we study in this paper and reviews its related works. In Section 4, we present CoCl, a novel context clustering based recommender system. Section 5 evaluates and compares CoCl against traditional recommender system. Finally, in Section 6, we draw conclusions and make suggestions of possible future work.

II. Literature Review for Recommendation Systems

In this section, we briefly summarize the academic knowledge on collaborative filtering as well as context-aware recommender system.

A. Collaborative Filtering

The basic idea behind collaborative filtering is that the users who have similar preferences in the past tend to behave similarly in the future [1]. The recommendations made by this methodology are based on information about similar users and items. Collaborative filtering methods only rely on user ratings or user interactions. That means there is no need to have additional information about items or users. Moreover, the user's ratings can be acquired explicitly or implicitly (e.g., products bought, songs heard, movies watched, visited pages) [4], so collaborative filtering methods can be used even when the user does not explicitly provide ratings for the items. In the literature, collaborative filtering methods can be grouped in two general classes (i) memory-based techniques and (ii) model-based techniques.

In memory-based technique, the rating history is directly used to predict rating of items that the user has not yet seen. This can be done in two ways: (i) user-based collaborative filtering and (ii) item-based collaborative filtering. In the former method, a set of neighbor users will be selected based on similarity in their rating history to the targeted user. Then, the recommendations will be produced based on top-rated products liked by neighbor users. The item-based collaborative filtering is just an analogous procedure to the previous method. Here, for each item, a set of k-nearest items will be selected. Then, for every product that the target user has not rated before, we estimate the rating using the closest neighbors which are rated previously by the target user. It is important to note that every neighbor has a weight, which reflects the degree of similarity, that will be used in process of rating estimation. However, the most popular metrics used to calculate the similarity between different users, or items, are cosine similarity and Pearson correlation.

In contrast to memory-based technique, which uses the stored ratings directly in the prediction, the model-based technique use these ratings to learn a predictive model. Basically, the learning process is based on matrix factorization which uses the rating history to learn the latent preferences of users and items in order to make a prediction for the missing ratings. Matrix factorization is an unsupervised learning method for dimensionality reduction. The most popular techniques applied for dimensionality reduction are Principal Component Analysis (PCA), Singular Value Decomposition (SVD), Probabilistic Matrix Factorization (PMF), Matrix Completion Technique, Latent Semantic methods, and Regression and Clustering [2] [5].

B. Context-aware Recommender System

The basic idea behind context-aware recommender systems (CARS) is to incorporate the contextual information into recommendation process in order to recommend more relevant items to users under certain circumstances [2] [3]. Many researchers and practitioners have recognized that it is very important to consider relevant contextual information, such as weather, time, location and mood, when providing recommendations. For example, the vacation packages proposed by a travel agency in the winter can be very different from the one proposed in the summer. Thus, the main goal of context-aware recommender system is to consider the contextual information when providing recommendations [6]. After gathering the relevant context, explicitly, implicitly or by inferring, the following question arises: how can we incorporate the context in the recommendation process?. However, Adomavicius and Tuzhilin [7] identified three different approaches to achieve this goal as follows:

- Contextual pre-filtering: here the context information is used to select only the most relevant data from the data set. In other words, information about the current context is used for selecting the relevant set of data records (i.e., ratings). Then, ratings can be predicted using any traditional 2D recommender system on the selected data [8].
- Contextual post-filtering: the context information is ignored during the recommendation process, only the resulting set is contextualized. In this approach, the contextual information is initially ignored, and the ratings are predicted using any traditional 2D recommender system on the entire data. Then, the resulting set of recommendations is adjusted (contextualized) for each user using the contextual information [8].
- Contextual modeling: the recommendation algorithm is altered to include the context and consider it when calculating recommendations. In other words, the contextual information is used directly in the modeling technique as part of rating estimation [8] [9].

III. Related Works

Before presenting CoCl, we review some of the research literature related to contextual collaborative filtering approaches that utilize contextual information to improve recommendation quality.

Over the past decade, a lot of research concerned with context-aware recommender systems has been presented. Palmisano et al. [10] has proved that contextual information, such as age, time, and location, is very useful when predicting customer behavior. Recently, many researches started focusing on the use of context for user-item sub-grouping. Zhong et al. [11] and Liu et al. [12] used decision trees to partition the original rating matrix hierarchically by grouping similar users and items together, and then using the matrix factorization technique to predict missing preference of a user for an item using the partitioned matrix. The previous sub-grouping methods can only handle categorical contexts, and to mitigate this problem, Xiaolin Zheng et al. [13] proposed the use of spectral clustering for user-item sub-grouping, which can handle both categorical and continuous contexts. A new recommender system which is based on matrix factorization has been proposed by Xiaoyao et al. [14]. They considered many factors while building the recommender system, such as contextual information, user ratings and item feature. Using K-modes algorithm, they optimized the process of building the recommender system by clustering user-item dataset which eventually reduces the computation complexity. However, they performed extensive experiments to demonstrate that their method improves the accuracy of generated recommendations.

According to previous findings, sub-grouping has been proved to be valuable for better performance in collaborative filtering methods, but we believe there is space left for further improvements by discovering more advanced grouping approaches.

IV. CoCl Recommender System

In this section, we present CoCl, a Context Clustering based recommender system. We first formalize the context-aware recommendation problem. Then, we describe our proposed contextual clustering model that is used in CoCl.

A. Problem Definition

The main problem we address in this paper is to improve the traditional collaborative filtering approach by incorporating the context in the process of building the recommender system. The main idea is to produce new forms of user-item matrices, also known as utility matrices, by clustering, aggregating and splitting the records in this matrix. However, two approaches will be provided for grouping or clustering the rating records in utility matrix. In the first approach, called RateClust, the ratings in the utility matrix will be grouped in such a way that the ratings with similar contextual information will be together. In the second approach, called UserClust, the users in the utility matrix will be grouped based on their ratings in dedicated contexts.

B. Contextual Clustering Model

In this section, we introduce the reader to our proposed recommendation model. As mentioned before, CoCl proposes a hybrid model that utilizes the contextual information and KMeans clustering algorithm to create new forms of user-item matrices. Then, we apply the traditional collaborative filtering approach on these new matrices to get many recommender systems which give us more accurate results than building one recommender system using the original dataset. In this experiment, we present two approaches to utilize the contextual information for the purpose of clustering the data in user-item matrix: (i) RateClust, and (ii) UserClust.

In RateClust, we aim to group the ratings that are given in similar contexts. As we mentioned in previous section, the contextual information in our dataset describes the situation in which the user consumed the item. In this approach, the contextual information space is represented by an array of vectors where every vector, i.e. C = (c1, c2, · · ·, cl) represents the contexts associated with one rating in user-item matrix. The values in this vector describes the situation for every context variable. For instance, the first context variable in our dataset is time which is represented by five values as follows: morning = 1, afternoon = 2, evening = 3, night = 4, missing value = -1. After creating this array of vectors, we passed it to KMeans algorithm that helps us to cluster the ratings, and subsequently divide the user-item matrix into smaller parts.

On the other hand, our goal in UserClust, is to group the users that share the same behaviour in similar contexts. In other words, we aim to cluster the users based on their ratings in particular context. For instance, in our experiment, we select the mood context which is represented by positive, neutral, negative and missing (unknown). Then, for each user, we calculate the average rating given in every mood possible value. So, for every user, we have average rating for positive, neutral, negative and missing values. The result of applying previous step for all users is an array of vectors which is used to cluster the users. Finally, the output of users' clustering is utilized to split the user-item matrix into smaller groups which contain the users who rate the items similarly in the same context.

After clustering the records in the original user-item matrix using previous approaches, we generate two new versions of user-item matrix by aggregating the ratings given for each movie in each cluster. For instance, if the same movie has been rated by two users who belong to the same cluster, then the ratings given by both users will be replaced by their average. The new generated matrices can be utilized in different ways while building the recommender system. One way is to divide the aggregated user-item matrix into smaller matrices based on the cluster the records belong to. Then, many recommender systems can be build using these smaller matrices. However, this approach is useful when we have enough number of records belong to each cluster. Another way is to just build one recommender system without dividing the aggregated user-item matrix. More information about these approaches will be provided in the next section while comparing the performance of CoCl models with traditional collaborative filtering model.

V. Evaluation for CoCl

In this section, we conduct comprehensive experiments to evaluate the performance of CoCl by comparing the recommendations accuracy with classical collaborative filtering recommender system.

A. Dataset

In our experiments, we used LDOS-CoMoDa[1] dataset which is presented by KOŠIR et al. [15]. LDOS-CoMoDa is a context rich movie recommender dataset that consists of 200 users, who gave 2296 ratings for 4138 movies in twelve pieces of contextual information. However, the contextual information is explicitly acquired from the users directly after watching the movies. Moreover, this dataset is collected from real user-item interaction and not from hypothetical situation or user's memory of past interactions. The context variables in LDOS-CoMoDa dataset are presented in Table I. The values of context variables in this dataset is represented by numerical values. For example, in daytaype variable, Working day is represented by 1, Weekend by 2, Holiday by 3.

TABLE I: Context variables in LDOS-CoMoDa dataset

time	Morning, Afternoon, Evening, Night
daytype	Working day, Weekend, Holiday
season	Spring, Summer, Autumn, Winter
location	Home, Public place, Friend's house
weather	Sunny / clear, Rainy, Stormy, Snowy, Cloudy
social	Alone, My partner, Friends, Colleagues, Parents, Public, My family
endEmo	Sad, Happy, Scared, Surprised, Angry, Disgusted, Neutral
dominantEmo	Sad, Happy, Scared, Surprised, Angry, Disgusted, Neutral
mood	Positive, Neutral, Negative
physical	Healthy, Ill
decision	User decided which movie to watch, User was given a movie
interaction	first interaction with a movie, n-th interaction with a movie

[1] https://www.lucami.org/en/research/ldos-comoda-dataset/

Also, it is worth to note that the entire dataset is splitted into training, testing and validation sets where:

- Training set is used to build the recommender systems using several algorithms, where each algorithm has predefined set of input parameters. The optimal values of these parameters is discovered using the validation set.
- Testing set is used to compare the accuracy of different recommender systems.
- Validation set is used to discover (i) the optimal parameters settings for each recommender system algorithm and (ii) the optimal number of clusters. More details about these two steps will be provided in next subsection.

B. Parameters Selection

In order to evaluate our model, various collaborative filtering algorithms have been applied to generate recommendations based on user-item rating matrix. We use Surprise library which provides various implementations of collaborative filtering algorithms [16]. These algorithms are (i) matrix factorization-based algorithms, such as singular value decomposition (SVD) and non-negative matrix factorization (NMF) and (ii) k-nearest neighbors-based algorithms, such as KNNBasic, KNNBaseline, KNNWithMeans and KNNWithZScore and (iii) other types of algorithms, such as CoClustering, SlopeOne, NormalPredictor and BaselineOnly. The main challenge here is to determine the optimal values of hyper-parameters for every algorithm. This is extremely important since the performance of the recommender system will be impacted based on those values. Moreover, selecting optimal parameters settings for every algorithm is also important to conduct fair and reliable experiments. To tackle this challenge, we use grid search with cross-validation (GridSearchCV) tool which automates the process of tuning the hyper-parameters for every algorithm mentioned before.

On the other hand, we used silhouettes score to select the optimal number of clusters for both versions of CoCl, RateClust and UserClust. The silhouettes score is calculated for each instance based on below formula:

$$SilhouetteScore = (x - y)/\max(x, y)$$

where, y is the mean distance to the other instances in the same cluster (mean intra-cluster distance), x is mean distance to the instances of the next closest cluster (mean nearest-cluster distance).

The silhouettes score, or silhouettes coefficient, varies between 1 and -1. A value close to 1 implies that the instance is far away from the neighboring clusters; hence, it is a part of the right cluster. Whereas, a value close to -1 indicates that the value is assigned to the wrong cluster. A value close to 0 means that the instance is very close to the decision boundary between two neighboring clusters [17].

Fig 1 shows that 8 is the optimal number of clusters that needs to be selected to group the users based on their ratings in given context.

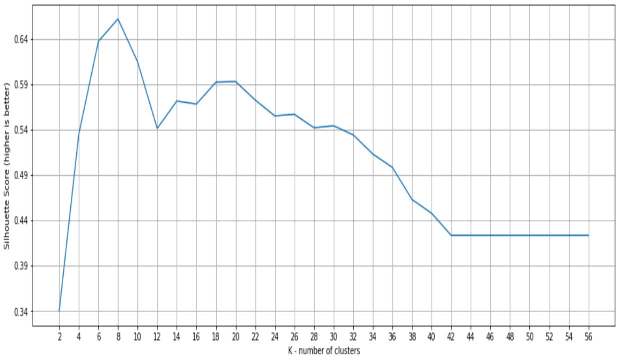


Fig. 1: Calculating mean silhouettes score over all samples for different number of clusters

C. Performance Comparison and Analysis

We use the standard Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) metrics to measure and compare the performance of various recommendation models. The RMSE imposes a penalty over the larger errors:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - x_i^{`})^2} \qquad (1)$$

While MAE measures the average magnitude of the errors in a set of predictions, without considering their direction:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|x_i - x_i^{`}| \qquad (2)$$

After generating the new utility matrices, aggregating the ratings, and calculating the weights, we evaluate the recommendation accuracy of context clustering based models by comparing it with classical collaborative filtering model. For the sake of more accurate evaluation, we use different approaches in our comparisons.

We start evaluating CoCl recommender system which utilizes the contexts to build two types of recommendation systems by clustering the ratings (RateClust), and clustering the users (UserClust) in utility matrix. The recommendation systems produced by CoCl will be evaluated using four methods.

In the first one, we use cross-validation method to check how well the model is able to make new predictions for data which has not seen before. Using k-fold cross-validation is very useful in such scenario when the size of dataset is considered as small. In this method, we use the entire aggregated dataset to build each recommender system in CoCl. we split this data into two parts, training, and testing. The training, which is equal to 85 percent of entire dataset, is used to evaluate every model using repeated cross-validation method. While the rest of the data is used as testing set for final general evaluation. The same criteria is applied to split the original, not aggregated, dataset which is used to evaluate classical recommender system. With the objective of conducting fair and reliable comparison, we ensure that the same records have been used in every fold while evaluating all models using cross-validation method. The only difference between the folds is that the aggregated ratings have been used to evaluate CoCl models while the original ratings have been used to evaluate traditional model. We compare the accuracy between CoCl and classical models by calculating the average of RMSE and MAE which are generated in each fold. However, we repeat the same comparison by using different algorithms to build the recommender system. The comparison results of this evaluation methodology for RateClust and UserClust are shown in Tables II and III respectively. Moreover, Fig 2 shows the generalization assessment of final models fit on entire training set. It is important to note that the performance of CoCl models outperforms the performance of traditional model in every iteration of cross-validation method. In conclusion, this experiment shows that CoCl models reduce MAE/RMSE by around 6% on average compared to classical collaborative filtering model.

In the second evaluation methodology, we use holdout evaluation method where the entire aggregated dataset is splitted into training and testing sets. The training and testing records have been selected in a way such that from each cluster we select 85 percent of data as training and 15 as testing. In this way, we guarantee that records from all clusters have been included in training and testing sets. For classical recommender system, we select the same records which are selected before as training and testing but from original user-item matrix without any aggregation in ratings. After that, we compare the accuracy by calculating RMSE and MAE for each recommender system. However, we repeat the same comparison by using different algorithms to build the recommender system. The comparison results of this evaluation methodology are shown in Tables IV and V. This experiment demonstrates that using contextual information improves recommendation quality.

In the third evaluation methodology, we split the records in aggregated user-item matrix into smaller matrices based on the clustering results. Then, we build a clustering based recommender systems for each one of them. These recommender systems will be compared with classical recommender system which is created based on original dataset without any splitting or aggregation. The comparison results of this evaluation methodology are shown in Tables VI and VII. It is important to note that the recommender systems which are generated using the small aggregated utility matrices perform better than the one which is generated based on entire aggregated matrix.

In the fourth method, we create ensemble recommender systems for RateClust, UserClust and classical models. The main idea is to aggregate the ratings produced by each algorithm in order to produce the final ratings in the target recommender system. For the sake of improving the results, we select the best three algorithms that produce the most accurate results in previous evaluation methods. These algorithms are SVD, KNNBaseline and BaselineOnly. However, while building the clustering-based recommender systems; the entire aggregated dataset is used without any splitting. The comparison results of this evaluation methodology are shown in Tables VIII and IX. The results indicate that the clustering based recommender systems achieve better accuracy than the one which is produced using original dataset. However, RateClust models has slightly better accuracy than UserClust models.

We notice that in all experiment scenarios, CoCl models outperform traditional collaborative filtering model. Moreover, the experiment results demonstrate the advantage of considering the contextual information in the area of recommender systems.

VI. Conclusions and Future Work

In this paper, we have proposed CoCl, a novel context clustering based recommender system, which methodically incorporates the context in the process of generating the recommendations. Two versions of CoCl have been introduced, RateCust, rating-based clustering recommender system, and UserClust, user-based clustering recommender system. We proposed to use KMeans clustering algorithm to cluster the data in user-item matrix in order to produce new forms of utility matrices which can achieve better accuracy using collaborative filtering techniques. To evaluate our proposed models, we conducted comprehensive experiments on LDOS-CoMoDa dataset using Surprise library which provides various implementations of collaborative filtering algorithms that can be used for building and analyzing recommender systems. Moreover, multiple evaluation methodologies have been proposed to compare between models. The experimental results can reveal the answer for our research question stated above. The results illustrate that CoCl accuracy outperforms classical collaborative filtering approach in all experiments. However, experiments also indicate that RateClust approach has slightly better performance than UserClust approach.

In the future work, we are interested in applying CoCl to some real world application scenarios. For instance, CoCl can be integrated into a website where the recommender system can generate some recommendations in real time based on the current context. Also, we need to take into account the dynamics of evolving user preference by periodically updating the user-item matrices based on recent recommendations. Moreover, in the scenario of splitting the user-item matrix into smaller ones based on clusters, we need to determine in real time the proper recommender system that can produce the best recommendations for specific user and in dedicated context. Also, LDOS-CoMoDa is considered to be rather small dataset, and hence, another suggestion for future work is to evaluate CoCl against larger and more complex datasets.

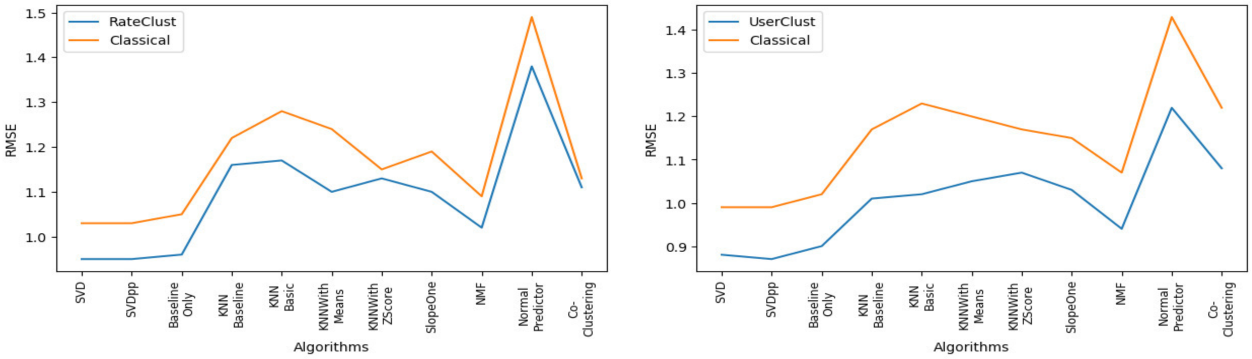
On the other hand, in our research we focused on memory based techniques in collaborative filtering, so another important aspect to consider is to evaluate CoCl using model based techniques. Furthermore, we are particularly interested in using more advanced machine learning techniques to incorporate the contexts in recommendation systems. Also, another direction of future work is to use distributed stream processing engines, like Apache Flink, to examine parallel

TABLE II: Cross Validation - Rating-based clustering VS Classical

Model	Metric	SVD	SVDpp	Baseline Only	KNN Baseline	KNN Basic	KNN WithMeans	KNN WithZScore	Slope One	NMF	Normal Predictor	Co-Clustering
RateClust	RMSE MAE	0.96 0.76	0.95 0.76	0.95 0.76	1.07 0.78	1.08 0.79	1.08 0.82	1.09 0.82	1.06 0.82	1.03 0.83	1.36 1.08	1.10 0.83
Classical	RMSE MAE	1.01 0.81	1.01 0.81	1.02 0.82	1.14 0.87	1.18 0.89	1.16 0.89	1.12 0.85	1.14 0.89	1.09 0.88	1.41 1.13	1.14 0.88

TABLE III: Cross Validation - User-based clustering VS Classical

Model	Metric	SVD	SVDpp	Baseline Only	KNN Baseline	KNN Basic	KNN WithMeans	KNN WithZScore	Slope One	NMF	Normal Predictor	Co-Clustering
UserClust	RMSE MAE	0.95 0.75	0.94 0.74	0.96 0.76	1.13 0.84	1.13 0.84	1.08 0.80	1.08 0.81	1.09 0.84	1.01 0.81	1.33 1.04	1.13 0.86
Classical	RMSE MAE	0.99 0.80	0.99 0.80	1.03 0.84	1.17 0.90	1.23 0.93	1.20 0.93	1.17 0.91	1.15 0.91	1.06 0.87	1.53 1.21	1.18 0.91



(a) Rating-based clustering VS Classical

(b) User-based clustering VS Classical

Fig. 2: Generalization assessment using testing set

TABLE IV: Performance Comparison - Rating-based clustering (one recommender system) VS Classical

Model	Metric	SVD	SVDpp	Baseline Only	KNN Baseline	KNN Basic	KNN WithMeans	KNN WithZScore	Slope One	NMF	Normal Predictor	Co-Clustering
RateClust	RMSE MAE	0.86 0.69	0.85 0.68	0.86 0.68	1.02 0.72	1.03 0.73	1.02 0.78	1.06 0.79	0.98 0.75	0.92 0.73	1.31 1.07	0.99 0.76
Classical	RMSE MAE	0.94 0.76	0.94 0.77	0.95 0.76	1.11 0.83	1.08 0.83	1.06 0.80	1.08 0.83	1.05 0.82	1.00 0.80	1.38 1.12	1.11 0.86

TABLE V: Performance Comparison - User-based clustering (one recommender system) VS Classical

Model	Metric	SVD	SVDpp	Baseline Only	KNN Baseline	KNN Basic	KNN WithMeans	KNN WithZScore	Slope One	NMF	Normal Predictor	Co-Clustering
UserClust	RMSE MAE	0.89 0.69	0.89 0.70	0.89 0.70	0.95 0.68	0.96 0.68	1.00 0.75	1.05 0.78	0.96 0.75	0.98 0.76	1.34 1.07	0.97 0.73
Classical	RMSE MAE	1.04 0.84	1.03 0.84	1.05 0.85	1.12 0.87	1.18 0.89	1.15 0.89	1.14 0.88	1.17 0.92	1.14 0.92	1.49 1.22	1.18 0.93

TABLE VI: Performance Comparison - Rating-based clustering (Multiple recommender systems) VS Classical

Model	Metric	SVD	SVDpp	Baseline Only	KNN Baseline	KNN Basic	KNN WithMeans	KNN WithZScore	Slope One	NMF	Normal Predictor	Co-Clustering
RateClust	RMSE	0.78	0.79	0.78	0.82	0.83	0.80	0.80	0.86	0.82	1.17	0.86
	MAE	0.61	0.62	0.60	0.61	0.62	0.56	0.56	0.66	0.62	0.95	0.67
Classical	RMSE	0.94	0.93	0.95	1.08	1.12	1.08	1.06	1.05	1.00	1.35	1.06
	MAE	0.77	0.76	0.76	0.83	0.83	0.83	0.80	0.82	0.80	1.06	0.83

TABLE VII: Performance Comparison - User-based clustering (Multiple recommender systems) VS Classical

Model	Metric	SVD	SVDpp	Baseline Only	KNN Baseline	KNN Basic	KNN WithMeans	KNN WithZScore	Slope One	NMF	Normal Predictor	Co-Clustering
UserClust	RMSE	0.88	0.89	0.88	0.89	0.90	0.88	0.89	0.93	1.04	1.26	0.91
	MAE	0.68	0.69	0.69	0.64	0.64	0.60	0.61	0.71	0.82	1.00	0.68
Classical	RMSE	1.04	1.03	1.05	1.12	1.18	1.15	1.14	1.17	1.14	1.39	1.15
	MAE	0.85	0.83	0.85	0.87	0.89	0.89	0.88	0.92	0.92	1.12	0.92

TABLE VIII: Rating-based clustering VS Classical (Ensemble Recommender System)

model	Ensemble Recommender System	
	RMSE	MAE
RateClust	0.95	0.74
Classical	1.03	0.79

TABLE IX: User-based clustering VS Classical (Ensemble Recommender System)

model	Ensemble Recommender System	
	RMSE	MAE
UserClust	1.00	0.80
Classical	1.08	0.86

implementations of CoCl, in order to make them scalable to infinite streams or large-scale datasets.

Acknowledgment

This paper was partially supported by the grant nr POIR.01.01.01-00-1307/17 entitled by "Development of a hybrid recommendation system for the fashion & cosmetics branch operating, among others based on contextual data using machine-learning mechanisms" co-financed by the European Regional Development Funds, a part of the Smart Growth Operational Programme.

References

[1] J.S. Breese, D. Heckerman, and C. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI1998), 1998.

[2] F. Ricci, L. Rokach, and B. Shapira, Eds., "Recommender Systems Handbook," Springer New York Heidelberg Dordrecht London, 2015, doi: 10.1007/978-0-387-85820-3.

[3] A. Lommatzsch, B. Kille, and S. Albayrak, "Incorporating context and trends in news recommender systems," In Proceedings of the International Conference on Web Intelligence (WI '17). ACM, New York, NY, USA, 1062-1068, 2017, doi: 10.1145/3106426.3109433.

[4] S.K. Lee, Y.H. Cho, and S.H. Kim, "Collaborative filtering with ordinal scale-based implicit ratings for mobile music recommendations," Information Sciences 180 (11) (2010) 2142–2155, doi: 10.1016/j.ins.2010.02.004.

[5] L.E.M. FERNÁNDEZ, "Recommendation System for Netflix," VRIJE UNIVERSITEIT AMSTERDAM, 2018.

[6] F. Shi, C. Ghedira, and J.-L. Marini, "Context Adaptation for Smart Recommender Systems," IEEE Computer Society 1520-9202/15/31.00 © 2015 IEEE, doi: 10.1109/MITP.2015.96.

[7] G. Adomavicius, and A. Tuzhilin, "Chapter 6: Context-Aware Recommender Systems," in Recommender Systems Handbook, F. Ricci, L. Rokach and B. Shapira, Eds., Springer, Boston, MA, 2015, doi: 10.1007/978-1-4899-7637-6_6.

[8] U. Panniello, A. Tuzhilin, M. Gorgoglione, C. Palmisano, and A. Pedone, "Experimental comparison of pre- vs. post-filtering approaches in context-aware recommender systems," Proceedings of the 2009 ACM Conference on Recommender Systems, 2009, doi: 10.1145/1639714.1639764.

[9] Y. Shen, Y. Deng, A. Ray, and H. Jin, "Interactive Recommendation via Deep Neural Memory Augmented Contextual Bandits," In Proceedings of RecSys 2018 – the ACM Conference Series in Recommendation systems, Vancouver, 2018, doi: 10.1145/3240323.3240344.

[10] C. Palmisano, A. Tuzhilin, and M. Gorgoglione, "Using context to improve predictive modeling of customers in personalization applications," Knowledge and Data Engineering, IEEE Transactions on 20(11):1535–1549, 2008, doi: 10.1109/TKDE.2008.110.

[11] E. Zhong, W. Fan, and Q. Yang, "Contextual collaborative filtering via hierarchical matrix factorization," In Proceedings of the SIAM International Conference on Data Mining, 744–755, 2012, doi: 10.1137/1.9781611972825.64.

[12] X. Liu, and K. Aberer, "Soco: a social network aided context-aware recommender system," In Proceedings of the 22nd international conference on World Wide Web, 781–802, 2013, doi: 10.1145/2488388.2488457.

[13] C. Chen, X. Zheng, Y. Wang, F. Hong, and Z. Lin, "Context- ware Collaborative Topic Regression with Social Matrix Factorization for Recommender Systems," In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence. 9-15, 2014.

[14] X. Zheng, Y. Luo, L. Sun, and F. Chen. 2016. "A New Recommender System Using Context Clustering Based on Matrix Factorization Techniques," Chinese Journal of Electronics. Vol.25, No.2, 2016, doi: 10.1049/cje.2016.03.021.

[15] A. Kosir, A. Odic, M. Kunaver, M. Tkalcic, and J. F. Tasic, "Database for contextual personalization," Elektrotehniski Vestnik/Electrotechnical Review, vol. 78, pp. 270–274, 2011.

[16] N. Hug, "Home," Surprise. [Online]. Available: http://surpriselib.com/.

[17] P.J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," Journal of Computational and Applied Mathematics 20, pp. 53-65, 1987, doi: 10.1016/0377-0427(87)90125-7.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 93–100 DOI: 10.15439/2020F141 ISSN 2300-5963 ACSIS, Vol. 21

Age-related Spike Timing Dependent Plasticity of Brain-inspired Model of Visual Information Processing with Reinforcement Learning

Petia Koprinkova-Hristova
Institute of Information and Communication Technologies,
Bulgarian Academy of Sciences
Sofia, Bulgaria
Email: pkoprinkova@bas.bg

Nadejda Bocheva
Institute of Neurobiology,
Bulgarian Academy of Sciences
Sofia, Bulgaria
Email: nadya@percept.bas.bg

***Abstract*—The paper summarizes our efforts to develop a spike timing neural network model of dynamic visual information processing and decision making inspired by the available knowledge about how the human brain performs this complicated task. It consists of multiple layers with functionality corresponding to the main visual information processing structures starting from the early level of the visual system up to the areas responsible for decision making based on accumulated sensory evidence as well as the basal ganglia modulation due to the feedback from the environment. In the present work, we investigated age-related changes in the spike timing dependent plastic synapses of the model as a result of reinforcement learning.**

I. Introduction

FOR centuries scientists were trying to discover the way we learn how to behave in an unknown environment without prior information about which the proper actions are. Every day humans make a large number of decisions based on sensory information as well as on previously accumulated experience. In decision making based only on sensory information the response is determined by the stimulus characteristics. However, the pioneering work of Pavlov on conditional and unconditional reflexes of living creatures revealed yet another intriguing characteristics of the natural intelligence: the ability to plane its future behaviour not only based on its current sensory information but also on its past experience accumulated via trial and error thus accounting for its past actions outcome in similarly sensed environment. It gives rise to development of theory of reinforcement learning starting with the seminal work of [1] and their actor-critic architecture that was able to learn from simple punish/reward feedback from the environment. It can be considered as one of the first artificial systems possessing a kind of artificial intelligence. Parallel to the research based on behavioral experiments neurologists tried to discover the brain counterparts of sensory information processing as well as of the reinforcement learning.

The visual system of human brain was probably the most investigated by neurobiologists. The hierarchical processing structures starting from our light sensors - the eyes - through optic nerve to visual cortex were well established. It also has been shown that several brain areas like lateral intraparietal area (LIP) accumulate evidence supporting the alternative decisions ([2]) and transform eye sensory information in a decision variable that directs action. Many models assume that a choice is made when the accumulated evidence for one of the sensory signals reaches a predefined value. It has been shown (e.g.[3]) that the basal ganglia can modulate this threshold level. Other studies (e. g. [4]) imply that the basal ganglia could also modify the rate of sensory evidence accumulation. These results suggest an important role of the basal ganglia in perceptual decision making. Existing evidence (e.g. [5], [6], [7]) suggests also a significant role of the basal ganglia on learning by trial and error to acquire a reward, i.e. reinforcement learning. The role of the basal ganglia in reinforcement learning is related to the differential responses of the dopaminergic neurons in one structure of the basal ganglia (substantia nigra compacta) to unexpected and predicted rewards and to the omission of an expected reward. Recently, several modeling attempts (e.g.[8], [9]) try to integrate these two functions of the basal ganglia - in decision making processes and in reinforcement learning in a common framework starting from the cortical input.

In contrast to these models, in previous work [10] we developed a spike timing neural network model that includes the major structures related to dynamic visual information processing i.e. including the structures that provide the sensory information for making a decision starting from the retinal input. The parallel structure of the model was adopted from [11] while the basal ganglia connectivity was adopted from [12]. Main advances in our work in comparison to these two previous works were as follows:

- While in [12] the model consisted of cellular network structures whose neurons are modelled by firing rate equations, in our model we used spike timing neurons organized in layers with the same connectivity.
- In [12] dopamine signal is calculated as temporal differ-

This work is supported by the Bulgarian Science Fund project No DN02/3/2016 "Modelling of voluntary saccadic eye movements during decision making".

ence error that was directly exploited to adjust dopamine synapses. In our model we have an additional layer of neurons whose spiking activity was equivalent to dopamine release into corresponding synapses having spike timing dependent plasticity.

- In [12] the layer responsible for visual information processing and generating sensory input to the basal ganglia is simplified and in [11] it is completely missing while our model includes multiple layers corresponding to hierarchical brain structures performing dynamic visual information processing.
- In contrast to [11] where lateral connections within layers are limited, our model has much more elaborated connectivity similar to that proposed in [12].

These characteristics of our model made it more realistic and provide greater opportunities for understanding the process of learning and decision making in human brain.

The model was implemented using NEST 2.12.0 simulator [13].

Further attempts to improve our model were as follows:

- Enhanced connectivity of visual information processing layers with multiple feedback connections and spike timing dependent plasticity in its synapses reported in [14].
- Enhanced feedback connectivity from basal ganglia back to visual cortex.

The present paper summarizes our model structure and connectivity and investigates age-related changes of its dynamic synapses induced by age-specific external reinforcement. It is organized as follows: Section II briefly describes the model structure; next simulation experiment with moving dot stimulus and external age-related reinforcement signal was presented and achieved after training values of spike timing dependent plastic (STDP) synapses were presented and commented; the paper finishes with concluding remarks and directions for future work.

II. Model Structure

Based on the available data about human brain structures playing role in visual motion information processing and decision making, as well as their connectivity, the hierarchical model proposed in [15] consists of two basic substructures: related to visual information perception and sensory-based decision making and the basal ganglia and their function on the perceptual decision via external reinforcement.

Each layer consists of neurons positioned in a regular two-dimensional grid. The receptive field (area of neurons from a given layer that are connected to a given neuron from the same or neighbour layer) of each individual neuron depends on the function of the layer it belongs to as well as on its position inside the layer. The neurons' dynamical models as well as intra- and inter-layer connectivity are described in consecutive subsections.

A. Visual information perception and sensory-based decision

The structure of perceptual layers up to LIP area involved in the sensory-based decision making reported in [15], [16], [17] is shown in Fig. 1. It consists of the following layers: Retinal ganglion cells (RGC); Lateral geniculate nucleus (LGN); Thalamic reticulate nucleus (TRN) and Interneurons (IN); Primary visual cortex (V1);Middle temporal (MT) area; Medial superior temporal (MST) area and Lateral intraparietal cortex (LIP).

Following the commonly accepted models from [18], [19], the reaction of retinal ganglion cells to luminosity changes was simulated by a spatiotemporal filter whose spatial component has circular shape modelled by a difference of two Gaussians and the temporal component has a bi-phasic profile determined by difference of two Gamma functions. The continuous signal generated by convolution of this spatiotemporal kernel with the visual stimuli (images falling on the retina) is the electrical output current of retinal cells. Each retinal ganglion cell generates input current and is connected to its corresponding LGN neuron.

The structure of LGN layer is the same as that of the retinal layer. We have two layers of retinal cells and their corresponding LGN neurons, having identical positions of "on-center off-surround" (ON) and "off-center on-surround" (OFF) cells placed in reverse order. Their positions are relative to the visual scene. For the LGN neurons we used the proposed in [20] model whose parameters were determined from in-vivo experiments. This layer sends forward signals to the next layer (V1) and receives excitatory feedback from it directly as well as via inhibitory connections through interneurons and TRN.

The structure of thalamic relay that prepossesses the feedback from the visual cortex (V1) to LGN has structure adopted from [21] as shown on Fig. 1. The interneurons receive excitatory inputs from both retinal neurons (feedforward) and primary visual cortex (feedback) and send inhibitory signal to their corresponding LGN neuron. The TRN neurons mediate excitatory feedback from visual cortex and send another inhibitory input to the corresponding LGN neurons. Since we have a TRN and an interneuron attached to each LGN neuron, their positions coincide on the LGN grid of neuron positions. For simplicity, in our model the feedback connectivity from V1 was the same as the feedforward connectivity from LGN to V1. In [16] the presence and strength of such feedback connectivity on the spiking activity of the primary visual cortex was investigated by simulations. It was demonstrated that it has modulatory effect on the selectivity of V1 neurons.

As in [19], the neurons in V1 layer are separated into four groups - two exciting and two inhibiting populations connected via corresponding excitatory and inhibitory intra-layer (lateral) connections. According to [19] and [18] the ratio of exiting to inhibiting neurons should be 4/1. All neurons are positioned at the same two-dimensional space and the inhibiting neurons are dispersed among bigger groups of exciting neurons. Since the neurons in V1 layer are orientation sensitive, they have elongated receptive fields defined by a Gabor probability function

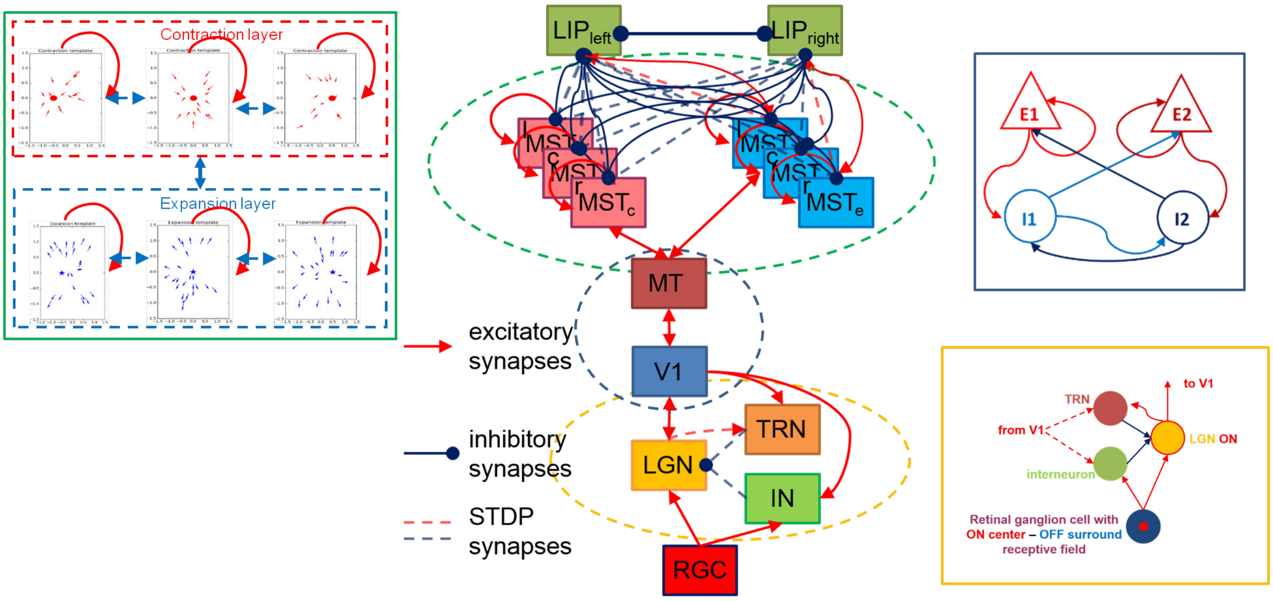


Fig. 1. Model of dynamic sensory information processing. Each box represents a two-dimensional layer of neurons. The color of connections corresponds to their type, i.e. red for excitatory and blue for inhibitory ones.

with orientation and phase parameters like in [22]. The 2D maps containing the neurons' orientations and phases at corresponding 2D grid of the V1 layer should have typical for the mammalian brain "pinwheel-structure". Among the proposed approaches for artificial design of such a structure, that of [23] is relatively new and easily implemented one. That is why we used it to design V1 orientation and phase maps of our model ([17]). The absolute values of lateral connection weights in V1 are determined on the basis of neuronal Gabor correlations with respect to their positions, phases and orientations. The sign of a connection weight depends on whether it is excitatory (positive) or inhibitory (negative). Besides, as in [19], neurons from inhibitory populations connect preferentially to neurons having a receptive field phase difference of about 180°. In our model, we defined the spatial frequencies and standard deviations of the Gabor filters for lateral connection weights so as to obtain approximately circular receptive fields for all neurons in the layer.

The next (MT) layer is the major motion information processing structure and it has identical structure to V1 layer. The lateral connections are designed in the same way while the connections from V1 cells depend on the angle between orientation preferences of each pair of cells according to [24]. The orientation and phase maps of this layer were generated in the same way as in the case of V1 layer.

The following Medial Superior Temporal Area (MST) was modeled like in [25] by two layers sensitive to expansion and contraction movement patterns that occur during the self-motion of the observer . Each MST cell has assigned contraction/expansion pattern template having circular shape and focal point at MT layer. Following [25], the MST neurons have on-center receptive fields. Each MST neuron collects inputs from MT cells corresponding to its pattern template. Both layers have intra- and interlayer excitatory/inhibitory recurrent connections between cells having similar/different sensitivity (see Fig. 1). These lateral connections are determined based on neurons' positions and template similarities. All neurons have Gaussian receptive fields. Connections within expansion/contraction layers are excitatory or inhibitory in dependence on their focal points' similarity. Connections between expansion and contraction layers are all inhibitory and depend both on similarities of their positions and focal points.

LIP area is the last layer of perceptual part of the model that is responsible for making decisions based on accumulated sensory evidence. Since our model aims to decide whether the expansion center of moving dot stimulus is left or right from the stimulus center, in [15] we proposed a task-dependent design of excitatory/inhibitory connections from MST expansion/contraction layers to the two LIP sub-regions whose increased firing rate corresponds to either of two motor responses - eye movement to the left or right. Both LIP areas are connected via excitatory connections to neurons in MST expansion layer having template focal points (left or right) corresponding to their motor responses (left or right). The rest of neurons are connected via inhibitory connections. There are also lateral inhibitory connections between both groups of LIP neurons.

B. Basal Ganglia

In order to modulate LIP decisions using external reinforcement signal, its output (considered as processed and accumulated sensory information) was further fed into a group of subcortical nuclei - Basal ganglia (BG). These include Striatum, Globus Pallidus externa (GPe), Subthalamic Nucleus

(STN), Substantia Nigra pars reticulata (SNr) and Substantia Nigra pars compacta (SNc) [12], [11]. The structure of BG in our model, shown on Fig. 2 combines ideas from both [12] and [11].We excluded the internal segment of the Globus Pallidis (GPi) from the model as it is an output of the Basal ganglia to the thalamus, while we are interested in the effects of BG activity on eye movement control. For this reason, we considered only the other output structure of the Basal ganglia - the SNr as it projects to the superior colliculus (SC), a structure controlling saccade generation.

Like in [12], our model incorporates layers of Striatum, GPe/STN structure and SNr. However, it consists of two parallel structures, receiving inputs from the left and right saccade selecting LIP areas respectively. These two channels (left and right) are connected via mutually inhibiting connections through their GPe areas like in [11]. Additionally, in contrast to [12], our model has a complete 2D layer of neurons producing dopamine neuromodulator (SNc) and dopamine-dependent synapses.

Striatum is divided into two sub-areas depending on the type of dopamine receptors they express (D1 and D2 on Fig. 2). Both are modelled as 2D layers of integrate and fire (IAF) neurons whose lateral connections have short-range excitation and long-range inhibition characteristics like in [12]. These two sub-areas form the inputs to the direct and indirect pathways that process signals through the basal ganglia. The cortical input (coming from the LIP layer) has dopaminergic synapses whose weights were randomly initialized and they are dynamically changed in dependence on the spiking activity of SNc area (considered as dopamine secreting structure). The NEST simulator offers dopaminergic synapse model from [26]. Since our model includes also anti-dopamine synapses (from LIP to D2 sub-area of the Striatum) whose dynamics has to be opposite to that of dopaminergic ones, we've modified the model from [26] by converting the amplitudes A_+ and A_- of the dopamine eligibility trace dynamics from positive to negative.

The Globus Pallidus externa (GPe) and Subtalamic Nucleus (STN) pairs consists of 2D grid of pairs of neurons connected one-to-one via glutamatergic (excitatory) and GABAergic (inhibitory) connections as shown on Fig. 2. The GPe layer has also lateral connections having negative center and positive surround shape as in [12]. The structure receives inhibitory input from the second part of the Striatum (D2) via GPe and sends its output through STN via dopamine-dependent synapses to SNr (so called indirect pathway from Striatum to the BG output layer).

SNr was modelled by a 2D layer having short-range excitatory and long-range inhibitory lateral connections like both Striatum layers. Its input comes from both D1 layer of the Striatum (direct pathway) and GPe/STN structure (undirect pathway) via dopamine dependent synapses. SNr generates BG output to the motor-reaction controlling structure (SC).

SNc is considered as the brain area producing the neuromodulator dopamine in dependence on external motivation (reinforcement) input signal. In contrast to [12], where the dopamine level is calculated using temporal difference error, here we incorporated another 2D layer of neurons. The input to SNc, coming from D1 area of the Striatum, was considered as the value function estimation like in [12]. Thus in order to "produce" the dopamine (temporal difference error) at the output of SNc, we set its inputs to be the value function for two consecutive time steps and the reinforcement signal as follows:

$$V(t) = \mathrm{D1}(t) \tag{1}$$

$$\delta(t) = \mathrm{r(t)} + \gamma V(t+1) - V(t) \tag{2}$$

$$\mathrm{SNc} = F(\delta(t)) \tag{3}$$

i.e. the dopamine release from the SNc is a function F of the temporal difference error δ as in [5]. Here reinforcement signal r(t) is external input current to the neurons in the SNc area ($\mathrm{r}_{\mathrm{left}}$ and $\mathrm{r}_{\mathrm{right}}$ respectively) and the value function V is associated with spiking activity in the D1 part of the Striatum. The discount factor γ was set to 0.9.

Since the SNc has the role of the critic within the model, its input connections from the Striatum were modelled as dopamine dependent synapses too.

Finally, the motor controlling structure SC was modelled by 2D layer of neurons receiving inputs directly from the LIP area (decision according to accumulated sensory information) as well as from the external reinforcement modulated output of BG (via SNr).

The overall model connectivity is also enhanced by excitatory feedback connections from SC to their corresponding D1 and D2 areas of the Stratum as well as to LIP areas following recently reported findings [27], [28], [29]. Moreover we introduced mutually inhibiting connections between the two SC groups.

III. Simulation Results and Discussion

The overall model structure was implemented in NEST [13] simulator. For the neurons in LGN layer conductance-based leaky integrate-and-fire neuron model as in [20] was adopted. For the rest of neurons leaky integrate-and-fire (IAF) model with exponential shaped post-synaptic currents according to [30] was used.

The adjustable parameters in presented simulation are the strengths of dopaminergic synapses that vary in dependence on spiking activity of both SNc layers as well as STDP synapses of the visual perception sub-structure. The reinforcement inputs $\mathrm{r}_{\mathrm{left}}$ and $\mathrm{r}_{\mathrm{right}}$ are both teaching signals that control the dopamine level. In contrast to our preliminary investigations, where both reinforcement signals were constant generating currents of both SNc structures, here they were proportional to the difference between desired SC activity (generating current as in [14]).

The experiments with human test subjects separated into three age groups: 12 young persons (19-34 years old), 11 middle age (36-52 years old) and 12 elderly people (57-84 years old) were conducted and mean reaction time of each

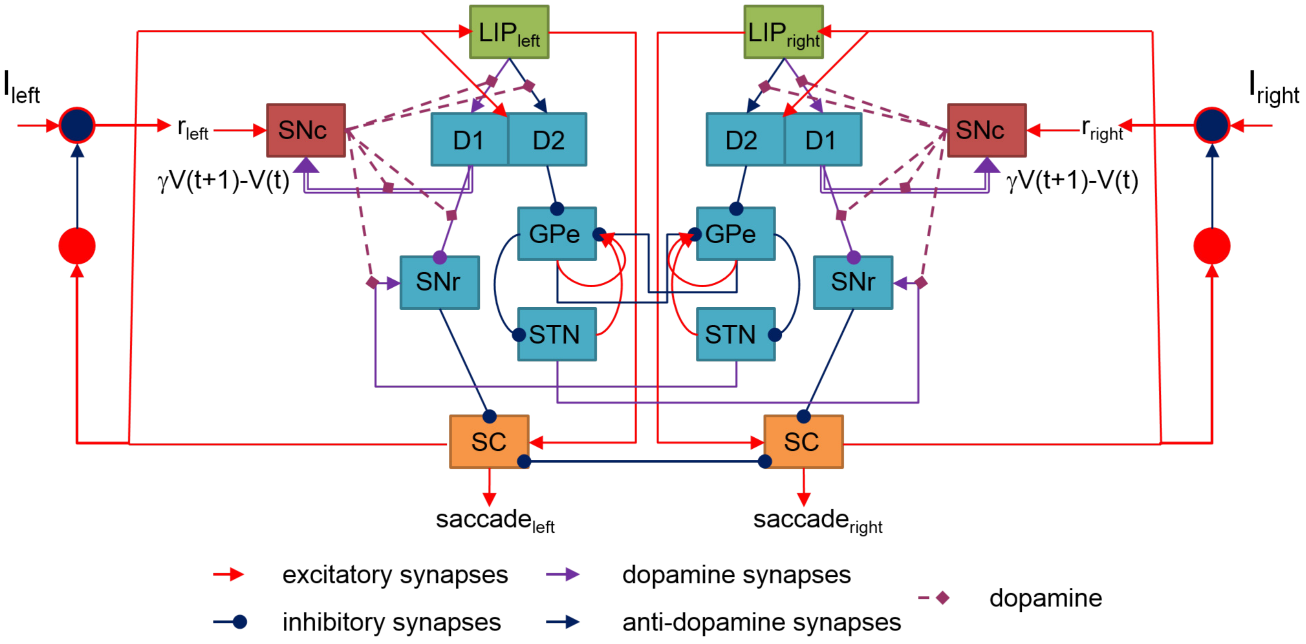


Fig. 2. Basal ganglia structure (in blue). It receives inputs from the decision-making area based on sensory information (LIP) as well as from the dopamine releasing area (SNc) and generates activity biasing saccades generation via SC.

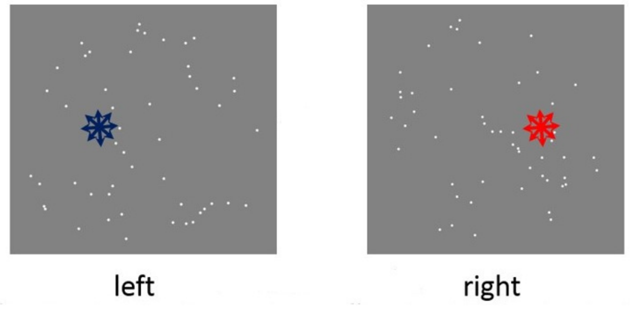


Fig. 3. Stimuli example.

TABLE I
MEAN AND VARIANCE OF WEIGHTS OF CONNECTIONS FROM MSTE TO LEFT LIP IN CASE OF REINFORCEMENT SIGNAL CORRESPONDING TO PERCEPTUAL DECISION (LEFT).

Group	exc. mean	exc. var	inh. mean	inh. var
Young	3.63096	2.17431	-5.82932	5.95606
Middle	3.63059	2.17837	-5.82684	5.95684
Elderly	3.62601	2.17272	-5.83082	5.95357

TABLE II
MEAN AND VARIANCE OF WEIGHTS OF CONNECTIONS FROM MSTE TO RIGHT LIP IN CASE OF REINFORCEMENT SIGNAL CORRESPONDING TO PERCEPTUAL DECISION (LEFT).

Group	exc. mean	exc. var	inh. mean	inh. var
Young	5.18551	5.03905	-5.68507	5.79413
Middle	5.21042	5.07128	-5.68448	5.79909
Elderly	5.19587	4.67182	-5.70433	5.81697

age group was estimated [31]. The visual stimulation consists of projection of moving dot patterns on a computer screen. The test subjects are asked to indicate perceived expansion center that is left or right from the screen center as shown on Fig. 3. Details about experimental set-up were reported in [15].

In current investigation we simulated our model presenting as input moving dot stimuli with expansion center to the left of the screen center. As in [14] we've created training signals as generating currents $\mathrm{I}_{\mathrm{left}}$ and $\mathrm{I}_{\mathrm{right}}$ for the left and right LIP neurons respectively as follows:

$$\mathrm{I}_{\mathrm{left/right}} = A_{left/right}/(1+\exp(k_{left/right}t)) \quad (4)$$

Amplitude $A_{left/right}$ defines maximal input current (in pA) while $k_{left/right}$ determines settling time of the exponent that corresponds to the mean reaction time determined from experiments for each age group. For all three age groups amplitude values were the same: $A_{left} = 200$ and $A_{right} = 100$. In order to achieve approximately the settling time determined from experimental data, parameter $k_{left/right}$ has different values for three age groups (Y - young, M - middle, O - old) with opposite signs for left and right case of stimulus respectively as follows: $k^{Y}_{left/right} = -/+0.02$; $k^{M}_{left/right} = -/+0.01$; $k^{O}_{left/right} = -/+0.005$.

We monitored the changes in STDP synapses of the model. Tables I - IV show the mean values and variances of connection weights between MSTe and LIP layers, while the

TABLE III
MEAN AND VARIANCE OF WEIGHTS OF CONNECTIONS FROM MSTE TO LEFT LIP IN CASE OF REINFORCEMENT SIGNAL OPPOSITE TO PERCEPTUAL DECISION (LEFT).

Group	exc. mean	exc. var	inh. mean	inh. var
Young	5.90917	5.82209	-5.83044	5.95340
Middle	5.90917	5.82209	-5.82591	5.96218
Elderly	5.90917	5.82209	-5.83086	5.95266

TABLE IV
MEAN AND VARIANCE OF WEIGHTS OF CONNECTIONS FROM MSTE TO RIGHT LIP IN CASE OF REINFORCEMENT SIGNAL OPPOSITE TO PERCEPTUAL DECISION (LEFT).

Group	exc. mean	exc. var	inh. mean	inh. var
Young	3.64700	2.18479	-5.95241	6.15924
Middle	3.64970	2.18400	-5.95241	6.15924
Elderly	3.64957	2.18176	-5.95241	6.15924

TABLE V
MEAN OF WEIGHTS OF CONNECTIONS FROM LGN TO TRN IN CASE OF REINFORCEMENT SIGNAL CORRESPONDING TO PERCEPTUAL DECISION (LEFT).

Group	ON1	ON2	OFF1	OFF2
Young	-0.99001	-0.97580	-0.86800	-0.85067
Middle	-0.99126	-0.97732	-0.86824	-0.85042
Elderly	-0.98803	-0.97743	-0.86915	-0.85175

TABLE VI
VARIANCE OF WEIGHTS OF CONNECTIONS FROM LGN TO TRN IN CASE OF REINFORCEMENT SIGNAL CORRESPONDING TO PERCEPTUAL DECISION (LEFT).

Group	ON1	ON2	OFF1	OFF2
Young	0.00518	0.00444	0.00619	0.00796
Middle	0.00514	0.00463	0.00625	0.00798
Elderly	0.00528	0.00437	0.00640	0.00795

Tables V - VIII show the mean values and variances of connection weights between four sub-structures within LGN (ON1, ON2, OFF1, and OFF2) and TRN obtained using external reinforcement signal for the three age groups with correct and reverse amplitude respectively. We did not observe changes in the inhibitory connections from MSTc area to both LIP left and right layers.

In case of reinforcement signal corresponding to perceptual decision both Tables I and II demonstrate tendency of decreased excitatiory connectivity and increased inhibitory connectivity with aging for both LIP areas. At the same time obtained excitatory connections are stronger and with bigger variance for the right LIP area (that corresponds to suppressed by reinforcement decision) while inhibitory connections achieved a little bit higher absolute values for the correct (left) LIP area.

In the case of reinforcement signal opposite to the perceptual decision Table III shows that excitatory connections to the suppressed by training left LIP area increase in comparison to previous case but remain the same for the three age groups while the absolute values of the inhibitory connections increased slightly with aging. Table IV demonstrates that when the reinforcement signal favors the right LIP area the excitatory connectivity increased with aging but has lower mean values and less variability than in the previous case. The inhibitory connections however remain the same for three age groups and achieved a little bit higher absolute values and variance.

These results might be explained with task-related connectivity between MST and LIP areas in our model. Since the excitatory connections are allowed only from MSTe templates to the corresponding to their focal points LIP areas and the rest of connections remain inhibitory, the reinforcement corresponding to perceptual decision increases inhibition from MSTe areas related to the wrong decision to both LIP areas while the opposite reinforcement tries to revert the strength of excitatory connections towards the one opposite to the perception decision.

Concerning the deep thalamic relay, in [14] we observed that only inhibitory feedback connections to LGN from TRN structure are subject to some changes so here we monitored only their changes. In contrast however to our previous results from STDP training of perceptual part of the model reported in [14], changes in case of reinforcement training reported here became visible after single presentation of teaching signal.

In case of reinforcement signal corresponding to perceptual decision (Tables V and VI) we observe slight decrease of inhibition with aging in connections only to first LGN layer having ON receptive fields (ON1) and slight increase for the rest of LGN layers (ON2, OFF1, and OFF2).

In case of reinforcement opposite to the perceptual decision (Tables VII and VIII) we observed a tendency towards a decrease of inhibitory connectivity with aging to all LGN layers. Achieved in this case connection weights however are a little bit smaller in comparison with previous case of reinforcement training.

TABLE VII
MEAN OF WEIGHTS OF CONNECTIONS FROM LGN TO TRN IN CASE OF REINFORCEMENT SIGNAL OPPOSITE TO PERCEPTUAL DECISION (LEFT).

Group	ON1	ON2	OFF1	OFF2
Young	-0.99001	-0.97830	-0.87068	-0.85187
Middle	-0.98981	-0.97705	-0.87082	-0.84828
Elderly	-0.98722	-0.97753	-0.86884	-0.84816

TABLE VIII
VARIANCE OF WEIGHTS OF CONNECTIONS FROM LGN TO TRN IN CASE OF REINFORCEMENT SIGNAL OPPOSITE TO PERCEPTUAL DECISION (LEFT).

Group	ON1	ON2	OFF1	OFF2
Young	0.00500	0.00440	0.00636	0.00835
Middle	0.00493	0.00458	0.00633	0.00766
Elderly	0.00522	0.00441	0.00663	0.00757

The aging however increased the variances of connections to ON1 and OFF1 and decreased those of connections to ON2 and OFF2 in both considered cases of reinforcement signal.

In summary, the ageing effects in thalamic relay demonstrated predominantly increased inhibition in case of reinforcement signal corresponding to perceptual decision and decreased inhibition for the opposite reinforcement training. This might be explained by the fact that the reinforcement signal suppressing the perceptual decision tries to invert the overall model perceptual attitude while the reinforcement signal corresponding to the perceptual decision leads to age-related differentiating in this structure positioned deep in the perceptual part of the model.

IV. Conclusions

The model, presented here incorporates all basic structures in the human brain responsible for decision making based on dynamic visual information in tasks with eye movement response starting from the visual information encoding, preprocessing, information extraction and accumulation and saccade generation biased by subcortical structures (BG) in the presence of external reinforcement.

The adjustment of the model parameters in the dynamic (dopamine and STDP) synapses by feeding reinforcement signal reflecting specific characteristics of the human performance provides further insight into the complicate interactions between different brain structures and their modification in the process of learning, acting and aging.

A future application of our model will be to investigate by simulations the behaviour of the brain structures involved in visual information processing and decision making in case of deterioration in any of its layers, i.e. to perform in-sillico modelling of brain lesions or other degenerative brain processes. Comparison of such simulated behaviour with patients' performance in visual tasks can support early and noninvasive diagnosis of some deceases of human brain.

References

[1] A. G. Barto, R. S. Sutton and C. W. Anderson, C.W., "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 13 (5), 1983, pp. 834-846. DOI: 10.1109/TSMC.1983.6313077

[2] M. N. Shadlen and W. T. Newsome, "Motion perception: seeing and deciding," *Proc. Natl. Acad. Sci. USA*, vol. 93 (2), pp. 628-633, 1996. DOI: 10.1073/pnas.93.2.628

[3] D. M. Herz, B. A. Zavala, R. Bogacz and P. Brown, "Neural correlates of decision thresholds in the human subthalamic nucleus," *Current Biology*, vol. 26 (7), pp. 916-920, 2016. DOI: 10.1016/j.cub.2016.01.051

[4] K. Dunovan, B. Lynch, T. Molesworth and T. Verstynen, T., "Competing basal-ganglia pathways determine the difference between stopping and deciding not to go," *eLife*, vol. 4, Article number e08723, 2015. DOI: 10.7554/eLife.08723

[5] A. G. Barto, "Adaptive critics and the basal ganglia," in J. C. Houk, J. L. Davis and D. G. Beiser, Editors, *Models of Information Processing in the Basal Ganglia*, MIT Press, Cambridge, MA; 1995, pp. 215-232.

[6] D. Joel, Y. Niv and E. Ruppin, "Actor-critic models of the basal ganglia: new anatomical and computational perspectives," *Neural Networks*, vol. 15, pp. 535-547, 2002. DOI: 10.1016/S0893-6080(02)00047-3

[7] M. J. Frank, L. C. Seeberger and R. C. O'Reilly, "By carrot or by stick: cognitive reinforcement learning in Parkinsonism," *Science*, vol. 306 (5703), pp. 1940-1943, 2004. DOI: 10.1126/science.1102941

[8] R. Bogacz and T. Larsen, T., "Integration of reinforcement learning and optimal decision-making theories of the basal ganglia,", *Neural Computation*, vol. 23 (4), pp. 817-851, 2011. DOI: 10.1162/NECO_a_00103

[9] K. Dunovan and T. Verstynen, "Believer-Skeptic meets actor-critic: Rethinking the role of basal ganglia pathways during decision-making and reinforcement learning,", *Frontiers in Neuroscience*, vol. 10, Article number 106, 2016. DOI: 10.3389/fnins.2016.00106

[10] P. Koprinkova-Hristova and N. Bocheva, "Spike timing neural model of eye movement motor response with reinforcement learning," Lecture Notes in Computer Science, in press.

[11] J. Igarashi, O. Shounob, T. Fukai and H. Tsujino, "Real-time simulation of a spiking neural network model of the basal ganglia circuitry using general purpose computing on graphics processing units," *Neural Networks*, vol. 24, pp. 950-960, 2011. DOI: 10.1016/j.neunet.2011.06.008

[12] R. Krishnan, S. Ratnadurai, D. Subramanian, V. S. Chakravarthy and M. Rengaswamyd, "Modeling the role of basal ganglia in saccade generation: Is the indirect pathway the explorer?," *Neural Networks*, vol. 24, pp. 801-813, 2011. DOI: 10.1016/j.neunet.2011.06.002

[13] S. Kunkel et al.,"NEST 2.12.0," *Zenodo*, 2017. DOI: 10.5281/zenodo.259534

[14] P. Koprinkova-Hristova, N. Bocheva, S. Nedelcheva, M. Stefanova, B. Genova, R. Kraleva and V. Kralev, "STDP plasticity in TRN within hierarchical spike timing model of visual information processing," *IFIP Advances in Information and Communication Technology*, vol. 583 IFIP, pp. 279-290, 2020. DOI: 10.1007/978-3-030-49161-1_24

[15] P. Koprinkova-Hristova, N. Bocheva, S. Nedelcheva and M. Stefanova, "Spike timing neural model of motion perception and decision making," *Frontiers in Computational Neuroscience*, vol. 13, Article number 20, 2019. DOI: 10.3389/fncom.2019.00020

[16] P. Koprinkova-Hristova, N. Bocheva and S. Nedelcheva, "Investigation of feedback connections effect of a spike timing neural network model of early visual system, " in *Innovations in Intelligent Systems and Applications (INISTA)*, Thessaloniki, Greece, 2018, DOI: 10.1109/INISTA.2018.8466292

[17] S. Nedelcheva and P. Koprinkova-Hristova, "Orientation selectivity tuning of a spike timing neural network model of the first layer of the human visual cortex," *Studies in Computational Intelligence*, vol. 793, pp. 291-303, 2019. DOI: 10.1007/978-3-319-97277-0_24

[18] T. W. Troyer, A. E. Krukowski, N. J. Priebe and K. D. Miller, "Contrast invariant orientation tuning in cat visual cortex: thalamocortical input tuning and correlation-based intracortical connectivity," *J. Neurosci.*, vol. 18, pp. 5908-5927, 1998. DOI: 10.1523/jneurosci.18-15-05908.1998

[19] J. Kremkow, L. U. Perrinet, C. Monier, J.-M. Alonso, A. Aertsen, Y. Fregnac and G. S. Masson, "Push-pull receptive field organization and synaptic depression: Mechanisms for reliably encoding naturalistic stimuli in V1," *Frontiers in Neural Circuits*, vol. 10, Article number 37, 2016. DOI: 10.3389/fncir.2016.00037

[20] A. Casti, F. Hayot, Y. Xiao and E. Kaplan, "A simple model of retina-LGN transmission," *J. Computational Neuroscience*, vol. 24, pp. 235-252, 2008. DOI: 10.1007/s10827-007-0053-7

[21] M. Ghodratia, S.-M. Khaligh-Razavic and S. R. Lehky, "Towards building a more complex view of the lateral geniculate nucleus: Recent advances in understanding its role," *Progress in Neurobiology*, vol. 156, pp. 214-255, 2017. DOI: 10.1016/j.pneurobio.2017.06.002

[22] P. Gleeson, R. Martinez and A. Davison, "Network models of V1," *Open Source Brain*, http://www.opensourcebrain.org/projects/111.

[23] S. Sadeh and S. Rotter, "Statistics and geometry of orientation selectivity in primary visual cortex," *Biol. Cybern.*, vol. 108, pp. 631-653, 2014. DOI: 10.1007/s00422-013-0576-0

[24] M.-J. Escobar, G. S. Masson, T. Vieville and P. Kornprobst, "Action recognition using a bio-inspired feedforward spiking network," *Int. J. Comput. Vis.*, vol. 82, pp. 284-301, 2009. DOI: 10.1007/s11263-008-0201-1

[25] O. W. Layton and B. R. Fajen, "Possible role for recurrent interactions between expansion and contraction cells in MSTd during self-motion perception in dynamic environments," *Journal of Vision*, vol. 17 (5), Article number 5, 2017. DOI: 10.1167/17.5.5

[26] W. Potjans, A. Morrison and M. Diesmann, "Enabling functional neural circuit simulations with distributed computing of neuromodulated plasticity," *Front. in Comp. Neuroscience*, vol. 4, 2010. DOI: 10.3389/fncom.2010.00141

[27] J. L. Plotkin and L. A. Goldberg, "Thinking outside the box (and arrow): Current themes in striatal dysfunction in movement disor-

ders," *The Neuroscientist*, vol. 25 (4), pp. 359-379, 2019. DOI: 10.1177/1073858418807887

[28] W. Wei, J. E. Rubin and X.-J. Wang, "Role of the indirect pathway of the basal ganglia in perceptual decision making," *The Journal of Neuroscience*, vol. 35 (9), pp. 4052-4064, 2015. DOI: 10.1523/JNEUROSCI.3611-14.2015

[29] H. Yan and J. Wang, "Quantification of motor network dynamics in Parkinson's disease by means of landscape and flux theory," *PLoS ONE*, vol. 12 (3), Article number e0174364, 2017. DOI: 10.1371/journal.pone.0174364

[30] M. Tsodyks, A. Uziel and H. Markram, "Synchrony generation in recurrent networks with frequency-dependent synapses," *The Journal of Neuroscience*, vol. 20 (1), pp. RC50, 2000. DOI: 10.1523/jneurosci.20-01-j0003.2000

[31] N. Bocheva, B. Genova and M. Stefanova, "Drift diffusion modeling of response time in heading estimation based on motion and form cues," *Int. J. of Biology and Biomedical Engineering*, vol. 12, pp. 75-83, 2018.

DOI: 10.15439/2020F137
ISSN 2300-5963 ACSIS, Vol. 21

Shallow, Deep, Ensemble models for Network Device Workload Forecasting

Cenru Liu
Ngee Ann Polytechnic, Singapore
liucenru@gmail.com

***Abstract*—Reliable prediction of workload-related characteristics of monitored devices is important and helpful for management of infrastructure capacity. This paper presents 3 machine learning models (shallow, deep, ensemble) with different complexity for network device workload forecasting. The performance of these models have been compared using the data provided in FedCSIS'20 Challenge. The R^2 scores achieved from the cascade Support Vector Regression (SVR) based shallow model, Long short-term memory (LSTM) based deep model, and hierarchical linear weighted ensemble model are 0.2506, 0.2831, and 0.3059, respectively, and was ranked 3^{rd} place in the preliminary stage of the challenges.**

***Index Terms*—Workload forecasting, Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Support Vector Regression (SVR), Hierarchical Linear Weighted Ensemble**

I. Introduction

EMCA Software is a Polish vendor of Energy Log server, which is capable of collecting data from various log sources and providing in-depth data analysis to its end-users. The objective of the FedCSIS'20 challenge is to explore reliable machine learning models to predict workload-related characteristics of monitored devices, based on historical data gathered from such devices, which is important for IT and technical teams to manage the capacity of their infrastructure [1].

Workload forecasting models have been developed based on machine learning methods in the literature. Future host load was predicted using 9 features extracted from historical workload values by using the Bayesian model in [2]. A forecasting model by combining neuro-fuzzy and Bayesian inference was developed for CPU workload forecasting in [3]. In [4], a workload forecasting model has been developed based on Artificial Neural Network (NN) and adaptive Differential Evolution (DE). Workload was predicted by using Autoregressive Integrated Moving Average (ARIMA) model in [5]. To consider temporal dependencies in workload sequence data, recently, Recurrent neural network (RNN) and its variant, Long short-term memory (LSTM), have been employed in workload forecasting and shown promising performance [8], [6], [7].

To explore the performance of shallow, deep and ensemble models and cater for FedCSIS'20 Challenge, we developed 3 network device workload forecasting models:

1) a cascade shallow model based on Support Vector Regression (SVR);
2) a deep learning model based on LSTM;
3) a hierarchical linear weighted ensemble model.

The performance of these 3 models were compared using the network device workload data provided in FedCSIS'20 Challenge. The hierarchical ensemble of LSTM achieved the highest R^2 score in the preliminary stage, while the cascade SVR model was more robust to overfitting.

This paper is organized as follows. The FedCSIS'20 challenge is briefly introduced in Section II. The cascade shallow model is presented in Section III, the LSTM based deep model is given in Section IV, and the hierarchical linear ensemble model is described in Section V. Section VI compares the performance of the 3 models. Conclusions are given in Section VII.

II. FedCSIS 2020 Challenge: Network Device Workload Prediction

In this section, we briefly introduced the FedCSIS'20 Challenge titled as Network Device Workload Prediction [1]. The task in this challenge is to predict future workload characteristics of a number of monitored devices based on the given historical data collected from these devices.

A. Data

The dataset provided in this challenge is in the format of a .csv file, which holds a table of over forty-four million rows and ten columns. The 10 columns include identifiers followed by the mean, standard deviation, and a candlestick aggregation of the corresponding values, as listed below:

- hostname: an ID of the device;
- series: a name of the considered characteristic;
- time window: a timestamp of the aggregation window;
- Mean: the mean of the values;
- SD: the standard deviation of the values;
- Open: a value of the first reading during the corresponding hour;
- High: the maximum of values;
- Low: the minimum of values;
- Close: a value of the last reading during the corresponding hour;
- Volume: the number of values.

B. Task

The data for each hostname-series pair can be arranged into 7 time series spanning over 80 days, which are values of mean, SD, open, high, low, close, and volume. The participants of the challenge were required to forecast the mean of the workload values in each of the next 168 hours after the end of the training data for ten thousand hostname-series pairs selected from these over twenty-four thousands pairs.

C. Evaluatoin

The solutions were assessed by the R^2 measure. The forecasts of each time series are compared to ground truth values and assessed using the R^2 score that is defined as:

$$R^2(f, y) = 1 - \frac{RSS(f, y)}{TSS(y)}. \tag{1}$$

RSS is the residual sum of squares of forecasts and TSS is the total sum of squares, given as

$$\begin{aligned} RSS &= \sum_i (y_i - f_i)^2, \\ TSS &= \sum_i (y_i - \frac{1}{N}\sum_i y_i)^2, \end{aligned} \tag{2}$$

where y_i and f_i are the ground truth and their prediction, respectively, and N is length of the time series. The score of a submitted solution is the average R^2 value over all time series from the test set.

The preliminary scores of the submitted solutions were evaluated externally and published on the challenge leaderboard computed on a small subset (10%) of the test time series that are fixed for all participants. The final evaluation will be published after completion of the competition using all of the test time series.

III. Shallow Model

Support Vector machine (SVM) was proposed by Vladimir Vapnik and his co-workers based on the statistical learning theory (or VC theory) [9], [10], [11], [12], [13], [14], [15], [16], [17]. The SVM has shown competitive generalization ability over many existing machine learning models in a number of fields, e.g. optical character recognition (OCR), object recognition, time series prediction, etc. [13], [18], [19], [20], [21]. The Support Vector Regression (SVR) is a powerful regression approach and successfully applied in numerous applications [22], [23], [24], [25]. In this work, a cascade shallow model has been developed based on the SVR with the Radial basis function kernel (RBF) for workload prediction.

Although 7 types of hourly aggregated workload values were provided in the challenge, only the hourly mean of the data was used in our method. The data were organized in a matrix, in which each row represents the time series of a hostname-series pair and each column stores the mean of workloads in one hour. The data were standardized to have zero mean and unit standard deviation, which is essential to non-linear machine learning models.

One difficulty in this challenge is arising from the fact that the devices considered in the data were not uniform and some of the devices were a part of the same system and it is likely that their workloads were highly correlated and cross-dependent [1]. To increase the diversity of training, the cascade SVR-based models are trained on the following two parts of the data provided:

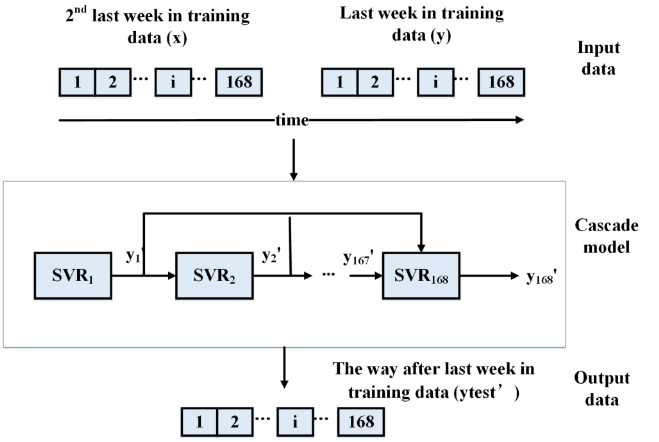

2nd last week in training data (x)
Last week in training data (y)
Input data
1
2
i
168
1
2
i
168
time
Cascade model
SVR1
y1'
SVR2
y2'
y167'
SVR168
y168'
The way after last week in training data (ytest')
Output data
1
2
i
168

Fig. 1. Structure of the cascade SVR model and composition of its input and output.

- training set 1: the 10k time series involved in testing;
- training set 2: the time series selected from the other 14k sequences based on the following rules: having less missing values and closer to the 10k testing sequences.

Instead of using all data in over 80 days, only the values in the last 2 weeks, i.e. 336 average hourly values, were used in training.

The cascade shallow model is composed of a series of connected single SVR models denoted as hourly models, each of which was trained to predict the mean of workloads in one hour. Let SVR_t represent an hourly model predicting the mean workloads in the t^{th} hour, where $t \in [1, 168]$. The input features to SVR_t can be the values in all of the hours before t. Assuming weekly periodic property of workloads, we took only the values from the previous 168 hours. Therefore, the feature length of an hourly model is 168.

In the cascade shallow model, there are a series connected hourly models that are trained one by one. The outputs from the previous hourly models will be used as partial inputs to all subsequent models. Fig. 1 illustrates the structure of the cascade SVR model and the composition of its input and output. In this way, the latter model can be adapted based on the predictions from its previous models, which conforms to the cognition that the previous values are correlated to the latter ones.

The hyper-parameters of the non-linear SVR models with RBF kernel were set as follows.

ϵ in the ϵ-insensitive loss function was set to be an estimate of a tenth of the standard deviation using the inter-quartile range of the response variable y, expressed as:

$$\epsilon = iqr(y)/13.49, \tag{3}$$

where $iqr(y)$ is the inter-quartile range of y.

The parameter C controls the trade off between training error and model complexity, which was set to be an estimate of the standard deviation of the response variable, expressed as:

$$C = iqr(y)/1.349. \tag{4}$$

γ is a free parameter used in the radial kernel. The radial basis function kernel, or RBF kernel on two samples x_i and x_j is defined as

$$K(x_i, x_j) = exp(-\gamma||x_i - x_j||^2). \tag{5}$$

The value of γ was optimized by the heuristic procedure using sub-sampling [26].

IV. Deep Learning Model

Considering temporal dependencies in workload sequence data, a workload forecasting model based on LSTM, a special kind of RNN, has been developed. The RNNs, derived from feedforward neural network, use memory to process sequence signals, which exhibit temporal dynamic behavior by connecting nodes to form a directed graph along a temporal sequence [27], [28], [29]. The LSTM, proposed by Hochreiter and Schmidhuber in 1997, unlike standard feedforward neural networks, has feedback connections, which allows the LSTM to process not only single data point, e.g. image, but also an entire sequence of data, e.g. speech or video [28], [30].

Similarly to the cascade SVR model, the average hourly workload values were used to train the LSTM model. The data given in the challenge were separated into two parts, one was used for training the LSTM networks, and the other for the purpose of validation in order to prevent overfitting:

- training set: the 10k time series involved in testing;
- validation set: the time series selected from the other 14k sequences based on the following rules: having less missing values and closer to the 10k testing sequences.

The data were standardized to have zero mean and unit standard deviation.

Due to the limited computation resource available for training sophisticated deep networks with multiple layers, just the data in the last 4-8 weeks were used in training and validation. The length of the input sequence was dependant on the size of the network, which was fixed in one LSTM network. The LSTM model was trained with sequence-input-sequence-output mode. The LSTM models have multiple LSTM-layers ranging from 2-5. Each layer has different number of hidden neurons, ranging from 128-640.

V. Hierarchical Linear Weighted Ensemble

In machine learning, ensemble of multiple independently trained models is expected to perform better than any base model by combining the advantage of base models and diluting their self-errors. In our ensemble model, the base models were linearly combined with different weights to yield final output, where the weights were estimated by linear regression. Note that only the deep models were used as base models since they gave highest preliminary scores. The dataset employed to train the linear regression models is the same as that used to train the cascade SVR model. A set of weights were trained for each of the 168 hours on the predictions from the base models, based on which the final output was generated.

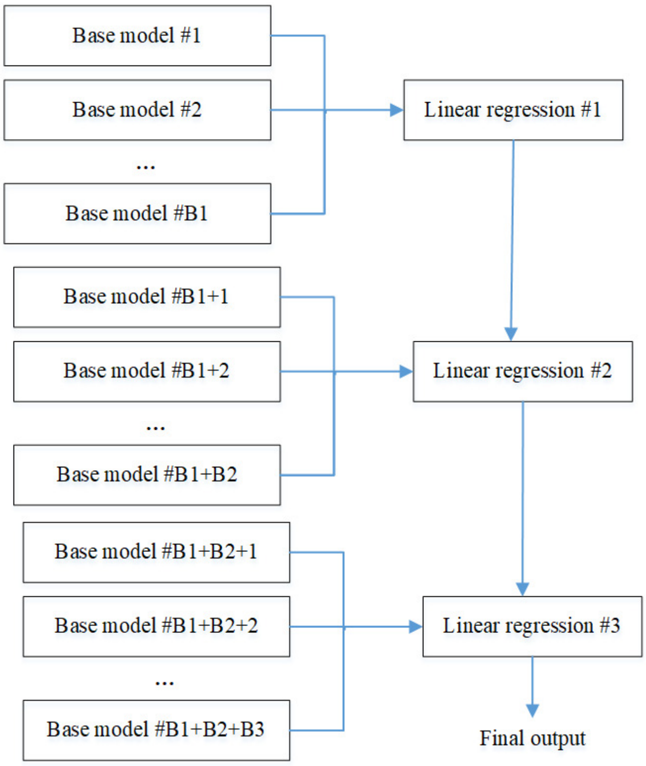


Fig. 2. Flowchart of hierarchical linear weighted ensemble.

When we observed the public scores of the solutions from the individual base models and the weights generated from the linear regression, it was found that some solutions with high scores got very low weights indicating that the importance of these solutions has been weakened, partially due to the variation of the given and unknown data. To address this issue, a hierarchical linear regression that combined various individual models in different stages has been developed. An example structure is shown in Fig. 2, where there are 3 linear regression stages, having B1, B2, and B3 base models, respectively. The B1 base models are firstly linearly combined, the output of which is then combined with the additional B2 base models, and similarly, the output from the second stage is then combined with the other B3 base models to yield the final output. The base models are arranged in ascending order of their public scores, e.g. the score of the model indexed B2+1 is higher than those of the models indexed from B1+1 to B2, by which the models with higher scores are combined in later stages so that the high-scored models are likely to have more priority in combination.

VI. Experiment Results

In this section, the performance of these 3 models is compared by using the data provided in the challenge.

A. Results of SVR models

When the cascade shallow model was trained on partial of the given data, e.g. the training set 2, the preliminary R^2 score was 0.2153. This was increased to 0.2506 if both training sets were used.

B. Results from LSTM networks

We have trained various LSTM models using different network structures. The performance was rather different. The highest preliminary R^2 score was 0.2831, which was achieved from a LSTM network having 3 LSTM layers each with 336 hidden neurons.

C. Results for Linear weighted ensemble

The preliminary R^2 score from the hierarchical linear weighted ensemble model was 0.2990 when trained on partial data, i.e. training set 1, while it was increased to 0.3059 when the LSTM network was trained on both training sets.

D. Discussion

Although the preliminary R^2 score, which was assessed based on 10% of the testing data, from the cascade SVR model is lower than those from both single and ensemble of LSTM models, its final score evaluated on the full testing set is 0.2365 that is higher than the baseline and published top score being 0.2295 and 0.1630, respectively. This indicates that the cascade shallow model is robust to overfitting. Both single and ensemble of LSTM can achieve higher preliminary R^2 score while they are likely to fall into overfitting. This implies that suitable implementation of shallow models can outperform deep models.

VII. Conclusions

This paper addresses forecasting workloads of network devices from historical sequence data. Three machine learning models, which are cascade SVR-based shallow model, LSTM-based deep model, and hierarchical linear weighted ensemble model, have been developed and verified using the data provided in the FedCSIS'20 Challenge. The preliminary evaluation on R^2 scores achieved from the shallow, deep and ensemble models are 0.2506, 0.2831, and 0.3059, respectively. Both the single and ensemble of LSTM models achieved much higher preliminary scores than the SVR model, while the SVR is more robust to overfitting.

References

[1] FedCSIS 2020 Challenge: Network Device Workload Prediction, *https://knowledgepit.ml/fedcsis20-challenge/*.

[2] S. Di, D. Kondo, W. Cirne, "Host load prediction in a Google compute cloud with a Bayesian model," *Proc. of IEEE Int. Conf. on High Performance Computing, Networking, Storage and Analysis*, 2012.

[3] F. Benhammadi, Z. Gessoum, A. Mokhtari, "CPU load prediction using neuro-fuzzy and Bayesian inferences," *Neurocomputing*, vol. 74, pp. 1606–1616, 2011.

[4] J. Kumar, A. Singh, "Workload prediction in cloud using artificial neural network and adaptive differential evolution," *Futur. Gener. Comput. Syst.*, vol. 81, pp. 41–52, 2018.

[5] R. Calheiros, E. Masoumi, R. Ranjan, R. Buyya, "Workload prediction using ARIMA model and its impact on cloud applications' QoS," *IEEE Trans. Cloud Comput.*, vol. 3, no. 4, pp. 449–458, 2014.

[6] Z. Huang, J. Peng, H. Lian, J. Guo, and W. Qiu, "Deep recurrent model for server load and performance prediction in data center," *Complexity*, 2017.

[7] J. Kumar, R. Goomer, and A. Singh, "Long short term memory recurrent neural network (LSTM-RNN) based workload forecasting model for cloud datacenters,". *Procedia Comput.(Elsevier)*, vol. 125, pp. 676–682, 2018.

[8] B. Song, Y. Yu, Y. Zhou, Z. Wang, and S. Du, "Host load prediction with long short-term memory in cloud computing," *The Journal of Supercomputing*, vol. 74, 6554–6568, 2018.

[9] B.E. Boser, I.M. Guyon, V. Vapnik, "A training algorithm for optimal margin classifiers," *Proceedings of the Annual Conference on Computational Learning Theory, ACM*, pp. 144–152, Pittsburgh, PA 1992.

[10] I. Guyon, B. Boser, and V. Vapnik, "Automatic capacity tuning of very large VC-dimension classifiers," *Advances in Neural Information Processing Systems 5*, pp. 147–155, Morgan Kaufmann Publishers, 1993.

[11] C. Cortes, and V. Vapnik, Support vector networks, Machine Learning, vol. 20, pp. 273–297, 1995.

[12] B. Schölkopf, C. Burges, and V. Vapnik, "Extracting support data for a given task," *Proceedings of First International Conference on Knowledge Discovery and Data Mining*, AAAI Press, 1995.

[13] B. Schölkopf, C. Burges, and V. Vapnik, "Incorporating invariances in support vector learning machines," *Artificial Neural Networks, Springer Lecture Notes in Computer Science*, Vol. 1112, pp. 47–52, Berlin, 1996.

[14] V. Vapnik, S. Golowich and A. Smola, "Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing," in M. Mozer, M. Jordan, and T. Petsche (eds.), Neural Information Processing Systems, vol. 9, MIT Press, Cambridge, MA., 1997.

[15] V. Vapnik and A. Chervonenkis, "Theory of Pattern Recognition" (in Russian), Nauka, 1974.

[16] V. Vapnik, "Estimation of dependences based on empirical data," Springer Verlag.

[17] V. Vapnik, "The Nature of Statistical Learning Theory," Springer, New York.

[18] B. Schölkopf, P. Simard, A. Smola, and V. Vapnik, "Prior knowledge in support vector kernels," *M.I. Jordan, M.J. Kearns, and S.A. Solla (Eds.), Advances in Neural Information Processing Systems 10*, MIT Press, Cambridge, MA, pp. 640–646, 1998.

[19] V. Blanz, B. Schölkopf, H. Bulthoff, C. Burges, V. Vapnik, and T. Vetter, "Comparison of view-based object recognition algorithms using realistic 3D models," *Artificial Neural Networks, Springer Lecture Notes in Computer Science*, vol. 1112, pp. 251–256, Berlin, 1996.

[20] B. Schölkopf, K. Sung, C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik, "Comparing support vector machines with Gaussian kernels to radial basis function classifiers," *IEEE Transactions on Signal Processing*, vol. 45, pp. 2758–2765, 1997.

[21] K.R. Muller, A. Smola, G. Ratsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik, "Predicting time series with support vector machines," *Artificial Neural Networks, Springer Lecture Notes in Computer Science*, vol. 1327, pp. 999–1004, Berlin, 1997.

[22] H. Drucker, C.J.C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Advances in Neural Information Processing Systems 9*, pp. 155–161, MIT Press, Cambridge, MA, 1997.

[23] M. Stitson, A. Gammerman, V. Vapnik, V. Vovk, C. Watkins, and J. Weston, "Support vector regression with ANOVA decomposition kernels," *Advances in Kernel Methods—Support Vector Learning*, MIT Press Cambridge, MA, pp. 285–292, 1999.

[24] A. Smola, and B. Schölkopf, "A Tutorial on Support Vector Regression," *STATISTICS AND COMPUTING*, vol. 14, pp. 199-222, 2003.

[25] D. Basak, S. Pal, and D. Patranabis, "Support Vector Regression," *Neural Information Processing – Letters and Reviews*, vol. 11, Non. 10, pp. 203-224, October 2007.

[26] fitrsvm: Fit a support vector machine regression mode, https://www.mathworks.com/help/stats/fitrsvm.html.

[27] S. Dupond. "A thorough review on the current advance of neural network structures,". *Annual Reviews in Control*, vol. 14, pp. 200–230, 2019.

[28] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[29] M. Miljanovic, "Comparative analysis of Recurrent and Finite Impulse Response Neural Networks in Time Series Prediction," *Indian Journal of Computer and Engineering*, vol. 3, no. 1, 2012.

[30] S. Hochreiter, J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

DOI: 10.15439/2020F189
ISSN 2300-5963 ACSIS, Vol. 21

Integrated Human Tracking Based on Video and Smartphone Signal Processing within the Arahub System

Jan Ludziejewski, Łukasz Grad
Uniwersytet Warszawski
Email: {jan.ludziejewski, lukasz.grad}@mimuw.edu.pl

Łukasz Przebinda
Arahub & Myled
Email: l.przebinda@myled.pl

Tomasz Tajmajer
QED Software
Email: tomasz.tajmajer@qed.pl

***Abstract*—Embedded platforms with GPU acceleration, designed for performing machine learning on the edge, enabled the creation of inexpensive and pervasive computer vision systems. Smartphones are nowadays widely used for profiling and tracking in marketing, based on WiFi data or beacon-based positioning systems. We present the Arahub system, which aims at integrating world of computer vision systems with smartphone tracking for delivering data useful in interactive applications, such as interactive advertisements. In this paper we present the architecture of the Arahub system and provide insight about its particular elements. Our preliminary results, obtained from real-life test environments and scenarios, show that the Arahub system is able to accurately assign smartphones to their bearers, based on visual and WiFi/Bluetooth positioning data. We show the commercial value of such system and its potential applications.**

I. Introduction

WHILE video monitoring systems are currently found everywhere, still, most of them are used for security applications. Systems installed in commercial zones, stores or cafes, could deliver valuable information to owners of such places, yet automatic analysis of such data requires advanced computer vision systems. Embedded platforms with GPUs for providing machine learning to the edge, enabled the creation of inexpensive and pervasive devices, that may process high-level data extracted from video streams.

As virtually every person is equipped with a smartphone these days, many companies are offering analytic services based on location tracking and mobile applications. Location-based marketing, geofencing or predictive analysis are all more widely used for companies to deliver personalized, targeted marketing. Yet this source of data has its limitations - it is difficult to deliver real-time information about a person which is at a particular place - and this is crucial if one wants to provide personalization and interaction, e.g. a dedicated advertisement displayed to a specific person.

In this work we present the Arahub project. It is focused on combining the world of computer vision systems with smartphone tracking for delivering data useful in interactive applications, where both location and profile of a person are required. The primary use-case for Arahub is digital marketing system that could be used for marketing campaigns delivered to specific persons at specific places.

In this paper the overall architecture of the Arahub system is described. We provide insights into particular elements of the system and methods used. We also present preliminary results, which we were able to obtain in real-life environments.

A. Principle of operation

The primary goal of Arahub is to provide statistical data about people present near an area of interest. Examples of such data are: the number of people watching a commercial on a display withing a specified time period, the gender of a person currently watching a shop exposition, shopping preferences of a person moving towards a display, etc. Such statistics may be based on data gathered from several sources: vision systems [1], [2], [3], indoor-positioning [4], [5] or mobile apps. The most interesting (and challenging) is the possibility of integrating data from multiple sources [6] to gather even more commercially valuable insight.

Let us consider the following scenario: a person with a smartphone has a loyalty application installed and running. This person is shopping in a store that is supported by the loyalty application. The owner of the store may have access to data provided by the application, such as the purchase history of the given customer. The owner, however, cannot directly match that data with a particular person currently visiting the store, as localization data may be too coarse. Yet, the owner of a store has access to a visual monitoring system, which could be used for precise visual tracking of all customers. Those two data sources, when properly linked together, could provide rich data attributed to a particular person currently visiting the store. Such a link could be established by combining the position of a person based on visual cues with the position of the mobile device owned by that person.

B. Motivation

Digital Out Of Home (DOOH) is a segment of marketing that is based on digital forms of advertising placed outdoors or in indoor public locations (out-of-home). The set of media types, including displays, LED screens and similar, used in DOOH, are referred to as Digital Signage (DS).

As DOOH and DS systems are becoming more common, there is a need for novel methods of targeting, interaction and content design, that could use the potential of this new type of

Supported by Innoventure and NCBiR (POIR.01.03.01-00-0022/16)

advertising. A particularly interesting ideas may be borrowed from the world of online advertising, which after decades of existence has become a mainstream advertising channel.

Existing DOOH systems are passive in terms of targeting - marketing content is selected based on long-term demography statistics or, in the best case, on custom surveys made for a particular location. It is obvious that such methods of audience analysis could not be compared to precise on-line targeting based on browser cookies or shopping history. Yet there is a high potential for using external data sources in DOOH. Such cases, using traffic or weather data, are already existing.

The biggest potential is in so-called "programmatic DOOH", which envisions a novel method of selling DOOH media - not by air time or by surface area, but by the number of views, or even views of the specified audience with particular interests or shopping history. To enable such operation, one needs to provide real-time data about the audience or particular viewers. Arahub is meant to provide such functionality and connect the advertising from online world with digital media existing in the real world.

Even though real-time, personalized DOOH is the main motivation behind the development of Arahub, there are many other, useful applications of such a system. The integration of multi-modal data sources for more accurate positioning and profiling may be used in smart-city and smart-home [7] environments, especially in healthcare or public services [8]. Also, security systems could benefit from more accurate analysis methods; facial recognition methods - despite rising privacy concerns - may also provide valuable insight if used with respect to legal regulations [9]. Finally, a system such as Arahub is a source of meta-data that could be used to learn about general behaviors and trends in the society, which can be used for making predictive models or inferring rules [10].

II. Related work

Positioning Systems based on WiFi and Bluetooth signals have been an active area of research over the last years. The two most common approaches to device localization based on a system of multiple WiFi access points or Bluetooth beacons are triangulation and fingerprinting.

Triangulation methods can be further divided into lateration and angulation [11]. These methods use the estimated distance from several transmitters or receivers based on signal attenuation [12], time characteristics of the propagated signal, e.g. Time of Arrival [13], Time Difference of Arrival [14] or are based on the direction of the received signal - Angle of Arrival [15]. Triangulation methods achieve good results in open space environments. However, they perform significantly worse in the indoor conditions where the signals may be reflected by several obstacles and there is no clear line-of-sight between the transmitting and receiving devices.

Fingerprinting methods work in two phases. In the first learning phase, a database of the signal characteristics at known locations is built [16], usually based on the Received Signal Strength Indicator (RSSI). This reference data set is then used in the second stage to perform localization, by comparing the measured signal characteristics with the fingerprints stored in the database. Several methods that improve on the standard fingerprint-based methods have been developed, e.g. statistical post-processing methods to estimate a continuous distribution of RSSI values based on Gaussian Process Theory [17] [18] or parametric estimation of the RSSI distribution [19]. Moreover, [20] presents a comparison between WiFi and Bluetooth localization system based on the fingerprinting approach and shows the advantages of BLE-based localization

In our work, we present a uniform approach for WiFi and Bluetooth signal modeling and develop two methods for estimating RSSI distribution along with a probabilistic Indoor Positioning System. The first approach is based on an extension of the Log-distance path loss model [21], the second method is based on a probabilistic fingerprinting-based model.

The two most common approaches for human tracking using video stream data are neural network based with subsequent box matching and motion detection. Motion detection can be further divided into Background Subtraction, Frame Differencing, Optical Flow and Temporal Differencing [22]. We utilize both approaches, in the second case merging Background Subtraction and Frame Differencing with a custom clustering method. However, multi-camera human tracking generally focuses on Probabilistic Occupancy Maps [23], developing a number of color-based or location-based techniques [24], while we propose a graph-based approach focused on location path similarity without dividing location space into clusters.

III. Arahub system overview

The architecture of the Arahub system consists of: a) distributed sensor network, which includes all equipment installed on-site; b) centralized data aggregation part, which includes multiple services running in the cloud environment. The overview of the architecture is presented in figure 1.

The distributed part of Arahub is based on a custom hardware solution - the Arabox, which integrates a vision system, WiFi monitoring hardware and GPU-enabled computing. In a typical scenario, several Araboxes are installed in one location for precise monitoring of a given point of interest. Moreover, Bluetooth Low Energy (BLE) beacons are also used to enhance the precision of indoor positioning. Araboxes provide high-level data about persons visible by the camera, such as their position on a 2D plane, they also provide the RSSI for WiFi clients connected to a specified WiFi Access Point (WiFi AP).

The data aggregation part has several functions. First of all, it provides interfaces for collecting the data from Araboxes and mobile applications, secondly it runs dedicated algorithms for filtering and combining multi-modal data, and finally, it provides services for accessing and interacting with the data. Arahub system also includes web services for management, visualization and diagnostics.

Another important elements of Arahub are the mobile devices carried by people in monitored locations. Arahub provides two methods for smartphone positioning: a) active - when the smartphone has a dedicated application running,

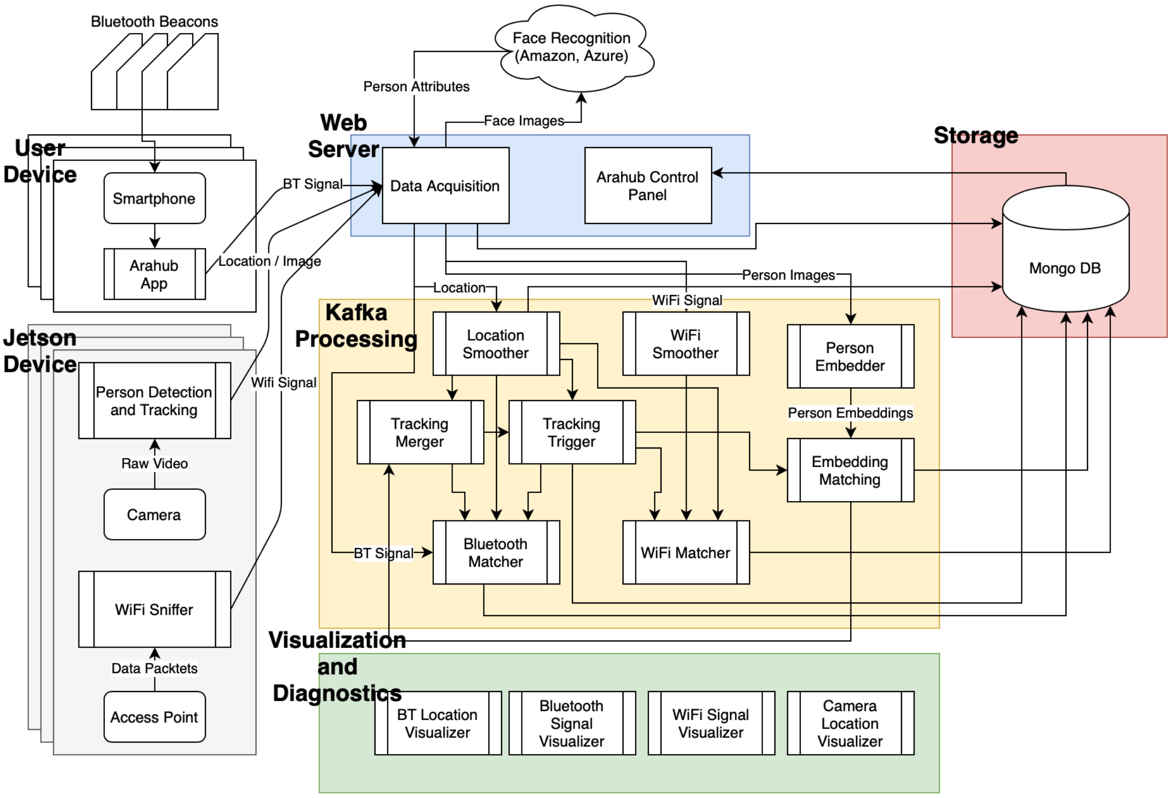


Fig. 1. Arahub architecture overview. A distributed sensor network is based on the Arabox devices installed on-site as well as mobile devices running dedicated Arahub application. Data from the sensor network is sent to the webserver and processed using a data acquisition module. We use Amazon and Microsoft Azure face recognition systems to enrich video data with personal attributes such as age and gender. Then, raw signal and location data are processed within a Data Aggregation system based on Kafka processing engine. We utilize Kafka connectors to save data in a Mongo database for the purpose of business analysis and model training. Arahub system also provides a number of visualization and diagnostics tools that enable monitoring of raw radio signals and locations received as well as tracking and signal-based indoor positioning systems.

b) passive - when the smartphone is connected to a dedicated WiFi network. The details of the operation of those methods are covered in section IV-E

A. Arabox - embedded platform for video and WiFi analysis

Arabox is a dedicated platform for gathering video streams and WiFi analysis. The goal of Arabox design was to create a compact, standalone device, that could locally perform computer vision tasks such as object detection. The device is meant to be installed in commercial zones, with no requirements as to existing infrastructure other than internet connectivity.

At the design stage, two main use-cases of Arabox were taken into consideration: 1) to be installed next to digital displays, where it could provide contextual information about the audience, 2) to be installed in passages such as corridors or stairways in commercial zones, where it would provide information about people visiting certain points of interests. For this reason, two versions of Arabox were developed: a large version (presented in figure 2), with two wide angle cameras integrated into a single enclosure, and a smaller version, with a single camera detached from the main enclosure.

In terms of the hardware platform, both versions of Arabox consist of the same elements. The core is an nvidia's Jetson Nano platform, with 4GBs or RAM and an integrated GPU, capable of CUDA operations. The video stream is provided by an RGB camera with dedicated optics, capable of recording full HD video at 30fps with low noise and in low light conditions. The third part is the WiFi adapter with an antenna dedicated for WiFi monitoring in 2,4GHz and 5GHz bands. Each Arabox also has a proper power adapter and ventilation system included. The enclosure of Arabox in the large version fits all elements inside and is waterproof, thus is suitable for outdoor installation. In this version, two cameras are placed such that their combined field of view angle is not less than 120 degrees. The cameras can be configured for different view angles if needed. The small version is dedicated for indoor installation - a single camera and WiFi adapter with an antenna are enclosed together separately from the Jetson Nano board. Both versions of Arabox have a dedicated mounting system, that allows for mounting to a ceiling or a wall.

The Arabox's embedded system - the Jetson Nano - is running a Linux system with custom software. The software

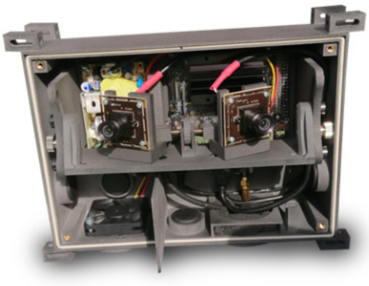


Fig. 2. Arabox prototype - the large version. A custom casing includes all elements: two cameras, Jetson Nano board, WiFi adapter, power supply and cables.

consists of three parts: video processing, WiFi processing and management.

Video processing is done in several steps: first, the raw data from the camera is normalized and throttled, to obtain a stable stream of video images. The stream may be then processed by several algorithms for object detection, such as GPU-based convolutional neural networks (described in more detail in section IV-B). The outputs of those algorithms are bounding boxes, based on which physical 2D positions of objects are calculated. Finally, the calculated positions are sent to the data aggregation system. Depending on the configuration, cropped images of detected objects may be also sent to the data aggregation system for further analysis, e.g. gender detection.

WiFi processing is based on monitor capabilities of an IEEE 802.11ac interface. The WiFi interface is configured to monitor data on channels used by a dedicated Access Point. The software reads control packets sent between that AP and all connected clients in range. It provides the RSSI (Received Signal Strength Indication) of the signal sent by clients, measured in the point where particular Arabox is installed. This data, containing the client's identifier, timestamp and RSSI is then forwarded to the data aggregation system.

A management system is used to provide software updates, configuration changes and to monitor the state of an Arabox. It is based on third-party software, that provides a centralized system for remote management of multiple devices with various internet connectivity (e.g. using third-party, NAT or cellular connections).

Arabox works in a semi-autonomic way - most data processing is done locally, so only high-level data is sent to the data aggregation system. Arabox needs to have constant internet connectivity, however as the data footprint is low, even cellular connections could be used for that purpose.

B. Mobile application

Arahub system uses a custom application developed for Android and iOS systems. The primary goal of this application is to enable indoor positioning based on BLE beacons. The application operates as follows: first, the application listens for familiar beacons IDs in slow scan mode; when it finds a beacon that operates in a zone observed by Arahub, the scanning mode is changed to fast. Now, the beacons are scanned with a 1 second period. The RSSI values from all beacons, that are registered to a particular zone, are read and immediately send to the data aggregation system. When a particular beacon from the list is not in range, then such information is also noted. After a long period without any signal form a known beacon, the application switches back to slow scan mode. An alternative version of the application is used in one of the test environments, where the user may also interact with the application to provide his preference related to a product being presented on a display connected to the Arahub system.

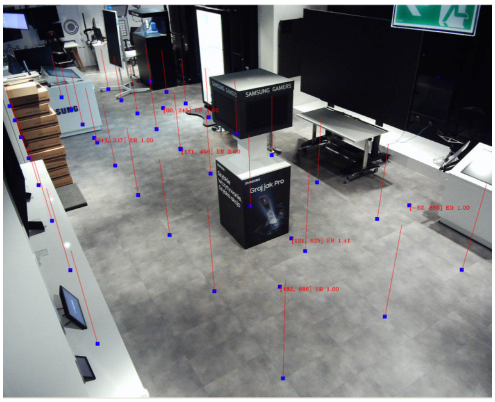


Fig. 3. A view from camera with calibration data shown. A uniform grid of points transformed using the calibration matrix is used to enable human validation of the process.

C. Calibration

In order to obtain physical positions of objects, a calibration procedure is required upon Arabox installation. The calibration is required for the purpose of both the visual and Bluetooth/Wifi positioning systems.

Visual system calibration is done independently for each camera in a particular location. For that, a dedicated chessboard pattern is used with the addition of several markers. The procedure requires placing the pattern and markers in the field of view of the camera - covering possibly the largest surface. Then the coordinates of markers and chessboard are provided to a particular Arabox configuration using a dedicated calibration tool, obtaining world-to-image-plane point correspondences. Using the point correspondences, a projection transformation from 3D world coordinates to the image plane can be calculated. In our work we assume the pinhole camera model. Thus, in order to perform camera calibration, we estimate both intrinsic and extrinsic parameter matrices along with radial and tangential distortion coefficients. We use the calibration method proposed in [25] implemented in the OpenCV [26] library. An example calibration result is presented in figure 3.

The camera calibration procedure is followed by an offline stage of creating a training data set for the purpose of Beacon/Wifi positioning systems. For this, the operator of the

Arahub system needs to use the mobile application to gather data about RSSI levels from BLE beacons in relation to his position predicted by the visual tracking system. Simultaneously, the WiFi signal strength is also recorded using Arabox WiFi monitors. To achieve the best results, the whole observed area should be covered multiple times.

Due to the possibility of errors or security concerns, some areas visible by the video tracking system needs to be excluded (e.g. areas "behind" mirrors). This is done as the last part of the calibration process.

IV. Data sources and processing

A. Location and height calculation

To improve the accuracy of location and height estimation we calculate, using the camera projection matrix, a line orthogonal to floor surface such that on the image, within some margin, it fits in the detected bounding box. To be considered a good prediction, this person candidate's height has to fit in a possible range. Moreover, the location has to be in an acceptable area defined by the union of convex polygons in spot configuration.

In real-world scenarios, especially in commercial zones, we find a number of objects partially covering customers (occlusion) - and it may not be possible or cost-effective to cover some areas with cameras without dealing with such obstacles. The most common scenario is people partially hidden by store shelves, desks or tables, with the upper body detected by the network and legs invisible, which significantly affects location predictions, especially when the camera angle is highly acute. However, if someone goes behind such an obstacle which cuts off the lower part of the box we are able to detect it because Intersection over Union (IoU) of successive boxes should be within the acceptable threshold, but location difference drastically increases and following three 2-dimensional points should approximately form a straight line: camera location (without height), expected location in current time and new location extracted from the cut-off box. Afterward, if we assume that the head is visible within the box and we know the height of this person, we can draw a line in 3-dimensional location coordinate space, such that it satisfies the following four assumptions forming a linear equation system: its length is equal to the height, projection of its start on camera image is equal to head location within the box, it is orthogonal to the ground and ends there.

B. Human tracking based on video data

Within one Jetson device, there are four stages of processing, each performed using separate thread:

1) *Reading frames from camera*
2) *Human detection* is performed using SSD mobilenet lite [27], fine-tuned on spot-specific data set labeled by full-size SSD, created using recordings from each camera.
3) *Box tracking* integrates detected boxes from each frame into a set of currently tracked persons. Firstly, similarity matrix between each box and person is calculated, then one-to-one assignment is performed [28] based on SciPy [29] implementation. Basing on the score used for this matching, *reliability* of each person is altered - ones that were not matched to anything receive a most severe drop, but if they were previously matched, they will still be able to survive several frames before they disappear. A new person with low *reliability* is created when unmatched box probability exceeds the given threshold. The Similarity between box and person is calculated as a weighted sum of: Intersection over Union of the proposed box with estimated person box in current time (calculated using velocity and previous boxes averaged with momentum), spot location difference and height difference.

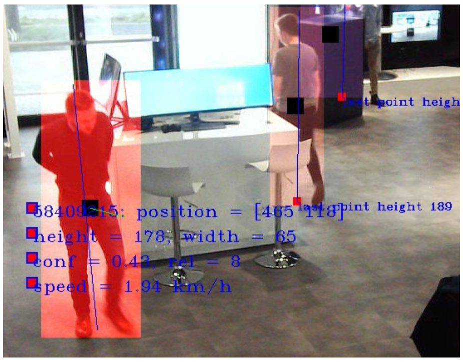


Fig. 4. Detecting real location of partially visible person (man on the right). Since his legs are mostly invisible on the picture, neural network detected only torso. Algorithm detected it and found an approximate point of his feet using head position and height.

4) *Sending* locations and cropped frontal images to server

As a result, the algorithm works with a stable speed of about 8 FPS.

As an alternative to the previous method, when it is possible to place the camera on the ceiling, we propose a tracking approach based on motion detection. This is suitable especially on narrow or crowded passages, where it is hard for people not to cover each other, looking from the side camera.

The first step is image processing to get points that will later be used for clustering. To initially remove noise we use manually implemented Sobel edge detector due to its fast computation on GPU. Afterward, for motion detection, instead of subtracting subsequent frames or saved background image, we use subtracting background computed as the average of previous frames with momentum. With the right parameters, this approach is both resistant to temporarily motionless people (contrary to subtracting subsequent frames) and changing environment i.e. in the form of objects left on the ground (contrary to subtracting saved background). Finally, we choose pixels meeting the given threshold and remove isolated ones that gives us noiseless image.

We tried multiple clustering algorithms using scikit-learn library [30], including hierarchical, OPTICS, Birtch, DBSCAN, K-means and a combination of the last two, however each failed to suit the task. DBSCAN was the closest match, but failed to separate people walking literally side by side. The need was for an algorithm that does not know the number of clusters, is fast with many points (not necessarily many clusters), with the only assumption about the distribution that clusters are denser in the middle, where clusters can touch with a local structure comparable to some clusters interior, however having approximately constant, circular size. Therefore we propose a simple custom approach to clustering based on these assumptions, with the only important parameter being the radius of the cluster and computational complexity $\mathcal{O}(n^2)$, also benefiting from distributed vectorized operations. We calculate the distance matrix between each pair of points, then check for each distance if it is smaller then radius, creating a connectivity matrix for a graph. Then, we iterate over vertices by descending degree and greedily assign a new cluster to check if it does not intersect with any previous (contain vertex already assigned to the cluster). Note that we want that greed because it fulfills the assumption that cluster centers are local maxima of density and without it, if we rewrite the problem into maximizing the number of non-intersecting clusters, two persons side by side are sometimes clustered as three.

To track these clusters, we use the same algorithm as with a neural network based approach, however, instead of IoU of boxes, the similarity of clusters is calculated as symmetric Kullback Leibler divergence, assuming that points form 2-dimensional normal distribution.

C. Merger - connecting the same person's paths from different cameras

To track a person for a longer period of time, we need to merge paths of the same person from different cameras. This is especially desirable in the context of person-device matching, since the longer the path we have, the easier it is to distinguish whether a person has a given device.

The state of the merger algorithm can be represented as a graph, where each path is a vertex and each edge represents the possibility of merging two paths. Within this set, when a new location is added to the path, we only need to update all edges connected to the corresponding vertex, performing computation with complexity independent of their length, unless this triggers merging paths. Managing merges of these vertices is handled using fast `Find-Union` algorithm [31]. In order to simplify the calculation and comparison of paths, locations in the paths are linearly interpolated so that the subsequent timestamps match fixed intervals. Note that the path is processed using the Kalman Filter, so it is enriched with information about the variance, interpreted as the certainty of location prediction. There are three events that can happen after receiving a new location:

1) Initialization: Initialization of a new path after receiving an unknown identifier. Assuming the local camera tracker does not already track this person with a different identifier, edges are added to each vertex, except the ones originating from the same camera.

2) Reject: Rejects are removals of edge from the graph. This happens, when corresponding locations (in time, with their variance) from different paths do not pass Two-Sample t-Test for Equal Means [32], so that within a certain confidence interval, we know that these locations do not originate from the same distribution.

3) Merge: Merges have lesser priority then *Rejects*, as we only take into consideration current, not removed edges. Therefore edges of a merged vertex are the intersection of component vertices neighborhood. This is intuitive and helpful because if given two paths were simultaneously tracked on the same camera in the past or separated significantly, we remember that they cannot originate from the same person also after merge with another path. In practice, in most cases we merge vertices connected by only edge left by *Rejects*, however this is not the case, when pair of people walks together tracked with two cameras, always maintaining close distance. When two paths coexist for a given time without *Reject*, similarity of paths X and Y is calculated as $(\|\mathbf{D}(\mathbf{X},\mathbf{Y})\|_\mathbf{2})^{-\mathbf{1}}$, where $\mathbf{D}(\mathbf{X},\mathbf{Y})$ is a vector of euclidean distances between corresponding in time path locations. When the value meets the given threshold, the edge is put on *Merge* priority queue with calculated similarity. The queue is resolved each several iterations, maximizing summed similarity of merged edges. Note that in general, it is `MAXIMUM WEIGHTED CLIQUE COVER` problem with weights on edges, which is at least NP-hard (as a generalization of `CLIQUE COVER`). However, since practical instances are generally small and without any complex structures, we found out that greedy heuristic, trying to merge priority queue starting from most similar edges is good enough.

D. Bluetooth / WiFi signal modeling

We propose two methods for WiFi and Bluetooth signal modeling based on Received Signal Strength Indication (RSSI). The first method is a parametric approach based on the Log-distance path loss model. The second approach is a novel non-parametric method similar to the existing probabilistic fingerprinting-based methods.

Since the received signal power generally decreases as the distance between the receiver and the transmitter increases, it is a valid source of information about the current location of the device of interest. However, RSSI values are heavily dependent on the surrounding environment and other factors such as the relative position of the device or the line of sight between the transmitting and receiving devices. Therefore, in both methods, we adopt a probabilistic approach to explicitly model the aforementioned uncertainty, where we are interested in the likelihood of observing an RSSI value conditioned on a current device location. It is important to note that the roles of the transmitter and the receiver in our models are switched when modeling WiFi and Bluetooth signals. For WiFi signals, we model the RSSI at one of our APs that is being transmitted from the person's device. Here, we know the position of the receiving AP, but the location of the transmitting mobile device is unknown. On the other hand, in case of Bluetooth, we model the RSSI value at the mobile device that is being transmitted from one of the BLE beacons. This way, we know the location

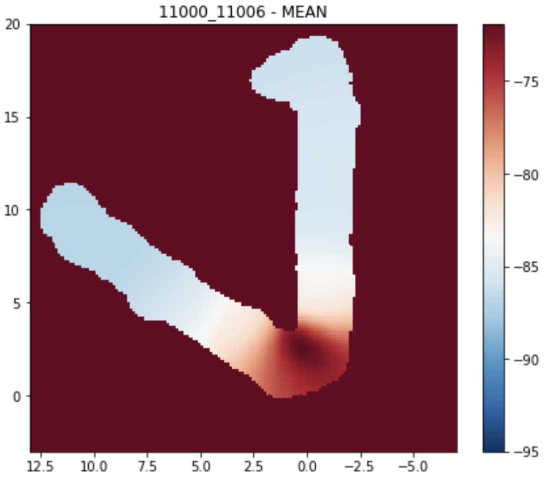


Fig. 5. Heatmap of estimated expected values of the RSSI distribution based on our non-parametric fingerprinting method.

of the transmitting beacon, but the location of the receiving device remains unknown. Another key difference in WiFi and Bluetooth modeling is the fact that in the case of the WiFi the transmitting power of the mobile device is unknown and can vary in time, whereas the transmitting power of the BLE beacon is known and does not change in time. In this work, however, in both cases, we assume that the transmitting power is constant. Thus, we lose on the quality of our WiFI models at the cost of a unified and more transparent approach.

Log-distance path loss model is a radio propagation model that predicts the loss in the signal strength, measured in decibels (dB), inside a building or densely populated areas over distance. We extend the standard log-distance model with the information about the cosine of the angle between the direction the person is facing and the direction of the AP or BLE beacon of interest. This way we can take into account the loss in the signal strength due to the body occlusion, assuming that the device is located at the front of a person. With a further assumption of homoscedasticity of variance and gaussian errors, the log-distance path loss model is a standard log-linear regression model:

$$f(s|x) = \mathcal{N}(s; \beta + \gamma log(d(x)) + w\cos(\alpha(x)), \sigma^2)$$

where s is the RSSI value, x is the device location, $d(x)$ is the distance between the transmitter and the receiver, $\alpha(x)$ is the above-mentioned angle, γ is the estimated path loss exponent that depends on the environment and σ^2 is the estimated variance based on residuals from the fitted model. The key advantage of this method over the second approach is its generalizability. Once we estimate the path loss exponent for a certain environment, we can reuse the fitted model in a different spot location with similar environmental properties, without the offline stage of model training.

Our second approach is similar to the existing fingerprinting-based methods. Here, we assume that we are given a training data set $\{(x_i, s_i)\}_{i=1}^n$ of locations x_i and corresponding RSSI values s_i that where gathered during the offline stage for each AP/BLE beacon in the spot. This data can be gathered efficiently with the help of the video tracking system described in section IV-B. We define a dense grid of point $G = \{x_{i,j}\}$ locations for which we will estimate locally the distribution of RSSI values. In our experiments, the grid had a size of 100×100 with a resolution of less than 0.5 meters. For each point in the grid x_i, we create the set of its nearest neighbors in a given radius r based on the euclidean distance. We define the reliability of each neighbor x_j using the squared exponential kernel with a fixed length scale l - $w_{i,j} = \exp - \frac{\|x_i - x_j|_2^2}{2l^2}$. Next we define unbiased weighted estimators for the mean and variance using the computed reliability weights:

$$\hat{\mu_i} = \frac{1}{V_1} \sum_j w_{i,j} s_j$$

$$\hat{s_i^2} = \frac{1}{V_1 - (V_2/V_1)} \sum_j w_{i,j} (s_j - \hat{\mu_i})^2$$

where $V1 = \sum_j w_{i,j}$ and $V_2 = \sum_j w_{i,j}^2$. Finally, the likelihood of observing a given RSSI value s for a new location x is estimated using the gaussian model with mean and variance of the closest grid point $x_i = \underset{x_j}{\operatorname{argmin}} \|x_j - x|_2$

$$f(s|x) = \mathcal{N}(s; \hat{\mu_i}, \hat{s_i^2})$$

Alternatively, when the spot area is substantially larger and the corresponding grid resolution is lower we can perform linear interpolation of the computed first and second moment estimators prior to likelihood calculation.

E. Human tracking based on radio data

Equipped with a probabilistic signal model we can efficiently tackle the problem of device localization and tracking using either WiFi or Bluetooth signal. We again adopt a probabilistic view of position estimation, i.e. we are interested in computing:

$$x_{1:n}^* = \underset{x_{1:n}}{\operatorname{argmax}}\ p(x_{1:n}|s_{1:n})$$

where each s_i is a set of RSSI measurements observed in a given time window and x_i^* is the estimated location. For notational brevity, we do not distinguish between the AP that received the signal or the transmitting BLE beacon, assuming that for each device we use the corresponding model.

Firstly we focus on estimating position for a single time window. Putting a uniform prior on location $\pi(x) \propto 1$ we calculate

$$p(x|s) \propto f(s|x)\pi(x) = f(s|x) = \Pi_i f(s_i|x)$$

where s_i is a single RSSI measurement. Therefore as the most probable location we simply take $x^* = \underset{x}{\operatorname{argmax}}\ \Pi_i f(s_i|x)$.

To account for spatio-temporal correlations in device localization we use a first-order Kalman Filter, where the underlying noise process models the acceleration of the tracked object.

As a result, for each time step, we obtain the estimated mean and variance of the device position as well as its velocity.

F. Person - device matching

Person - device matching is a key component of the Arahub system, as it enables combining the information extracted from visual cues, e.g. using face recognition systems, with a rich user history based on the advertising identifier or MAC address. We distinguish two tasks for the person - device matching. *Local matching* is focused on correctly assigning a device, from a pool of visible devices, to the user at the moment of entering a spot of interest, e.g. a LED panel. *Global matching* is a continuous process of performing global assignments of all visible devices to all persons currently tracked within a single spot.

Irrespective of the matching task being performed, we first focus on processing video tracking data together with the incoming signal data. To minimize the computation overhead when performing local matching, the process of combining the information about the location of a person at a given time with the incoming signal value is performed in an online fashion. We match a readout about the location with a given RSSI value if their corresponding time difference is less than a specified threshold, usually half a second. When a new signal readout from a device is received, we try to match it with all currently visible tracks. Similarly, when a new location readout is received, we try to match it with all active devices. After successfully matching a location x to a signal value s, the likelihood $f(s|x)$ is computed using one of the models described in section IV-D. The matching system also handles track merges, by taking the union of the location readouts for each track and computing new location-to-signal matches if necessary. Moreover, to provide stable performance over time, we clean up information about inactive tracks and devices.

To solve the *Local matching* task we once again refer to the probabilistic approach. Assigning a device to a person can be formulated as taking a device with the highest conditional probability of observing its signal conditioned on a given track $f(s^i_{1:n_i}|x_{1:m})$. However, to account for a varying number of received signal readout for each device n_i, we focus on maximizing the geometric mean of the total likelihood instead:

$$s = \operatorname*{argmax}_{s^i} f(s^i_{1:n_i}|x_{1:m})^{1/n_i}$$

To solve the *global matching* problem, we first define a cost matrix C, where each entry $c_{i,j}$ represents the cost of assigning a device i to a person j and is equal to the average log-likelihood of observing a total signal s^i conditioned on the tracking locations x^j. We assume independence between each device signal readouts, conditioned on the location, obtaining

$$C = [c_{i,j}]_{i,j} = \frac{1}{n_i} \sum_k \log(f(s^i_k|x^j_{1:m}))$$

Finally, we solve the linear assignment problem [28] using the matrix C to obtain person-device matching. In both local and global matching, if the resulting average log-likelihood of observing a given device signal conditioned on a track is lower than a predefined threshold, we omit this pair in the final assignment.

V. Evaluation

To provide automated testing for algorithms and adjust parameters, we created a simple video tagging procedure. We define convex polygons covering locations space and count for every person where it started and ended its walk and compare its path with manually annotated. This is suitable for both tracking methods. Also using this procedure, we can count how many people entered some room or provide statistical information on people flow around different areas in the commercial area or even shelves.

To reliably test the difficult cases of counting people entering and leaving the room using the motion-based camera mounted on the ceiling, we created a test at a hallway with three exits. In each pass, two people walked touching shoulder to shoulder and either diverted, or walked close together to one exit. The metric was, as described above, how many people passed between each pair of areas, creating a total of 28 manually tagged passes. The algorithm achieved an accuracy around 0.93. The test was carried out in this way because for an analogous, non-directed test in which people naturally and independently entered rooms with 35 passes, the effectiveness was errorless.

A. Use-cases and applications

In order to test the Arahub system in real-life scenarios, we installed it in two sites, that were similar to our target installation environments. Both sites were closed, private spaces yet a substantial number of different people were moving around, thus we could test the system without having control over the environment and people involved.

1) Office Lab: The first location was placed in an office space, that included about 30 persons. Arahub system was installed along a L-shaped corridor that connected all offices, conference rooms, reception, kitchen and utility rooms. The map of the location is presented in figure 6. In total, we used 5 Araboxes - two in each branch of the corridor and one on the bend. They were placed such that it was possible to observe a person entering through the main entrance in the reception and then moving along the corridor, passing all the offices and rooms till the end of the office space. Moreover, two digital displays were installed in the corridor: one at the entrance near the reception desk and the second one at the end of the corridor near a bathroom. In addition, 7 BLE beacons were installed in the corridor in order to uniformly cover it with BLE signal.

The office lab was used for our initial tests and tuning of the system. Our goal was to enable the following minimum requirements for the system:

1) track continuously three persons moving together with spacing between them not less than 3m.
2) track continuously three smartphones that have our custom application installed and running

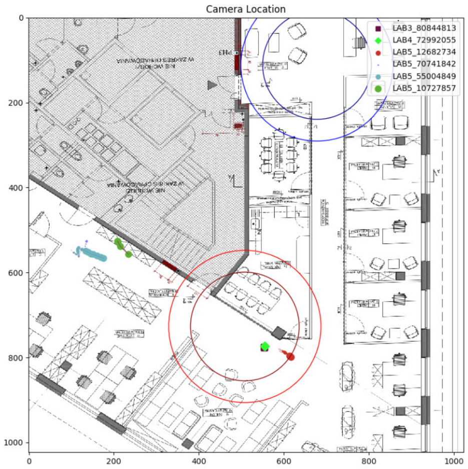


Fig. 6. A map of the office lab. The circles indicate places in which the Arahub system performed an action when human was present - in this case the information about that person was shown on a digital display.

3) be able to assign a smartphone to a person when it approaches a digital display, with accuracy of 80%, after each person walked the distance of the whole corridor length.

Eventually, Arahub system was able to perform according to those three requirements. However, the final accuracy depended heavily on the type of smartphones used in the test. Android-based devices were tracked accurately in about 70% of cases, while for iOS-based devices the accuracy was over 90%. The accuracy was calculated based on 30 trials - separately for both types of devices.

2) Showroom Lab: The second test location was placed in a showroom of one of our business partners. The showroom is a space dedicated to presenting new products to customers; it consists of a large hall with different displays on the walls and conference room. Arahub was deployed to cover the main hall were customers were guided by the showroom's employee. In total 7 Araboxes were installed in addition to 10 BLE beacons. Moreover, one extra Arabox was placed on the ceiling in a narrow part of the showroom - it was used for testing the person counting functionality. The showroom was occupied by 1-2 employees all the time and several times a day, a group consisting of up to 8 people was guided by them. Two digital displays already installed in the showroom were used for the needs of Arahub. Moreover, an alternative mobile application was created - in this version the user could choose one of three products in the application, then a video material, related to this product, was played as this person moved near one of the selected displays.

B. Experiments

The showroom lab was used for testing the performance of Arahub's person tracking capabilities (without person re-identification). In the test, the lab was divided into three sub-areas observed by seven araboxes with overlapping fields of view: 1) narrow corridor - visible by 2 araboxes, 2) large hall with multiple obstacles - visible by 4 araboxes, 3) small hall with one obstacle - visible by 3 araboxes. The goal was to continuously track persons moving between sub-areas. We performed tests in which from 2 to 4 persons were moving across the whole lab using different paths. Moving between sub-areas was counted as a transition. If the system was not able to track a person during a transition, it was counted as a tracking error. Additionally two cases were tested: persons moving separately (not touching each other) and persons moving jointly (without visible separation between them). The results are presented in table I. We may conclude that the arahub system is able to track separately moving persons with high accuracy. However, as re-identification functions were not used, it had difficulties to track persons moving in very close proximity.

TABLE I
EVALUATION RESULTS - CONTINUOUS TRACKING OF PERSONS MOVING BETWEEN PREDEFINED LOCATIONS IN AN AREA OBSERVED BY 7 ARABOXES WITHOUT PERSON RE-IDENTIFICATION

test no.	case	transitions	transition errors	number of persons	accuracy
1	joined	4	3	2	0,25
2	joined	8	3	2	0,63
3	joined	9	3	2	0,67
4	separated	4	1	4	0,75
5	separated	4	0	1	1,00
6	separated	9	0	2	1,00
7	separated	11	1	2	0,91

VI. CONCLUSIONS AND FUTURE WORK

In this work, we have provided a comprehensive description of the Arahub system. We have shown that it is possible to successfully integrate tracking data from video system and smartphones and use it for commercial purposes. Our work was tested in real-life environments, and however it is still at an advanced prototype level, we are able to deploy it in commercial applications. In our work we developed several novel methods for improving tracking and integration of multi-modal signals, we also focused heavily on optimization to provide a solution that is cost-efficient.

The Arahub system needs to be developed towards more versatile usage capabilities e.g. in outdoor environments, or for high-density crowd scenarios. Moreover, the biggest issues are connected to incompatibility between different smartphone brands and systems. Our tests show that even covering 80% of the smartphone brands currently available on the market, requires a substantial amount of fine-tuning. In order to scale the system, a more granular approach of data analysis could be introduced, e.g. person tracking could be done at the crowd level initially, but at a single-person level when more details are needed [33], [34], [35].

We are also developing methods for improving privacy concerns. The current version of Arahub is meant to be deployed in controlled environments, where users have may opt-in freely. There is a need to provide anonymization methods [36],

[37], which would ensure that even the system operator is not able to use the system for other means than statistical analysis of visitors. We are researching the possibility of using novel cryptography methods, that allows one to use data for machine learning purposes without revealing private information.

Acknowledgment

Let us thank Wojciech Rosiński for his great contribution to this work. We also appreciate the support offered by Jetline and Samsung who provided testing space for our experiments.

References

[1] R. Frączek, B. Cyganek, and K. Wiatr, "Parallelized algorithms for finding similar images and object recognition," *Computer Science*, vol. 14, no. 1, 2013.

[2] M. Meina, A. Janusz, K. Rykaczewski, D. Ślęzak, B. Celmer, and A. Krasuski, "Tagging firefighter activities at the emergency scene: Summary of aaia'15 data mining competition at knowledge pit," in *FedCSIS 2015*, 2015, pp. 367–373.

[3] J. Wilson, S. Chaudhury, and B. Lall, "Clustering short temporal behaviour sequences for customer segmentation using LDA," *Expert Syst. J. Knowl. Eng.*, vol. 35, no. 3, 2018.

[4] I. Rüb, M. Matraszek, P. Konorski, M. Perycz, A. Waśniowski, D. Batorski, and K. Iwanicki, "30 sensors to mars: Toward distributed support systems for astronauts in space habitats," in *ICDCS 2019*, 2019, pp. 1704–1714.

[5] J. Bułat, K. Duda, M. Duplaga, R. Frączek, A. Skalski, M. Socha, P. Turcza, and T. P. Zieliński, "Data processing tasks in wireless gi endoscopy: Image-based capsule localization navigation and video compression," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2007, pp. 2815–2818.

[6] H. Lu and M. A. Cheema, "Indoor data management," in *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*, 2016, pp. 1414–1417.

[7] J. Domaszewicz, S. Lalis, A. Pruszkowski, M. Koutsoubelias, T. Tajmajer, N. Grigoropoulos, M. Nati, and A. Gluhak, "Soft actuation: Smart home and office with human-in-the-loop," *IEEE Pervasive Computing*, vol. 15, no. 1, pp. 48–56, 2016.

[8] A. Krasuski, A. Jankowski, A. Skowron, and D. Ślęzak, "From sensory data to decision making: A perspective on supporting a fire commander," in *2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, vol. 3, 2013, pp. 229–236.

[9] D. H. Hepting, R. Spring, and D. Ślęzak, "A rough set exploration of facial similarity judgements," in *Transactions on Rough Sets XIV*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 81–99.

[10] M. Świechowski and D. Ślęzak, "Introducing logdl - log description language for insights from complex data," in *FedCSIS*, 2020.

[11] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1067–1080, 2007.

[12] Q. Dong and W. Dargie, "Evaluation of the reliability of rssi for indoor localization," in *2012 International Conference on Wireless Communications in Underground and Confined Areas*. IEEE, 2012, pp. 1–6.

[13] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal processing magazine*, vol. 22, no. 4, pp. 54–69, 2005.

[14] X. Li, K. Pahlavan, M. Latva-aho, and M. Ylianttila, "Comparison of indoor geolocation methods in dsss and ofdm wireless lan systems," in *Vehicular Technology Conference Fall 2000. IEEE VTS Fall VTC2000.*, vol. 6. IEEE, 2000, pp. 3015–3020.

[15] R. Peng and M. L. Sichitiu, "Angle of arrival localization for wireless sensor networks," in *2006 3rd annual IEEE communications society on sensor and ad hoc communications and networks*, vol. 1. Ieee, 2006, pp. 374–382.

[16] A. Zhang, Y. Yuan, Q. Wu, S. Zhu, and J. Deng, "Wireless localization based on rssi fingerprint feature vector," *International Journal of Distributed Sensor Networks*, vol. 11, no. 11, p. 528747, 2015.

[17] A. Golovan, A. A. Panyov, V. V. Kosyanchuk, and A. S. Smirnov, "Efficient localization using different mean offset models in gaussian processes," in *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2014, pp. 365–374.

[18] B. F. D. Hähnel and D. Fox, "Gaussian processes for signal strength-based location estimation," in *Proceeding of robotics: science and systems*. Citeseer, 2006.

[19] L. Pei, R. Chen, J. Liu, H. Kuusniemi, T. Tenhunen, and Y. Chen, "Using inquiry-based bluetooth rssi probability distributions for indoor positioning," *Journal of Global Positioning Systems*, vol. 9, no. 2, pp. 122–130, 2010.

[20] X. Zhao, Z. Xiao, A. Markham, N. Trigoni, and Y. Ren, "Does btle measure up against wifi? a comparison of indoor location performance," in *European Wireless 2014; 20th European Wireless Conference*. VDE, 2014, pp. 1–6.

[21] T. S. Rappaport *et al.*, *Wireless communications: principles and practice*. prentice hall PTR New Jersey, 1996, vol. 2.

[22] J. S. Kulchandani and K. J. Dangarwala, "Moving object detection: Review of recent research trends," in *2015 International Conference on Pervasive Computing (ICPC)*, 2015, pp. 1–5.

[23] R. L. F. Fleuret, J. Berclaz and P. Fua, "Multi-camera people tracking with a probabilistic occupancy map," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, p. 267–282, 2008.

[24] R. Iguernaissi, D. Merad, K. Aziz, and P. Drap, "People tracking in multi-camera systems: a review," *Multimedia Tools and Applications*, vol. 78, 09 2018.

[25] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[26] G. Bradski, "The opencv library. dr. dobb's journal of software tools," 2000.

[27] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," *Lecture Notes in Computer Science*, p. 21–37, 2016.

[28] D. F. Crouse, "On implementing 2d rectangular assignment algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 4, pp. 1679–1696, 2016.

[29] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[30] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[31] B. A. Galler and M. J. Fisher, "An improved equivalence algorithm," *Commun. ACM*, vol. 7, no. 5, p. 301–303, May 1964.

[32] D. H. Jones, "Book review: Statistical methods, 8th edition george w. snedecor and william g. cochran ames: Iowa state university press, 1989. xix + 491 pp," *Journal of Educational and Behavioral Statistics*, vol. 19, no. 3, pp. 304–307, 1994.

[33] W. Świeboda, A. Krauze, and H. S. Nguyen, "A granular evacuation modeling framework," *Annals of Computer Science and Information Systems*, vol. 2, p. 337–342, 2014.

[34] M. Świechowski and D. Ślęzak, "Granular games in real-time environment," in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, 2018, pp. 462–469.

[35] M. Przyborowski, T. Tajmajer, L. Grad, A. Janusz, P. Biczyk, and D. Ślęzak, "Toward machine learning on granulated data – a case of compact autoencoder-based representations of satellite images," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018, pp. 2657–2662.

[36] P. Szczuko, "Simple gait parameterization and 3d animation for anonymous visual monitoring based on augmented reality," *Multimedia Tools and Applications*, vol. 75, no. 17, pp. 10 561–10 581, Sep 2016.

[37] B. Cyganek, "Change detection in multidimensional data streams with efficient tensor subspace model," in *Hybrid Artificial Intelligent Systems*. Cham: Springer International Publishing, 2018, pp. 694–705.

DOI: 10.15439/2020F213
ISSN 2300-5963 ACSIS, Vol. 21

Deep Bi-Directional LSTM Networks for Device Workload Forecasting

Dymitr Ruta
EBTIC, Khalifa University, UAE
dymitr.ruta@ku.ac.ae

Ling Cen
EBTIC, Khalifa University, UAE
cen.ling@ku.ac.ae

Quang Hieu Vu
Zalora, Singapore
quanghieu.vu@zalora.com

***Abstract*—Deep convolutional neural networks revolutionized the area of automated objects detection from images. Can the same be achieved in the domain of time series forecasting? Can one build a universal deep network that once trained on the past would be able to deliver accurate predictions reaching deep into the future for any even most diverse time series? This work is a first step in an attempt to address such a challenge in the context of a FEDCSIS'2020 Competition dedicated to network device workload prediction based on their historical time series data. We have developed and pre-trained a universal 3-layer bi-directional Long-Short-Term-Memory (LSTM) regression network that reported the most accurate hourly predictions of the weekly workload time series from the thousands of different network devices with diverse shape and seasonality profiles. We will also show how intuitive human-led post-processing of the raw LSTM predictions could easily destroy the generalization abilities of such prediction model.**

***Index Terms*—Workload prediction, time series, Long Short-Term Memory (LSTM), ensemble averaging**

I. Introduction

PREDICTIVE analytics on workload-related characteristics has become increasingly important. Reliable workload prediction of monitored devices becomes critical in order to proactively manage the capacity of connected infrastructure, mitigate cyber security risks and simply respond early to the anomalous behaviour of the monitored IT infrastructure [1]. Accurate forecasting of the future host workload plays also a central role for robust scheduling and resources management in data centers and cloud computing and among many expected benefits could lead to reduced operational cost, for example in a form of eliminated or cut idle time of the devices [2], [3], [4].

Prediction of future workload characteristics has received considerable research interests in both academia and industrial applicationss. Simple linear techniques like (auto-regressive) moving average (ARMA) models have been used heavily in this field [5], [6], [7], and enjoyed relatively good performance at the very low computational cost. As the complexity of the time-series increases, a subtle dependence of the future on the past may be non-linearly implied in the uni- or multi-variate series and linear models struggle or completely fail to efficiently unscramble such dependence. In recent years, workload prediction has also been attempted using non-linear machine learning models, e.g. Bayesian model [8], neuro-fuzzy and Bayesian inference [9], Neural Network (NN) [10]. Despite the fact that these non-linear models are not particularly suited to learn temporal dependencies in sequences, their strong regression capabilities often led to the predictive performance improvement measured in the static conditions shifted over moving window. The developments of the Recurrent Neural Networks (RNN) managed to better capture internal correlations along the data series and in an instant became naturally suited and applied to to sequential data analysis, including workload prediction. In [2], an adaptive model was developed for highly-variable workloads prediction by integrating a Top-Sparse Auto-Encoder (TSA) and Gated Recurrent Unit (GRU) blocks into RNN. In [3], workload sequences in Cloud and Grid systems were predicted by developing a model of stacking prediction algorithms using RNN and Autoencoder. An approach based on the Long Short-Term Memory (LSTM) encoder-decoder network with attention mechanism was proposed in [11]. In [12], a GRU-based encoder-decoder network containing 2 gated recurrent neural networks was implemented for prediction of multi-step-ahead host workload in cloud computing.

Accurate workload prediction is a challenging problem. Different time series from various devices typically have varied patterns that not only lack of well pronounced stationarity but also are often full of sudden spikes, dropouts, staircase and other complex temporal patterns. This makes them very difficult to model and predict using the same predictor or even the same class of predictive models. This paper presents an ensemble model based on Bi-Directional Long Short-Term Memory (BiLSTM) networks developed and pre-trained for prediction of a large class of network device workload time series. At its core we have proposed a regression network with our own architecture containing 3 BiLSTM layers that appears to perform very well for a very diverse workload time series. Its prediction accuracy evaluated on thousands of different devices' workload series was acknowledged to generate best results in the FedCSIS'2020 Data Mining Challenge, details of which are further elaborated in sections below.

The remainder of the paper is organized as follows. The FedCSIS'2020 Challenge is briefly described in Section II. Data pre-processing is presented in Section III, followed with the description of the core BiLSTM network and its ensemble aggregation model in Section IV and the experimental results in Section V. Concluding remarks are provided in Section VI.

II. FEDCSIS'2020 CHALLENGE

The aim of this challenge was to predict workload-related characteristics of monitored devices based on historical data collected from these devices. Accurate prediction of device workload can be quite useful to manage infrastructure capacity, mitigate cyber security risks and early respond to anomalous events. The data provided in the competition were collected by EMCA Software that is a Polish vendor of Energy Logserver, a globally operating system capable of collecting data from various log sources to provide in-depth data analysis and alerting to its end-users [1].

The training data were organized in hourly aggregated values of various workload characteristics extracted from device logs, provided in the form of a CSV table also containing device identifiers and timestamps of the aggregation windows. Overall there were 24249 devices' time series, each containing 7 sets of workload related statistics including mean, standard deviation and the candlestick intra-hour aggregates of open, high, low, close of each hour. Each of such multivariate time series was captured along 1924 subsequent hours spanning over 80 days from 2019-12-2 07:00:00 to 2020-2-20 10:00:00, however, a significant number of values were missing.

Based on these data the task of the competition was to predict the following week i.e. a 168-hourly future sequences starting at 2020-02-20 11:00:00 - directly after the end of the training data, of the mean workload for only the selected subset of 10000 series. The competitors' solutions provided in a form of 10000 168-element vectors y_i were evaluated against the true values f_i using the R^2 score defined as follows:

$$R^2(y,f) = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \overline{y})^2} \tag{1}$$

During the competition only partial feedback on the performance of the competitors' models was provided by the knowledgepit.ml platform, on which the competition was hosted. This feedback was in a form of the preliminary R^2 score that was computed over a small unknown subset of the complete testing set of 10000 series.

It is important to note that although $\overline{y}$ is supposed to stand for the mean of the true values of the testing series, in fact in this competition the scope of the mean has been extended to include a complete series, i.e. it is a mean of both training and testing parts of the series. This has been necessitated by the risk of infinite R^2 scores reported over the testing week that could easily happen for a single flat or dropped out signal that would hijack the complete score of 10000 series.

R^2 score scaled between $-\infty$ and 1 is considered to be a very "tough" or penalizing measure of the regression performance compared to the mean squared error (MSE) or relative MSE measurements. This is simply because it is open to the large negative scores observed for random predictions and to even reach the score of 0 one has to correctly match the mean between predicted and actual values.

III. DATA PROCESSING

Considering that the task of the competition concerned the prediction of only the mean values of devices' workload characteristics, there was a founded temptation to only use the mean values time series rather than the whole candlestick series and the signal volatility as features. Following a rapid prototyping experiments with a couple of standard and simple regression models we have concluded that none of the time series other than the mean, even in the multivariate regression setup, bring any visible improvement in predicting the future hourly mean values of the series, while their inclusion only multiplies the computational cost of the regression model. Backed up by these results we have reduced the data to include only the mean workload series for both training and testing sets. Our task was therefore simplified to predicting 168 subsequent values of the 10000 univariate time series based on their own history of 1924 hourly values as well as other univariate time series available in the training set containing in total 24249 series of 1924 hourly mean workload values. For computational simplicity the whole data were represented as a matrix $X^{[24249\times1924]}$ of values in a single precision format to reduce memory requirements.

A. Filling missing values

Out of the 46655076 values, 2265379, which is almost 5%, were missing. Since the models we intended to apply to the regression problem did not accept missing values we have developed a simple scheme to fill all missing values that simply fills the average values from the same hour in the same week-day across all weeks if there is at least one values given for this hour, otherwise, it is filled with the average of its closed hour, i.e. previous or next hour in the same weekday.

B. Normalization

After all missing values are filled, each time series is then normalized to zero mean and unit standard deviation, expressed as:

$$\tilde{x} = \frac{x-\mu}{\sigma}, \tag{2}$$

where μ and σ are the mean and standard deviation, yielding the output time series with

$$\tilde{\mu} = 0 \ , \ \tilde{\sigma} = 1. \tag{3}$$

C. Data partition

Competition participants are required to predict workload for only selected 10000 among the 24249 device time series. Accordingly, the whole data set X was partitioned into training and validation sets.

- The training set contains the selected 10000 time series, which will be used for model training and testing.
- The validation set contains all the remaining 14249 workload time series after ensuring that:
  - all of the values used for validation are positive;
  - all time series have positive standard deviation.

IV. The universal time series regression network

Note that after the preprocessing we are left with a set of time-aligned but independent series of data, each of which needs forward prediction in time. Using traditional time series approach one would in fact build a separate model for all of them independently and also independently use them to forecast the time series' future. The novelty of our proposed solution is that it intends to build a single model that is trained on all diverse time series and once pre-trained it is effectively expected to be able to predict the future of any other time-aligned time-series, based on its past, without any further (re)-training necessary. We have decided to build such universal time series prediction model using Long-Short-Term-Memory (LSTM) networks that are particularly suited for predicting deep futures of the variety of diverse time series data.

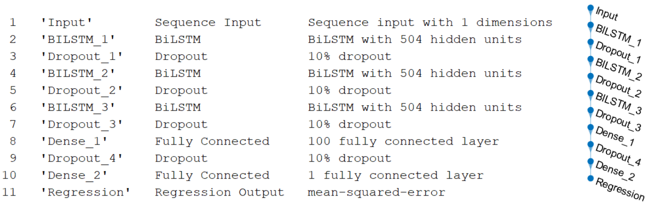
A. Long-Short-Term-Memory networks

LSTM networks are powerful family of models based on deep recurrent learning regression networks that are very flexible with a freedom of layered architecture design and powerful gated mechanism of LSTM layers that give them the ability to manipulate its memory state to extract complex patterns over long sequentially arranged input feature space. Due to these features LSTMs are known to successfully capture multitude of seasonalities, autocorrelations and other subtle time dependencies in both uni or multivariate mode and are reported to maintain stable accurate forecast deep into the future. As per the application recommendations we used bi-directional version of the LSTM (BiLSTM), that can learn from both past and future, which is suited to our problem when the predicted series is long and therefore has enough space to accommodate forward and backward learning patterns.

B. BiLSTM network architecture

There has been an iterative refinement process of constructing the final BiLSTM network architecture [13]. This process was guided by the observation of improved predictions with an increasing number of hidden units and a number of BiLSTM layers. Expanding the network along this tendency had two issues, however. Firstly the computational cost and hence the time of training grew very quickly, exponentially if expanding both the number of layers and their sizes. The second drawback is, that unless validation set was perfectly representative, the expanded network showed the tendency of becoming over-trained very quickly, although without clear and reliable rule as to when is the best moment to stop training. On top of this, allowing frequent validation evaluation during training is very costly and additionally slows down the experimentation phase of the network build.

To address the above issues we have noticed that we can compensate the additional cost of expanding the network by reducing the training set down the the k-last weeks. This process brought significant performance benefits up to when we tried to shrink the training series below 4 last weeks indicating that on average there is no predictive gain from learning from more than 3 weeks back. Eventually, we have expanded the BiLSTM layers up to 504 (3×168) hidden units and included 3 BiLSTM layers followed with 10% dropout layers. We have also tried to include ReLu layers that eliminate negative signals but eventually their impact turned out not to influence the results hence we dropped them. The 3 rounds of BiLSTM and dropout layers followed with two sets of dense layers separated with another dropout layer before eventually reaching the final (MSE) regression layer. The final network architecture, with which we have generated the final predictions, is presented in Figure 1:



```
 1   'Input'        Sequence Input      Sequence input with 1 dimensions
 2   'BILSTM_1'     BiLSTM              BiLSTM with 504 hidden units
 3   'Dropout_1'    Dropout             10% dropout
 4   'BILSTM_2'     BiLSTM              BiLSTM with 504 hidden units
 5   'Dropout_2'    Dropout             10% dropout
 6   'BILSTM_3'     BiLSTM              BiLSTM with 504 hidden units
 7   'Dropout_3'    Dropout             10% dropout
 8   'Dense_1'      Fully Connected     100 fully connected layer
 9   'Dropout_4'    Dropout             10% dropout
10   'Dense_2'      Fully Connected     1 fully connected layer
11   'Regression'   Regression Output   mean-squared-error
```

Figure 1. Deep BiLSTM Network Architecture

To take full advantage of the BiLSTM layers that require to look forward and backward in relation to the tested point we have trained the network in the sequence-to-sequence mode rather than the standard one-next-and-update mode. It is also worth noting that our validation strategy evolved from initially evaluating on the additional series not used in testing, through validating on the last 2 weeks of the available 10000 tested series, up to validating on just the last week of the available data. The training proceeded on the mini-batches of 64 randomly selected series and terminated when validation error has not been reduced within the last 50 iterations.

V. Experiments and post-processing

Once the network architecture has been established the experiments followed an iterative process of final parametric optimization guided by the regression performance measured on the validation set as indicated above. The generated validation and testing set predictions have been provisionally inspected particularly in terms of the signal and its trend continuity. For the vast majority of series that visually follow an established pattern the validation set predictions are very accurate as shown in Figure 2.

It is reassuring to observe a correct flat mean prediction whenever a signal resembles a random noise. Even unexpected sudden changes of the signal are to a certain extent reflected in the predictions. Overall the presented BiLSTM model in its standalone form received the preliminary R^2 score of 0.31

A. post-processing

Analysis of the predictions revealed some perceived issues for the time series that have sudden change of the signal near the end of the series as well as occasional signal dropouts to 0 or near 0. In response to these observations we have developed a set of additional post-processing techniques that were supposed to fix these issues.

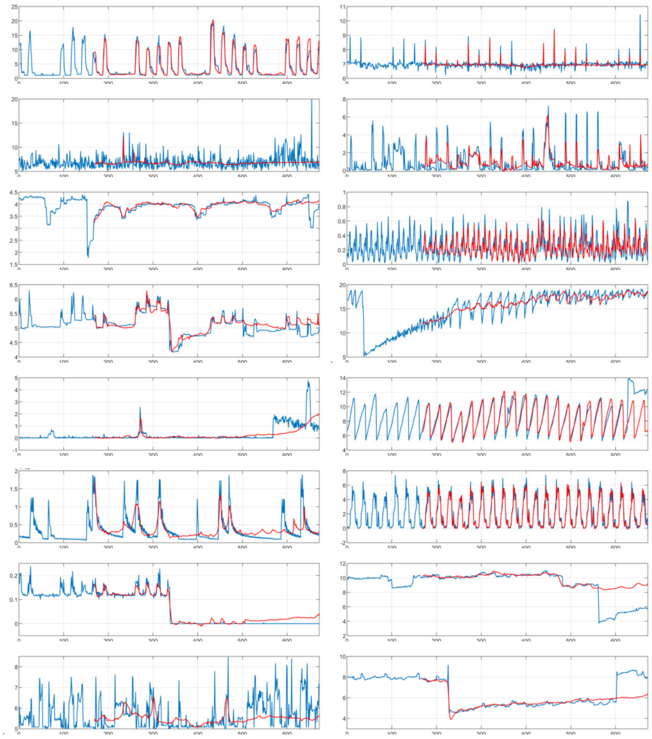


Figure 2. Validation set series samples (blue) and their predictions (red)

We have identified two categories of significant disparity between the last week of the training series and the following predictions that we have decided to address in post-processing:

- Signal dropouts: observed when all values of the training time series in the last week are (near) 0 but the predictions are not completely 0. In such case we have concluded that the signal which drops at its end to 0, should maintain its prediction also at the 0 level for the whole week.
- Excessive difference: observed when the mean difference between the last weeks' values and the predicted values is excessively large i.e. exceeds 3 standard deviations of the complete training series. In such case we have introduced two adjustments: moderation that simply computes the average between the last week and the predictions and trend alignment that replaces the network predictions with a simple average of the 3-weekly trend matched to continuously extrapolate the end of the series in the direction of the gradient between the last two weeks.

The above post-processing techniques have been applied to the predictions and their effects measured in a form of preliminary and, later revealed, final testing scores. Interestingly, all of these techniques improved the R^2 score but only reported over the validation set, also reflected by the gains in the preliminary scores. Unfortunately, as the final testing revealed after the end of the competition, none of the pre-processing techniques improved the performance measured over the complete testing set. The comparison of the preliminary and final testing R^2 scores are shown in the Table I.

Table I
PRELIMINARY AND FINAL R^2 SCORES OF THE BILSTM MODEL WITH VARIOUS POST-PROCESSING TECHNIQUES

model	post-proc	preliminary R^2	final R^2
BiLSTM	none	0.309	0.290
BiLSTM	dropout	0.312	0.288
BiLSTM	moderation	0.318	-2.38
BiLSTM	trend alignment	0.321	-0.779

B. Ensemble averaging

As a final step an arbitrary number (20-30) of the top solutions have been aggregated with the simple mean operator. Although this operation expectedly resulted with the average performance improvements, notably R^2 score was elevated to 0.322, these gains were negligible compared to the performance degradation caused by the signal post-processing.

VI. CONCLUSIONS

In this paper, we presented a powerful 3-layer BiLSTM network that has the capability to deliver deep predictions of a very wide and diverse spectrum of time series. The model achieved the performance of the R^2 score of nearly 0.3 beating all other competitive solutions in the FedCSiS'2020 competition. Although the subsequent post-processing introduced to the model turned out be a bad idea, this lesson learnt gives us confidence in the native capability of the deep BiLSTM networks to reliably predict diverse time series that the arbitrary and selective human corrections can only damage.

REFERENCES

[1] FedCSIS 2020 Challenge: Network Device Workload Prediction, *https://knowledgepit.ml/fedcsis20-challenge/*.

[2] Z. Chen, J. Hu, G. Min, A. Zomaya, and T. El-Ghazawi, "Towards Accurate Prediction for High-Dimensional and Highly-Variable Cloud Workloads with Deep Learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 4, pp. 923-934, April 2020.

[3] H. Nguyen, S. Woo, J. Im, T. Jun, and D. Kim, "A Workload Prediction Approach Using Models Stacking Based on Recurrent Neural Network and Autoencoder," *IEEE Int. Conference on High Performance Computing and Communications, IEEE International Conference on Smart City, IEEE International Conference on Data Science and Systems*, Dec. 2016.

[4] K. Qazi and I. Aizenberg, Towards quantum computing algorithms for datacenter workload predictions, *IEEE Int. Conf. on Cloud Comput.*, 2018.

[5] P. Saripalli, G. Kiran, R. Shankar, H. Narware, and N. Bindal, "Load prediction and hot spot detection models for autonomic cloud computing," *IEEE Int. Conf. in Utility and Cloud Computing*, pp. 397–402, 2011.

[6] R. Calheiros, E. Masoumi, R. Ranjan, R. Buyya, "Workload prediction using ARIMA model and its impact on cloud applications' QoS," *IEEE Trans. Cloud Comput.*, vol. 3, no. 4, pp. 449–458, 2014.

[7] P. Dinda, and D. O'Hallaron, "Host load prediction using linear models," Cluster Computing," vol. 3, no. 4, pp. 265–280, 2000.

[8] S. Di, D. Kondo, W. Cirne, "Host load prediction in a Google compute cloud with a Bayesian model," *Proc. of IEEE Int. Conf. on High Performance Computing, Networking, Storage and Analysis*, 2012.

[9] F. Benhammadi, Z. Gessoum, A. Mokhtari, CPU load prediction using neuro-fuzzy Bayesian inferences. Neurocomputing 74, 1606–1616 (2011)

[10] J. Kumar, A. Singh, Workload prediction in cloud using artificial neural net. and adaptive diff. evolution, *Futur. Gen. Comput. Syst.* 81:41–52, 2018.

[11] Y. Zhu, W. Zhang, Y. Chen and H. Gao, "A novel approach to workload prediction using attention-based LSTM encoder-decoder network in cloud environment," *EURASIP Journal on Wireless Communications and Networking*, Article number: 274, 2019.

[12] C. Peng, Y. Li, Y. Yu, Y. Zhou and S. Du, "Multi-step-ahead host load prediction with GRU based encoder-decoder in cloud computing," *IEEE Int. Conference on Knowledge and Smart Technology*, pp. 186–191.

[13] S. Hochreiter, J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 119–128 DOI: 10.15439/2020F202 ISSN 2300-5963 ACSIS, Vol. 21

Generating Fuzzy Linguistic Summaries for Menstrual Cycles

Łukasz Sosnowski
Systems Research Institute, Polish Academy of Sciences
Newelska 6, 01-447 Warsaw, Poland
sosnowsl@ibspan.waw.pl

Tomasz Penza
OvuFriend Sp. z o.o.,
Złota 61/100, 00-819 Warsaw, Poland
tomasz.penza@ovufriend.com

***Abstract*—This paper presents a method of generating linguistic summaries of women's menstrual cycles based on the set of concepts describing various aspects of the cycles. These concepts enable description of menstrual cycles that are readable for humans, but they also provide high-level information that can be used as control input for other data processing actions such as e.g. anomaly detection. The labels signifying these concepts are assigned to cycles by means of multivariate time series analysis. The corresponding algorithm is a subsystem of a bigger solution created as a part of an R&D project.**

I. Introduction

INFERTILITY is becoming a civilization disease. Statistics say that every fifth couple that is trying to conceive (TTC) has a problem to achieve pregnancy in the first 12 months of efforts, and this tendency is increasing [1]. In addition, the age of women trying for the first child statistically shifts towards 35. This is a problem because with age the risk of pregnancy problems increases, including the birth of a child with defects, and according to official terminology, pregnancies of women aged over 35 are referred to as ”geriatric pregnancy”.

OvuFriend[1] is a platform for women trying to conceive that allows them to document their menstrual cycle and receive feedback aimed at helping them successfully conceive. Using a mobile app, users provide declarative data about specific parameters of their body and subjective feelings recorded at specific times of the day. By providing this data, they gain access to the algorithms designed to help them conceive, a supportive community and other such tools.

The platform has collected data of over 400,000 menstrual cycles, e.g. symptoms felt in various stages of the cycle and measurements of basic factors used to determine the phase of the cycle and its fertility on a given day [4]. This data include, among others, measurements of baseline body temperature (BBT), type of cervical mucus occurring on particular days of the cycle, parameters of the cervix as well as the results of ovulation tests which measure the concentration of the LH hormone in a woman's body.

The company is conducting a research and development project co-financed from the National Center for Research and Development in Poland, aimed at eliminating barriers related to pregnancy and facilitating effective family planning at home environment. One of the elements prepared under the project is an AI algorithm dedicated to the prediction and confirmation of ovulation [15]. Its approach is to use a set of independent detectors that analyze time series of different parameters of the menstrual cycle to detect parameter-specific information about ovulation. The results of their analyses are aggregated with a set of weights that depends on the phase of the cycle to facilitate differentiation of the ovulation designation into two phases: prognostic and retrospective. Another issue that the project deals with is discovering the vulnerability of medical anomalies from data and sustaining intelligent communication between the system and the women using the OvuFriend's platform.

Automatic description of the menstrual cycle is an additional goal of the project that facilitates understanding of the processes occurring in a woman's body. The description is to describe the parameters of one's own menstrual cycle in such a way that they are understandable to the average woman of childbearing age without medical experience.

This last issue is the subject of this article. However, for a better understanding of the context, Fig. 1 presents a general diagram of the entire solution covered by the R&D project.

A. *Overview of the OvuFriend's AI platform*

The central element of the architecture is the AI module, which integrates the developed algorithms into a coherent interface that exchanges data between individual elements at the level of processing. This module is powered by data from a data warehouse built to store the cycle data provided by women. The most commonly used functionality is the ovulation algorithm [15]. In the part where insightful comparisons are made with available historical cycles both at the level of a single woman as well as a group of women characterized by similar cycles in terms of selected features, the algorithm uses networks of compound object comparators (NoC) described in detail in [14]. Determining the state of ovulation and the date of its eventual occurrence is the key information for further processing. The descriptions of the cycle that are generated help not only to make the decision regarding ovulation, but they are also used by other parts of the system, e.g. by the detection algorithm for the most common medical anomalies related to menstrual cycles (e.g. endometriosis [3]). However, this work will describe the part of the solution that generates

[1] www.ovufriend.pl

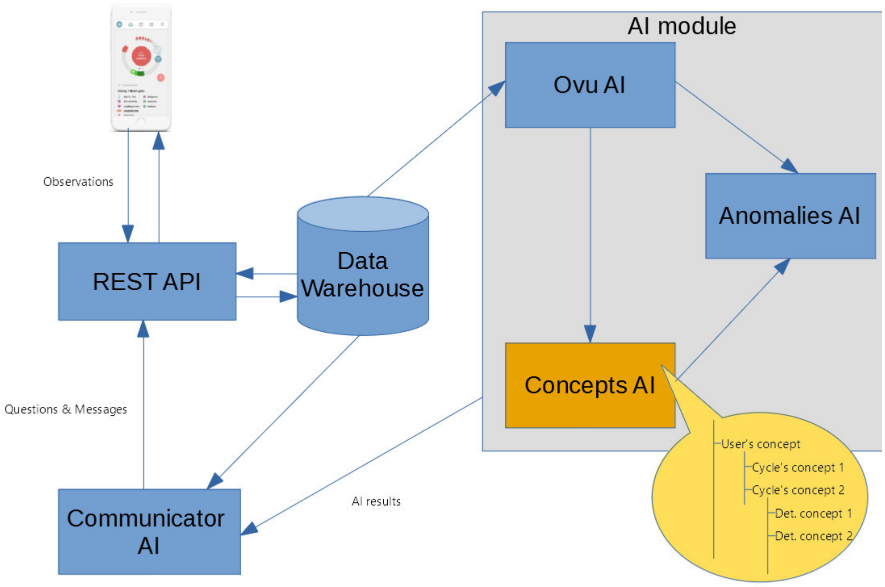


Fig. 1. General scheme of the architecture of the AI module linked with other interfaces inside the OvuFriend's platform

descriptions understandable to women who are the users of the system.

B. Goal description

The issue of cycle description applies to both open (ongoing) menstrual cycles as well as those that are already closed (past cycles). The algorithm provides, for both types of cycles, an accessible and automatically generated description of the given menstrual cycle, that pays attention to a number of features that may be medically important. This description should include local aspects of the cycle, as well as a slightly broader perspective of the whole cycle. The description is to contain the initial classification of the values of the features, compared with the applicable standards. The last aspect is the analysis in a wider time window that captures trends and tendencies arising from the repetition of certain phenomena in a defined historical window. Calculating approximated results based on meta-descriptions and summaries are widely applied in many areas of interest, such as analytical databases [13], large relational data sets [12], redesigning and accelerating machine learning algorithms [2] or systems for monitoring health conditions for members of nursing homes [5]. A slightly different approach is to use Japanese candles as summaries [9], and then compute and process that data. One example is the annual AI competition, which this year is based on such summaries [6].

C. The algorithm for automatic cycle description

The cycle description algorithm analyzes and automatically labels the menstrual cycles of women who are users of the OvuFriend's platform. The analysis takes into account various aspects important from the point of view of confirming the correctness of the entered data or probability of the occurrence of certain specific symptoms associated with medical anomalies. Finally, linguistic summaries describing the given cycle are generated and, optionally, descriptions concerning the user herself are generated (inter-cycle analysis). Generated labels are processed into natural language (messages in a language understood by the end user). This description is intended to facilitate the understanding and interpretation of the user cycle as well as to improve the quality of the data entered. In addition, it is a source of input data in the Anomalies AI module included in Fig. 1, dealing with the analysis of the possibility of medical anomalies. Generation of descriptions is a fuzzy linguistic summary by using a number of quantifiers in accordance with the techniques described in the works [8]. The basis for generating linguistic summaries are label collections, but in order to correctly present the final text, various summary generation techniques are used, ranging from simple static ones, to dynamic ones and ones that aggregate other variables.

D. Layout of the paper

This article consists of five sections. The second section presents the processed data that together constitute the compound object. This description will allow the reader to understand the complexity and relationship of the individual elements making up the representation of the menstrual cycle in the system. The third section presents the methods used to build the solution, the formal foundations and the definitions of individual components of the solution. The fourth section describes the method of evaluating the correctness of the solution and the results achieved. The last section presents the discussion of the results and the plan of further work on the issue.

II. Menstrual cycle's representation

The central object of interest is the menstrual cycle described by the ensembles of time series inter-correlated with each other, constructed from observations taken by women. The individual component time series are indexed with the same time quanta representing particular days of the cycle. Depending on the cycle and the woman's behavior, there are many possible combinations of data types that constitute this multivariate time series [4]. Time series are typically associated with financial applications [11] but in this case we are dealing with a specific case of a cyclic time series, where a single cycle in multivariate version is considered. The set of features contains: BBT, cervical mucus, cervix parameters, LH urinary tests, pregnancy tests, statistics and occurrence of user-specific symptoms that may signify approaching ovulation.

BBT data consists of temperature values. The measurements are compared to the mean temperature of the previous 6 days. At the same time other factors are computed (eg. mean, relative difference, etc.) and stored together with the BBT time series for later processing.

Cervical mucus is defined by one of five possible values taken from the enumerative scale: dry, sticky, creamy, watery, stretchy. Each value describes different state of the mucus. Making use of this parameter requires detection of patterns in its variability. Therefore it is not enough to get a single measurement. The data should be collected day by day in a certain range.

Cervix has three parameters that can be tracked: opening, position and texture. Each of them has three values respectively: {open, medium, closed}, {high, medium, low}, {soft, medium, hard}. The observations are collected independently, but the interpretation of the whole state depends on all these values combined (at least two of them). These data create an additional nested three dimensional time series that describes one feature.

Ovulation test has a binary value: positive or negative. However there are some difficulties with interpreting its result which sometimes leads to wrong classification as one of these two states on part of the user. In this type of data a series of measurements is also required, in particular one containing a transition from negative to positive values. A single positive measurement is often not enough to accurately determine ovulation day.

Pregnancy test also has binary positive and negative values. If a woman got pregnant during the cycle, the pregnancy test will come out positive, but only if it was taken an appropriate amount of time after the ovulation. Thus both positive and negative values of pregnancy test in such a cycle may convey some information on the date of the ovulation.

Statistics are useful, because the length of the luteal phase is expected to be constant across a given woman's menstrual cycles. Simple statistical data particular to the user are used: average cycle and luteal phase lengths, as well as typical values based on clustering concerning luteal phase, cycle length, ovulation days, etc.

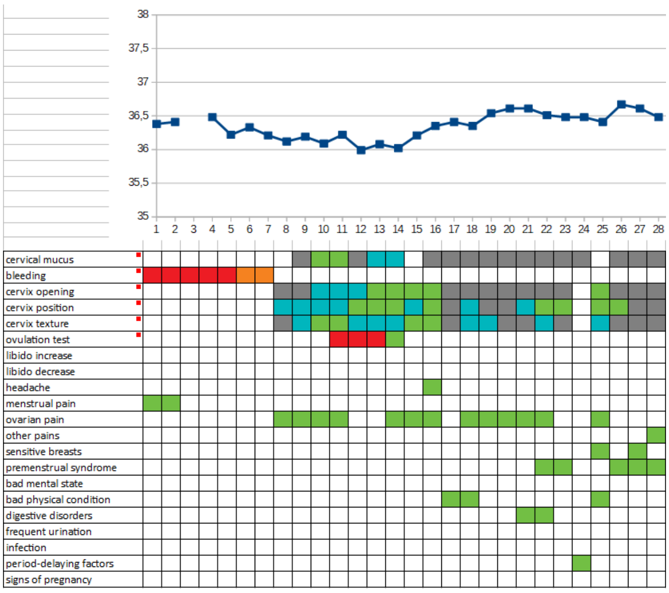


Fig. 2. Multivariate time series of menstrual cycle

Symptoms are the most complex feature in terms of stored information. It consists of more than 80 elements which describe symptoms (e.g. various pains, mental states, infections, libido, etc.) on a single day of the cycle. Most of them are binary, but together they create a complex structure. Elementary symptoms are granulated and combined into groups of similar elements.

A representation of a single menstrual cycle can be any combination of these data. Moreover, each of the time series independently may require handling of missing values and of imprecision of processed values [8]. An important element is a correlation of the particular sub-time-series. Thus the need arises to create a representation whose values are determined by the mutual influence of the individual parts of the multivariate time series.

An example of a graphical interpretation of this multivariate time series representing the menstrual cycle is shown in Fig. 2.

Such combined time series for each cycle constitutes a *compound object* described by various features and consisting of many sub-objects in the sense of the definition in [14].

III. Methods used

A. Ontology of concepts

The cycles are described through the lenses of concepts defined at three levels of the hierarchy. The lowest level of the hierarchy concerns concepts assigned at the level of a single dimension of the particular time series of the input object, e.g. the menstrual cycle described by the given data type (mucus, cervix, BBT, etc.). The second level aggregates the previous one and concerns observations for the whole cycle. On the third - highest - level of the hierarchy are the concepts obtained as a result of the analysis of user's historical cycles in a fixed time window. The concepts are connected by relationships among themselves, which generally falls under the definition of ontology.

The name of the *ontology* comes from philosophy, but now it is also frequently found in the field of artificial intelligence (AI). The formal definition (one of many) was introduced in 2001, described in [16]. Its meaning is as follows: ontology is a system marked as $O = \{C, R, H_c, rel, A, L\}$, which specifies the structure of concepts, relationships between them as well as theory defined on a model, where: C is the set of all concepts of the model and the concept is called the idea of representing a group of objects with common characteristics. R is a set of non-taxonomic relations defined as named connections between concepts [16], H_c - a collection of taxonomic relationships between concepts, rel - defined non-taxonomic relationships between the concepts, A - a set of axioms, L - lexicon defining the meaning of concepts (including relations). L is a set of the form $\{L_c, L_r, F, G\}$, where L_c - lexicon of definitions for concepts, L_r - lexicon of definitions of a set of relationships, F - references to concepts, G - references to relationship.

There are many interesting applications of ontology described in the literature covering many fields, e.g. pattern recognition, image analysis or modelling situational awareness by AI systems [17]. In all these cases ontology is a tool for modelling the structure of concepts and relationships describing a selected part of the local context in which the system is described [18].

In the simplest sense, ontology is a set of concepts connected one with another through named relationships. Ontological concepts can create hierarchies by grouping more specific concepts into more general entities. This form is used, for example, to model mereologic relations, which describe dependencies between parts of objects [10].

In the context of this work, ontology is used as a set of concepts describing menstrual cycles with its structure and relations. It is used for preparing meta-representation of the object, ready to further processing, e.g. comparing each other, clustering or generating human readable linguistic descriptions.

A fragment of the ontology is presented on Fig. 3 as an example. It shows in particular how concepts of higher levels are obtained from the concepts of lower levels using the algebra of labels.

B. Overview of the designed solution

First, the designed algorithm designates labels for particular detectors, the cycle and the user with terms corresponding to various ontological concepts, and then it generates the cycle description in natural language. The concepts are organized in a three-level hierarchy and processing consists of three steps. At the first level there are simple atomic concepts regarding directly the aspects of the cycle related to the parameters analyzed at the level of a given data type (one dimension of the cycle time series). Higher levels of the ontological concepts are built on the basis of atomic concepts. First, the concepts of a single cycle are built - this is the second level of the hierarchy. These concepts constitute knowledge of the cycle using calculations based on labels from the previous level. The comprehensive description of a single cycle obtained in this way allows defining the concepts of the third level of the hierarchy regarding the behavior and condition of a woman in time. These concepts are defined based on the occurrence of individual concepts related to the cycle in user cycles in a historical time window. They allow detecting the persistent features of cycles that characterize a woman (her condition and the functioning of the biological mechanisms).

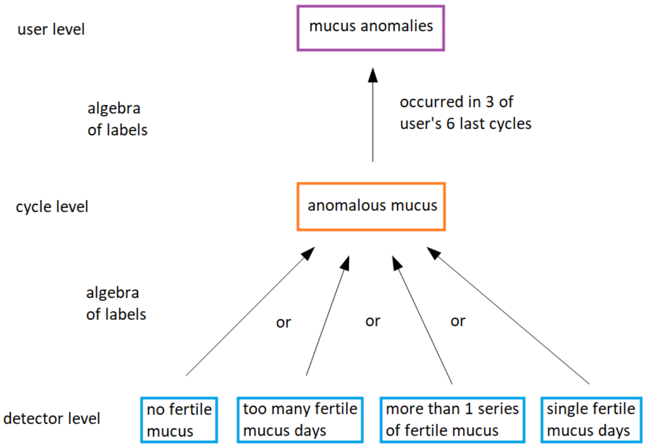

user level
mucus anomalies
algebra of labels
occurred in 3 of user's 6 last cycles
cycle level
anomalous mucus
algebra of labels
or
or
or
detector level
no fertile mucus
too many fertile mucus days
more than 1 series of fertile mucus
single fertile mucus days

Fig. 3. A fragment of used ontology together with an example of using the algebra of labels to obtain higher level concepts.

Concepts are defined in natural language as various features that cycles may possess. This set was developed by medical experts who identified interesting aspects in the cycle that should be monitored. Then these concepts were defined in the form of predicates that are verified during processing of the data. If the predicate is satisfied, the object (cycle, detector representing the given data type or woman herself) is assigned a label. If condition is not met, the label does not appear in the context. Due to the fact that labels address both positive (normative) features and anomalies (non-normative), in both cases certain subset of labels will be allocated.

The rest of the section provides ontological definitions of concepts at individual levels of the hierarchy and some details of the associated conditions that have to be met in order for a concept label to be assigned. The descriptions given for these conditions are general and short, they don't go into the details. Many of the conditions are fuzzy - they can be satisfied to a degree. The label then is assigned to that same degree. As an example consider the label *temperature jump detected*. The occurrence of the temperature jump is decided in a fuzzy way - a clear-cut jump is recorded if the temperature raises at least 0.2°C over the mean of the last 6 days. But the definition is extended in the fuzzy way to accommodate raises of even 0.18°C. If the temperature jump is satisfied to a certain degree, then the label *temperature jump detected* is considered to have such degree of being assigned.

Labels corresponding to the lowest level concepts are set by the detectors from the ovulation detection algorithm, because the same set of input data is used to determine them. Some labels are common to all detectors, and some dedicated to specific detectors. Below, the lowest level concepts are listed, divided by the type of data for which the label is issued.

1) Concepts of level 1 - data types: Common concepts (labels) appearing in all types of analyzed data at the first level of the hierarchy:

- *No data* - the analyzed data type has not been entered at all or the boundary conditions for the occurrence of a given data type have not been met (for different types, the boundary conditions specify the minimum amount of data necessary to perform calculations).
- *Few measurements* - the measurements analysis showed that the occurrence of data in critical areas of the cycle is low. The label is set if the amount of data entered is relatively low according to the specifications of the given detector (unit to process data of a given type in ovulation detection).
- *Cycle irregular* - the cycle shows a deviation from the standard pattern in terms of the parameter being processed (type of data). Label is set if the data entered deviate from the standard according to the specifications of the given detector.
- *No ovulation detected* - measurement analysis did not lead to detection of ovulation. The label is set if the detector has not determined ovulation by exceeding the activation threshold. Assigning this label does not mean that the cycle is anovulatory.

Concepts dedicated to temperature analysis:

- *Not double phased* - the cycle is not divided into a lower temperature phase and a higher temperature phase. Label is set if the temperature detector does not detect a temperature jump that has been confirmed.
- *Temperature fluctuation* - temperature fluctuations occur in the cycle. Label is set if the temperature detector has discovered at least two temperature jumps that didn't persist.
- *Imprecise thermometer* - the thermometer used for measurements has low precision - it measures the temperature only to one decimal place. Label is set, if no temperatures entered by the user have a non-zero value in second place after the decimal point, while the minimum requirement for the number of is attained.
- *Jump detected* - A temperature jump has been detected in the cycle. Label is set if the temperature jump is detected and there is no confirmed temperature jump yet.
- *Confirmed jump detected* - there is a confirmed temperature jump in the cycle.

Concepts dedicated to cervical mucus analysis:

- *No fertile mucus* - there are no days in the cycle when the mucus is fertile (stretchy or watery). Label is set if there isn't any fertile mucus, even though the cycle is closed or its length has exceeded the upper predicted limit of fertile days.
- *Too many fertile mucus* - there are too many fertile mucus days in the cycle. The label is set if the period between the first and last day of fertile mucus exceeds the specified threshold value.
- *Menstruation long* - the length of menstruation exceeds the established norm, but it falls within the extended norm (acceptable from medical point).
- *Menstruation too long* - the length of menstruation exceeds the extended norm.
- *Single fertile mucus* - single fertile mucus days occur in the cycle. Label is issued if there are single days of fertile mucus surrounded by days of infertile mucus.
- *Fertile mucus series occurred* - a series of fertile mucus appeared in the cycle. The label appears if in the cycle there were two days with fertile mucus next to each other or separated at most by one infertile day.
- *Fertile mucus series finished* - label is set, if after a fertile series there were at least two infertile days (or no data).
- *More than 1 mucus series* - label is set, if more than one series occurred in mucus data. The label means the irregularity in the cervical mucus data.
- *Vaginal infection* - vaginal infection appeared in the cycle. The label is set if at least one day has appeared in the cycle with vaginal infection.

Concepts dedicated to cervix analysis:

- *No fertile cervix* - there are no days in the cycle when the cervix is in a fertile phase, even though the cycle is closed or its length has exceeded the upper forecasted limit of fertile days.
- *Too many fertile cervix days* - there are too many days in the cycle when the cervix is fertile.
- *Single fertile cervix after series* - there are single days in the cycle when the cervix is in the fertile phase surrounded by days when the cervix is in the infertile phase.
- *Fertile cervix series occurred* - a series of fertile cervix appeared in the cycle. Label is issued if two consecutive days appeared in the cycle indicating fertile cervix parameters.
- *Fertile cervix series finished* - the series of fertile cervix indications ended. Label is issued if after a fertile series there were at least two days of infertile cervix (or no data).

Concepts dedicated to ovulation test analysis:

- *No positive ovulation test* - there are no days in the cycle when the ovulation test is positive, even though the cycle is closed or its length has exceeded the predicted limit of fertile days.
- *Too many days of positive tests* - there are too many days in the cycle when the test is positive. Label is set if the period between the first and last day when the test is positive exceeds the specified threshold.

- *First positive ovulation test* - a positive ovulation test appeared in the cycle.
- *Series of positive tests finished* - the first negative test appeared after positive tests.
- *LH hormone irregular* - generated pattern from ovulation tests results indicates irregularity.

Concepts dedicated to the ovulation monitor analysis:

- *Measurements started too late* - measurements were performed in a given cycle, but contrary to the instructions of the ovulation monitor, they were started too late (after the 10'th day of the cycle).

Concepts dedicated to symptoms analysis:

- *A lot of pain* - there are a large number of days with marked pain symptoms in the cycle. The label is issued if pain symptoms occur in the percentage of days in the cycle exceeding a certain threshold.
- *Positive pregnancy test* - a reliable positive pregnancy test appeared in the cycle.
- *Menstruation phase* - the cycle is currently in the menstrual phase. Label is issued while current cycle day is in the range of the bleeding series which started at the beginning of the cycle.
- *Follicular phase* - phase of the cycle after the end of menstruation, but before ovulation, determined for ongoing cycles. The label is issued when the menstruation is over and ovulation symptoms have not yet occurred.
- *Ovulation phase* - concept assigned usually for one day. Label is issued if the current day of the cycle coincides with the ovulation forecast.
- *Luteal phase* - phase after ovulation in the cycle. Label is issued for open cycles with designated ovulation.

Concepts dedicated to the user's history analysis:

- *No personal reference set* - there are no historical cycles of the user that are completed, ovulatory and the credibility of ovulation is at least on a certain level defined with a parameter.
- *Small personal reference set* - there are only few historical cycles of the user that meet the criteria of being closed, ovulatory and of an appropriate level of reliability of the determined ovulation.

Concepts dedicated to the history of the user's profile analysis:

- *No profile reference set* - there are no historical cycles of users from the user's profile that meet the criteria of closure, ovulatory and the required value of reliability for determining ovulation.
- *Small profile reference set* - there are only few historical cycles from user's profile that meet the condition of being closed, ovulatory and of an appropriate level of reliability of the determined ovulation.

2) Concepts of level 2 - cycles: Based on level 1 concepts (section III-B1), more general concepts are built at the level of the entire cycle. The analysis confronts the occurrence of premises in various types of data. In this way, more and more general knowledge is obtained based on the processing of individual levels of concepts. The construction of generalized concepts is performed using the so-called *algebra of labels*, e.g. the mechanism of constructing higher-level concepts based on the presence of specific lower-level concepts using logical operations (alternative, conjunction, etc). Concepts of cycles are assigning only for already closed cycle. The definitions are given below along with some details required for calculating conditions.

- *No data* - no measurements of any type have been entered in the cycle. Label is issued if each detector has assigned a label *No data* in the first level.
- *Few measurements* - few measurements have been entered to indicate ovulation reliably. Label is issued if each detector has assigned *Few measurements* or *No data* labels, and at least one has assigned the label *Few measurements*.
- *Anovulatory* - the multivariate analysis of the input time series did not show behavior characteristic for particular types of given data, or the indications were completely divergent. On this basis, it is assumed that such a cycle is anovulatory. The label is issued if the aggregation of the responses of individual detectors did not determine the day of ovulation with a certainty exceeding the learned threshold and the labels *No data* and *Few measurements* are not set.
- *Luteal phase too long* - the luteal phase exceeds the standard length and falls outside the extended norm. The label is issued if ovulation with certainty exceeding the learned threshold has been determined and the obtained luteal phase is longer than the specified value.
- *Luteal phase long* - the luteal phase exceeds the standard length but is within the enlarged norm and the ovulation has been determined with credibility exceeding the learned threshold.
- *Luteal phase ok* - the luteal phase is normal. The label is issued if ovulation credibility exceeds the learned threshold and the obtained luteal phase is 12-16 days long.
- *Luteal phase short* - the luteal phase is shorter than the specified norm but falls within the enlarged norm and ovulation was determined with credibility exceeding the learned threshold.
- *Luteal phase too short* - the luteal phase is shorter than the specified norm and does not fall within the increased norm, at the same time the ovulation was determined with a credibility exceeding the learned threshold.
- *Biochemical pregnancy* - the possibility of biochemical pregnancy occurred. The label is issued if ovulation has been designated without the luteal filter and at an appropriate interval from ovulation a pregnancy test was performed with a positive result, but menstruation occurred no later than 46 days after the start of the cycle.
- *Intermenstrual bleeding* - bleeding occurs within the cycle. Label is set if bleeding occurs after the menstruation, but not in the vicinity of ovulation (which is normal).

- *Anomalous temperature* - time series analysis of temperature shows deviations from the defined double phased pattern. Label is issued if the temperature detector has assigned the labels *Irregular cycle* or *No double phase* or *Temperature fluctuation*.
- *Anomalous mucus* - mucus time series analysis shows deviations from the defined pattern of changes in cervical mucus.
- *Anomalous cervix* - analysis of the time series of cervix parameters shows deviations from the defined pattern of changes in cervix parameters.
- *Anomalous hormones* - time series analysis of ovulation test results or ovulation monitor measurements shows deviations from the defined pattern of changes in the test results.
- *Anomalous symptoms* - prolonged persistence of individual symptoms.
- *Sufficient variety of data* - the variety of entered measurements is sufficient - one can try determine ovulation in a reliable way.
- *Good variety of data* - the variety of entered measurements is good - one can try determine ovulation in a reliable way.
- *Intercourse on ovulation day* - there was an intercourse on the ovulation day.
- *No intercourse in ovulation day* - there was no intercourse on the ovulation day.
- *Intercourse in the area of ovulation* - the woman had an intercourse close enough to ovulation to have a chance of pregnancy.
- *No intercourse in the area of ovulation* - the woman didn't have an intercourse close enough to ovulation for a chance of pregnancy.
- *The X parameter and the Y parameter do not match* - There is a mismatch between the given parameters in terms of ovulation indications.
- *The X parameter does not match the rest of the parameters* - a given parameter deviates from the rest of the parameters in terms of ovulation indications.
- *PCOS symptoms* - symptoms characteristic of PCOS were detected in the cycle.
- *Endometriosis symptoms* - symptoms characteristic of endometriosis were detected in the cycle.
- *Thyroid disease symptoms* - symptoms characteristic of thyroid disease were detected in the cycle.
- *Hyperprolactinaemia symptoms* - symptoms characteristic of hyperprolactinaemia were detected in the cycle.

3) Concepts of level 3 - woman's health: The third level of the labels hierarchy is based on the previous two levels. Top-level labels - for a woman (user of the platform) - are assigned based on the labels of her latest cycles history. The history is considered in the fixed length window, controlled by the parameter e.g. 3 months, 6 months, etc.

The most generalized concepts (labels) are defined as follows:

- *Short cycles* - in the history of the user analyzed in a fixed time window, most of the cycles have length below the established norm.
- *Long cycles* - in the history of the user analyzed in a fixed time window, most of the cycles have length above the established norm.
- *Short luteal phases* - in the history of the user analyzed in a fixed time window, most cycles have a luteal phase length below the established norm.
- *Long luteal phases* - in the history of the user analyzed in a fixed time window, most cycles have a luteal phase length above the established norm.
- *Chronic anovulation* - in the history of the user analyzed in a fixed time window, most of the cycles have the characteristics of an anovulatory cycle.
- *Temperature anomalies* - in the history of the user analyzed in a fixed time window, most of the cycles have temperature anomalies.
- *Mucus anomalies* - in the user's history analyzed in a set time window, most cycles have mucus anomalies.
- *Cervix anomalies* - in the user's history analyzed in a set time window, most cycles have cervix anomalies.
- *Hormonal anomalies* - in the user's history analyzed in a set time window, most cycles have hormonal anomalies.
- *Symptom anomalies* - in the history of the user analyzed in a fixed time window, most cycles have symptom anomalies.
- *Menstruations long* - in the user's history analyzed in a fixed time window, most cycles have menstruation length exceeding the norm.
- *Chronic pain* - in the user's history analyzed in a fixed time window, most cycles have prolonged periods of pain symptoms.
- *Does not enter parameter X* - in the history of the user analyzed in a fixed time window, in most cycles, the user did not enter data on a given parameter.
- *Insufficiently enters parameter X* - in the history of the user analyzed in a fixed time window, in most cycles, the user did not enter enough data of a given parameter to be able to reliably indicate ovulation.
- *No measurements* - in the history of the user analyzed in the fixed time window, in most cycles, the user did not enter any relevant data that would allow calculation of ovulation detection.
- *Insufficient measurements* - in the history of the user analyzed in a fixed time window, in most cycles, the user did not enter enough data to be able to reliably indicate ovulation.
- *Incorrectly measures the ovulation monitor* - in the history of the user analyzed in the fixed time window, in most cycles the user performed ovulation monitor measurements contrary to the instructions in the manual.
- *Possible PCOS* - in the history of the user analyzed in a fixed time window, most cycles have symptoms typical for PCOS.
- *Possible endometriosis* - in the user history analyzed in

a fixed time window, most cycles have symptoms typical for endometriosis.
- *Possible thyroid disease* - in the user history analyzed in a fixed time window, most cycles have symptoms typical for thyroid disease.
- *Possible hyperprolactinaemia* - in the user history analyzed in a fixed time window, most cycles have symptoms typical for hyperprolactinaemia.

The set of labels obtained in this way is a linguistic summary. Operators (quantifiers) analyzing a certain window of woman's historical cycles and counting occurrences are used to designate individual labels. This summary is fuzzy, because the predicates used to assign the labels are based on fuzzy rules. It is a very important that higher levels perform operations on labels (linguistic summaries) of lower levels, so the higher in the hierarchy of concepts, the higher level of generalization is obtained. A very important feature is the decomposability of labels, which provides the functionality of selecting subsets of cycles to be searched using more advanced methods, having a designated top-level label. The user's level label covers a certain group of cycles, and these cycles consist of an even larger set of decomposed 1-dimensional time series corresponding to a given data type (processed by specific ovulation detectors). In other cases, such an aggregated linguistic description is sufficient to perform some calculations. An example can be the summary *Short cycles*, from which you can easily conclude that most cycles are shorter than the assumed norm. This information can be used to predict the length for a new cycle. In the presented algorithm, the processing does not end at this point, another step is introduced to generate descriptions in natural language.

C. *Fuzzy linguistic summaries of menstrual cycles*

After determining the labels, each cycle is described by a set of concepts at individual levels of the hierarchy. These sets of information allow for generating of natural language description that can easily be understood by a woman trying to conceive. The next stage of the algorithm for generating linguistic summaries is based on dynamic templates responsible for defining the sets of possible options used during the generation. The template is responsible for the structure of the description, elements taken into account, their order and the information scope of the researched summaries. These templates can be built from several types of summaries. The simplest form is singleton summaries, which are responsible for mapping the label to a sentence in natural language. This corresponds to the *exists* quantifier for simple fuzzy linguistic summaries [7]. If the label appears in the cycle or user representation set then a simple summary related to this information will be generated. These are relatively simple summaries that do not take into account interrelationships and context. There can be any number of the singleton summaries in the template and they can be arranged in any order.

The second type of summaries used in the template are the so-called aggregated summaries. These summaries concern the occurrence of a given feature for many elements simultaneously. Using singleton summaries in this case would create an unnatural text with repetitions. So, it is better to use an aggregated summary that has combining capabilities, e.g. the main part of the summary states that specific premises are met, and the second specifies which objects or data it concerns. These summaries use the method of grouping and enumerating values. If a given label exists in multiple data types, then one summary sentence will be produced covering the different data types. This makes the text more user-friendly.

The third type of summaries used in the described solution are generalizing summaries. They use quantifiers such as:

- for all - requires fulfillment of fuzzy predicate for all elements,
- exist - requires at least one elements which fulfills fuzzy predicate,
- most - requires fulfillment of fuzzy predicate for a majority of elements,
- at least two - requires fulfillment of fuzzy predicate for two or more elements,
- at least three - requires fulfillment of fuzzy predicate for three or more elements,
- almost majority - requires fulfillment of fuzzy predicate for number of elements which is nearly a majority.

These quantifiers are able to count the occurrences of appropriate labels and evaluate them in relation to each other (depending on the number of data types present, e.g. the *most* operator refers to the data types defined in a given menstrual cycle and not all possible ones). Depending on the fulfillment of individual quantifiers (it is worth noting here that the conditions may be met for more than one), a different form of summary can be returned. Such a summary, using different quantifiers, can be defined in the template in the form of an alternative or a conjunction. In the first case, the order in which the components are set controls the order in which the conditions are checked. So, the first satisfied summary in this type of alternative is returned. If the conjunction is used, each of the conditions generated by the quantifiers must be met.

The description template is therefore any combination of the summaries of these three types. If the data for a given summary does not meet the condition given by using the appropriate quantifier, it does not return any value. Despite appearing in the template, it does not interfere with the generation of text in natural language for the cycle or the user.

By using three types of summaries and combining conditions using conjunctions or alternatives, it is possible to define very complex label-based schemes that generate various linguistic summaries that stylistically differ very much, depending on the input data, despite using the same description template.

D. *Illustrative example*

Fig. 4 presents an example of a real menstrual cycle coded as a multivariate time series. Summaries generated for the level 1 of the hierarchy of the ontology are shown in Table I.

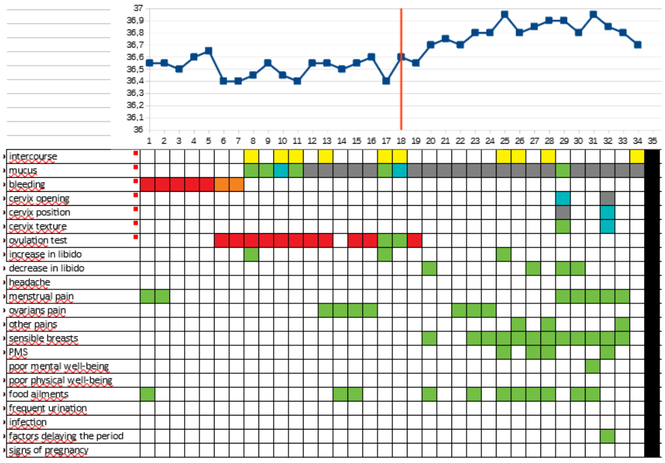


Fig. 4. Example of a real menstrual cycle retrieved from OvuFriend's platform

TABLE I
LINGUISTIC SUMMARY OF THE MENSTRUAL CYCLE BASED ON THE LEVEL 1 OF THE HIERARCHY OF THE ONTOLOGY.

Data type	Label
Ovulation test	series of positive tests finished
Ovulation monitor	no data
Cervix	few measurements
Cervix	fertile cervix series finished
Mucus	cycle irregular
Mucus	more than 1 mucus series
Mucus	fertile mucus series finished
Mucus	single fertile mucus
BBT	cycle irregular
BBT	confirmed jump detected

Next, the *algebra of labels* provided higher level of summaries, which are presented in Table II. In this simple example there is no historical cycles, so the third level of summaries cannot be designated. Even though using the pattern of linguistic description one can generate a human readable text, which for this particular cycle will be in the following form: *Cycle is ovulatory. The cycle length is normal. The length of the luteal phase is normal. The provided data is of good variety. The cervical parameters differ from the rest of the data entered. Make sure that you measure it correctly. There are anomalies in cervical mucus and temperature. Intercourse around ovulation has been observed, which gives a chance of getting pregnant.*

TABLE II
LINGUISTIC SUMMARY OF THE MENSTRUAL CYCLE BASED ON THE LEVEL 2 OF THE HIERARCHY OF THE ONTOLOGY.

ovulatory cycle
cycle length Ok
good data variety
cervix doesn't agree with the rest
anomalous mucus
cycle irregular
intercourse in fertile period
anomalous temperature
luteal phase ok

TABLE III
EVALUATION OF FUZZY LINGUISTIC SUMMARY GENERATION IN THE FORM OF LABELS FOR MENSTRUAL CYCLES. EVALUATION PERFORMED ON A SUBSET OF LEVEL 1 LABELS THAT MOST INFLUENCE PREGNANCY. EVALUATION MADE ON 100 CYCLES TAGGED BY MEDICAL EXPERTS. WHOLE SET WAS A TESTING SET.

Label	TP	TN	FP	FN	Pr	Rec	F1	Acc
Cycle too short	3	97	0	0	1	1	1	1
Cycle short	10	90	0	0	1	1	1	1
Cycle length ok	72	28	0	0	1	1	1	1
Cycle long	8	92	0	0	1	1	1	1
Cycle too long	7	93	0	0	1	1	1	1
Ovulation cycle	72	21	2	5	0.97	0.94	0.95	0.93
No double phase	20	74	4	2	0.83	0.91	0.87	0.94
Menstruation too short	5	95	0	0	1	1	1	1
Menstruation long	8	91	0	1	1	0.89	0.94	0.99
Menstruation too long	4	96	0	0	1	1	1	1
Intermenstrual bleeding	21	77	1	1	0.95	0.95	0.95	0.98
No fertile mucus	5	94	1	0	0.83	1	0.90	0.99
Too many fertile mucus	5	94	0	1	1	0.83	0.91	0.99
Mucus more than 1 series	15	83	1	1	0.94	0.94	0.94	0.98
Single fertile mucus	12	85	1	2	0.92	0.86	0.89	0.97
No fertile cervix	4	95	1	0	0.80	1	0.89	0.99
Too many fertile cervix	39	61	0	0	1	1	1	1
Cervix more than 1 series	19	78	2	1	0.90	0.95	0.92	0.97
Single cervix days	5	95	0	0	1	1	1	1
No positive o. t.	6	93	0	1	1	0.86	0.92	0.99
Too many positives o. t.	6	93	1	0	0.86	1	0.92	0.99
O. t. more than 1 series	7	93	0	0	1	1	1	1
Mucus irregularity	32	62	1	5	0.97	0.86	0.91	0.94
Cervix irregularity	48	48	2	2	0.96	0.96	0.96	0.96
LH hormone irregular	18	79	2	1	0.90	0.95	0.92	0.97
Averaged	451	2007	19	23	0.96	0.95	0.96	0.98

IV. EVALUATION AND RESULTS

As part of the project, medical experts selected a set of labels at the lowest level. The designation of these labels was carried out by the described algorithm. The development of the algorithm and its initial fine-tuning was developed on a set of 200 cycles tagged by medical experts. This set has been treated as a learning set (whole), although in this case it is not a classical learning mechanism with feedback. The algorithm has been tuned for operation on the tagged set and pre-validated in terms of content-related correctness of operation. The next step was to draw a new set of cycles having an empty intersection with the previous one and return the set for tagging. In this case, the collection of labels has been limited to those most related to pregnancy, that is, among others, labels directly or indirectly related to the anomalies. Although in the classic approach to learning, the training set is usually smaller than the testing set, here, however, due to the non-typical nature of the process, the opposite proportions were used (66% training set, 34% testing set, respectively). Therefore, the results of the experiment are given for a set of 100 menstrual cycles, which were tagged in a second attempt by medical experts to evaluate the quality of the automatic cycle description algorithm. The other higher-level labels are derivative concepts, well defined by the *algebra of labels* mentioned above, therefore they were not evaluated in this experiment. The table III presents the results of the experiment dividing into individual labels, as well as the summary efficiency calculated based on the sum of the contingency tables for individual labels. After the calculation of the summary table, the Precision, Recall, F1Score and Accuracy evaluation measures were calculated.

The obtained results are very good. All tested labels attained the minimum requirement of at least 0.8 value on both Precision and Recall, and on most labels these values are much higher.

V. Conclusions

The described solution shows how in a relatively simple way, using fuzzy quantifiers, one can create an effective algorithm that generates linguistic summaries and patterns from complex multidimensional data. The developed algorithm can be used both to generate natural language text that describes compound objects, as well as to generate high level information used to control other processes in the system. In this approach, the processed information is largely aggregated and compressed. The need to analyze decomposed data occurs sporadically, and in most cases it is enough to control the process or even make decisions based on linguistic descriptions constructed in this way.

It is worth noting that the working compliance of the mechanism with decisions of medical experts is very high. An additional difficulty in this case was the correlation with the second algorithm and the results it achieved - the algorithm of prediction and confirmation of ovulation. Linguistic summaries were generated while performing those calculations and many of them depended on the decisions made by the ovulation algorithm. Therefore, the achieved results confirm that the operation of both algorithms is compatible with the intuition and knowledge of medical experts. The average precision at the level of 0.96 and the recall at the level of 0.95 allow to treat all generated linguistic summaries and the final generation of description in natural language with enough confidence.

Further work will focus on the development of the described algorithm at the stage of higher-level labels, so that more and more information can be deduced and processed based on the generated patterns (made of linguistic summaries). In addition, in terms of implementation work, the algorithm will be implemented on the production platform and will work in fully real conditions, which will be an extension of the current state, which was developed in conditions similar to real ones (real data but supported from a backup environment).

Acknowledgement

Co-financed by the EU Smart Growth Operational Programme 2014-2020 under the project ”Development of New World Scale Solutions in the Field of Machine Learning Supporting Family Planning and Overcoming the Infertility Problem”, POIR.01.01.01-00-0831/17-00.

References

[1] L. Bablok, W. Dziadecki, I. Szymusik, and et al., “Patterns of infertility in Poland - multicenter study,” *Neuro Endocrinol Lett.*, vol. 32, no. 6, pp. 799–804, 2011.

[2] A. Chadzynska-Krasowska, P. Betlinski, and D. Slezak, “Scalable machine learning with granulated data summaries: A case of feature selection,” in *Proceedings of ISMIS 2017, Warsaw, Poland*, ser. Lecture Notes in Computer Science, vol. 10352. Springer, 2017, pp. 519–529. [Online]. Available: https://doi.org/10.1007/978-3-319-60438-1_51

[3] W. Damian, S. Iwona, W. Miroslaw, and P. Bronislawa, “The impact of endometriosis on the quality of life and the incidence of depression-a cohort study,” *Int. J. Environ. Res Public Health*, vol. 17, no. 10, p. 3641, 2020.

[4] J. Fedorowicz, L. Sosnowski, D. Slezak, I. Szymusik, and et al., “Multivariate ovulation window detection at OvuFriend,” in *Proceedings of IJCRS 2019, Debrecen, Hungary*, ser. Lecture Notes in Computer Science, vol. 11499. Springer, 2019, pp. 395–408. [Online]. Available: https://doi.org/10.1007/978-3-030-22815-6_31

[5] A. Jain, M. Popescu, J. M. Keller, M. Rantz, and B. Markway, “Linguistic summarization of in-home sensor data,” *J. Biomed. Informatics*, vol. 96, 2019. [Online]. Available: https://doi.org/10.1016/j.jbi.2019.103247

[6] A. Janusz, M. Przyborowski, P. Biczyk, and D. Ślęzak, “Network device workload prediction: A data mining challenge at knowledge pit.” in *Proceedings FedCSIS 2020, Sofia, Bulgaria*, 2020.

[7] J. Kacprzyk and R. R. Yager, “Linguistic summaries of data using fuzzy logic,” *International Journal of General Systems*, vol. 30, no. 2, pp. 133–154, 2001. [Online]. Available: https://doi.org/10.1080/03081070108960702

[8] J. Kacprzyk and S. Zadrozny, “Fuzzy logic-based linguistic summaries of time series: a powerful tool for discovering knowledge on time varying processes and systems under imprecision,” *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 6, no. 1, pp. 37–46, 2016. [Online]. Available: https://doi.org/10.1002/widm.1175

[9] W. Kosiński and A. Chwastyk, “Ordered fuzzy numbers in financial stock and accounting problems,” in *2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS)*, 2013, pp. 546–551.

[10] L. Polkowski and P. Artiemjew, *Granular Computing in Decision Approximation - An Application of Rough Mereology*, ser. Intelligent Systems Reference Library. Springer, 2015, vol. 77. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-12880-1

[11] M. Romaniuk and P. Nowak, “Monte carlo methods : Theory, algorithms and applications to selected financial problems,” Warszawa, 2015.

[12] D. Slezak, J. Borkowski, and A. Chadzynska-Krasowska, “Ranking mutual information dependencies in a summary-based approximate analytics framework,” in *2018 International Conference on High Performance Computing & Simulation, HPCS 2018, Orleans, France, July 16-20, 2018*. IEEE, 2018, pp. 852–859. [Online]. Available: https://doi.org/10.1109/HPCS.2018.00137

[13] D. Slezak, R. Glick, P. Betlinski, and P. Synak, “A new approximate query engine based on intelligent capture and fast transformations of granulated data summaries,” *J. Intell. Inf. Syst.*, vol. 50, no. 2, pp. 385–414, 2018. [Online]. Available: https://doi.org/10.1007/s10844-017-0471-6

[14] L. Sosnowski, “Compound objects comparators in application to similarity detection and object recognition,” *Trans. Rough Sets*, vol. 21, pp. 169–300, 2019. [Online]. Available: https://doi.org/10.1007/978-3-662-58768-3_6

[15] L. Sosnowski, I. Szymusik, and T. Penza, “Network of fuzzy comparators for ovulation window prediction,” in *Proceedings of IPMU 2020*, ser. Communications in Computer and Information Science, vol. 1239. Springer, 2020, pp. 800–813. [Online]. Available: https://doi.org/10.1007/978-3-030-50153-2_59

[16] S. Staab and A. Maedche, “Knowledge Portals: Ontologies at Work,” *AI Magazine*, vol. 22, no. 2, pp. 63–75, 2001.

[17] J. Stepaniuk and A. Skowron, “Ontological framework for approximation,” in *Proceedings of RSFDGrC 2005*, ser. Lecture Notes in Computer Science, vol. 3641. Springer, 2005, pp. 718–727. [Online]. Available: https://doi.org/10.1007/11548669_74

[18] M. Świechowski and D. Ślęzak, “Introducing LogDL - Log Description Language for Insights from Complex Data,” in *Proceedings FedCSIS 2020, Sofia, Bulgaria*, 2020.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 129–136 DOI: 10.15439/2020F57 ISSN 2300-5963 ACSIS, Vol. 21

Exploration into Deep Learning Text Generation Architectures for Dense Image Captioning

Martina Toshevska*, Frosina Stojanovska*, Eftim Zdravevski*, Petre Lameski* and Sonja Gievska*
*Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University - Skopje,
North Macedonia
Email: {martina.toshevska, frosina.stojanovska, eftim.zdravevski, petre.lameski, sonja.gievska}@finki.ukim.mk

***Abstract*—Image captioning is the process of generating a textual description that best fits the image scene. It is one of the most important tasks in computer vision and natural language processing and has the potential to improve many applications in robotics, assistive technologies, storytelling, medical imaging and more. This paper aims to analyse different encoder-decoder architectures for dense image caption generation while focusing on the text generation component.**

Already trained models for image feature generation are utilized with transfer learning. These features are used for describing the regions using three different models for text generation. We propose three deep learning architectures for generating one-sentence captions of Regions of Interest (RoIs). The proposed architectures reflect several ways of integrating features from images and text. The proposed models were evaluated and compared with several metrics for natural language generation. The experimental results demonstrate that injecting image features into a decoder RNN while generating a caption word by word is the best performing architecture among the architectures explored in this paper.

I. Introduction

Describing images, also known as image captioning, is the process of generating a textual description that best explains the image scene. Automatically describing the content of an image is a problem in artificial intelligence that connects computer vision and natural language processing. The textual description is expected to represent not only the presence of objects but also the interaction between them, as well as their characteristics and relationships [1], [2], [3].

Recognizing and describing the content of images is a very important task in many applications, including assistance to people with visual impairment (e.g., for text-to-voice guidance), robotic systems, vision-based search engines, and more. For most applications, the image captioning system must give an accurate description of the scene [4]. Additionally, image captioning can be used for automated scene description and its output can be used for automated training of models for other domains, such as other assistive technologies, storytelling, medical imaging, health-care, behaviour analysis, visual surveillance, and more.

Generating caption for a given image requires a strong understanding of its content. With the rise of deep learning techniques, understanding the content of an image relies upon convolutional neural networks. Detecting objects, as well as their properties and relations, is the primary concern for caption generation. Object detection is a widespread research area which comprises of many well-performing models [5].

The problem of automatically describing images can be split into two sub-problems: understanding the content of the image, which is considered as a computer vision task, and generating text sequences, a natural language processing task. Various approaches are used in the area of object detection, and the most successful ones are based on deep learning techniques. Models like R-CNN [6], Fast R-CNN [7], Faster R-CNN [8], Mask R-CNN [9] utilize region proposal networks (RPNs) to detect objects in an image. On the other hand, the method described in [10], often referred to as VGG, named by the group that proposed it, focuses on classifying the image scene with Very Deep Convolutional Networks.

The problem of image captioning can be split into two main approaches: (1) generation of a single description of an image, and (2) describing different Regions of Interest (RoIs) from a single image, also known as dense captioning [4]. The dense image captioning describes several regions of the image that contain objects and some relations between them. Therefore, the problem is considered as a more informative strategy when describing images, but also a more difficult one.

Concerning the problem of image captioning, many researchers are using hybrid deep learning models, that is, a combination of a convolutional neural network (CNN) and a recurrent neural network (RNN). The models developed for object detection, RoI proposal, image segmentation, and related problems are achieving great performances [11]. These models are used as feature extractors of images and specific regions in the images. The main question is, can we use the already designed models as feature extractors, and then describe the regions in an image with models designed for text generation.

To answer this question we conducted experiments with three deep learning architectures for generating text captions of RoIs in the image[1]. We apply a transfer learning approach using a pre-trained object detection network from Mask R-CNN for determining RoIs and their corresponding features. The integration of features describing images and the context of previously generated text was performed using three different models for text generation. We evaluated and compared

[1] The code for this research is available at https://github.com/frosinastojanovska/image-captioning

the models using well-known evaluation metrics for natural language generation. The discussion of the performance of the models is introduced along with the evaluation results.

The rest of the paper is organized as follows. Section II reviews the relevant related work. Section III describes the utilized dataset, the proposed architecture for extraction and captioning of regions of interest (RoIs) in images and the used evaluation metrics. Next, in Section IV, we present and discuss the results of our experiments. Finally, Section V concludes the paper and identifies directions for future research.

II. Related Work

There is an abundance of models introduced in the domain of image captioning, originating from the models that generate single image caption to models that generate multiple captions for an image. The first group includes models that generate a single caption for the whole image [12], [13], [14], [15].

To describe the entire image with one sentence, the NIC approach [12] uses a CNN pre-trained for an image classification task. This method encodes images into a compact representation, followed by an LSTM network that generates a corresponding sentence. The model is trained to maximize the likelihood of the sentence for a given image, which is fed into the LSTM only once.

Attention-based models focus on a specific image part (i.e., region or object). A visual attention-based model with hard and soft attention alternatives is proposed in [13]. As the model generates each word, its attention changes to reflect the relevant parts of the image. A semantic attention-based model is proposed in [14]. This model learns to selectively attend to semantic concept proposals and fuse them into hidden states and outputs of RNNs. The selection and fusion form feedback combining a top-down approach, which starts from a gist of an image and converts it into words, and a bottom-up approach, which combines words describing various aspects of an image.

The model presented in [15] consists of object detection and localization model to extract the information of objects and their spatial relationship, and RNN with attention mechanism to generate sentences. The encoder first uses Faster R-CNN to detect objects and then applies VGG to create feature representation for detected object regions. Captions are generated with an LSTM conditioned on the attention of the detected object regions, previously generated tokens and a previous hidden state.

Recent approaches [16], [3], [17], [18] incorporate the Transformer [19] architecture instead of traditional RNNs for caption generation. The underlying architecture remains Encoder-Decoder, but the structure differs from previous CNN-RNN approaches. Faster R-CNN [8] is used as image encoder in [17], [3], ResNext [20] in [16], and a novel Image Transformer in [18]. For all methods, Transformer is applied as a decoder to generate the caption.

The second group consists of models intended for dense image captioning. These models, unlike the models described above, generate a caption for each region of an image. The DenseCap model [21] achieved exceptional results in describing image regions. It consists of convolutional and recurrent networks responsible for detecting RoIs and their vector representation, respectively. DenseCap is a convolutional localization layer based on VGG similar to the one applied in Faster R-CNN with several modifications. The localization layer identifies spatial regions of interest and extracts a fixed-sized representation from each region. The second part is an LSTM for creating descriptions.

Another approach for dense captioning is presented in [22]. It relies upon Faster R-CNN for region features extraction. This model is an improvement of the DenseCap model. The improvement is two-fold: (1) incorporate global context feature of the image, and (2) late fusion of the region features. Authors in [23] present a Multimodal RNN that uses visual-semantic alignments. This alignment method is based on a combination of a CNN that processes image regions, a bidirectional RNN that processes sentences, and a structured objective that aligns the two modalities through a multimodal embedding.

Novel approaches [24], [25] rely upon object context features for generating a caption. CAG-net [24] uses Faster R-CNN for region extraction and custom contextual feature extraction for extracting features of the target region as well as a global feature of the whole image and features of neighbouring regions. The features are then fused and fed into an LSTM network to generate region caption. Another approach presented in [25], proposes two different architectures. The first architecture, COCD, uses an LSTM to decode the object context. It is then concatenated with caption LSTM in order to generate the final description. In the second architecture, COCG, the object context is fed into caption LSTM as guidance information for generating the region description. For both architectures, the object context is obtained with gLSTM module [26] with region features, extracted with Faster R-CNN, as guidance information.

In this paper, we focus on the caption decoder of an encoder-decoder based architecture for dense captioning. We employ the Mask R-CNN module [9] for region extraction with a transfer learning approach. We explore different architectures for decoding region features into region captions.

III. Methods and Analysis

A. Dataset

In the experiments, we used the Visual Genome dataset [27], consisting of 108,077 images with 5,408,689 region descriptions. An exemplary image with three regions is shown in Fig. 1. Each image region (i.e., a RoI) is described with the following parameters: width, height, x coordinate, y coordinate, and caption. The distribution of the number of regions per image and caption length is shown in Fig. 2.

B. Feature extraction based on Mask R-CNN

The dense image captioning is the problem of generating descriptions of RoIs in an image. Therefore, the RoIs need to be extracted from the image and described with a fixed-length feature vector. This vector then is an input into another part of

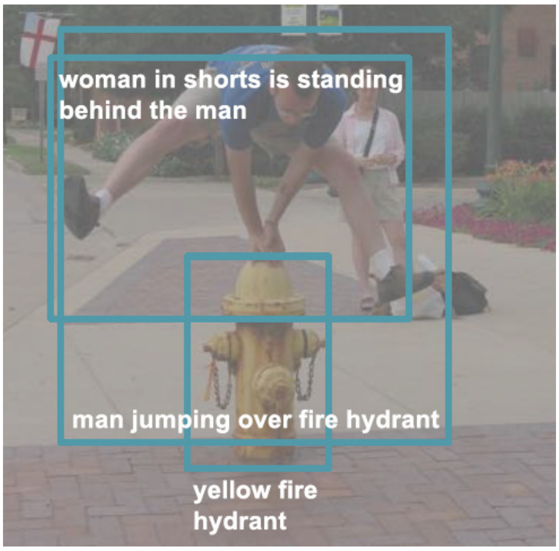

woman in shorts is standing behind the man
man jumping over fire hydrant
yellow fire hydrant

Fig. 1: Sample image, regions of interests and their corresponding captions from Visual Genome [27]

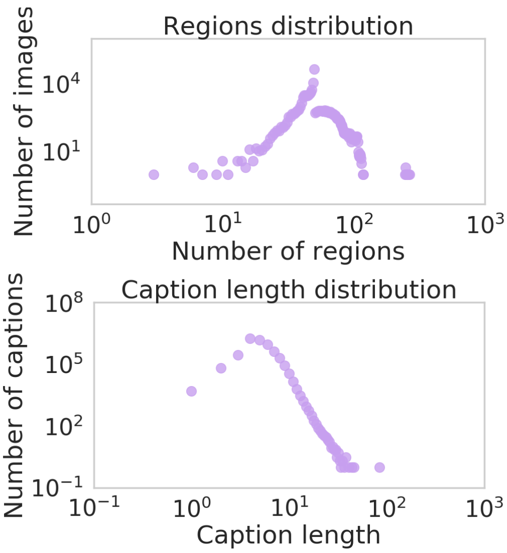


Fig. 2: Distribution of the number of regions per image (top) and caption length (bottom)

the model for text generation. There are several deep learning convolutional models for this problem. The R-CNN [6] model is improved with the next version of the Fast R-CNN [7] that facilitate feature extraction from RoIs with any dimension into a fixed-sized feature vector. Then, the Faster R-CNN [8] is the next improvement that adds a Region Proposal Network (RPN) for detection of RoIs which are fed to the Fast R-CNN model.

Mask R-CNN model [9] is a method for object detection and segmentation. It extends the Faster R-CNN [8] model by adding a new branch for mask detection and introducing RoIAlign technique. RoIAlign is a modification of the RoIPool technique [7], which extracts a feature map with quantization from each RoI. RoIAlign replaces the quantization with bilinear interpolation aligning the extracted features with the RoIs.

In this paper, to generate the RoI feature representations, we utilize the first stage of the Mask R-CNN model (RPN) and the first part of the second stage (RoIAlign). The Mask R-CNN modules for object detection and segmentation are ignored. We use transfer learning in the following way. The Mask R-CNN is pre-trained on an object detection problem, and the segmentation model was pre-trained for detecting and encoding image regions on the MS COCO dataset [28].

The input of the model are images with varying sizes. Therefore, the images are resized with a scale that ensures that the smaller dimension is at least 800 and the longer dimension is maximum 1024 pixels. We apply padding to the scaled image to fix the image dimensions to 1024×1024.

The image is processed with the ResNet feature pyramid network of the ResNet-FPN convolutional backbone architecture for feature extraction of an entire image. This bottom-up approach extracts the features of the image with five stages of the ResNet [29] architecture, which has 101 layers. Each stage is incorporated into a top-down Feature Pyramid Network (FPN) network [30], which constructs higher resolution feature maps.

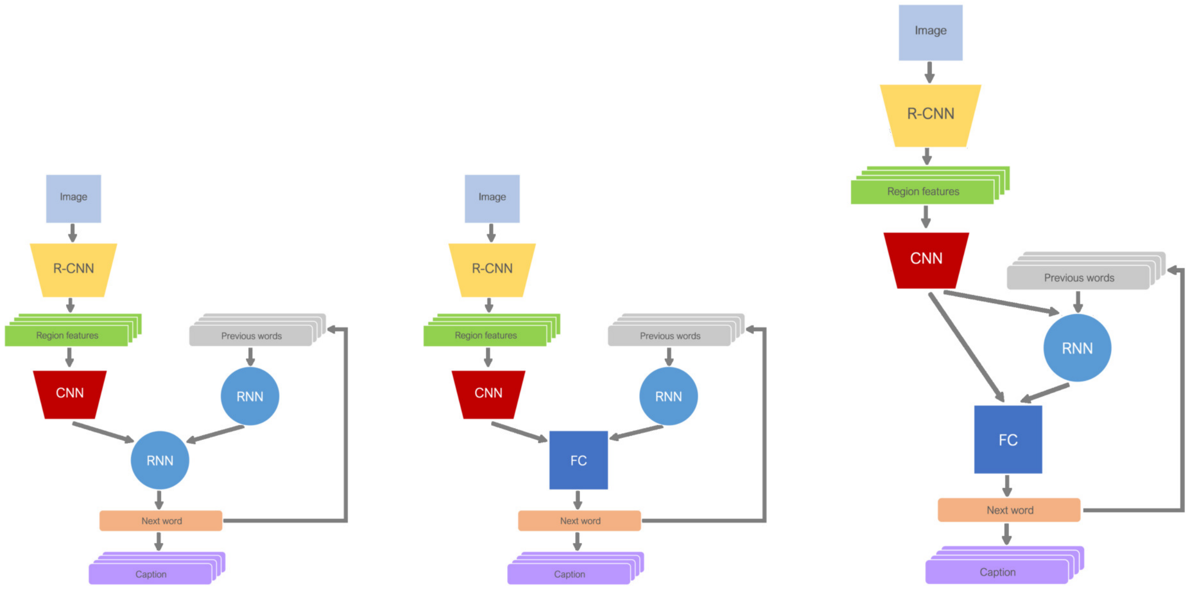
The proposed boxes, called anchors, are generated given a sliding window with proper scale and ratio. The RPN network ranks the anchors and chooses the ones that most likely contain objects. This process involves predicting foreground and background boxes from the anchors and their refinement. The output regions of the RPN are then processed with non-maximum suppression (NMS) to remove the highly overlapping regions. With an Intersection over Union (IoU) threshold of 0.7 of the NMS method, the region proposals are filtered according to their class probability of being positive (foreign) region. RoIs can be with different sizes, so the RoIAlign layer is proposed to generate small feature maps with size $7{\times}7{\times}256$ by applying bilinear interpolation. The outputs of the RoIAlign layer are the feature maps for every RoI.

C. Text generation deep learning architectures

The architectures of the proposed models are shown in Fig. 3. The application of recurrent neural networks (RNNs) in image captioning problems is discussed in [31]. An RNN can be used as either a decoder (generating words) or an encoder (encoding preceding words). In the proposed architectures, we utilize RNNs in both ways, as described below.

All three proposed architectures start similarly, by feeding the image through the R-CNN network that we reuse from the Mask R-CNN model (the yellow block named R-CNN in Fig. 3). This network creates features for each RoI of an image provided at the input. In Fig. 3, the FC blocks denote feed-forward networks which are represented by fully-connected dense networks, as described in the following text.

1) Inject Model (M1): Fig. 3a presents the diagram of our first model, M1. First, as mentioned earlier, the image is fed through the R-CNN network that we reuse from the Mask R-CNN model (the yellow block named R-CNN on



(a) Inject model - M1

(b) Merge model - M2

(c) Hybrid model - M3

Fig. 3: Diagrams of the proposed model architectures

Fig. 3a), thus generating features for each RoI of an image. A convolutional neural network (CNN) then processes these features and encodes each RoI. In parallel, a recurrent neural network (RNN) encodes the previous words (the blue circle in Fig. 3a). The features representing both the RoI of the image and the previous word embeddings (i.e., features) are fed into a decoder RNN (the second blue circle in Fig. 3a) that decodes the next word. Because the image features are injected into this RNN, we denote this model as the inject version. In our evaluations of this architecture, the decoder RNN is an LSTM with 256 units. The caption is created word by word in a loop with a predefined padding size.

2) Merge Model (M2): The second proposed architecture which we denote as the merge version or M2 is shown in Fig. 3b. It is identical to M1 in the way in which it creates features for the RoIs of the input images. However, unlike in the previous architecture, it uses a fully connected layer as a decoder. The features representing the image and the previous words are merged and passed to a fully connected (i.e., dense) layer, behaving as a decoder (instead of the second RNN used in M1), as shown by the blue FC block in Fig. 3b. The number of units in the fully connected layer is equal to the vocabulary size. Identical to the previous model, the caption is created word by word in a loop with a predefined padding size.

3) Hybrid Model (M3): Additionally, we propose a third architecture called a hybrid model (M3), which is shown in Fig. 3c. Leveraging the ideas from the former two models, the image features are concatenated with the word embeddings of the previously generated words and fed into the RNN network that encodes the previous context. The encoded context is concatenated with the image features, and two fully connected layers decode this vector representation into predicted word. The difference in the training between this model and the prior two is that this model is trained one-way, i.e., a caption by caption, unlike the multi-way training, a word by word, of the other methods.

In our experiments, the hybrid model (M3) uses two LSTM layers with 512 units for encoding the previous words, and two fully connected layers for generating captions: one fully connected layer with 1024 units and second fully connected layer with the number of units equivalent to the vocabulary size for the FC block in Fig. 3c.

D. Scoring metrics

Evaluating the output of a natural language generation model is a fundamentally difficult task. The most common way to assess the quality of automatically generated texts is a subjective evaluation by human experts [32]. However, human evaluation is not always attainable. Another approach is to use automatic evaluation metrics, such as METEOR [33] and BLEU [34], which were developed for machine translation. ROUGE [35], which was developed for text summarization, and CIDEr [36] and SPICE [37] which were developed for evaluating image captions. All these measures compute a score that indicates the similarity between the system output and one or more human-written reference texts.

E. Training details

We partitioned the dataset into three subsets of size $90,000$, $10,000$ and $8,077$ images for training, validation and testing, respectively. Captions of the images in the validation and training subset were used for creating the vocabulary. All words are converted into lowercase. Words representing punctuation were removed. The final vocabulary has $36,413$ tokens.

Consequently, words are represented with one-hot encodings of size $36,413$. Each word is related to an integer that maps the word with its corresponding one-hot encoding. An embedding layer is used to encode the words into a representation with size 300. This layer's weights are frozen and initialized with weights from the GloVe (Global Vectors for Word Representation) [38] model, which is pre-trained on the Wikipedia corpus[2].

All models were trained with categorical cross-entropy loss function, Adam optimizer [39] with 0.001 learning rate and batch size 1024. Regions of the image are characterized by three-dimensional features of size $7 \times 7 \times 256$). For previous words, we use a frame with a padding size of 10, so for each word, we utilise the previous 10 words as features. If there are less than 10 previous words, features are padded with zeros to the required padding size.

All the models are implemented using the Python deep learning library Keras[3] with Tensorflow[4] backend. The training and testing were performed on NVIDIA Tesla K80 GPU on Windows Azure. Some experiments were also performed on an on-premises NVIDIA Titan V GPU.

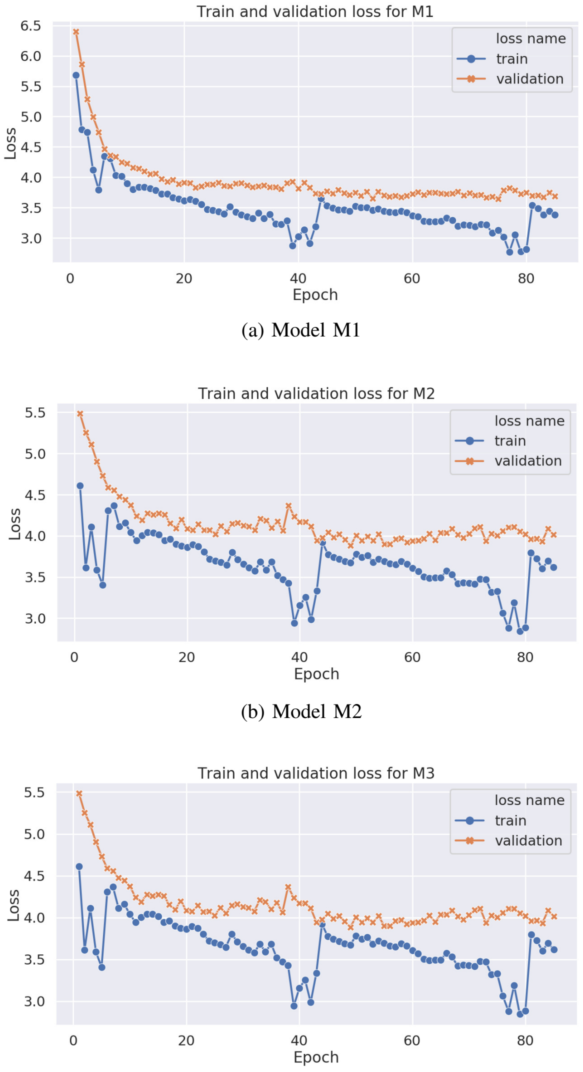
IV. Results and Discussion

We evaluated the three models using the evaluation metrics described in Section III-D. For each RoI of each image in the test set, all metrics were calculated and then averaged to get an average score the image.

The first two models, inject (M1) and merge (M2), were trained in 85 epochs using the ground truth regions of the images. Train and validation losses are shown in Fig. 4a (inject model - M1) and Fig. 4b (merge model - M2). Fig. 4a shows that for M1 in the first epochs both validation and training loss decrease. After about 40 epochs, the training loss starts oscillating between 3 and 4. Similarly, for model M2 the validation loss is decreasing and training loss is oscillating between 3 and 4, as shown in Fig. 4b. Fig. 4c shows oscillating training loss and decreasing validation loss for the hybrid model - M3.

The average evaluation scores for each metric on the test set are shown in Table I. The table also includes information about the number of weights that need to be trained for each model.

The comparison of the inject (M1) and merge (M2) models highlights that the inject model has better performance. Although the merge model has comparable results, it achieves lower average scores for all metrics except BLEU-1. This contradicts the findings of [31], which showed that the merge version generally outperforms the inject version of models. We could hypothesize that the RNN decoder outperforms the fully connected decoder, as opposed to the findings of [31] which demonstrate that applying fully connected layer as a decoder leads to better performance. However, their task differs from ours since we predict multiple captions for an image as opposed to predicting a single caption. Moreover, both problems require different dataset types, that is "image - single caption pairs" for single caption generation and "image - multiple caption pairs" for multiple caption generation. Therefore, we cannot precisely determine if one architecture is better than another.



(a) Model M1

(b) Model M2

(c) Model M3

Fig. 4: Train and validation loss of (a) the inject model (M1), (b) the merge model (M2) and (c) the hybrid model (M3)

Even though the hybrid model (M3) was trained differently than models M1 and M2, it was evaluated with the same test set. The evaluation shows that the M3 model achieves lower scores. We could hypothesize that the reason for the low predictive performance of this model could be the fact

[2] https://nlp.stanford.edu/projects/glove/, last visited: 22.05.2020
[3] https://keras.io/, last visited: 22.05.2020
[4] https://www.tensorflow.org/, last visited: 22.05.2020

TABLE I: Evaluation results for the proposed models M1 (inject model), M2 (merge model) and M3 (hybrid model).

Model	#Weights	SPICE	ROUGE-L	METEOR	CIDEr	BLEU-1	BLEU-2	BLEU-3	BLEU-4
M1	17M	**0.1387**	**0.3593**	**0.1411**	**0.9935**	0.3073	**0.1600**	**0.0946**	**0.0526**
M2	80M	0.1274	0.3476	0.1349	0.9142	**0.3136**	**0.1600**	0.0941	0.0521
M3	32M	0.0452	0.1643	0.0524	0.2553	0.1561	0.0664	0.0386	0.0277

that it was trained a caption by a caption and could benefit from more training.

BLEU [34] measures how close a candidate sequence is to a reference sequence, more concisely, the hits of n-grams of a candidate sequence to the reference. According to the results, we could hypothesize that M2 performs better in terms of matching smaller n-grams, that is, unigrams and possibly bigrams. However, for matching longer n-grams, M1 achieves better results. This is confirmed with ROUGE-L [35], which applies the concept of the Longest Common Subsequence (LCS). The value of this metric is higher for M1.

CIDEr [36] measures how often n-grams in the candidate sentence are present in the reference sentences, while METEOR [33] is based on the harmonic mean of unigram precision and recall, where recall is weighted higher. Both metrics map the words in their stem or root forms. M1 shows better performance for both metrics. Therefore we can infer that this model generates words that perhaps may not be the exact match of the reference words, but they nevertheless have the same root form.

SPICE [37] measures how effectively image captions recover objects, attributes and the relations between them. It is based on the agreement of the scene-graph tuples of the candidate sentence and all reference sentences. The M1 model, again, achieves the best performance leading to the conclusion that this model effectively describes the image scene.

The number of trainable weights of the models is included in Table I. The M1 model is the smallest model out of the three models. That could be the reason for the best performance of this model, which is learning fewer weights given the same training time. Having a bigger model with many trainable weights has been the preferred way for learning more complex relationships in images. However, larger models also require more training time for learning all the weights. Therefore, with the results from these experiments, we can infer that the smaller model is more practical and has the best performance in this setting. Also, regarding the weaker performance of the M3 model, we can conclude that the multi-way training (word by word) is preferred over the more difficult process of one-way training (caption by caption).

A. Extensive analysis of the capability of the models

Evaluating the models based on the n-gram evaluation metrics limits us to understand the relative strengths and weaknesses of the models. Therefore, we use the property of the SPICE metrics that enables us to divide the metric value into meaningful categories.

In Table II, we review the performance of the models from different aspects. The table contains F-scores for the subcategories from which SPICE is calculated, that is objects, their attributes, and relations between them. The M1 model surpasses the other models for all of the categories, except the size category, where the M2 model is finer. This effect means that the M2 model caption generator is better for capturing the size of the objects than the other models. The M3 model is inferior in these settings, and we can see that the crucial shortcoming of the model is the cardinality, so the model is not able to count while generating the captions.

From the evaluation results, we can conclude that the models perform well at capturing objects present at the image and their cardinality. However, they fail to describe the attributes of the objects. We could hypothesize that such behaviour is expected since the part of the models that extracts image features is pre-trained on an object detection task and therefore could potentially be biased towards detecting objects rather than describing them in details. Therefore, because the text generation models are separated from the CNN model for creating the image features, one way of improving is to refine the features of the CNN model by additionally training the model on attribute prediction, not solely on object detection. In this way, the image features should be expected to include more information about the attributes of the object and hence help the text generation models to create better captions.

B. Qualitative results

We present example predicted captions for ground truth regions from models M1 (inject) and M2 (merge) in Fig. 5. Predictions from M1 are shown on the left, while predictions from M2 in the right. For brevity, we plot only one region caption per image. For each model, one good, one quite good and one not good example are displayed.

The first row presents captions classified as good. The predictions are made with padding size 10, i.e. each generated caption has length 10. However, from the examples, we can infer that for some regions, this padding size is too big. Both models generate descriptive captions with specific length and fill the rest with words unrelated to the image. Nevertheless, we classify such captions as good. Quite good captions are those related to the image with minor errors (second row). For example, M2 generates the following caption "child wearing a blue shirt". As we can see, the child is wearing a white shirt. We can conclude that even though the colour is incorrect, the main context of the region is described. The last row presents captions classified as not good. These captions are unrelated to the region, which they describe.

TABLE II: F-scores from SPICE by semantic proposition subcategory. The models models M1 (inject model), M2 (merge model) and M3 (hybrid model) are compared with the SPICE metric for object, relation, attribute, color, cardinality and size.

Model	SPICE	Relation	Cardinality	Attribute	Size	Color	Object
M1	**0.1387**	**0.0595**	**0.1241**	**0.0618**	0.0493	**0.0627**	**0.1989**
M2	0.1274	0.0456	0.1151	0.0548	**0.0502**	0.0499	0.1854
M3	0.0452	0.0197	0.0000	0.0219	0.0066	0.0408	0.0671

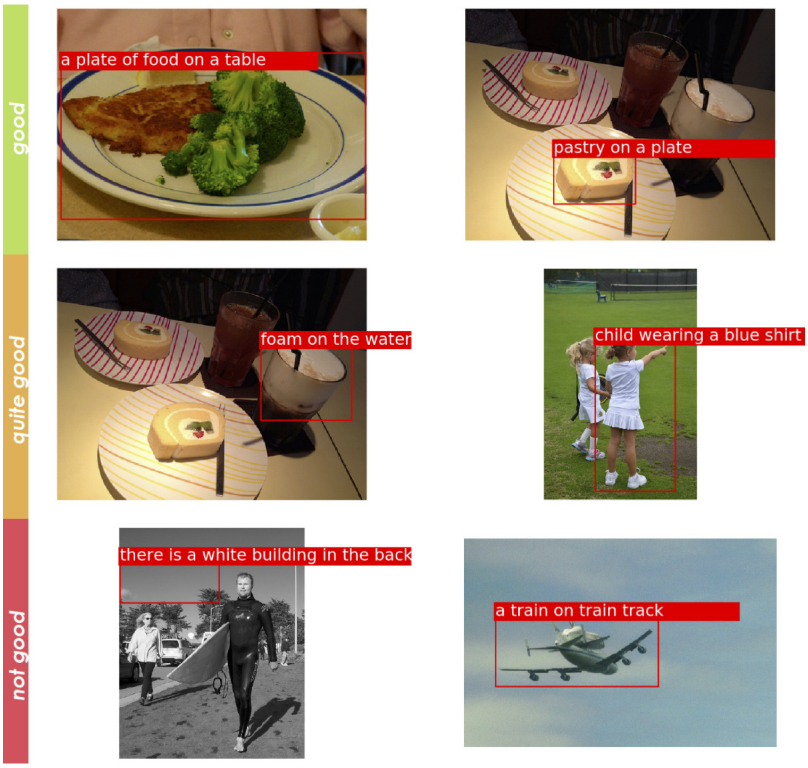

good
quite good
not good
a plate of food on a table
pastry on a plate
foam on the water
child wearing a blue shirt
there is a white building in the back
a train on train track

Fig. 5: Examples of generated region captions for the inject model, M1, (left) and the merge model, M2 (right)

V. Conclusions

This paper investigated the problem of automatically generating descriptions for RoIs in images. The aim is to investigate the appropriate model for generating text that describes RoI in images. The Mask R-CNN model trained for image classification was modified and used for RoI feature extraction. For caption generation, three model versions were proposed. In the first version, called inject model, image features are injected into a decoder RNN. In the second version, called merge model, image features and previous words features are concatenated and fed into a fully connected layer as a decoder. The third version, called hybrid model, the image features are fed into a decoder RNN but the caption is generated one-way instead of generating word by word as in the previous two models.

We evaluated the proposed models with several text evaluation metrics. The results show that the models M1 (inject model) and M2 (merge model) are better than M3 (hybrid model), with M1 having the best performance. Also, the M1 model has the smallest number of trainable weights out of the three models and still is the best performing model. The experimental results demonstrate that injecting image features into a decoder RNN while generating a caption word by word is the best performing architecture among the architectures explored in this paper. The extended evaluation represents the shortcomings of the models to describe the attributes of the objects in the images. Hence, future experiments should examine the models after training the CNN feature extractor on attribute prediction.

The visual text generation models are impressive in most of the cases, but they also have faults. Show-and-Fool [40] is a model created for attacking image captioning models with adversarial perturbations in machine vision and perception to produce randomly chosen captions that are not relevant to the image. Therefore, the future work could focus on applying such attacking model for evaluating the robustness of the proposed model. An alternative implementation of this model is to build a more robust image captioning model using an attack model into a GAN network. Likewise, for caption generation it could be interesting to investigate the capability of networks consisted of attention only, such as the Transformer [19] approach, to encode and decode both the context of the image and the text.

Acknowledgment

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss. Cyril and Methodius University, Skopje, North Macedonia. We also gratefully acknowledge the support of Microsoft Azure for Research and the NVIDIA Corporation through grants providing GPU resources for this work.

References

[1] T. Yao, Y. Pan, Y. Li, and T. Mei, "Exploring visual relationship for image captioning," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 684–699.

[2] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3588–3597.

[3] S. Herdade, A. Kappeler, K. Boakye, and J. Soares, "Image captioning: Transforming objects into words," in *Advances in Neural Information Processing Systems*, 2019, pp. 11 135–11 145.

[4] M. Hossain, F. Sohel, M. F. Shiratuddin, and H. Laga, "A comprehensive survey of deep learning for image captioning," *ACM Computing Surveys (CSUR)*, vol. 51, no. 6, p. 118, 2019.

[5] W. Hechun and Z. Xiaohong, "Survey of deep learning based object detection," in *Proceedings of the 2nd International Conference on Big Data Technologies*, 2019, pp. 149–153.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[7] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[9] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[10] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, 2014.

[11] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 11, pp. 3212–3232, 2019.

[12] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.

[13] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.

[14] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4651–4659.

[15] Z. Yang, Y.-J. Zhang, S. ur Rehman, and Y. Huang, "Image captioning with object detection and localization," in *International Conference on Image and Graphics*. Springer, 2017, pp. 109–118.

[16] X. Zhu, L. Li, J. Liu, H. Peng, and X. Niu, "Captioning transformer with stacked attention modules," *Applied Sciences*, vol. 8, no. 5, p. 739, 2018.

[17] J. Yu, J. Li, Z. Yu, and Q. Huang, "Multimodal transformer with multi-view visual representation for image captioning," *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[18] S. He, W. Liao, H. R. Tavakoli, M. Yang, B. Rosenhahn, and N. Pugeault, "Image captioning through image transformer," *CoRR*, 2020.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[20] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[21] J. Johnson, A. Karpathy, and L. Fei-Fei, "Densecap: Fully convolutional localization networks for dense captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4565–4574.

[22] L. Yang, K. Tang, J. Yang, and L.-J. Li, "Dense captioning with joint inference and visual context," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2193–2202.

[23] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3128–3137.

[24] G. Yin, L. Sheng, B. Liu, N. Yu, X. Wang, and J. Shao, "Context and attribute grounded dense captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6241–6250.

[25] X. Li, S. Jiang, and J. Han, "Learning object context for dense captioning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 8650–8657.

[26] X. Jia, E. Gavves, B. Fernando, and T. Tuytelaars, "Guiding the long-short term memory model for image caption generation," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2407–2415.

[27] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma **et al.**, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International Journal of Computer Vision*, vol. 123, no. 1, pp. 32–73, 2017.

[28] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[30] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2117–2125.

[31] M. Tanti, A. Gatt, and K. P. Camilleri, "What is the role of recurrent neural networks (rnns) in an image caption generator?" in *The 10th International Natural Language Generation conference*, vol. abs/1708.02043, 2017, p. 51. [Online]. Available: http://arxiv.org/abs/1708.02043

[32] R. Bernardi, R. Cakici, D. Elliott, A. Erdem, E. Erdem, N. Ikizler-Cinbis, F. Keller, A. Muscat, and B. Plank, "Automatic description generation from images: A survey of models, datasets, and evaluation measures," *Journal of Artificial Intelligence Research*, vol. 55, pp. 409–442, 2016.

[33] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the ninth workshop on statistical machine translation*, 2014, pp. 376–380.

[34] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[35] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text Summarization Branches Out*, 2004.

[36] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575.

[37] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic propositional image caption evaluation," in *European Conference on Computer Vision*. Springer, 2016, pp. 382–398.

[38] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[40] H. Chen, H. Zhang, P.-Y. Chen, J. Yi, and C.-J. Hsieh, "Attacking visual language grounding with adversarial examples: A case study on neural image captioning," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 2587–2597. [Online]. Available: https://www.aclweb.org/anthology/P18-1241

DOI: 10.15439/2020F187 ISSN 2300-5963 ACSIS, Vol. 21

Pythagorean Fuzzy Analytical Network Process (ANP) and Its Application to Warehouse Location Selection Problem

Tutku Tuncalı Yaman
Beykent University Management Information
Systems Department Istanbul, Turkey
Email: tutkuyaman@beykent.edu.tr

Abstract—**Although new techniques are added to multi-criteria decision-making (MCDM) techniques every day, fuzzy applications of current and proven methods also take a large place in the literature. The main subject of this study is to propose an extension of Pythagorean fuzzy sets (PFS), which are useful to overcome the uncertainty in multi-criteria decision processes, to the well-known Analytical Network Process (ANP) technique. For this purpose, an empirical application of the proposed method was carried out in defining criteria weights of the warehouse location selection problem in the medical sector.**

I. Introduction

ANALYTICAL approaches to decision-making processes have introduced to the academic world by the pioneers of the field and then applied to real business problems since the 80s. [1-4] made significant contributions to the field of fuzzy logic in order to reflect the human factor in methodical decision-making processes.

After the 2000s, the fuzzy logic concept has been diversified with developments such as type-2 fuzzy sets, Intuitionistic Fuzzy Sets (IFS), Pythagorean Fuzzy Sets (PFS), Neutrosophic Sets (NS) and Hesitant Fuzzy Sets (HFS). Thus, the uncertainty, which is caused by the human factor, has been tried to be covered as much as possible. Many MCDM methods have been proposed with different fuzzy approaches [5]. [6] introduced ANP as a MCDM technique for decision problems, which have interdependence between criteria and alternatives. The method is applied as a successor to DEMATEL [7], which is used especially in exploratory studies, in determining the causal relationships between the criteria and creating a network structure of them. Then the relative importance levels of the criteria can be obtained by sorting the determined weight values. It is possible to use these weight values as inputs for different MCDM methods. ANP and AHP have a common theoretical application steps with different perspective and outputs. Albeit, AHP recognizes independency among criteria, in ANP, correlations among criteria have an important role. AHP is not sufficient to address the complexity of real world problems on its hierarchical structure; ANP presents a problem in a network of criteria and alternatives, which are strongly intercorrelated [8].

The adaptations of the ANP method for the different fuzzy approaches such as type-2 and IFS have already been realized. In this study, it is expected to fill the gap in the literature by proposing the application of a Pythagorean Fuzzy ANP method (PFANP). Thus, the Pythagorean fuzzy Analytic Hierarchy Process (PFAHP), which was detailed in [9], was adapted to ANP. The application of the proposed technique was carried out in the context of the location selection problem (WLSP), which was previously handled by a few researchers. In this paper, an empirical application of WLSP is made specifically for the medicine/pharmaceutical industry. Although it is a crucial factor in supply chain management processes [10], the choice of warehouse location in this sector has not been handled as an MCDM problem before.

The content of the article is organized as follows; in the following section, the state-of-art WLSP and aim of the study are detailed. In Section III, a literature review is carried out by referring academic articles about the WLSP. Section IV includes the methodological perspective of the proposed technique in detail. Section V consists of the empirical results of the study. Conclusion and details of planned future research are given in the last section.

II. Objective of the Study

The main motivation of this study is proposing a Pythagorean fuzzy extension of the ANP method. As the application area of the method, the WLSP, which plays a critical role in the effective realization of all logistics activities, was chosen. The criteria that are important for the selection of the storage location, with availability and cost priority, are given for the medical sector considered within the scope of the research: C1: Proximity to target markets (hospitals, pharmacies), C2: Proximity to the ports and customs, C3: Proximity to the pharmaceutical production centers, C4: The location decision of a warehouse must be submitted together with capacity and demand estimation, C5: The proximity of qualified workforce, C6: The infrastructure of the area (electricity, water, sewage, transportation, natural gas, etc.), C7: The climate of the location, C8: Ground properties of the location (impact of construction on excavation cost), C9: Leasing cost of the location, C10: Traffic density of location.

The proposed method will be detailed in the methodology section and the literature review carried out within the scope of the subject will be included in the next section.

III. Literature Review

With the review purposes, a search was made in the Scopus database on 1st of August, 2020, with "Article Title, Abstract, and Keywords" gives the following frequencies: 9600 for Analytic Network Process (ANP), 503 for Pythagorean fuzzy sets, 318 for fuzzy ANP, and 0 Pythagorean fuzzy ANP. Similarly, when the WLSP is examined, a total of 21 studies have been found in the same database since 2011. The most recent study carried out in the context of this problem was performed in the field of humanitarian relief logistics with the multi-objective fuzzy mathematical programming method [11]. In another study, same problem was solved with the spherical fuzzy CODAS method [12]. In [13], authors attempted to solve WLSP about the storage of agricultural products by MCDM methods such as SAW, AHP and TOPSIS. For further studies on the subject, please refer to [14-32].

IV. Methodology

In cases where there are complex causality relationships between the criteria at the beginning of the MCDM problems, the network structure between the criteria is determined in order to identify and address them in the model, and then the importance levels of the criteria are determined by considering the degree of influence on each other [33]. In this study, by following the same process, the network structure for 10 criteria recommended by experts for the storage location selection problem was determined by the Pythagorean fuzzy DEMATEL (PFDEMATEL) method and then the weight values of the criteria were determined through the proposed Pythagorean fuzzy ANP (PFANP) method. Preliminaries of PFS, and detailed calculation steps of PFDEMATEL, which were used in calculation steps of this study, can be found in [34, 35] respectively. The use of the ANP method with the PFSs and its extensions has not proposed in the literature, and also the application of PFS based ANP has never been used in a medical sector WLSP before. Calculation steps of PFANP framework are described below.

A. Pairwise Comparisons

Based on the network structure created for the criteria with the PFDEMATEL method [35], the criteria affecting each criterion are subjected to pairwise comparisons in the context of the related criterion. These comparisons are made by experts of the subject on the basis of the linguistic variables [9] given in Table I below. Here, membership and non-membership degrees of PFNs are denoted as μ and v, respectively. Given linguistic expressions are converted to PF values to obtain PF pairwise matrices for each criterion and for each expert.

B. Aggregated Pairwise Comparison Matrix

The PF weighted power geometric (PFWPG) operator [36] is used to obtain the Aggregated Pairwise Comparison Matrix ($R = (r_{ik})_{m \times m}$), by averaging the evaluations made by experts. It is possible to assign different weight values to different experts evaluating here. However, the sum of these weights should be equal to 1. Let $\widetilde{P_i} = \langle \mu_i | v_i \rangle, i = 1,2, \dots, n$ be a collection of PFNs and $W = (w_1, w_2, \dots, w_n)^T$, is the weight vector of P_i with $\sum_{i=1}^{n} w_i = 1$.

Then the PFWPG operator is, $PFWPG(\widetilde{P}_1, \dots, \widetilde{P}_n) = \left(\sqrt{(1 - \prod_{i=1}^{n}(1-\mu_i^2)^{w_i})}, \sqrt{(1 - \prod_{i=1}^{n}(1-v_i^2)^{w_i})}\right)$ (1)

C. Priority Vector

According to [37], the idea of a priority vector has much less validity for an arbitrary positive reciprocal matrix than for a consistent and a near consistent matrix. Here, [9]'s point of view is followed and the Priority Vector $A = (a_i)_{1 \times m}$ is created on a relative dominance basis. The calculation steps are as follows:

First, differences matrix $D = (d_{ik})_{m \times m}$is constructed using Equations 2 and 3 below.

$$d_{ik_L} = \mu_{ik_L}^2 - v_{ik_U}^2 \quad (2)$$

$$d_{ik_U} = \mu_{ik_U}^2 - v_{ik_L}^2 \quad (3)$$

Then, interval multiplicative matrix $S = (s_{ik})_{m \times m}$ is found using Equations 4 and 5.

$$s_{ik_L} = \sqrt{1000^{d_{ik_L}}} \quad (4)$$

$$s_{ik_U} = \sqrt{1000^{d_{ik_U}}} \quad (5)$$

Determinacy value $\tau = (\tau_{ik})_{m \times m}$of the Aggregated Pairwise Comparison Matrix is calculated using following Equation 6.

$$\tau_{ik} = 1 - \left(\mu_{ik_U}^2 - \mu_{ik_L}^2\right) - \left(v_{ik_U}^2 - v_{ik_L}^2\right) \quad (6)$$

Matrix of Weights $T = (t_{ik})_{m \times m}$ is obtained using Equation 7 below.

$$t_{ik} = \left(\frac{s_{ik_L} + s_{ik_U}}{2}\right) \tau_{ik} \quad (7)$$

Normalization of Matrix of Weights gives us the Priority Vector A of each criterion. The normalization operator is,

$$a_i = \frac{\sum_{k=1}^{m} t_{ik}}{\sum_{i=1}^{m} \sum_{k=1}^{m} t_{ik}} \quad (8)$$

D. Super Matrix

After creating a Priority Vector (a_i) for each criterion, as described in previous steps, the Super Matrix $(W = (w_i)_{m \times m})$ is created by listing local priority vectors in the appropriate columns of W.

E. Global Weights

Once the Super Matrix is created, a stationary Limit Matrix (LM) is obtained by multiplying W with infinite times using Equation 9 below.

$$\lim_{k \to \infty} (W)^k \quad (9)$$

The idea behind that, obtaining the cumulative influence of each element on every other interacted element. In practice, it is necessary to raise the super matrix to the power $k = 2n + 1$ where n is an arbitrary large number [38]. Each column of the resulting LM will be equal to 1. Any of the columns of the LM will give us the Priority Vector of the criteria in our problem. By performing previously detailed five steps of PFANP algorithm, an empirical application is done in order to determine criteria weights by concerning casual relationships between criteria of WLSP. In the following section, the results of the application are given in detail.

V. Empirical Results

In order to demonstrate the empirical application of the PFANP method, the WLSP, which is one of the important supply chain management problems, is discussed. In order to obtain the network structure on the basis of 10 criteria given in the Section II, PFDEMATEL method [35] was applied after obtaining linguistic evaluations of 5 experts in the field. After revealing cause and effect groups of criteria, the Super Matrix design was obtained [39]. Then criteria based pairwise comparisons were made by 3 experts through linguistic expressions are given in Table I and Pairwise Comparison Matrices were obtained for each criterion on expert basis. Since the matrices contained linguistic expressions, they were converted to PF values, as presented in Table I. Then R was calculated using the PFWPG operator provided in Equation 1 by giving equal weight $(0.3\bar{3})$ to each expert. Then expert assessments for each criterion were combined. Priority Vectors for each criterion are calculated using Equations 2-8, respectively. These vectors were used to construct the Super Matrix. A stationary Limit Matrix was found using Equation 9. Any column of the Limit Matrix can be used as Global Weights' of criteria. These weights are detailed in Table II below. Since in PFDEMATEL results, the cause group consists of C2, C3, C4 and C8. The most important criterion in cause group is C4 with the highest (*c*+*r*) value. C1, C6, C7, C9 and C10 are listed in effect group [39]. According to the global weight figures of PFANP, the first four important criteria in WLSP are found as C7, C10, C5 and C2, respectively.

VI. Conclusion

This paper aimed to demonstrate the Pythagorean fuzzy extension of the well-known ANP method. In line with the mentioned objective and due to complexity of the problem, a combined application of PFDEMATEL and PFANP is performed. The use of DEMATEL's outputs as inputs in ANP is suggested as a solution [17] to the problem of dependence and feedback among each measurement criteria. As an illustrative example, the WLSP was held. It is important for companies to determine the optimal locations of warehouses, which have a critical role in supply chain and logistics management. On the contrary, the WLSP has not been dealt with much in the literature, and its application has also not been encountered especially for the medical sector. In this context, the example was found to be appropriate to be given as a medical sector application by hoping to guide the professionals of the field.

For future directions, the proposed method can be applied in different problems, and its effectiveness can be evaluated by dealing with different types of fuzzy set extensions. A sensitivity analysis, which can evaluate different weight values of different fuzzy extensions of ANP, can help us in finding the superior approach for the handled problem. Group decision-making perspective of [40] can also be pursued in pair-wise comparisons' evaluations of experts. Plus, one step after the proposed method, an MCDM technique in the selection of potential alternatives can be implemented that uses global criteria weights of the PFANP as input. Hence, it is hoped that a comprehensive decision-making system can be created.

Table I.
Rating Scales of Linguistic Terms

Linguistic terms	PFN equivalents IVPF numbers			
	μ_L	μ_U	v_L	v_U
Certainly Low Importance	0	0	0.9	1
Very Low Importance	0.1	0.2	0.8	0.9
Low Importance	0.2	0.35	0.65	0.8
Below Average Importance	0.35	0.45	0.55	0.65
Average Importance	0.45	0.55	0.45	0.55
Above Average Importance	0.55	0.65	0.35	0.45
High Importance	0.65	0.8	0.2	0.35
Very High Importance	0.8	0.9	0.1	0.2
Certainly High Importance	0.9	1	0	0
Exactly Equal	0.1965	0.1965	0.1965	0.1965

Table II.
Global Weights

Criteria	Weight	Rank
C1. Proximity to target markets	0.0444	8
C2. Proximity to the ports and customs	0.1106	4
C3. Proximity to the pharmaceutical…	0.0958	5
C4. The location decision of a warehouse must…	0.0428	9
C5. The proximity of qualified workforce	0.1172	3
C6. The infrastructure of the area	0.0424	10
C7. The climate of the location	0.2038	2
C8. Ground properties of the location	0.0918	6
C9. Leasing cost of the location	0.0452	7
C10. Traffic density of location	0.2056	1

References

[1] L. Zadeh, "Fuzzy-set-theoretic interpretation of linguistic hedge", *Journal of Cybernetics,* vol. 2, 1972, pp. 4–34. https://doi.org/10.1080/01969727208542910

[2] L. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning-1", *Information Sciences*, 1975, pp. 199–249. https://doi.org/10.1016/0020-0255(75)90036-5

[3] R.R. Yager, "A characterization of the extension principle," *Fuzzy Sets and Systems*, vol. 18, no. 3, 1986, pp. 205–217. https://doi.org/10.1016/0165-0114(86)90002-3

[4] K. Atanassov, "Intuitionistic fuzzy sets", *Fuzzy Sets and Systems*, vol. 20, no.1, 1986, pp.87–96. https://doi.org/10.1016/S0165-0114(86)80034-3

[5] E. Bolturk, "Pythagorean fuzzy CODAS and its application to supplier selection in a manufacturing firm", *Journal of Enterprise Information Management*, vol.31, no.4, 2018, pp.550–564. https://doi.org/10.1108/JEIM-01-2018-0020

[6] T. L. Saaty, *Decisions with the analytic network process (ANP).* University of Pittsburgh, USA: ISAHP, 1996.

[7] S-L. Si, X-Y. You, H-C. Liu, and P. Zhang, “DEMATEL technique: a systematic review of the state-of-the-art literature on methodologies and applications.” *Mathematical Problems in Engineering*, vol. 2018, 2018, pp. 1-33. https://doi.org/10.1155/2018/3696457

[8] M. Reisi, A. Afsaneh, and A. Lu,"Applications of analytical hierarchy process (AHP) and analytical network process (ANP) for industrial site selections in Isfahan, Iran." *Environmental Earth Sciences*, vol.77, no. 537, 2018, pp.1-13. https://doi.org/10.1007/s12665-018-7702-1

[9] E. Ilbahar, A. Karaşan, S. Cebi, and C. Kahraman, “A novel approach to risk assessment for occupational health and safety using Pythagorean fuzzy AHP & fuzzy inference system”, *Safety Science*, vol. 103, 2018, pp. 124-136. https://doi.org/10.1016/j.ssci.2017.10.025

[10] J. Korpela, and M. Tuominen, "A decision aid in warehouse site selection." *International Journal of Production Economics*, vol. 45. no.1-3, 1996, pp. 169-180. https://doi.org/10.1016/0925-5273(95)00135-2

[11] C. Boonmee, Chawis, and C. Kasemset. "The Multi-Objective Fuzzy Mathematical Programming Model for Humanitarian Relief Logistics." *Industrial Engineering & Management Systems*, vol. 19, no.1, pp.197-210, 2020. https://doi.org/10.7232/iems.2020.19.1.197

[12] F. Kutlu Gündoğdu, and C. Kahraman, *Spherical Fuzzy Sets and Decision Making Applications.* In: International Conference on Intelligent and Fuzzy Systems. Springer, Cham, 2019, pp. 979-987. https://doi.org/10.1007/978-3-030-23756-1_116

[13] M. Khaengkhan, C. Hotrawisaya, B. Kiranantawat, and M. R. Shaharudin, “Comparative analysis of multiple criteria decision making (MCDM) approach in warehouse location selection of agricultural products in Thailand”, *International Journal of Supply Chain Management, vol*. 8, no. 5, pp. 168-175, 2019.

[14] I. Otay, and M. Jaller. "Multi-criteria and multi-expert wind power farm location selection using a pythagorean fuzzy analytic hierarchy process." International Conference on Intelligent and Fuzzy Systems. Springer, Cham, 2019. https://doi.org/10.1007/978-3-030-23756-1_108

[15] F. Kutlu Gündoğdu, and C. Kahraman, "A novel VIKOR method using spherical fuzzy sets and its application to warehouse site selection", *Journal of Intelligent & Fuzzy Systems*, vol. 37, no.1, , 2019, pp. https://doi.org/1197-1211.10.3233/JIFS-182651

[16] S. Y. Roh, Y. R. Shin, and Y. J. Seo, “The Pre-positioned warehouse location selection for international humanitarian relief logistics”, *The Asian Journal of Shipping and Logistics*, vol.34, no.4, pp.297-307, 2018.

[17] R. K. Singh, N. Chaudhary, and N. Saxena, “ Selection of warehouse location for a global supply chain: A case study”, *IIMB Management Review*, vol. 30, no.4, pp. 343-356, 2018.

[18] N. Foroozesh, R. Tavakkoli-Moghaddam, and S. M. Mousavi, “A novel group decision model based on mean–variance–skewness concepts and interval-valued fuzzy sets for a selection problem of the sustainable warehouse location under uncertainty”, *Neural Computing and Applications*, vol.30, no.11, 2018, pp.3277-3293. https://doi.org/10.1007/s00521-017-2885-z

[19] Ş. Emeç, and G. Akkaya, "Stochastic AHP and fuzzy VIKOR approach for warehouse location selection problem", *Journal of* Enterprise Information Management, vol. 31 no. 6, 2018, pp. 950-962. https://doi.org/10.1108/JEIM-12-2016-0195

[20] B. Dey, B. Bairagi, B. Sarkar, and S. K. Sanyal, “Group heterogeneity in multi member decision making model with an application to warehouse location selection in a supply chain”, *Computers & Industrial Engineering*, vol.105, 2017, pp.101-122. https://doi.org/10.1016/j.cie.2016.12.025

[21] G. T. Temur, “A novel multi attribute decision making approach for location decision under high uncertainty”, *Applied Soft Computing*, vol. 40, 2016, pp.674-682. https://doi.org/10.1016/j.asoc.2015.12.027

[22] B. Dey, B. Bairagi, B. Sarkar, and S. K. Sanyal, “Warehouse location selection by fuzzy multi-criteria decision making methodologies based on subjective and objective criteria”, *International Journal of Management Science and Engineering Management*, vol.11, no.4, 2016, pp. 262-278. https://doi.org/10.1080/17509653.2015.1086964

[23] B. Malmir, A. Aghighi, M. N. Bisheh, A. Ala, B. A., Avilaq, and S. Dehghani, “Application of a new multi criteria decision making method for warehouse location problem”, *International Journal of Value Chain Management,* vol.7, no.3, 2015, pp. 255-270. https://doi.org/10.1504/IJVCM.2016.079211

[24] C. Karmaker, and M. Saha, “Optimization of warehouse location through fuzzy multi-criteria decision making methods”, *Decision Science Letters*, vol.4, no.3, 2015, pp.315-334. https://doi.org/10.5267/j.dsl.2015.4.005

[25] B. Malmir, R. Moein, and S.K. Chaharsooghi, “Selecting warehouse location by means of the balancing and ranking method with an interval approach.” 2015 International Conference on Industrial Engineering and Operations Management (IEOM), IEEE, 2015, pp. 1-7. https://doi.org/10.1109/IEOM.2015.7093911

[26] F. Uysal, and Ö. Tosun, “Selection of sustainable warehouse location in supply chain using the grey approach”, *International Journal of Information and Decision Sciences*, vol.6, no.4, 2014, pp.338-353. https://doi.org/10.1504/IJIDS.2014.066633

[27] I. U. Sarı, B. Öztayşi, and C. Kahraman, Fuzzy analytic hierarchy process using type-2 fuzzy sets: An application to warehouse location selection. In Multicriteria decision aid and artificial intelligence, John Wiley & Sons, Ltd., 2013, pp. 285-308. https://doi.org/10.1002/9781118522516.ch12

[28] B. Dey, B. Bairagi, B. Sarkar, and S. K. Sanyal, “A hybrid fuzzy technique for the selection of warehouse location in a supply chain under a utopian environment”, *International Journal of Management Science and Engineering Management*, vol.8, no.4, pp. 250-261, 2013.

[29] B. Dey, B. Bairagi, B. Sarkar, and S. K. Sanyal, “A MOORA based fuzzy multi-criteria decision making approach for supply chain strategy selection”, *International Journal of Industrial Engineering Computations,* vol.3, no.4, 2012, pp. 649-662. https://doi.org/10.1080/17509653.2013.825075

[30] T. Özcan, N. Çelebi, and Ş. Esnaf, “Comparative analysis of multi-criteria decision making methodologies and implementation of a warehouse location selection problem”, *Expert Systems with Applications,* vol.38, no.8, 2011, pp.9773-9779. https://doi.org/10.1016/j.eswa.2011.02.022

[31] T. Demirel, Ç. Demirel, and C. Kahraman, “Multi-criteria warehouse location selection using Choquet integral”, *Expert Systems with Applications,* vol. 37, no.5, 2010, pp. 3943-3952. https://doi.org/10.1016/j.eswa.2009.11.022

[32] Q. Cao, X. Di, and X. Zhang, A simulated annealing methodology to estate logistic warehouse location selection and distribution of customers' requirement. In 2009 International Workshop on Intelligent Systems and Applications, IEEE, May 2009, pp. 1-4. https://doi.org/10.1109/IWISA.2009.5072676

[33] Ö. Senvar, U. R. Tuzkaya, and C. Kahraman, Supply chain performance measurement: an integrated DEMATEL and Fuzzy-ANP approach. In Supply Chain Management Under Fuzziness, Springer, Berlin, Heidelberg, 2014, pp. 143-165. https://doi.org/10.1007/978-3-642-53939-8_7

[34] R.R. Yager, Properties and applications of Pythagorean fuzzy sets. In: Imprecision and Uncertainty in Information Representation and Processing. Springer, 2016, pp.119–136. https://doi.org/10.1007/978-3-319-26302-1_9

[35] L., Abdullah and P. Goh, “Decision making method based on Pythagorean fuzzy sets and its application to solid waste management”, *Complex & Intelligent Systems*, vol. 5, no.2, 2019, pp. 185-198. https://doi.org/10.1007/s40747-019-0100-9

[36] R.R Yager, and A.M. Abbasov, “Pythagorean membership grades, complex numbers, and decision making”, *International Journal. Intelligent Systems,* vol. 28 no.5, 2013, pp. 436–452. https://doi.org/10.1002/int.21584

[37] T. L. Saaty, “Fundamentals of the analytic network process—Dependence and feedback in decision-making with a single network”, *Journal of Systems Science and Systems Engineering*, vol.13, 2004, pp.129-157. https://doi.org/10.1007/s11518-006-0158-y

[38] G. Büyüközkan, and G. Gizem, "A novel fuzzy multi-criteria decision framework for sustainable supplier selection with incomplete information." *Computers in Industry*, vol. 62 no.2, 2011, pp. 164-174. https://doi.org/10.1016/j.compind.2010.10.009

[39] T. Tuncalı Yaman, and G. R. Akkartal, “Warehouse location selection decision systems for medical sector (In press)”, In: *2020 Fourth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4),* London, United Kingdom, 2020.

[40] A. Łodziński, "Multicriteria support of choosing a group decision," *2015 Federated Conference on Computer Science and Information Systems (FedCSIS),* Lodz, 2015, pp. 1597-1602, https://doi.org/10.15439/2015F58.

DOI: 10.15439/2020F101
ISSN 2300-5963 ACSIS, Vol. 21

A Framework for Time Series Preprocessing and History-based Forecasting Method Recommendation

Marwin Züfle, Samuel Kounev
University of Wuerzburg, Germany
Email: {marwin.zuefle, samuel.kounev}@uni-wuerzburg.de

***Abstract*—The complexity of managing the capacities of large IT infrastructures is constantly increasing as more network devices are connected. This task can no longer be performed manually, so the system must be monitored at runtime and estimations of future conditions must be made automatically. However, since using a single forecasting method typically performs poorly, this paper presents a framework for forecasting univariate network device workload traces using multiple forecasting methods. First, the time series are preprocessed by imputing missing data and removing anomalies. Then, different features are derived from the univariate time series, depending on the type of forecasting method. In addition, a recommendation approach for selecting the most suitable forecasting method from this set of algorithms for each time series based only on its historical values is proposed. For this purpose, the performance of the forecasting methods is approximated using the historical data of the respective time series under consideration. The framework is used in the FedCSIS 2020 Challenge and shows good forecasting quality with an average R^2 score of 0.2575 on the small test data set.**

I. Introduction

OVER the last decades, the network load in large IT systems has grown considerably. Thus, coping with the increased data traffic is becoming more and more difficult. Typical reactive mechanisms that adapt the system to the current condition are no longer applicable, as this leads to temporary overload situations with resulting delays. To overcome this problem, proactive adaptation algorithms are required that analyze historical data and automatically forecast future conditions to enable early decision making. However, the decision making component is beyond the scope of this paper, as this paper is part of the *FedCSIS 2020 Network Device Workload Prediction Challenge* [1]. To achieve sufficient forecasting performance, no single method can be used since the “No-Free-Lunch-Theorem” states that there cannot be a single algorithm that outperforms all others on every kind of data [2]. For this reason, we developed a hybrid approach that recommends the best forecasting method for a given time series based only on its known historical values. In addition, we introduce an algorithm for missing data imputation and a technique for eliminating anomalies to preprocess the time series in advance.

The remainder of this paper is structured as follows: In Section II, we present related work on time series forecasting. The foundations of the applied forecasting methods are described in Section III. In Section IV, we introduce the preprocessing steps that were applied prior to the modelling part of the approach (Section V). Experimental results are presented in Section VI. Finally, Section VII concludes the paper.

II. Related Work

Forecasting time series is a widely studied field of research for which many different approaches exist. Firstly, individual methods can be used to forecast time series. This area ranges from the application of statistical methods [3] to machine learning models [4]. However, according to the “No-Free-Lunch-Theorem”, there is no single method that surpasses all other methods for every type of data [2]. Therefore, more sophisticated approaches implement hybrid forecasting methods. That is, several individual forecasting methods are applied and the final result is either a weighted combination [5], [6], a sequential execution of methods on different parts of the time series [7], [8], or the forecast of a recommended method.

First approaches towards forecasting method recommendation use manually created expert systems [9]. One of the first works using automatic rule induction methods is by Arinze et al. [10]. More recent approaches to the recommendation of forecasting methods are by Wang et al. [11] and Züfle et al. [12]. However, all of these automatic rule learning approaches calculate characteristics of the time series in a large training data set and assess the forecasting accuracy of the available methods on them. Then, a rule induction technique is applied to map the characteristics of the time series to the best performing forecasting method. Therefore, these approaches require a large training data set that equally covers all time series characteristics. In contrast, the approach presented in this paper does not require such a large training data set. Instead, the performance of the different forecasting methods is estimated on a part of the time series to be forecast. Furthermore, no rule learning approach is required because our framework selects the forecasting method with the highest R^2 score on the validation part of the considered time series.

III. Background

In this section, the applied forecasting methods and the FedCSIS 2020 Challenge data set are briefly presented.

A. Time Series Forecasting Methods

In this paper, we apply six different forecasting methods from three different categories. The first category consists of two simple statistical features, i.e., median and mode. For time series with very little information content, forecasting these constant values can achieve a better accuracy than using more sophisticated forecasting methods. The second category are machine learning methods that require derived features as input. Here, we apply two regression techniques, i.e., Random Forest [13] and XGBoost [14]. We also use Random Forest in classification mode when the time series meets a certain requirement. In some previous works, we have already developed a novel forecasting method for seasonal, univariate time series [15], [16]. This method is called *Telescope* and is available on our GitHub repository[1]. We use Telescope in two alternative ways, which forms our third category of forecasting methods. Telescope does not require feature generation by the user. Instead, Telescope includes an internal feature generation mechanism. First, Telescope estimates the frequency of the seasonal pattern and removes anomalies in an internal preprocessing step. Next, Telescope generates features by splitting the univariate time series into seasonal, trend, and residual components. Here, a heuristic is implemented that estimates whether the time series exhibits an additive or multiplicative composition. If the composition is multiplicative, a logarithm is applied to the time series to transform the composition into additive mode. Each of the components is then forecast separately. The fine-grained seasonal pattern is continued, since the definition of seasonality states that the seasonal pattern must not change. In addition, categorical information is extracted and forecast using an artificial neural network. The trend is predicted using an ARIMA model. Finally, the categorical information, the seasonal forecast, and the trend forecast are passed to XGBoost, which recombines the forecasts and predicts the residual component. For more details on how this forecasting method works, see [15] and [16].

B. Challenge Data Set

The data set of the FedCSIS 2020 Challenge consists of network device workload traces. Each trace consists of the target variable *Mean*, the corresponding *time_window* in hourly resolution, and up to eight additional features captured between 2 December 2019 and 20 February 2020. Thus, each feature of a trace consists of a maximum of 1924 monitoring entries. However, the traces also contain missing data. These data gaps range from individual entries up to several hours. Finally, the goal of the challenge was to forecast the *Mean* one week in advance, i.e., 168 values, for 10,000 time series.

IV. Preprocessing

As our approach relies on time series forecasting, we did not go deeper into the features other than *time_window*, nor did we examine time series other than those needed to be forecast.

A. Missing Data Imputation

After analyzing the data, we found that most of the missing values are at the beginning of the time series or that only a few consecutive data points are missing. We did not reconstruct missing values at the beginning of a time series, since these gaps can extend to several hundred values. The reconstruction of such long data series is typically highly error-prone and would therefore worsen our model. In addition, missing data at the beginning is not critical, since it merely shortens the time series. To impute the missing values within the time series, we assume a daily pattern within the data. Since the data are aggregated hourly, we set this seasonal time offset to 24. In addition to the daily pattern, we analyze whether there is a trend between the day of the missing data and the next or previous day. Then, the algorithm implants the missing value by multiplying, respectively dividing, the known value one season before, respectively after, the missing value by the derived trend factor. We apply this procedure in chronological order so that imputed values can be used to impute subsequent missing values. In case that there are still few missing values after applying this method (i.e., the values one season before or after the missing value are also missing), the value is imputed by linear interpolation between the last known value before and the first known value after the missing value.

B. Anomaly Removal

While analyzing the time series, we also saw that some time series had high spikes. Since these outliers worsen the learned models, we apply a method for detecting anomalies. Here, we use a modified version of the well-known “three-sigma rule”. In contrast to the typical three sigma rule, we use the median instead of the mean value as a baseline, since the distributions of the time series are not necessarily symmetrical. Furthermore, we calculate the standard deviation only between the 1st and 99th percentile of the data, since potential outliers would already influence the standard deviation if it was calculated over the entire time series. We also set the tolerance multiplier to a more conservative value (i.e., 10-sigma rule) because we do not want to remove the normal peaks in a daily workload pattern. After detecting outliers, the algorithm overwrites these values with a linear interpolation between the non-anomalous predecessor and the non-anomalous successor.

V. Modelling

After imputing missing values and removing outliers, the main part, i.e., the modelling, takes place. Fig. 1 shows the simplified overall workflow of our approach.

A. Frequency Estimation

In the modelling part, we first estimate the seasonal frequency of the time series. Telescope already provides such a frequency estimation method, which uses a periodogram to extract the most dominant frequencies and searches for meaningful human-based frequencies nearby. Here, we have limited the possible results of the frequency estimation method to -1 (no frequency found), 24 (daily), and 168 (weekly).

B. Feature Generation

Afterwards, the lags of the univariate time series are generated. We used the lags one to six for all time series and if the time series has a seasonal pattern (i.e., the frequency is 24 or 168), we also added the lags 24 and 168 to provide not only the most recent data as features for the machine learning models, but also those from a day ago and a week ago.

In addition to the delayed time series, we also provide the hour of the day, the day of the week, and whether the day is

[1] GitHub link to Telescope: https://github.com/DescartesResearch/telescope

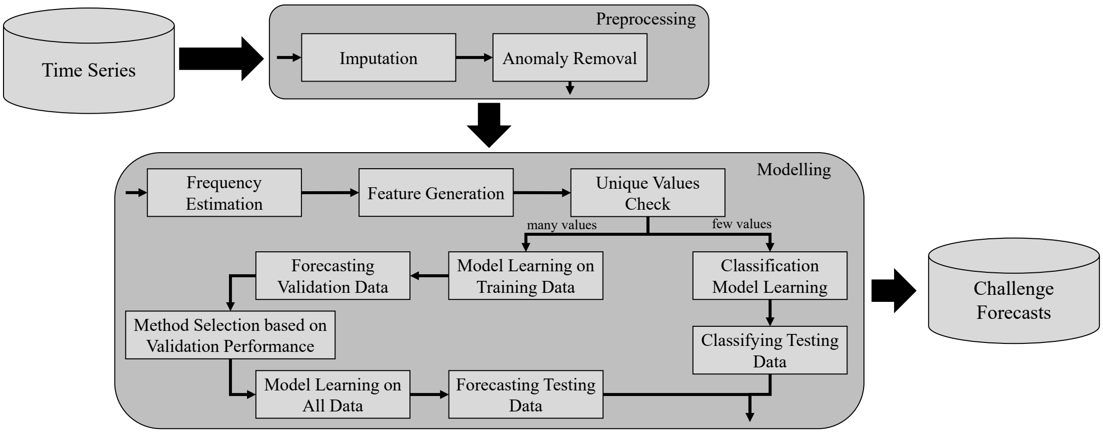


Fig. 1. The overall workflow of the framework including preprocessing and modelling.

a holiday or not as features for the machine learning models. These features are required as they can contain additional seasonal information or explain deviations from normal behavior.

C. Classification

The algorithm then determines the number of unique values within the time series. We have found that the data set contains several time series with only a few different values and, most importantly, no trend pattern. In such cases, classification can be advantageous over regression models. Thus, if we observe that a time series consists of less than six different values, we learn a random forest [13] multi-class classification model with each class representing the corresponding value.

In the specific case that the time series consists of only a single value, we predict exactly this value since the available training data does not contain any further information.

D. Regression Method Recommendation

In contrast, we apply six different regression methods when we find six or more different values in a time series ("No-Free-Lunch-Theorem"). For this purpose, we have again split the training data into a training and a validation set. The validation set consists of the last 168 values, while the training part contains all previous values. In the following, we use the term training data for this subset of the FedCSIS 2020 Challenge training data and validation data for this horizon within the FedCSIS 2020 Challenge training data. For the test data, we still refer to the unknown data that was used by the creators of the FedCSIS 2020 Challenge for the final evaluation.

The used methods are median, mode, Telescope with and without enabled frequency estimation, Random Forest, and XGBoost. The first two methods forecast the median, respectively the mode, of the training part for the entire horizon. Both versions of Telescope learn internal features but do not use the features created above, while Random Forest and XGBoost only get the lagged time series, hour of day, day of the week, and holiday as features. For both machine learning methods, we have carried out a hyper-parameter optimization. Since we predict 168 values at once, we have to fill our lag features during runtime by starting with the original values given by the training data and gradually filling them with our forecasts. That is, we forecast each value in the horizon as a one-step-ahead forecast, and after each of these one-step-ahead forecasts, we must create the feature set for the next value. However, the model remains the same, only the feature set must be recreated for each value in the forecast horizon.

To estimate the best method for a given time series, we use the validation data to calculate the R^2 score of each method. Then, we select the method that achieved the highest R^2 score and learn a new model using the entire time series (i.e., training and validation data). Finally, we forecast the 168 values using the presumably best forecasting method and adjust the forecasts under the assumption that the data should not contain negative values. Therefore, we set negative forecasts to zero if all values in the training data are non-negative. There are only few time series with negative values in the training data. For these time series, we set the forecasts that are smaller than the minimum in the training data to exactly this minimum as we interpret it as a kind of zero-baseline.

VI. Experimental Results

This section presents the experimental results of our framework based on the FedCSIS 2020 Challenge data set. First, Fig. 2 shows a time series with a gap of seven missing values. Here, the original time series is depicted in black, while the green color indicates the imputation generated by our algorithm. It can be seen that the imputation creates reasonable reconstructions. In particular, the imputation algorithm even reconstructs a first spike for the double-spiked seasonal pattern, similar to the other seasonal highs, because the algorithm considers precursors and successors with a distance equal to the frequency of the seasonal pattern.

Fig. 3 shows an exemplary anomaly removal in one of the competition time series. The black line shows the corrected time series, while the red line shows the anomalous values from the original time series. It can be seen that the peak value of the daily pattern significantly exceeds the normal range and

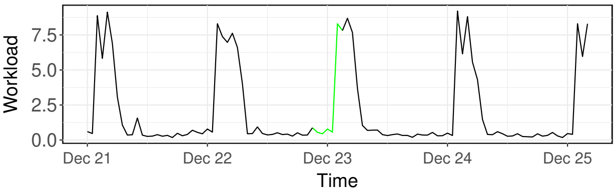


Fig. 2. An example time series with imputed values.

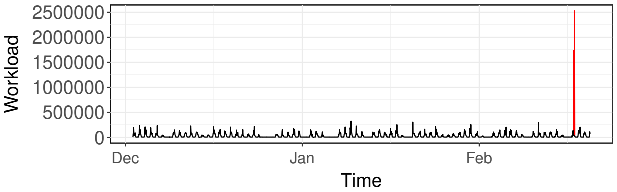


Fig. 3. An example time series with removed anomalies.

our anomaly detection method therefore overwrites these values by interpolating between the first non-anomalous precursor and the next non-anomalous successor. If such anomalies were not removed before modelling, the forecasting methods could learn a different, incorrect behavior. In particular, if the anomaly is at the end of a time series, as shown in Fig. 3, the trend component can be manipulated so that the approximation would erroneously detect an exponential trend.

The measure of the FedCSIS 2020 Challenge is the R^2 value with f_i and y_i representing the forecast and real value at time i, respectively, while $\overline{y}$ is the mean value of the time series:

$$R^2(f, y) = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \overline{y})^2}$$

The following results are based only on the small test data set of the FedCSIS 2020 Challenge. Using only single forecasting methods with the features explained in Section V, XGBoost yielded the best results with an R^2 score of -0.0072. By adding mode, median, and Random Forest regression together with the recommendation strategy, the R^2 score rose to 0.2012. After including both versions of Telescope into the set of possible forecasting methods, our framework achieved an R^2 score of 0.2544. Finally, by using Random Forest classification for time series with only a few different values, we achieved our highest R^2 score of 0.2575. Since the baseline has an R^2 score of 0.2267, our last two versions clearly surpass the baseline on the small test data set.

The distribution of forecasting methods recommended by our framework is as follows: 124 time series show no variation and therefore, the constant value is forecast. Random Forest classification is applied for 104 time series that have more than one and less than six individual values. For the remaining 9772 time series, regression is used. Mode and median are used 593 and 768 times, respectively. Although XGBoost performed best as a single method, it is only used 1510 times for the entire data set, while Random Forest regression is used most often, i.e., 3280 times. Both Telescope alternatives are applied almost equally often. Telescope without internal frequency estimation is used for 1809 time series, while the Telescope with internal frequency estimation is used for 1812 time series.

VII. Conclusion

In this paper, we introduced our approach used for the FedCSIS 2020 Data Mining Challenge. First, we imputed the time series as they contained missing values and removed anomalous peaks. To tackle the “No-Free-Lunch-Theorem”, our approach uses the corrected data to learn several models, from median and mode to univariate time series forecasting and machine learning models with lags and time information as features. Furthermore, our approach applies a recommendation to estimate the best of these methods based on the training performance for each time series. For time series with only a few different values, we apply Random Forest classification instead of regression. For the small testing set, we obtained an R^2 score of 0.2575, which clearly exceeds the baseline.

References

[1] A. Janusz, M. Przyborowski *et al.*, “Network Device Workload Prediction: A Data Mining Challenge at Knowledge Pit,” in *Proceedings of FedCSIS 2020, Sofia, Bulgaria*, 2020.

[2] D. H. Wolpert and W. G. Macready, “No free lunch theorems for optimization,” *IEEE Trans. on Evol. Computation*, vol. 1, no. 1, 1997. doi: 10.1109/4235.585893

[3] R. N. Calheiros, E. Masoumi *et al.*, “Workload prediction using arima model and its impact on cloud applications’ qos,” *IEEE Trans. on Cloud Computing*, vol. 3, no. 4, 2014. doi: 10.1109/tcc.2014.2350475

[4] K. Cetinski and M. B. Juric, “Ame-wpc: Advanced model for efficient workload prediction in the cloud,” *Journal of Network and Computer Applications*, vol. 55, 2015. doi: 10.1016/j.jnca.2015.06.001

[5] J. M. Bates and C. W. Granger, “The combination of forecasts,” *Journal of the Oper. Res. Society*, vol. 20, no. 4, 1969. doi: 10.2307/3008764

[6] R. T. Clemen, “Combining forecasts: A review and annotated bibliography,” *Int. Journal of Forecasting*, vol. 5, no. 4, 1989. doi: 10.1016/0169-2070(89)90012-5

[7] G. P. Zhang, “Time series forecasting using a hybrid arima and neural network model,” *Neurocomputing*, vol. 50, 2003. doi: 10.1016/s0925-2312(01)00702-0

[8] N. Liu, Q. Tang *et al.*, “A hybrid forecasting model with parameter optimization for short-term load forecasting of micro-grids,” *Applied Energy*, vol. 129, 2014. doi: 10.1016/j.apenergy.2014.05.023

[9] F. Collopy and J. S. Armstrong, “Rule-based forecasting: Development and validation of an expert systems approach to combining time series extrapolations,” *Management Science*, vol. 38, no. 10, 1992. doi: 10.1287/mnsc.38.10.1394

[10] B. Arinze, S.-L. Kim, and M. Anandarajan, “Combining and selecting forecasting models using rule based induction,” *Comp. & Oper. Research*, vol. 24, no. 5, 1997. doi: 10.1016/s0305-0548(96)00062-7

[11] X. Wang, K. Smith-Miles, and R. Hyndman, “Rule induction for forecasting method selection: Meta-learning the characteristics of univariate time series,” *Neurocomputing*, vol. 72, no. 10-12, 2009. doi: 10.1016/j.neucom.2008.10.017

[12] M. Züfle, A. Bauer *et al.*, “Autonomic forecasting method selection: examination and ways ahead,” in *Proceedings of ICAC 2019*. IEEE, 2019. doi: 10.1109/icac.2019.00028

[13] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, 2001.

[14] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in *Proceedings of SIGKDD 2016*, 2016. doi: 10.1145/2939672.2939785

[15] M. Züfle, A. Bauer *et al.*, “Telescope: A Hybrid Forecast Method for Univariate Time Series,” in *Proceedings of ITISE 2017*, September 2017.

[16] A. Bauer, M. Züfle *et al.*, “Telescope: An automatic feature extraction and transformation approach for time series forecasting on a level-playing field,” in *Proceedings of ICDE 2020*, April 2020. doi: 10.1109/icde48307.2020.00199

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 145–154 DOI: 10.15439/2020F168 ISSN 2300-5963 ACSIS, Vol. 21

Introducing LogDL – Log Description Language for Insights from Complex Data

Maciej Świechowski
QED Software, Warsaw, Poland
Email: maciej.swiechowski@qed.pl

Dominik Ślęzak
Institute of Informatics
University of Warsaw, Poland

***Abstract*—We propose a new logic-based language called *Log Description Language* (LogDL), designed to be a medium for the knowledge discovery workflows over complex data sets. It makes it possible to operate with the original data along with machine-learning-driven insights expressed as facts and rules, regarded as so-called *descriptive logs* characterizing the observed processes in real or virtual environments. LogDL is inspired by the research at the border of AI and games, precisely by *Game Description Language* (GDL) that was developed for *General Game Playing* (GGP). We emphasize that such formal frameworks for analyzing the gameplay data are a good prerequisite for the case of real, "not digital" processes. We also refer to *Fogs of War* (FoW) – our upcoming project related to AI in video games with limited information – whereby LogDL will be used as well.**

I. Introduction

COMPUTER languages have played crucial role in the way how people use computers and interact with them. The most common types of languages are general-purpose programming languages such as Java or C++, query (data manipulation) languages such as SQL, which are often domain specific, and description (markup) languages such as XML.

In this paper, we present a new language – *Log Description Language* (LogDL). The term "Log" was chosen deliberately as it refers to both *logic*, because LogDL is logic-based, and *logs*, i.e. information obtained from and about some process (a network activity, a video game, etc.). This ambiguity accentuates the fact that LogDL is perfect for representing both the static knowledge stored in a form of database and dynamic knowledge, i.e. new insights, inferred dynamically by reasoning mechanisms based on AI, logic, machine learning (ML) as well as computational intelligence (CI).

LogDL allows us not only for richer data representation – in terms of *descriptive logs* (d-logs in short) – but also for formal logical reasoning, spatio-temporal analysis and interactions. Because we keep LogDL as human-friendly as possible, it may be used to guide the discovery algorithms as well as for preferences specification, complex querying, data labelling and augmentation. It is also designed to be evolvable to make it a good fit for large-scale evolutionary algorithms (EA).

Modern knowledge discovery approaches need to deal with large multimodal process-related and spatio-temporal data sources such as games, sensors, monitoring, UI controllers, etc. Though the original data shall remain unstructured or multi-structured, the layer of insights can take a form of collections of well-established facts, rules and formulas expressed in LogDL – the aforementioned d-logs describing the observed processes and activities in real or virtual environments. Figure I illustrates the usage of LogDL and how it can be involved in various data-related operations and activities.

LogDL provides building blocks (e.g. facts, operators, rules) which algorithms may use, constraints (expressed by means of e.g. domains and rules) within which they operate and some built-in concepts such as time that can be interpreted automatically. Any language that is not text-based, e.g. with dynamic elements, needs a dedicated interpreter. LogDL has a few dynamic elements – already-mentioned rules and operators, interactive querying and logical reasoning based on a current knowledge base. In automated scenario, an interpreter is used by algorithms that can operate with a much higher rate than humans. Therefore, we take into consideration yet another aspect – any constructs having negative impact on performance of LogDL interpreters must be avoided.

In summary, LogDL shall enable us to: 1) be both human-friendly and computer-friendly (like scripting languages); 2) represent knowledge, e.g. from diagnostic logs, in a structured way (like SQL or *Game Description Language* – GDL); 3) perform analytical queries directly in LogDL (like in dedicated software for analytics) and represent results of those queries; 4) perform logical reasoning (like in Prolog); 5) enrich the data by custom rules and facts that can be formulated directly in LogDL; 6) provide a framework for AI/ML/CI-based knowledge discovery algorithms (like numpy in Python).

The paper is organized as follows. Section II outlines the related works. Sections III-IV are devoted to GDL and its limitations. Sections V-VII introduce some of fundamental notions related to LogDL. Sections VIII-IX are devoted to potential applications of LogDL in video games and "real world". As this is the first article about LogDL, we let these sections occupy its significant portion in order to motivate a new language. Section X concludes our work with some open questions and comments. More in-depth specifications and properties will be published in future papers.

II. Related Work

The most closely related work concerns GDL that has inspired us to develop LogDL. GDL was proposed as a way to represent

This work was co-financed by EU Smart Growth Operational Programme 2014-2020 under GameINN project POIR.01.02.00-00-0184/17-00.

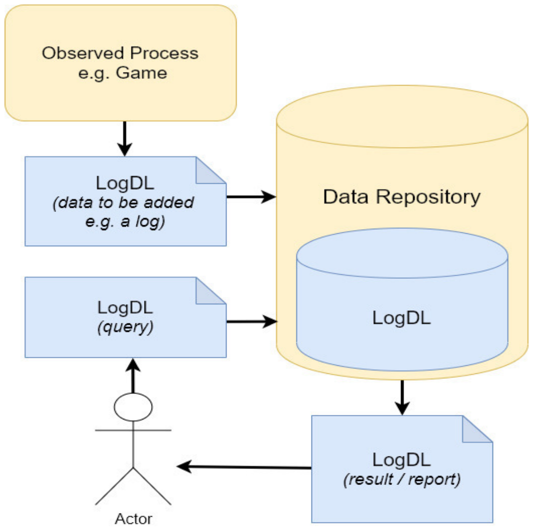


Fig. 1. The usage of LogDL from the data and knowledge processing perspectives. It stands as a medium to represent insights (in form of d-logs) derived from the original data, store them in a way which is efficiently integrated with that data and provide the means for human-computer interaction.

game rules [1]. We devote Section III entirely to it. GDL has been used to this day in the aforementioned GGP research. It is a first-order logic language that is heavily inspired by Datalog which is a logical database language [2]. GDL and Datalog are similar to each other, although not equivalent because GDL has constructions that are not a part of Datalog syntax.

One can emulate Datalog in Prolog [3], although these languages use different semantic conventions. A conversion to Prolog requires three things. Firstly, the notation is different, so each GDL element must be mapped to a Prolog counterpart. Secondly, additional code has to be written in Prolog to handle game-specific logic. Thirdly, there are special cases of negation that are handled differently in Prolog and GDL, so they have to be rewritten for Prolog. There have been numerous extensions to Datalog proposed, such as e.g. Datafun [4] – a functional oriented version and Dedalus [5] which is aimed at rich distributed services and tests for correctness.

Such languages as e.g. Ludemic-GDL [6] and Answer Set Programming [7] were proposed as well. Generally, GDL is a significant step forward as it allows to deal with the aforementioned game rules in an abstracted way decoupled from any particular game. It made it possible to create universal game-playing programs that accept games as input parameters. Nevertheless, GDL was used only in research so far.

We designed LogDL with the aim of taking the best from GDL and optimizing the rest to make it useful for the game industry and "real-world" applications. The usefulness of a language is often reflected in how efficient interpreters or compilers can be developed. Works such as [8] focus on the process of creating such interpreters and provide a good source of knowledge about GDL-style languages too. In [9], the performance of a few interpreters is compared.

Outside of the game research, there are commercial data analytics solutions such as e.g. Splunk [10]. These are services designed for a different purpose than LogDL, although it is certainly worth combining those two conceptual layers of data processing and reasoning to create efficient AI pipelines. Additionally, it is worth comparing some ideas of LogDL to those behind Complex Event Processing (CEP) [11].

LogDL is a logic-based language with a built-in reasoning mechanism. There exist commercial logical languages such as the already-discussed Prolog or 4QL [12]. Most of our comparison between GDL and LogDL translates to those languages as they are symbolic-based. From the perspective of *Fogs of War* (FoW) – our upcoming project related to AI in video games with limited information – it is also useful to look at some formal frameworks utilizing e.g. 4QL to reason about unknown and inconsistent situations [13].

There are numerous formal representations of insights derivable from the complex data. Let us mention about GVGDL – one more extension of GDL aimed at dealing with video games [14]. One may think about it as a step toward modelling "real world", though it still refers to "digital reality". There are also attempts to adapt spatio-temporal logics to model machine-generated processes, e.g. network events [15].

Last but not least, we shall refer to a collection of inspiring use-cases and potential applications of LogDL that revolve around learning new concepts and extracting new knowledge through logical reasoning, association rule mining as well as applying search-based and learning-based methods to construct new LogDL-based rules and facts. Such rules and facts constitute new knowledge that can be used for prediction, approximation and explanation [16]. This kind of strategy fits well into some of hot trends in AI, such as e.g. metaconcept learning and neuro-symbolic machine learning [17].

III. Game Description Language

GDL includes predefined keywords: *role*, *init*, *true*, *next*, *legal*, *does*, *terminal*, *goal*, *distinct*. Most of them can be treated as domain-specific extensions required to build a forward-model, i.e. simulate a game. When a GDL program is interpreted at run-time, facts can appear in three ways [18]:

- *Constant facts* are defined directly in the GDL code and they are considered *true* for the whole game. They can be regarded as "the laws of physics".
- *State facts* are parts of dynamic game states. They are initialized by *init* rules. They are cleared in each game's step, their new set is derived by *next* rules.
- *Temporary facts* are produced by non-keyworded rules. The set of such facts is derived dynamically. They are needed only temporarily in the logical resolution process initiated by one of keyworded rules.

In GGP, GDL is written using prefix Knowledge Interchange Format (KIF). It can also be represented by infix KIF or Lisp S-expressions. They are all syntactically equivalent.

A. Facts

On the top-most level, any GDL transcript consists only of *facts* and *rules*. For instance, the fact that a soldier a with rocket launcher is present in region denoted by coordinates 2 and 3 could be defined using the following form:

```
    (region 2 3 soldier rocket_launcher)
```

This is a proposition with symbols. The first symbol denotes the proposition's name. The remaining ones are arguments (also called attributes). Each proposition with the same name must have the same number of arguments. A proposition can be viewed as relation, i.e. all symbols together are in relation. There cannot be multiple facts with exactly the same symbols. Such facts would be interpreted as a single one.

GDL allows for nested facts, e.g "(weapon silver sword)" below describes a weapon of a soldier:

```
    (soldier1 2 3 (weapon silver sword))
```

Symbols have no type – they are all plain text. Apart from predefined keywords, symbols have no meaning – their interpretation is purely due to humans. In particular, game rules in GGP are often obfuscated in order to avoid any game-specific reasoning based on the choice of words, e.g. *board*. Game mechanics do not change if each unique symbol that is not a keyword is consistently changed to another one.

B. Rules

A rule in GDL can be specified as follows:

```
(<= (empty_region ?x ?y)
(true(region ?x ?y soldier ?weapon))
```

Rules are defined using the <= operator. The first proposition after it is the consequence. A rule is *true* or *false*. In addition it may produce results in form of facts that become *true*. The consequence defines a structure of such facts.

The remaining propositions are conditions that have to be satisfied in order for the rule to hold. Conditions, in contrast to facts defined directly in GDL description, can contain variables denoted by a symbol starting with ?. The variables are substituted by constant symbols in the resolution process. For example, ?x and ?y variables will be substituted by symbols 2 and 3, if the following fact holds:

```
 (region 2 3 soldier <anything>)
```

If many facts of type *region* satisfy the query, then there will be many variable bindings produced. GDL realizes the *variable unification property*: variables with the same name receive the same bindings in the rule's scope. This rule would only consider regions with coordinates equal to each other:

```
(<= (empty_region ?x ?x)
(true(region ?x ?x soldier ?weapon))
```

Conditions may refer to other rules. Facts may be *true* either through explicit specification or through rules, e.g.:

```
(cat lion)
(<= (cat ?y) (true(mammal ?y)
             (true(domestic ?y)
             (true(not(dog ?y))
```

The complete game world is defined by propositions which are *true*. There exists the completeness property which means that everything what cannot be derived as *true* from the available rules and facts at particular moment is *false*. Thus, GDL follows so-called *closed world assumption*.

IV. Limitations of GDL

Although GDL became useful in the game AI research, it has several drawbacks that hamper its wider usage. We point out the major limitations of GDL that have inspired us to develop LogDL in order to make it more applicable.

A. Poor Interpretation Performance

This is one of crucial limitations of GDL to make it more applicable in the AI workflows. The fastest GDL interpreters are based on Prolog and propositional networks [19] which instantiate all possible variables and lead to massive structures. There are cases (e.g. bigger games in GGP) in which the propositional network representation is totally infeasible.

Simulations of games in GDL are usually significantly slower than those performed by dedicated implementations. One of the reasons is that GDL is purely symbolic language. There are no built-in types that allow for taking advantage of CPU optimizations. Every piece of logic has to be written as GDL rules, whereas some aspects could be implemented more efficiently using a lower level language. Many well-established algorithms cannot be implemented efficiently because of lack of data structures such as heap, priority queue, etc.

A good example is lack of simple integer comparison operator. In GDL, it is usually implemented as follows:

```
(succ 0 1) (succ 1 2) (succ 2 3)
(<= (greater ?a ?b) (succ ?b ?a))
(<= (greater ?a ?b) (distinct ?a ?b)
                    (succ ?c ?a)
                    (greater ?c ?b))
```

B. Lack of Continuous Domains and Infinity

Another consequence of GDL being a symbolic language is that it cannot deal with continuous domains such as real numbers. For instance, it is not possible to define multiplication on real numbers, as it would require to define all possible results. There is no way of emitting new symbols that would represent the results of such operations and, even if the was, the algorithm of multiplication would have to implemented from scratch purely based on symbolic logic.

C. Lack of Stochasticity

GDL suits *finite*, *deterministic* and *synchronous* games. Accordingly, each rule in GDL is deterministic. It is either *true* or *false* given the current state. There are is no concept of randomness and no random number generators. It is debatable whether the world is deterministic or not, nevertheless, stochasticity and fuzziness are useful in modelling many real world phenomena. Moreover, there are many video games with randomness and incomplete information.

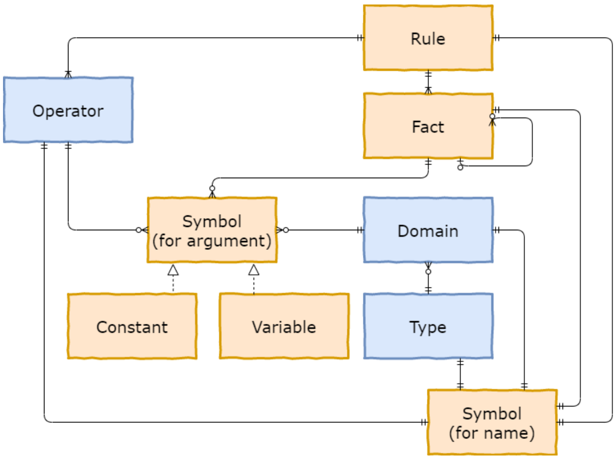


Fig. 2. Basic elements of LogDL.

D. Lack of Time-based Reasoning

In GDL, the term "synchronous" means that all players submit their actions simultaneously and then the game state is updated. Such updates are performed in consecutive frames. There is no notion of continuous time flow. We can tell that some fact appeared later than the other, but we cannot tell when exactly it happened and how much time have elapsed. There is no concept of time interval between frames.

E. Issues with Advanced Algorithms

Symbolic description without reflection, types and metadata is not well-suited for more sophisticated algorithms. For instance, let us consider the problem of rule evolution – quite important in the game industry and e.g. process mining – which could be achieved using evolutionary algorithms (EA) [20]. Consider a specific case of mutation operator that randomly perturbs a value of a certain argument. Such operator would greatly benefit from having a domain to choose values from. GDL does not support numerical domains.

As another example, crossover operator could replace a rule's condition to a different one that fits it. However, in GDL it is hard to determine whether a condition "fits" – there is nothing that could be tested against the existing rules for potential variable unifications. Moreover, such replacement could result in unexpected behavior such as a long computational time or even an infinite loop because of recursion without a proper stop condition. In GDL, there are no control statements such as *IF* and recursion caused by the interplay of rules and conditions terminates only when all its branches are evaluated as *false* at some point in the resolution process.

That said, programs written in generic programming languages such as C++ can be even more difficult to manipulate by EA, for different reasons. Although there are classes and types, there are too many degrees of freedom in how the code can be constructed. We believe that the GDL structure with rules, facts and conditions would be suitable for EA-style manipulation if only the language was extended by additional metadata. This is one of inspirations for LogDL.

F. Bloated Description

Due to lack of domain-specific operators the reasoning performance can be higher compared to programs in general purpose languages. GDL descriptions can be also extensive if they rely on concepts that are not easy to map to logical rules.

V. Log Description Language

In this section, we outline some selected ideas behind LogDL. Whenever useful, we do it in comparison to GDL. Figure 2 depicts the components of our new language. Operators, domains and types are distinguished using colors because they are not present in GDL. Symbols are similar to thosw in GDL. Symbols for arguments can either be constants (fixed literals) or variables (starting with "?"). Names can only be fixed literals and they are subject to additional uniqueness constraints, e.g. in order to avoid rules and operators with the same names or other kinds of ambiguities.

A. Facts

Facts in LogDL are similar to the GDL ones. However, we simplified the notation to make it more similar to the JSON notation and we also introduced *metadata* for arguments. The metadata consists of the argument's *name* and *domain* over a *type*, so type is indirectly part of metadata too. Domains are discussed in the next subsection. Names can be declared in a few ways, e.g. globally with simple configuration.

In GDL, arguments are specified in order of appearance, both in queries (conditions) and in implicit definitions. In LogDL, arguments can be addressed by name. The name is mapped onto the corresponding argument's index. Unnamed arguments are mapped onto those that have not yet been mapped. Here are equivalent fact definitions in LogDL:

```
1: region{2 3 soldier sword}
2: region{x: 2 y: 3 unit: soldier
        weapon: sword}
3: region{x: 2 3 soldier sword}
4: region{y: 3 2 soldier sword}
5: region{weapon: sword 2 3 soldier}
6: region{weapon: sword y: 3 x: 2 soldier}
7: region{x: 2 3 weapon: sword soldier}
```

To avoid ambiguity, i.e. which fragment is part of name or value of a specific argument, there are lows regarding a use of quotation marks and how interpreter reads characters.

The names of arguments are part of metadata. Figure 3 shows a simplified way how the data can be attributed to a particular type of fact. It also puts forward the idea of decoupling names from the data. In the implementation of LogDL, there will be optimizations for data storage such as using hash functions for logical reasoning purposes and indices that increase the performance of the expected queries.

B. Domains

Domains are associated with arguments of facts, i.e. the allowed range of their possible instantiations and the whole

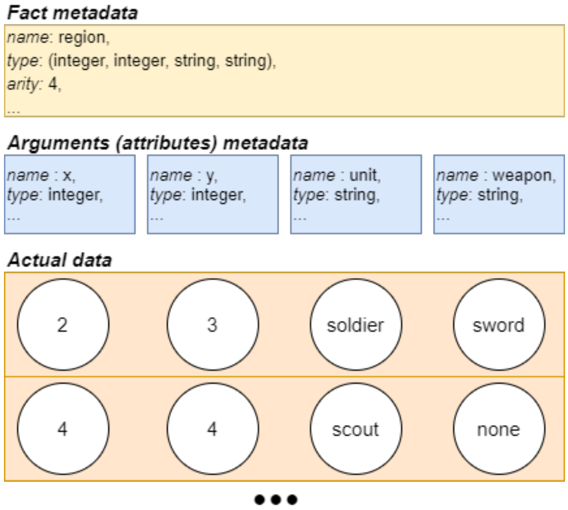


Fig. 3. Data and metadata stored for a given type of facts.

facts (complex domains). A domain is defined over a specific type. The following types are possible in LogDL:

1) Basic types – *integer*, *double*, *boolean*, *char*, *string*.
2) Complex types – tuples of other types, e.g. (*integer*, *integer*). They can be implicitly created by rules producing facts with the respective types of arguments used.

Types have to be defined as part of metadata description. Domain specifications are optional. If domains are not specified, then LogDL interpreters will make educated guess based on the data and arguments' usage. *Boolean* domain comprises always of two values: *true*, *false*. For numerical types such as *integer*, *double*, *char* (0-255), domains are defined by:

- The minimum and maximum values.
- Whether min/max values are included or excluded.
- Stride, i.e. distance between consecutive values (for *integer* and *char* it must be an integer number).

Domains can be also defined by the sets of allowed and disallowed values. If a symbol appears in both above sets, then it is considered disallowed. For numerical types, such set specifications can be combined with the range definitions. For example, we can consider a domain that has integer values from the $[0, 10]$ interval, but excluding 2 and 3.

Types and domains are part of LogDL for two main reasons:

1) They enable to introduce operators that work with specific types, which in turn allows for optimized implementation. Low performance, especially in case of math operations, is one of the main limitations of GDL.
2) They mimic what is called reflection in programming languages. A formalized structure enables us to use the AI/CI techniques that operate on the LogDL description, in particular search-/population-based EA methods.

C. Operators

Operators in LogDL can be treated as predefined functions. They are not present in GDL and we have already discussed why we believe they are important. The idea behind them is to perform certain operations more efficiently than by generic symbol manipulation and to be able to extend our language with a specialized application-specific logic.

LogDL is designed to be easily extendable by functions that can be used as conditions for rules. We are not saying that the language schema is extendable but rather a way the things are computed. This is a practical approach that also plays along with using LogDL in automated AI pipelines. Operators (just like rules) are building blocks to be used by the AI/CI algorithms, in particular EA approaches. Due to the introduction of domains and types, the algorithms can be aware of the context in which a particular operator is used.

Operators may use variables, facts or results of other operators as input parameters. The user specifies types for inputs and outputs. From LogDL point of view, the implementation is treated as a black-box. There are two categories:

1) *Built-in operators* have reserved names, i.e. keywords, in the LogDL language. By the standardization, their implementation has to be already provided by a LogDL interpreter. They are ready to be used.
2) *Custom operators* are not part of the standard. They can be declared and implemented as third-party extension to the interpreter for specific application.

Examples of built-in operators are: *logical* (conjunction, alternative, negation, etc.), *fact-related* (count, top N, getArity, etc.) and *mathematical* ($+$, $-$, square root, mean, etc.).

LogDL is still in its R&D phase, so the list of operators will continue to grow. Custom operators are provided to the LogDL interpreter by code that is compatible with the particular interpreter implementation. For example, if an already compiled interpreter is used e.g. as .NET Assembly or C++ library on Windows, then a dynamic link library (DLL) with the custom operators implementation should be provided.

D. Time and Data Organization

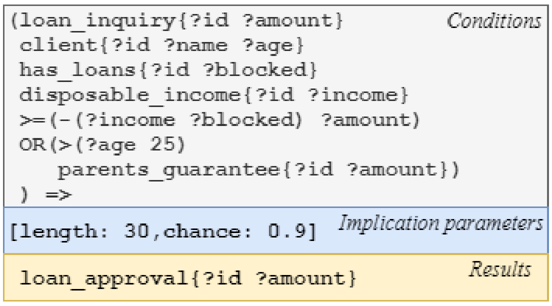
In contrast to GDL, LogDL includes the notion of time. It is defined by the reserved fact with the name "time", which has one argument of type *double*. For example:

```
time{5.0}
```

This is a time-stamp. It makes sense only if certain assumptions are made about the process that LogDL describes.

As GDL followed some constraints required to build a simulator of a game written in it, LogDL uses its specific conventions as well. First, it is assumed that LogDL is used to express the knowledge from and about processes. This suits use-cases that will be outlined in further sections.

Second, it is assumed that the process states are grouped within so-called *activities*. The concept of time is valid within the scope of the activity. Examples of activities can be: a game session, logs from a networking device, changing prices of specific stocks, health data of a specific patient. Any activity is described by a changing state in time. Another way to interpret activity is the correlated data observed chronologically. It enables to perform causal inference between events that



```
(loan_inquiry{?id ?amount}
 client{?id ?name ?age}
 has_loans{?id ?blocked}
 disposable_income{?id ?income}
 >=(-(?income ?blocked) ?amount)
 OR(>(?age 25)
    parents_guarantee{?id ?amount})
 ) =>
[length: 30,chance: 0.9]
 loan_approval{?id ?amount}
```

Fig. 4. Simple example of a rule in LogDL.

happen within the activity. There is a strong analogy between an activity and a Markov Decision Process [21].

Activities are grouped together within so-called *worlds*. With worlds, general rules (i.e. law of physics) of certain process are associated. For example, a world can denote a particular game, map in the game or a specific type of a networking device (e.g. router). The idea is that the data within the same *world* but distinct *activities* can be used to find patterns and extract knowledge about the process.

VI. Rules in LogDL

Rules are valid in the scope of *worlds*. Each rule consists of the following three parts: *conditions*, *implication parameters* and *results*. LogDL rule's structure is as follows:

```
(conditions) =>
[implication_parameters]{results}
```

The respective parts are distinguished with different colors in Figure 4. Implication parameters are optional to specify – there are default values in our LogDL language.

A. Conditions

In LogDL, conditions are either *fact propositions*, like in GDL, or *operators* that specifically return a boolean value. This is new compared to GDL. Facts and operators used as conditions may contain variables. Each condition is evaluated as *true* or *false*. If it contains variables, it provides instantiations (bindings) of those variables just like in GDL.

There are certain constraints imposed on rule conditions, which makes reasoning simpler and potentially faster. Conditions are checked in the order they are defined. Each condition may introduce new variables and use the already introduced ones. Naturally, a condition may be just a check, i.e. without any variables at all. The introduced variables are those with names that have not been used so far by conditions checked earlier. We refer to Table I to see how introduction of variables and their usage in conditions is defined. Variables in boldface are the ones that are introduced by a condition. The results of checking conditions in LogDL are:

- A boolean value indicating whether conditions as a whole are evaluated as *true* or *false*.
- If *true*: a list of records satisfying all variables.

Let us now list constraints that allow us for efficient implementation of LogDL interpreters. They are imposed on conditions starting from the second one in the order of appearance:

1) Operators cannot introduce new variables. However, they can use the already existing ones.
2) Conditions defined by facts should either: a) use at least one already introduced variable; b) do not have variables at all; c) have only one valid instantiation.

B. Results

This part of our language is analogous to GDL, i.e. it consists of specification of facts that become *true* if the rule is satisfied. The produced facts can include constants and any variables that have been introduced in the previous subsection. Implication parameters that are discussed below define the probability and time constraints, i.e. "from when" and "for how long" the results are considered *true*.

C. Implication Parameters

This aspect is new compared to GDL. It is a construction that describes additional context how results are created. All such parameters have default values in order to allow for concise expressions. They consist of the following elements:

1) *resultStartTime* is the expected delay time between the moment when conditions are met and results are produced. It is measured in the same units as LogDL's *time*. The default value is equal to 0.
2) *length* is the time that results are expected to hold, i.e. from *resultStartTime* to *resultStartTime + length*. This can be set to a real value or two special values *UPKEEP* and *PERSISTENT*. *UPKEEP* denotes that the rule effects will hold as long as its conditions do. The *PERSISTENT* option denotes that the rule effects will be persistent in the scope of the context the rule is launched.
3) *chance* is conditional probability. If conditions are met, then results will be produced with probability equal to this value. The default is 1, i.e. if not specified otherwise, rules will always produce results immediately and results will be valid as long as conditions are valid.

We may also consider a fuzzy component, e.g. by reserving the first argument of each fact as its satisfaction degree $\in [0, 1]$. In contrast to purely symbolic GDL, LogDL could handle fuzzy membership functions, fuzzy literals and the overall *computing with words* paradigm [22]. It might be worthwhile to make it possible to configure the logical reasoning mechanism, so it uses fuzzy norms to determine whether a rule is satisfied and to what degree. Rules and operators could be then used to perform the fuzzification and defuzzification processes.

VII. LogDL Compilation

Figure 5 shows typical environment for the LogDL usage. *LogDL description* is a code written in LogDL that is based on facts, rules and operators. *LogDL metadata* are definitions of global aspects such as domains and types. Both the LogDL program and metadata together form *data repository*. One of unique aspects of languages such as LogDL or GDL is that

TABLE I
AN EXAMPLE OF CONDITIONS OF A RULE. EACH ROW IS A TOP-LEVEL CONDITION. THE LAST TWO ROWS CONTAIN NESTED CONDITION.THE THIRD COLUMN DENOTES THE TRACKING OF THE INTRODUCED VARIABLES AFTER EACH CONDITION IS TAKEN INTO ACCOUNT.

Condition	Condition type	No. of variables new / total
loan_inquiry{**?id ?amount**}	fact	2 / 2
client{?id **?name ?age**}	fact	2 / 4
has_loans{?id **?blocked**}	fact	1 / 5
disposable_income{?id **?income**}	fact	1 / 6
>=(-(?income ?blocked) ?amount)	operator using nested operator	0 / 6
OR(>(?age 25) parents_guarantee?id ?amount))	operator nested operator nested fact	0 / 6

the code and the data are essentially inseparable, e.g. a standalone definition of a fact means that it is *true*, therefore it is a part of data.

In our proposal, which is still in development, we designed various ways to provide *LogDL description* to a repository, e.g. using a file (*.logdl), interactively as in e.g. Python console, or programmatically through a dedicated API. The last case can be useful e.g. when a game, or another data provider, is able to log the structured data in LogDL format.

LogDL metadata can be either provided through files or in a GUI-based *administration suite* which is the preferred approach The idea behind it is to have an easy to use, graphical tool to define the structure of particular problem to be modelled using LogDL. It can allow for defining all kinds of metadata, organizing the data by *worlds* and *activities* (see Section V-D), viewing / updating / deleting all kinds of the data as well as providing LogDL compiler with implementation of custom operators and making them visible for LogDL.

We assume dedicated LogDL compilers written in a few programming languages. Having a compiler for specific language allows for two features: (1) extending LogDL with custom operators with a native implementation as well as (2) having access to the interpreter from a code in the host language. (1) requires some metadata to bind LogDL expression with functions exported from a library that contains implementation. This includes providing a way to call the function, reserve its corresponding operator name as well as provide metadata for its arguments. All of this will be possible to set up via the GUI-based *administration suite*.

When a LogDL program is available, the user can utilize the interpreter either as interactive program (like in Python console) or directly from the user's application through API. The latter is possible if there exists an interpreter for a programming language of the host application. We will provide bindings to the most popular languages. First and foremost, the interpreter allows for interactive queries based on rules, facts and operators. The user may e.g. wish to perform a logical resolution, check a hypothesis or just fetch or count the specific data. The program can also be compiled without the interactive interpreter. In such case, queries must be provided beforehand, e.g. via a file. Then the compiler will compile the file to a program returning results of specified queries.

VIII. MOTIVATION – VIDEO GAMES

The subsequent sections are devoted to potential LogDL use-cases. First, we motivate why logic, structured logs and LogDL can be useful for the game industry domain. In particular, we intend to apply it in our upcoming projects. We have already carried out a prototype implementation aimed at verification of the expressive power of LogDL, its integrity and ease of use. Selected aspects have also been implemented in an optimized way, together with the tests measuring what kind of performance can be expected when the whole ecosystem outlined in Section VII is developed and integrated.

A. AI Development

The traditional approach to AI in commercial video games is extensively based on heuristics. A heuristic can be part of search algorithms, functionally realized by finite state machines or in form of behavior trees [23]. Heuristics require the expert knowledge expressed by means of some important aspects of the game environment, preferences, weights, probabilities, threshold values and utility values. For example – *it is worth shooting the opponent with weight X if the distance to it is less than Y and otherwise it is worth running away*. All components and parameters are usually chosen by a repetitive trial-and-error method or chosen arbitrarily.

LogDL promotes a different, evidence-based approach. Human game testers are usually the biggest group of people involved in the game production process. Logs from such test runs can provide a valuable data source, based on which game creators may build and tune heuristics. LogDL is particularly useful to represent facts from game replays and new insights that can be discovered from them. For example it may find choke-points on maps, usefulness of in-game items and how different strategies work against each other.

LogDL was in part inspired by our experience with the *Grail* library aimed at developing AI in video games [24]. Grail supports algorithms such as Utility AI and Monte Carlo Tree Search (MCTS) [25] which can be used as action-selection mechanism for AI players. Utility AI is based on curves that define relationship between an action's utility and a given consideration. Identification of considerations can be extracted from logs thanks to LogDL. Similarly, in video games MCTS is typically optimized with heuristics that provide early cut-off (i.e. scores in non-terminal states), limit the number of considered actions or guide the search process.

B. Game Testing and QA

When logs from games are available, LogDL can be a valuable tool for Quality Assurance (QA) [26]. Firstly, it can be used for balancing, i.e. identifying too strong aspects of the game, e.g. a weapon that inflicts too much damage or an enemy that cannot be reliably defeated. This is a similar case to developing the AI, but this time we are interested in other types of insights from the data. Secondly, it can be used to verify hypotheses about the game. For example a hypothesis may state that *60%*

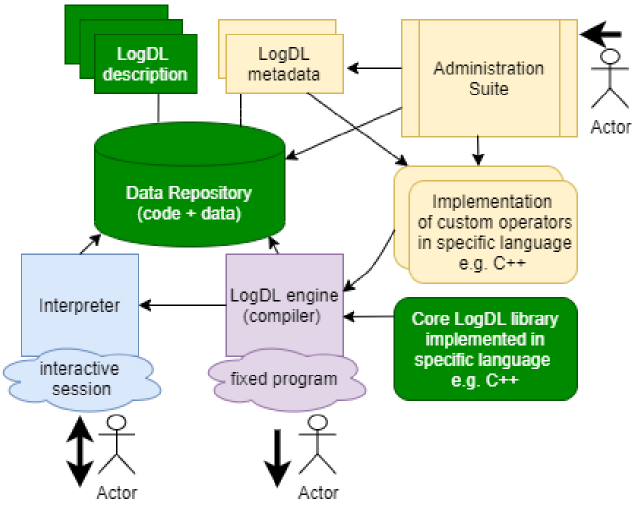

LogDL description
LogDL metadata
Administration Suite
Actor
Data Repository (code + data)
Implementation of custom operators in specific language e.g. C++
Interpreter
LogDL engine (compiler)
Core LogDL library implemented in specific language e.g. C++
interactive session
fixed program
Actor
Actor

Fig. 5. Technical overview of the usage of LogDL.

of the time a player is able to finish a particular level without losing life. LogDL is particularly suitable to query the game data with its structured form and the notion of time and space. Thirdly, the QA requirements can be expressed as queries and rules directly in LogDL. This enables to build an automated or semi-automated QA pipeline similar to continuous software integration systems. Lastly, LogDL can aid automated QA provided that the testing agents (bots) are available. The data in LogDL can be analyzed on the fly thanks to the reasoning mechanisms and custom rules provided once for the testing process. It makes it possible to guide agents in real-time in their testing behavior. For example, it can be revealed that certain game areas or interactions need to be tested more thoroughly. Although the automated testing is not common in the video game industry yet, it will be more popular in future with the AI becoming smarter.

C. Game Analytics

Game analytics and e-sport are one of the hottest topics not only in games but in entertainment, in general. LogDL allows for transformation of raw information collected from games into useful information that can be presented to players, teams, sponsors or game development companies. On a technical level, this use-case is related to the previous ones, i.e. developing AI, testing and QA. LogDL is designed to be queried interactively, used in the knowledge discovery processes and to provide new insights from the data. Such insights can be utilized in game analytics and coaching to increase human players' skills [27] or to comment e-sport games.

Fact-based and rule-based description can integrate the gameplay data with metadata about players as well as maps and concepts that are related to the given game but not actually present within it. It also includes the data obtained as input from players, game developers and data scientists. The rules can be continuously refined in an incremental self-improving process with human feedback in the loop. For example, logical analysis could find inconsistencies, potential candidates for anomalies, unexpected correlations, new strategies or just new concepts that human experts could use and name.

D. Explainability in Games

AI behavior that is understood and trusted is not only a requirement for high-risk applications such as e.g. those in medical field. Game studios extensively test the AI in their games to minimize the risk of players encountering unexpected or unnatural behavior in the shipped game [28].

The other facet of explainability refers to "cheating AI". In multi-player games with bots, players often do not believe that their AI opponents played with the same rules. Cheating AI is so prevalent in games without perfect information (especially in real-time strategy games) to make up for poor strategic skills, that if a bot in a particular game is competent, then it is often accused of cheating. Hidden information is important, because players cannot verify during the game whether the bot plays along the same game rules and limitations.

FoW – our already-mentioned new R&D project – will concentrate on believable AI bots in games with hidden information and on explaining their limitations and reasoning processes behind actions. LogDL is good for such applications, particularly when it serves for representing game logs too. It can be used to express the knowledge of bots and explanatory rules of their behaviors in an accurate or approximate way depending on what kind of AI algorithms are applied.

E. Multi-Agent Communication

Let us refer to FoW once again, now from the viewpoint of creating tools for game developers, which are aimed at reasoning under uncertainty (hidden and stochastic information), managing and updating beliefs of computer agents as well as their coordination and communication in a multi-agent environment [29]. One of our goals is to simulate human-like behavior in games with imperfect information with human-plausible simulation of perception. Formal languages, e.g. the aforementioned 4QL, can be used for modelling multi-agent interactions. However, they are often too difficult to apply in the industry. LogDL shares similarities to 4QL and other logic-based languages, however, it is simpler and reasoning is faster what fits better into game production pipelines.

F. Mechanics, Prototyping, Narrative Design

Tools aiding designers to create a plot in games have gained popularity in recent years [30]. They are typically based on graphs that contain events, branches, triggers and milestones that define progression in the game. They often allow for specifying the "keys and doors" systems, i.e. the goals that have to be achieved to unlock a certain game's aspect. LogDL can help in two ways in such use-case. All elements of the narrative can be expressed in it in a form of facts and rules. However, more importantly, due to the Prolog-style logical reasoning, it can provide immediate feedback to the designers. For example, LogDL queries can check whether the game can be completed, in how many ways, what is the most probable or efficient way, etc. Thanks to the extensive nature and ability to add custom operators, a narrative system build on top of LogDL can be fine tuned for a particular game.

In GGP, game rules are described in GDL and game-playing algorithms use such descriptions to conduct simulations. We believe that LogDL could be employed in a similar fashion to express the core mechanics and rules in video games. Due to complexity of such games, their descriptions would have to be high-level game approximations. Nevertheless, it could be useful for rapid prototyping aimed at testing the soundness of ideas in the creative design process [31].

IX. Motivation – Real World

Below we present some real-world use-cases that we intend to investigate. By a use-case we mean an area, wherein LogDL can bring extra value. The data and ML-related challenges in video games and "real world" are similar to each other [32]. This is why LogDL can be useful in both cases.

A. Process Mining

In general, LogDL is suitable for applications where the data needs to be stored, analyzed and new – most likely difficult to predict – insights from the existing data can be discovered. Good examples are innovative R&D applications, where new knowledge can emerge and provide technological advantage. For instance, it can help to gather information about any given process in the form of rules that describe it [33]. Imagine using process mining to discover weather patterns, laws of physics, proving hypotheses, making scientific discoveries, analyzing art, texts, making reverse-engineered models. Works such as [34] illustrate that the analytics of processes requires significant effort at the level of concept formation. Once appropriate concepts are specified, one can reason about their occurrences in the data in a logical fashion.

B. Business Intelligence

This is a field wherein the state-of-the-art data processing approaches are often applied. LogDL can be used as a glue that binds together logs, domain-specific concepts which subject matter experts understand and the automated data science that they usually are not deeply familiar with. A system that incorporates LogDL can be developed in such a way that technical AI/ML details are hidden and friendly interfaces are exposed. Actually, our aforementioned system that advises players how to improve their skills can be treated as an example of business intelligence development in the game industry [27]. Similarly, the analytics can be conducted in "real world" over complex multimodal data sources [35]. In both scenarios, the original data needs to be first digested/enriched – in the already-discussed form of d-logs – and then the main business intelligence layers can be employed.

C. Hierarchical Learning

In a typical ML scenario, languages such as JSON are used only in the first step of the data processing pipeline – to store the input data. However, the phase of reasoning about structured information expressed by LogDL-based d-logs can be still a part of a discovery process. Such an approach, i.e. to combine numerical and symbolic ML techniques (in our case: deriving d-logs from the raw data and reasoning about them) has long stayed under the radar, but recently it has been attracting researchers' attention. This is first, to provide the end-users with the explainable ML models and second, to utilize the layer of d-logs for other purposes, such as the above-discussed process mining or business intelligence.

The logical layer can also reinforce ML-based algorithms with reasoning and rule mining performed on a higher level, e.g. on definitions, features and concepts discovered by the ML-based algorithm such as neural networks [17]. This way, a natural hierarchical system can be achieved [36].

Good examples can be found in applications, where the data from videos or pictures is analyzed by convolutional neural networks (CNN) that are suitable for extracting low level and local features [37]. Methods that manipulate a LogDL description could take it from there and use those local features to induce/infer higher-level concepts. Works such as [38] demonstrate efficiency of such hybrid approach with respect to multimodal spatio-temporal data, whereby LogDL could be additionally used to express the domain knowledge of subject matter experts at that higher level of abstraction.

D. Interactive Analytics

LogDL enables to be interactively queried and manipulated by the AI/CI/ML algorithms which typically run in a continuous loop. It provides advantages of logic-based languages in terms of reasoning as well as robustness and flexibility of database languages. We believe that there is a need for such a medium that can be used interactively by humans algorithms.

In particular, one may consider utilizing LogDL rules for *real-time annotations* or *feedback* that consists of comments of the data. Examples of the corresponding applications refer to augmented reality, virtual reality, streaming platforms, self-driving cars, etc. Methods operating with logic and rules could help human investigators analyze large chunks of machine-generated or sensory data [39]. Such data usually consists mostly of the cases that are not particularly interesting and only at certain points there are photos or video frames that need human attention [40]. Rule-based systems combined with feature extraction may help to narrow down the cases and show only the most interesting ones to human operators.

X. Conclusions

We introduced a new logic-based language – LogDL – for complex data and knowledge discovery workflows. Examples of usefulness have been shown by motivations from both, video games and "real world" applications outside of the entertainment industry. The GDL language from the game research domain was the protoplast for LogDL. However, there are significant differences between them, e.g. types, domains, custom operators, time, probabilistic elements, etc.

Our goal was to combine advantages of data representation languages such as JSON and query languages such as SQL and introduce a necessary formalism to take advantage of the AI-driven knowledge discovery. In future, we plan to create highly efficient LogDL-based system integrating manual management and the AI/CI methods in the process of discovering new concepts/insights from the complex data. We believe that at foundations of such system there should be a language that can allow for logical reasoning, can be interactively queried and manipulated by intelligent (e.g. EA) algorithms.

A feature that could be beneficial to such manipulations is built-in granularity (level of detail) of rules [41]. It is already possible to define rules that operate with various granularities, but it is up to human interpretation and transparent from the LogDL's viewpoint. We could introduce a convention, e.g. that rules with the same name and a dedicated argument denoting the level of granularity describe the same concept.

Another aspect refers to our aforementioned project FoW, whereby LogDL will be used for reasoning (under uncertainty) by AI players and explanations for human players. Therein, we will need to decide whether to follow the GDL-style closed world assumption or rather make the world "open", like e.g. in case of Action Description Language (ADL) [42].

We also plan to integrate modern ML techniques (e.g. deep learning) which are powerful but difficult to explain with a logical/symbolic top-layer. Such combination not only would make using such a system more interpretable and trustful for human operators but it could also lead to discovery of new knowledge using the concepts defined within LogDL.

REFERENCES

[1] M. R. Genesereth, N. Love, and B. Pell, "General Game Playing: Overview of the AAAI Competition," *AI Magazine*, vol. 26, no. 2, pp. 62–72, 2005.

[2] S. Greco and C. Molinaro, "Datalog and Logic Databases," *Synthesis Lectures on Data Management*, vol. 7, no. 2, pp. 1–169, 2015.

[3] V. S. Costa, R. Rocha, and L. Damas, "The YAP Prolog System," *Theory and Practice of Logic Programming*, vol. 12, pp. 5–34, 2012.

[4] M. Arntzenius and N. R. Krishnaswami, "Datafun: A Functional Datalog," in *Proc. of ICFP 2016*, pp. 214–227.

[5] P. Alvaro, W. R. Marczak, N. Conway, J. M. Hellerstein, D. Maier, and R. Sears, "Dedalus: Datalog in Time and Space," in *Proc. of Datalog 2010*, pp. 262–281.

[6] E. Piette, M. Stephenson, D. J. Soemers, and C. Browne, "An Empirical Evaluation of Two General Game Systems: Ludii and RBG," in *Proc. of CoG 2019*, 2019, pp. 1–4.

[7] M. Thielscher, "Answer Set Programming for Single-Player Games in General Game Playing," in *Proc. of ICLP 2009*, pp. 327–341.

[8] J. Kowalski and M. Szykuła, "Game Description Language Compiler Construction," in *Proc. of Australasian AI 2013*, pp. 234–245.

[9] Y. Björnsson and S. Schiffel, "Comparison of GDL Reasoners," in *Proc. of GIGA@IJCAI 2013*, pp. 55–62.

[10] M. Okumura and S. Fujimura, "Constructing a Log Collecting System using Splunk and its Application for Service Support," in *Proc. of SIGUCCS 2016*, pp. 103–106.

[11] O. Etzion and P. Niblett, *Event Processing in Action.* Manning Publications, 2010.

[12] J. Małuszyński and A. Szałas, "Logical Foundations and Complexity of 4QL, a Query Language with Unrestricted Negation," *Journal of Applied Non-Classical Logics*, vol. 21, no. 2, pp. 211–232, 2011.

[13] B. Dunin-Kęplicz and A. Strachocka, "Paraconsistent Multi-party Persuasion in TalkLOG," in *Proc. of PRIMA 2015*, pp. 265–283.

[14] M. Ebner, J. Levine, S. M. Lucas, T. Schaul, T. Thompson, and J. Togelius, "Towards a Video Game Description Language," in *Artificial and Computational Intelligence in Games.* Dagstuhl, 2013, pp. 85–100.

[15] I. Haghighi, A. Jones, Z. Kong, E. Bartocci, R. Grosu, and C. Belta, "SpaTeL: A Novel Spatial-Temporal Logic and Its Applications to Networked Systems," in *Proc. of HSCC 2015*, pp. 189–198.

[16] D. Pedreschi, F. Giannotti, R. Guidotti, A. Monreale, S. Ruggieri, and F. Turini, "Meaningful Explanations of Black Box AI Decision Systems," in *Proc. of AAAI 2019*, pp. 9780–9784.

[17] M. H. Segler and M. P. Waller, "Neural-Symbolic Machine Learning for Retrosynthesis and Reaction Prediction," *Chemistry – A European Journal*, vol. 23, no. 25, pp. 5966–5971, 2017.

[18] M. Świechowski and J. Mańdziuk, "Fast Interpreter for Logical Reasoning in General Game Playing," *Journal of Logic and Computation*, vol. 26, no. 5, pp. 1697–1727, 2016.

[19] C. F. Sironi and M. H. M. Winands, "Optimizing Propositional Networks," in *Proc. of CGW@IJCAI 2016*, pp. 133–151.

[20] J. C. Tay and N. B. Ho, "Evolving Dispatching Rules using Genetic Programming for Solving Multi-Objective Flexible Job-Shop Problems," *Computers & Industrial Engineering*, vol. 54, no. 3, pp. 453–473, 2008.

[21] D. J. Lizotte and E. B. Laber, "Multi-Objective Markov Decision Processes for Data-Driven Decision Support," *Journal of Machine Learning Research*, vol. 17, pp. 211:1–211:28, 2016.

[22] L. A. Zadeh, *Computing with Words – Principal Concepts and Ideas.* Springer, 2012.

[23] D. Mark, *Behavioral Mathematics for Game AI.* Cengage Learning, 2009.

[24] M. Świechowski and D. Ślęzak, "Grail: A Framework for Adaptive and Believable AI in Video Games," in *Proc. of WI 2018*, pp. 762–765.

[25] M. Świechowski, T. Tajmajer, and A. Janusz, "Improving Hearthstone AI by Combining MCTS and Supervised Learning Algorithms," in *Proc. of CIG 2018*, pp. 445–452.

[26] D. Irish, *The Game Producer's Handbook.* Cengage Learning, 2005.

[27] A. Janusz, D. Ślęzak, S. Stawicki, and K. Stencel, "SENSEI: An Intelligent Advisory System for the eSport Community and Casual Players," in *Proc. of WI 2018*, pp. 754–757.

[28] M. Świechowski, "Game AI Competitions: Motivation for the Imitation Game-Playing Competition," in *Proc. of FedCSIS 2020*, pp. 155–160.

[29] B. Dunin-Kęplicz and R. Verbrugge, *Teamwork in Multi-Agent Systems – A Formal Approach.* Wiley, 2010.

[30] G. N. Yannakakis and J. Togelius, "A Panorama of Artificial and Computational Intelligence in Games," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 7, no. 4, pp. 317–335, 2014.

[31] J. Ruan, W. Van Der Hoek, and M. Wooldridge, "Verification of Games in the Game Description Language," *Journal of Logic and Computation*, vol. 19, no. 6, pp. 1127–1156, 2009.

[32] J. Togelius, "AI Researchers, Video Games Are Your Friends!" in *Proc. of IJCCI 2015*, pp. 3–18.

[33] W. Van Der Aalst, "Process Mining," *Communications of the ACM*, vol. 55, no. 8, pp. 76–83, 2012.

[34] T. Kawamura, T. Kimura, and S. Tsumoto, "Estimation of Service Quality of a Hospital Information System Using a Service Log," *The Review of Socionetwork Strategies*, vol. 8, no. 2, pp. 53–68, 2014.

[35] L. Dey, I. Verma, A. Khurdiya, and S. B. H., "A Framework to Integrate Unstructured and Structured Data for Enterprise Analytics," in *Proc. of FUSION 2013*, pp. 1988–1995.

[36] P. MacAlpine, M. Depinet, and P. Stone, "UT Austin Villa 2014: RoboCup 3D Simulation League Champion via Overlapping Layered Learning," in *Proc. of AAAI 2015*, pp. 2842–2848.

[37] M. Przyborowski, T. Tajmajer, Ł. Grad, A. Janusz, P. Biczyk, and D. Ślęzak, "Toward Machine Learning on Granulated Data – A Case of Compact Autoencoder-based Representations of Satellite Images," in *Proc. of Big Data 2018*, pp. 2657–2662.

[38] J. Ludziejewski, Ł. Grad, Ł. Przebinda, and T. Tajmajer, "Integrated Human Tracking Based on Video and Smartphone Signal Processing within the Arahub System," in *Proc. of FedCSIS 2020.*

[39] G. J. Nalepa, E. Brzychczy, and S. Bobek, "On the Opportunities for Using Mobile Devices for Activity Monitoring and Understanding in Mining Applications," in *Proc. of IDEAL (2) 2018*, pp. 75–83.

[40] C. Han, J. Mao, C. Gan, J. Tenenbaum, and J. Wu, "Visual Concept-Metaconcept Learning," in *Proc. of NeurIPS 2019*, pp. 5002–5013.

[41] M. Świechowski and D. Ślęzak, "Granular Games in Real-Time Environment," in *Workshop Proc. of ICDM 2018*, pp. 462–469.

[42] R. Reiter, *Logical Foundations for Specifying and Implementing Dynamical Systems.* MIT Press, 2001.

DOI: 10.15439/2020F126
ISSN 2300-5963 ACSIS, Vol. 21

Game AI Competitions: Motivation for the Imitation Game-Playing Competition

Maciej Świechowski
QED Software
Warsaw, Poland
maciej.swiechowski@qed.pl
0000-0002-8941-3199

***Abstract*—Games have played crucial role in advancing research in Artificial Intelligence and tracking its progress. In this article, a new proposal for game AI competition is presented. The goal is to create computer players which can learn and mimic the behavior of particular human players given access to their game records. We motivate usefulness of such an approach in various aspects, e.g., new ways of understanding what constitutes the human-like AI or how well it fits into the existing game production workflows. This competition may integrate many problems such as learning, representation, approximation and compression of AI, pattern recognition, knowledge extraction etc. This leads to multi-directional implications both on research and industry. In addition to the proposal, we include a short survey of the available game AI competitions.**

I. Introduction

EVER since the inception of the first computers, making machines capable of playing games has been viewed as an opportunity to test their intelligence. Alan Turing was one of the pioneers of this idea [1]. The first games that became frameworks for Artificial Intelligence (AI) were checkers [2] and chess [3]. The latter has even been referred to as "The Drosophila of AI", because it has been studied extensively and this is a parallel to a type of fly in biology that was often featured in research. Games have become one of the most important testbeds for *Aritifical Intelligence* (AI). The main reasons are that they are relatively *cheap*, *deterministic*, *easily repeatable and controllable* as well as *enterntaining* testing environments. Moreover, many problems encountered in games, reflect some real-world problems, which is especially significant in modern video games.

We have gone a long way from those early research to famous competitions between a man and a machine. The most notable ones were IBM's Deep Blue triumph over Garry Kasparov [4] in chess and IBM Watson winning against human champions in Jeopardy! [5]. Research in computer chess after the last Ultimate Computer Chess Challenge [6] in 2007 shifted towards deep learning [7] and human-like approaches. More recently, one of the major breakthroughs, not only in games but in AI in general [8], was highlighted with the match between Lee Sedol and Google DeepMind's AlphaGo [9] in Go. For decades, computers could defeat the top human players only in simpler abstract games. Many of them have been solved, e.g. Connect-4 [10] and Checkers [11]. However, with successes of DeepMind's AlphaStar [12] in Starcraft and OpenAI Dota-Five [13] in Dota, computer bots have finally started to achieve human-level performance in video games too.

In this paper, we propose a new type of competition for game-playing bots. The main idea is to construct computer programs that are capable of playing a game in the most similar way to a given human player. There are numerous reasons of why we think it is useful for research community and game development industry as well. The idea is based on both research experience related to game AI [14], [15], [16] as well as commercial experience in working with AI engine for game studios [17].

The next section is devoted to a short survey of the major modern competitions regarding game-based AI. In Section III, we present the motivation for the new competition. Finally, the last section concludes the paper.

II. Annual Game AI Competitions

In this section, we refer to the major modern competitions for AI in games. We focus on the goal of each competition. Readers interested in particular competitions are advised to follow the references given. There have not been suitable research references to the last two of the presented competitions, so we present the URLs instead.

General Game Playing (GGP) [15] - proposed by Stanford Logic Group [18], is one of the oldest on this list, being first hosted in 2005. It has been organized during either AAAI or IJCAI conferences. In 2006 and 2007, there was a $10K prize available for the winner. The idea is to create computer programs capable of playing any finite, deterministic, synchronous games, even previously unknown ones. Our program, named *MINI-Player* [19], has reached the quarter-final level twice. The games are given in the so-called Game Description Language (GDL). Abstract combinatorial games have been used including mostly simple board games. Participating agents are given relatively short amount of time (e.g. 1 minute) for preparation before each match and some shorter time for each move (e.g. 20 seconds). The

Research co-funded by the Smart Growth Operational Programme 2014-2020, financed by the European Regional Development Fund under GameINN project POIR.01.02.00-00-0184/17, operated by the National Centre for Research and Development (NCBR).

strongest players are based on the Monte Carlo Tree Search algorithm [20].

General Video Game AI (GVG-AI) [21] - first hosted in 2014, now part of the IEEE Conference on Games (CoG). The idea stemmed from GGP and both competitions share many similarities. The motivation is to develop universal, online learning-based methods with as little of game-dependent heuristics as possible. However, instead of combinatorial games as in GGP, GVG-AI employs simple video games. Most of them resemble the old games played on the Atari computer. They are represented in Video Game Description Language (VGDL), which has been inspired by GDL. Participating bots have 1 second for initialization and 40-50*ms* per each move, so the action is more fast-paced compared to GGP. However, similar techniques seem to work the best such as MCTS or Rolling Horizon Evolutionary Algorithms. Currently, the competition runs in a few tracks such as Single Player Planning Track, 2/N-Player Planning Track and Level Generation Track.

Arimaa Challenge [22] - Arimaa is a game designed to be playable with a standard set of chess but much more difficult for computer agents. However, for human players the game is not considered more difficult to play than chess despite having a much higher branching factor of approx. 17000. For reference, the branching factor of chess is 35. The competition has been oficially held since 2004. There was a prize available for the authors of the computer programs to beat human experts, called *defenders*, using a standard, off-the-shelf hardware. The prize ranged from $10K to $17.5K depending on the year. The prize was first claimed in 2015 by the program named *Sharp*. The competition has been discontinued since 2020.

Starcraft AI [23] - Starcraft is a very succesful real-time strategy game developed by Blizzard Entertainment. It has been particularly interesting for research community [24]. It is a popular e-sport game as well, what makes it even more appealing. The original full game of Starcraft is used thanks to the API made available for programmers, called *BWAPI*, to develop their own agents. The competition started in 2010. It has been organized alongside AIIDE and IEEE CIG competitions. The goal is to create strong Starcraft bots that are capable of both strategic and tactical reasoning, resource gathering, base building, managing build orders and battle micro-management. The organizes provide an open-source implementation of the agent called *UAlbertaBot*, which participated in all competitions so far and won in 2013. It has become both the entry point for new developers and the baseline to compare against. Despite the fact that this competition uses a specific game, unlike GGP and GVGGP, the winning solutions display a variety of techniques including multi-agent systems, MCTS, real-time planning, hierarchical task networks, graph searching algorithms, path-finding, dynamic scripting, neural networks, Q-learning, decision trees, and lots of heuristics or hard-coded strategies.

microRTS AI [25] - proposed as an alternative to Starcraft Competition with the goal of being more abstract and more accessible. Starcraft is a complex commercial game with a relatively difficult API to work with in order to create bots. Lots of setup is required, so the learning curve is steep. MicroRTS involves common challenges found in RTS games such as strategic and tactical reasoning, resource management, recruiting units, expanding bases etc. Another difference to Starcraft is that the agents have access to a *forward model* (a simulator) of the game. The goal is to create a bot that is able to defeat the enemies. The winners display an interesting blend of techniques such as game-tree search, fast heuristics, grammars, dynamic scripting and, more recently, machine learning.

Fighting Game AI [26] - this competition uses an abstract representation of fighting games such as *Mortal Kombat* or *Street Fighter*. The game is played asynchronously in real time. The AI has maximum of 16.67*ms* per frame to make a decision. There are 56 possible actions such as *high punch*, *low kick*, or *block*. The succesful agents are based on techniques such as Monte Carlo Tree Search, evolutionary algorithms or hierarchical reinforcement learning [27].

Visual Doom AI Competition (VizDoom) [28] - this AI platform is based on an old first-person shooter (FPS) game called *Doom*, which is now open-source. It is a very unique challenge as bots are given raw pixels (i.e., what the player sees) instead of some form of structured state representation as in the case of other game AI competitions. The agents have to reason about the surroundings, navigate through the levels, find interesting spots and weapons and fight with the opponents. There are two tracks of the competition. In the first one, the goal is to finish the game level in the shortest time. The second track is a typical deathmatch, in which the goal is to kill as many enemies as possible. This competition is advertised to be a benchmark for reinforcement learning agents [29]. The state-of-the-art techniques are based on deep learning [30]. However, even the most successful bots cannot compete with humans yet.

Hanabi [31] - Hanabi is a cooperative card game created by French game designer Antoine Bauza. The goal of the Hanabi AI Competition, hosted alongside IEEE CoG, is to create bots that can cooperate and win. The players have the option to give information, play a card or discard a card. Imperfect information plays crucial role in this game and therefore it is challenging for computer players [32]. It is considered a new AI framework for research in multi-agent learning. The agents submitted to the competition use various techniques such as Monte Carlo Tree Search, neural networks, reinforcement learning and rule-based systems.

Hearthstone AI [14] - Hearthstone is an extremely popular online collectible card game developed by Blizzard Entertainment. The first competition was run during IEEE CIG in 2018. It had the biggest number of entries among all competitions held during CIG. The agents played with predefined decks and were pitted against each other. Therefore, this is another example of a competition aimed at creating as strong computer players as possible in a particular game. Competitions in popular games such as Hearthstone have additional value to them, e.g. bots can generate data for learning-based algorithms aimed at solving particular game-related problems [33].

Strategy Card Game AI [34] - proposed as an alternative to Hearthstone just like microRTS competition has been introduced as a simpler alternative to Starcraft. This competition is based around a game called *Legends of Code and Magic (LOCM)*. It is a small and relatively generic framework for research in the area of collectible card-games. The first installment of the competition was run in 2019. The bots are required to be capable of drafting cards (building a deck) as well as playing them.

Geometry Friends [35] - this is a two-player cooperative puzzle platformer game. One player plays as a circle, that can jump and roll and the other one as a rectangle that can change shape (but preserving the area) and slide. Physics with gravity and friction plays an important role. The goal is to complete levels and collect all diamonds that are placed on each level. The main challenges are motion planning and cooperation. The competition has been held in association with three different conferences in various years - IEEE CIG, Genetic and Evolutionary Computation (GECCO) and EPIA Conference on Artificial Intelligence.

Bot Bowl [36] - proposed in 2018 and based on the board game Blood Bowl. This sport game draws inspirations from fantasy and football. Agents have to control 11 players, which poses a challenge of having multiple actions to perform in a coordinated fashion. The competition uses Fantasy Football AI framwork written in Python. It was designed with the Python's machine learning ecosystem in mind. Currently, held during IEEE CoG.

Angry Birds Level Generation [37] - although this competition is built upon a game framework, it has a distinct goal - to create interesting and fun levels in the game of Angry Birds. It belongs to the area of procedural content generation (PCG). The winners of this competition predominantly rely on methods such as evolutionary approaches [38].

Generative Design in Minecraft Competition [39] - like the previous competition, this one is not about the AI for bots, but rather procedural content generation. It is held as part of The Foundations of Digital Games (FDG) conference. The framework of choice is Minecraft - an extremely popular game based on voxels and using them to build objects, houses, settlements, complete maps. The goal of the competition is to use algorithms to produce content which will be both aesthetically pleasing and will evoke an interesting narrative. The evaluation is performed by human judges. That is an interesting fact which usually distinguishes PCG competitions from game-playing AI competitions. If there existed an acceptable way of automated evaluation of maps, then the PCG could directly employ it as part of the techniques, e.g. as fitness function in the evolutionary algorithm.

Halite Competition - available at https://halite.io/. This is both an AI competition and a programming contest. Halite is a resource management game. The goal is to create computer players that gather resources and navigate through 2D game map more effectively than their opponents. The authors of the competition report that participants from over 100 countries played more than 4 million games.

BattleCode - available at https://www.battlecode.org/. A competition hosted by MIT with $30K tournament prize pool. Battlecode is a two-player real-time strategy game. It is both an AI challenge and programming competition aimed for student teams. This competition is not specially designed to advance the state-of-the-art in AI. The computer agents have to solve problems such as resource management, positioning, pathfinding, communication, finding proper offensive strategies.

III. Imitation Game AI Competition

A. Description

We think that a new competition should be based around making computer bots that can effectively imitate any given human player. The term "imitation" requires elaborated definition. There are two viable approaches to measure the rate of imitation. The first one, more suitable for bigger competitions, is to do it in the same way as it is done in data-mining competitions, e.g. by having training data and hidden testing data. Examples of such data-mining competition platforms are $Kaggle$ [40] and $KnowledgePit$ [41]. The second one, has even featured game-based competition e.g. aimed at advising players [42] in card games. The second option, suitable for competitions with lesser amounts of entries, would be to have human referees judging the bots as shown in the *Generative Design in Minecraft* Competition. Although with this approach there is lesser objectivity, the human experts could see nuances that are difficult to grasp with automatic verification. It is also useful for the game industry - where all it matters is whether the bots' behavior *feels* right and whether the bots act the way game designers have envisioned.

The second aspect of the competition is how the input (training) data should be provided. The training data are past game records of the player to be imitated. They can have the form of videos, sequence of screenshots or structured logs. A particularly suitable format for such logs is in LogDL [43], which was inspired by GDL mentioned in the context of

General Game Playing. We think that the most important thing is to not give the bots any more information than the human players had. Apart from that, they should be given as much information as possible amongst the information human players can see. We propose to combine the video footage with logs containing the most important numerical parameters that describe the game. This is a multi-modal approach. In this way, the bots can do feature extraction and figure out what describes the style of a particular player the most accurately. At the same time, they do not have to do the basic necessary extraction of the obvious parameters such as the amount of resources in strategy games. Therefore, video recognition methods could be focused on patterns, maps and geometric dependencies.

B. Advantages

In this section, we present the motivation behind the *Imitation Game AI*. After introducing each reason, we put annotations: *Research* or *Industry* in brackets. The former indicates that the particular reason mostly concerns advancing the state-of-the-art of the Artificial Intelligence field. The latter refers to advantages for the game development industry.

1) Human-likeness (Research, Industry): What makes human-like AI is a question that has been asked by many researchers, e.g. [44]. Although it is an interesting concept by itself from the cognitive viewpoint, it also has practical implications, e.g. in trusted human-robot interaction [45]. A competition that revolves around this topic would not only spawn new methods of implementing human-like AI, but also new ways of measuring it. Such methods could be more objectivized that those that are based on the Turing's Test as they would rely less on the judgement performed by a relatively small number of referees.

Human-like bots are very valuable for computer games as well for numerous reasons. First of all, they can act as believable NPC characters [46]. This greatly increases the immersion in the game. More immersive and interesting games lead to more amount of time spent by people playing them and a better overall reception. Second of all, they can take part in multi-player matches if there are not enough human players available at given moment or they can take over when one of the human players disconnects from the game. Third of all, human-like bots can be used as virtual testers specialized to predict interactions real (human) players will make. If a goal of the AI challenge is to develop methods and techniques that can accurately capture the style of play of humans, then the property of human-likeness is an inherent part of the challenge.

2) Explainability (Research, Industry): As applications of machine learning models such as deep neural networks are growing in numbers, the explainable AI (XAI) is becoming more and more important [47]. In the game industry, the most commonly implemented AI techniques such as Behavior Trees, Finite State Machines or Utility AI are still highly explainable and interpretable. However, they are relatively limited in terms of complexity and competence levels of the AI. It seems inevitable that methods such as deep reinforcement learning will be applied more often not only in research but in shipped video games too. When the goal of the AI is to mimic specific human players, then if a bot achieves high accuracy regardless of how much of a black-box it is, we can explain it by asking the same human players for explanation of their reasoning. Expert human players tend to make thoughtful actions aimed at gaining some particular advantages in the game.

3) Controlled Difficulty (Industry): Superhuman bots are unacceptable in commercial video games. After all, the games need to be entertaining and possible to complete. On the other end of the spectrum, it is often difficult to create competent bots without letting them "cheat", e.g. have access to hidden information or gather resources more effectively than human players. The idea of mimicking human skills is a solution to both superhuman or incompetent bots. Here an important distinction can be seen between the goal of imitation of human skill in the game and just using human players as teachers in supervised learning. The latter case may, by chance, lead to the superhuman level of play. However, in our proposal, not surpassing human skill level inherently goes along with the aim of reproducing their skill.

4) Personification of AI (Industry): A persona system for bots means that even if multiple computer players are meant to have similar intelligence, they have certain individual characteristics, e.g., risk taker, explorer, conqueror, defender, builder etc. Therefore, they display various behaviors and are, in general, more interesting characters in games. Algorithms that are aimed at optimizing a value function such as Monte Carlo Tree Search or neural networks tend to converge (with faster or slower rate) to their optimality conditions. In order to introduce more variety with techniques such as MCTS, it was proposed to define game logic with varying granularity [48]. Here comes an important property of our Imitation Game AI Competition, i.e., the requirement to be able to learn and mimic the behavior of particularly chosen human players. This is a different case than learning based on a general corpus of game records played by humans as this way we would lose the individual traits.

5) Commercially Viable Workflow (Industry): Human game testers are usually the biggest group of people involved in the game production process [49]. In fact, when a game requires complex AI, sometimes it would be easier and cheaper to train it by letting it observe human players. It takes a lot of time of experienced AI designers and programmers to create a competent AI in games, where it plays an important role, e.g. shooter games with bots or strategy games. The Imitation Game AI competition solves enables an alternative way of coming up with the problem of creating the AI. Moreover, this approach fits into the existing game production pipeline, because games are thoroughly tested by people playing it, so many records of the played

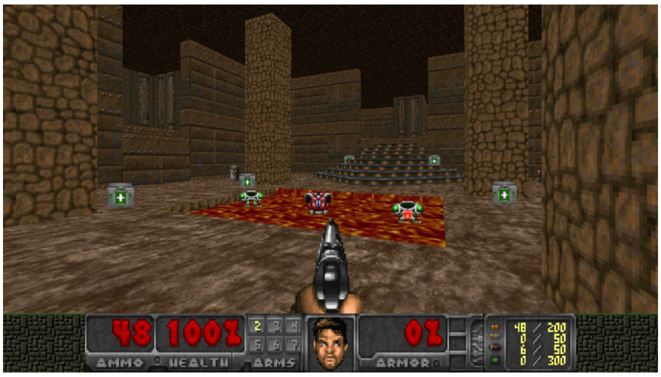


Fig. 1. Raw pixels shown to bots in the VizDoom Competition.

games are produced anyway for the internal companies' usage.

6) Vision Recognition (Research): The Imitation Game AI competition can be run with various forms of game records for the bots to observe and learn. One possibility is to give the bots raw pixels of the screen, just like humans observed the game. This is a similar case to the Visual Doom AI competition mentioned in Section II as shown in Figure 1. The other possibility is to provide structured logs of game records or even multi-modal data, e.g., combination of pixels, high-level representation of game states and data about the UI input state (mouse, keyboard, pad). In the first case, i.e., with raw visual data, the competition would benefit to the computer vision field. It would require effective methods of image classification [50], concepts extraction [51], object recognition [52] and visual reasoning [53].

7) Step Towards AGI (Research): Developing Artificial General Intelligence (AGI or "Strong AI") [54] is one of the distant challenges for the AI researchers. First and foremost, general AI competitions such GGP and GVG-AI with goals of creating universal game playing programs fit well into the AGI stream of research. We argue that the Imitation Game AI competition, in its general variant, is a step closer to AGI than a competition that revolves around making bots as strong as possible. AGI is not about making AI do something more effectively than humans - that is Artificial Superintelligence (ASI). The aim of AGI is to make an AI that can learn, reason and perform any task on par with humans. The focus is on generality in contrast to specialized Artificial Narrow Intelligence (ANI or "Weak AI").

IV. Conclusions

In this paper, we proposed an AI benchmark competition that is aimed at creating bots capable of learning from humans to mimic their skill and style of play. Such an approach to learning is similar to supervised-learning. However, in a typical supervised-learning scenario, training data must be properly prepared and labelled. We leave it as an open problem to solve by the bot, i.e., the algorithms must figure out how to extract useful knowledge from the observed human players. Moreover, the the players that produce the training data and the bots have various objectives. Human players play according to the goals of the particular game. The goal for the bots is to play in a similar way to the human player. This can be particular beneficial to video-game industry. Most games are tested for hundreds or even thousands of man-hours, so why not to take advantage of it and use the game records for training bots. Moreover, the problem of creating AI is transferred from AI specialists to specialized algorithms. Bots created this way can be, by definition, more human-like, can have different personas based on who trained them and can be more explainable for humans. Lastly, let's not forget that imitation learning brings interesting challenges and it is an interesting concept by itself. We are pursuing the goal of Artificial General Intelligence, but smaller steps have to be made first. Game-based AI challenges may continue being very useful for measuring progress in the field.

References

[1] A. M. Turing, "Can a Machine Think," *The World of Mathematics*, vol. 4, pp. 2099–2123, 1956.

[2] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," *IBM Journal of Research and Development*, vol. 3, no. 3, pp. 210–229, 1959, DOI=10.1147/rd.33.0210.

[3] A. Newell, J. C. Shaw, and H. A. Simon, "Chess-Playing Programs and the Problem of Complexity," *IBM Journal of Research and Development*, vol. 2, no. 4, pp. 320–335, 1958, DOI=10.1147/rd.24.0320.

[4] M. Newborn, *Kasparov versus Deep Blue: Computer Chess Comes of Age*. Springer Science & Business Media, 2012, DOI=10.1007/978-1-4612-2260-6.

[5] D. Ferrucci, A. Levas, S. Bagchi, D. Gondek, and E. T. Mueller, "Watson: Beyond Jeopardy!" *Artificial Intelligence*, vol. 199, pp. 93–105, 2013, DOI=10.1109/ICCI-CC.2013.6622216.

[6] M. Newborn, "2007: Deep Junior Deep Sixes Deep Fritz in Elista, 4–2," in *Beyond Deep Blue*. Springer, 2011, pp. 149–157.

[7] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel *et al.*, "A General Reinforcement Learning Algorithm that Masters Chess, Shogi, and Go Through Self-Play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018, DOI=10.1126/science.aar6404.

[8] F.-Y. Wang, J. J. Zhang, X. Zheng, X. Wang, Y. Yuan, X. Dai, J. Zhang, and L. Yang, "Where does AlphaGo Go: from Church-Turing Thesis to AlphaGo Thesis and Beyond," *IEEE/CAA Journal of Automatica Sinica*, vol. 3, no. 2, pp. 113–120, 2016, DOI=10.1109/JAS.2016.7471613.

[9] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton *et al.*, "Mastering the Game of Go Without Human Knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017, DOI=10.1038/nature24270.

[10] H. J. Van Den Herik, J. W. Uiterwijk, and J. Van Rijswijck, "Games Solved: Now and in the Future," *Artificial Intelligence*, vol. 134, no. 1-2, pp. 277–311, 2002, DOI=10.1016/S0004-3702(01)00152-7.

[11] J. Schaeffer, N. Burch, Y. Björnsson, A. Kishimoto, M. Müller, R. Lake, P. Lu, and S. Sutphen, "Checkers is solved," *Science*, vol. 317, no. 5844, pp. 1518–1522, 2007, DOI=10.1126/science.1144079.

[12] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev *et al.*, "Grandmaster Level in StarCraft II using Multi-Agent Reinforcement Learning," *Nature*, vol. 575, no. 7782, pp. 350–354, 2019, DOI=10.1038/s41586-019-1724-z.

[13] S. McCandlish, J. Kaplan, D. Amodei, and O. D. Team, "An Empirical Model of Large-Batch Training," *arXiv preprint arXiv:1812.06162*, 2018.

[14] M. Świechowski, T. Tajmajer, and A. Janusz, "Improving Hearthstone AI by Combining MCTS and Supervised Learning Algorithms," in *2018 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2018, pp. 1–8, DOI=10.1109/CIG.2018.8490368.

[15] M. Świechowski, H. Park, J. Mańdziuk, and K.-J. Kim, "Recent Advances in General Game Playing," *The Scientific World Journal*, vol. 2015, 2015, DOI=10.1155/2015/986262.

[16] M. Świechowski and J. Mańdziuk, "Specialized vs. Multi-Game Approaches to AI in Games," in *Intelligent Systems' 2014*. Springer, 2015, pp. 243–254, DOI=10.1007/978-3-319-11313-5_23.

[17] M. Świechowski and D. Ślęzak, "Grail: A Framework for Adaptive and Believable AI in Video Games," in *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE, 2018, pp. 762–765, DOI=10.1109/WI.2018.00012.

[18] M. R. Genesereth, N. Love, and B. Pell, "General Game Playing: Overview of the AAAI Competition," *AI Magazine*, vol. 26, no. 2, pp. 62–72, 2005, DOI=10.1609/aimag.v26i2.1813.

[19] M. Świechowski and J. Mańdziuk, "Self-Adaptation of Playing Strategies in General Game Playing," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 6, no. 4, pp. 367–381, Dec 2014, DOI=10.1109/TCIAIG.2013.2275163.

[20] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A Survey of Monte Carlo Tree Search Methods," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 1, pp. 1–43, 2012, DOI=10.1109/TCIAIG.2012.2186810.

[21] D. Perez-Liebana, S. Samothrakis, J. Togelius, T. Schaul, S. M. Lucas, A. Couëtoux, J. Lee, C.-U. Lim, and T. Thompson, "The 2014 General Video Game Playing Competition," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 8, no. 3, pp. 229–243, 2015, DOI=10.1109/TCIAIG.2015.2402393.

[22] O. Syed and A. Syed, *Arimaa - A New Game Designed to be Difficult for Computers*. Institute for Knowledge and Agent Technology, 2003, vol. 26, no. 2, DOI=10.3233/icg-2003-26213.

[23] S. Xu, H. Kuang, Z. Zhi, R. Hu, Y. Liu, and H. Sun, "Macro Action Selection with Deep Reinforcement Learning in Starcraft," in *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 15, no. 1, 2019, pp. 94–99.

[24] S. Ontanon, G. Synnaeve, A. Uriarte, F. Richoux, D. Churchill, and M. Preuss, "A Survey of Real-Time Strategy Game AI Research and Competition in StarCraft," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 5, no. 4, pp. 293–311, 2013, DOI=10.1109/TCIAIG.2013.2286295.

[25] S. Ontañón, N. A. Barriga, C. R. Silva, R. O. Moraes, and L. H. Lelis, "The First MicroRTS Artificial Intelligence Competition," *AI Magazine*, vol. 39, no. 1, pp. 75–83, 2018, DOI=10.1609/aimag.v39i1.2777.

[26] Y. Takano, H. Inoue, R. Thawonmas, and T. Harada, "Self-Play for Training General Fighting Game AI," in *2019 Nicograph International (NicoInt)*. IEEE, 2019, pp. 120–120, DOI=10.1109/NICOInt.2019.00034.

[27] I. P. Pinto and L. R. Coutinho, "Hierarchical Reinforcement Learning with Monte Carlo Tree Search in Computer Fighting Game," *IEEE Transactions on Games*, vol. 11, no. 3, pp. 290–295, 2018, DOI=10.1109/TG.2018.2846028.

[28] M. Wydmuch, M. Kempka, and W. Jaśkowski, "Vizdoom Competitions: Playing Doom from Pixels," *IEEE Transactions on Games*, vol. 11, no. 3, pp. 248–259, 2018, DOI=10.1109/TG.2018.2877047.

[29] M. Kempka, M. Wydmuch, G. Runc, J. Toczek, and W. Jaśkowski, "Vizdoom: A Doom-based AI Research Platform for Visual Reinforcement Learning," in *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2016, pp. 1–8, DOI=10.1109/CIG.2016.7860433.

[30] K. Shao, D. Zhao, N. Li, and Y. Zhu, "Learning Battles in ViZDoom via Deep Reinforcement Learning," in *2018 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2018, pp. 1–4, DOI=10.1109/CIG.2018.8490423.

[31] N. Bard, J. N. Foerster, S. Chandar, N. Burch, M. Lanctot, H. F. Song, E. Parisotto, V. Dumoulin, S. Moitra, E. Hughes *et al.*, "The Hanabi Challenge: A New Frontier for AI Research," *Artificial Intelligence*, vol. 280, p. 103216, 2020, DOI=10.1016/j.artint.2019.103216.

[32] J.-F. Baffier, M.-K. Chiu, Y. Diez, M. Korman, V. Mitsou, A. van Renssen, M. Roeloffzen, and Y. Uno, "Hanabi is NP-hard, even for cheaters who look at their cards," *Theoretical Computer Science*, vol. 675, pp. 43–55, 2017, DOI=10.1016/j.tcs.2017.02.024.

[33] A. Janusz, Ł. Grad, and D. Ślęzak, "Utilizing Hybrid Information Sources to Learn Representations of Cards in Collectible Card Video Games," in *2018 IEEE International Conference on Data Mining Workshops, ICDM Workshops, Singapore, Singapore, November 17-20, 2018*. IEEE, 2018, pp. 422–429, DOI=10.1109/ICDMW.2018.00069.

[34] J. Kowalski and R. Miernik, "Evolutionary Approach to Collectible Card Game Arena Deckbuilding using Active Genes," *Accepted to IEEE Congress on Evolutionary Computation 2020*, 2020. [Online]. Available: arXivpreprintarXiv:2001.01326

[35] D. M. G. Verghese, S. Bandi, and G. J. Jayaraj, "Solving the Complexity of Geometry Friends by Using Artificial Intelligence," in *Advances in Decision Sciences, Image Processing, Security and Computer Vision*. Springer, 2020, pp. 528–533, DOI=10.1007/978-3-030-24318-0_62.

[36] N. Justesen, L. M. Uth, C. Jakobsen, P. D. Moore, J. Togelius, and S. Risi, "Blood Bowl: A New Board Game Challenge and Competition for AI," in *2019 IEEE Conference on Games (CoG)*. IEEE, 2019, pp. 1–8, DOI=10.1109/CIG.2019.8848063.

[37] J. Renz, X. Ge, S. Gould, and P. Zhang, "The Angry Birds AI competition," *AI Magazine*, vol. 36, no. 2, pp. 85–87, 2015, DOI=10.1609/aimag.v36i2.2588.

[38] A. Irfan, A. Zafar, and S. Hassan, "Evolving Levels for General Games Using Deep Convolutional Generative Adversarial Networks," in *2019 11th Computer Science and Electronic Engineering (CEEC)*. IEEE, 2019, pp. 96–101, DOI=10.1109/CEEC47804.2019.8974332.

[39] C. Salge, M. C. Green, R. Canaan, and J. Togelius, "Generative Design in Minecraft (GDMC) Settlement Generation Competition," in *Proceedings of the 13th International Conference on the Foundations of Digital Games*, 2018, pp. 1–10.

[40] J. Whitehill, "Climbing the Kaggle Leaderboard by Exploiting the Log-Loss Oracle," in *Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[41] A. Janusz, D. Slezak, S. Stawicki, and M. Rosiak, "Knowledge Pit - A Data Challenge Platform," in *CS&P*, 2015, pp. 191–195.

[42] A. Janusz, T. Tajmajer, M. Świechowski, Ł. Grad, J. Puczniewski, and D. Ślęzak, "Toward an Intelligent HS Deck Advisor: Lessons Learned from AAIA'18 Data Mining Competition," in *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2018, pp. 189–192, DOI=10.15439/2018F386.

[43] M. Świechowski and D. Ślęzak, "Introducing LogDL - Log Description Language for Insights from Complex Data," in *Proceedings of the 15th Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2020, pp. 145–154.

[44] S. D. Baum, B. Goertzel, and T. G. Goertzel, "How Long until Human-Level AI? Results from an Expert Assessment," *Technological Forecasting and Social Change*, vol. 78, no. 1, pp. 185–195, 2011, DOI=10.1016/j.techfore.2010.09.006.

[45] J. Fink, "Anthropomorphism and Human Likeness in the Design of Robots and Human-Robot Interaction," in *International Conference on Social Robotics*. Springer, 2012, pp. 199–208, DOI=10.1007/978-3-642-34103-8_20.

[46] P. Hingston, *Believable Bots: Can Computers Play Like People?* Springer, 2012, DOI=10.1007/978-3-642-32323-2.

[47] J. Zhu, A. Liapis, S. Risi, R. Bidarra, and G. M. Youngblood, "Explainable AI for designers: A human-centered perspective on mixed-initiative co-creation," in *2018 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2018, pp. 1–8, DOI=10.1109/CIG.2018.8490433.

[48] M. Świechowski and D. Ślęzak, "Granular Games in Real-Time Environment," in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2018, pp. 462–469, DOI=10.1109/ICDMW.2018.00074.

[49] D. Irish, *The game poducer's handbook*. Course Technology Press, 2005, DOI=10.5555/1209055.

[50] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu, "CNN-RNN: A Unified Framework for Multi-label Image Classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2285–2294, DOI=10.1109/CVPR.2016.251.

[51] N. A. Bennett, Q. He, C. Chang, and B. R. Schatz, "Concept Extraction in the Interspace Prototype," *University of Illinois at Urbana-Champaign, Champaign, IL*, 1999, DOI=10.5555/871248.

[52] R. M. Cichy, D. Pantazis, and A. Oliva, "Resolving Human Object Recognition in Space and Time," *Nature Neuroscience*, vol. 17, no. 3, p. 455, 2014, DOI=10.1038/nn.3635.

[53] C. Han, J. Mao, C. Gan, J. Tenenbaum, and J. Wu, "Visual Concept-Metaconcept Learning," in *Advances in Neural Information Processing Systems*, 2019, pp. 5002–5013.

[54] B. Goertzel, "Artificial General Intelligence: Concept, State of the Art, and Future Prospects," *Journal of Artificial General Intelligence*, vol. 5, no. 1, pp. 1–48, 2014, DOI=10.2478/jagi-2014-0001.

5^{th} International Workshop on Language Technologies and Applications

DEVELOPMENT of new technologies and various intelligent systems creates new possibilities for information processing. Natural Language Processing (NLP) addresses problems of automated understanding, processing, evaluation and generation of natural human languages. LTA workshop provides a venue for discussion and presenting innovative research in NLP domain, but not restricted, to: computational and mathematical modeling, analysis and processing of any forms (spoken, handwritten or text) of human language, interactions via Virtual Reality and Augmented Reality, Computational Intelligence models and applications but also other various applications in decision support systems. We welcome papers covering innovative applications and practical usage of theoretical aspects. The LTA workshop will provide an opportunity for researchers and professionals to discuss present and future challenges as well as potential collaboration for future progress in the field.

TOPICS

The submitted papers shall cover research and developments in all NLP aspects, such as (however this list is not exhaustive):

- Computational Intelligence methods applied to language & text processing
- text analysis
- language networks
- text classification
- language networks, resources and corpora
- document clustering
- various forms of text recognition
- machine translation
- intelligent text-to-speech (TTS) and speech-to-text (STT) methods
- authorship identification and verification
- author profiling
- plagiarism detection
- sentiment analysis
- NLP applications in education
- knowledge extraction and retrieval from text and natural language structures
- multi-modal and natural language interfaces
- innovative language-oriented applications and tools
- interactions models and applications via Virtual Reality and Augmented Reality
- NLP for text analysis in forensic linguistics and cybersecurity

TECHNICAL SESSION CHAIRS

- **Damasevicius, Robertas,** Kaunas University of Technology, Lithuania
- **Martinčić – Ip&scaron;ić, Sanda,** University of Rijeka, Croatia
- **Napoli, Christian,** Department of Mathematics and Informatics, University of Catania, Italy
- **Sanada, Haruko,** Rissho University, Japan

PROGRAM COMMITTEE

- **Artiemjew, Piotr,** University of Warmia and Mazury, Poland
- **Burdescu, Dumitru Dan,** University of Craiova, Romania
- **Harbusch, Karin,** Universität Koblenz-Landau, Germany
- **Kapočiūtė-Dzikienė, Jurgita,** Vytautas Magnus University, Lithuania
- **Kurasova, Olga,** Vilnius University, Institute of Mathematics and Informatics, Lithuania
- **Marszałek, Zbigniew,** Silesian University of Technology, Poland
- **Maskeliūnas, Rytis,** Kaunas University of Technology, Lithuania
- **Matson, Eric T.,** Purdue University, United States
- **Połap, Dawid,** Institute of Mathematics, Silesian University of Technology, Poland
- **Starczewski, Janusz,** Czestochowa University of Technology, Poland
- **Tambouratzis, George,** Institute for Language and Speech Processing, Athena Research Centre, Greece

DOI: 10.15439/2020F192
ISSN 2300-5963 ACSIS, Vol. 21

Automatic Generation of Annotated Corpora of Diagnoses with ICD-10 codes based on Open Data and Linked Open Data

Svetla Boytcheva
Institute of Information and
Communication Technologies
Bulgarian Academy of Sciences, Sofia, Bulgaria
Email: svetla.boytcheva@gmail.com

Boris Velichkov, Gerasim Velchev, Ivan Koychev
Faculty of Mathematics and Informatics
Sofia University "St. Kliment Ohridski", Sofia, Bulgaria
Email: {bobby.velichkov, gerasim.petrov.velchev}@gmail.com
ivan.koychev@fmi.uni-sofia.bg

***Abstract*—We propose methods for automatic generation of corpora that contains descriptions of diagnoses in Bulgarian and their associated codes in ICD-10-CM (International Classification of Diseases, 10th revision, Clinical Modification). The proposed approach is based on the available open data and Linked Open Data and can be easily adapted for other languages. The resulted corpora generated for the Bulgarian clinical texts consists of about 370,000 pairs of diagnoses and corresponding ICD-10 codes and is beyond the usual size that can be generated manually, moreover it was created from scratch and for a relatively short time. Further updates of the corpora are also possible whenever new open resources are available or the current ones are updated.**

I. Introduction

THE AUTOMATIC processing and extraction of knowledge from medical texts is a task of public importance. The majority of healthcare documents are still available mainly in free text format, on the local language. Natural Language processing of clinical text require to be developed specific language resources that requires expert knowledge and validation, which is quite difficult to achieve, especially in a situation where health workers are overwhelmed with other more important daily responsibilities. Clinical Natural Language Processing (NLP) is quite challenging task for non-English language [1]. For low resource languages such as Bulgarian the majority of the required resources are not available, or there are some limited versions. The question is "how we can develop automatically or semi-automatically such resources from scratch for relatively short time?". Medical terminology in Bulgarian has very specific nature, because it is a mixture between terminology in Bulgarian, Latin and transliterated Latin terms in Cyrillic [2].

Diagnosis is one of the most important complex data on the patient's health in clinical texts. On the other hand, due to the complexity of the information they contain, there is a wide variety of ways to describe it - using different terms, paraphrases, abbreviations and various details to describe the stage of the disorder, its location, cause and severity,

Further processing of the extracted diagnosis information requires unification/normalization of the data according to some standard nomenclatures to avoid ambiguities. One of the widely used International Classification of Diseases is ICD-10 [1] that has also translations to many languages. The classification task for association of ICD-10 codes to textual descriptions of diagnosis requires training corpora, and because there are about 11,000 different codes the corpora should be relatively large in size. We focused on the task for automatic generation of training corpora of diagnosis descriptions in Bulgarian and their corresponding ICD-10 codes.

Already there are various research for other languages. Wang, Qiong, et al. [5] present a study that aims to develop and evaluate effective methods that can normalize diagnosis and procedure terms written by physicians to standard concepts in ICD in Chinese using an entity-linking framework and two manually annotated datasets (8,547 diagnoses and 8,282 procedures). Marovac, Avdić et al. [6] present in their paper the process of creating medical lexical resources for the Serbian language and they achieve mapping to certain ICD-10 codes with precision over 80%. Almagro, Unanue et al. [7] have been carried out an exploration on 7254 Spanish hospital discharge reports for a period of 3 years resulting in total 76,525 identified codes with approximately 7,000 unique ones. Bagheri, Sammani et. al. [8] sought to implement a system to help 3-digit Dutch ICD-10 coding of discharge letters via machine learning algorithms. Dalianis [9] addresses the automatic assignment of Portuguese ICD-10 codes for causes of death by analyzing 114,228 free-text descriptions containing a total of 1,418 distinct codes.

We propose methods for automatic generation of corpora that contains descriptions of diagnoses in Bulgarian and their associated codes in ICD-10. The proposed approach is based on the publicly available resources and can be easily adapted

This research is partially funded by the Bulgarian Ministry of Education and Science, grant DO1-200/2018 'Electronic health care in Bulgaria' (e-Zdrave) and the Bulgarian National Science Fund, grant DN-02/4-2016 'Specialized Data Mining Methods Based on Semantic Attributes' (IZIDA). We are grateful to anonymous reviewers for useful comments and suggestions.

[1] https://icd.who.int/browse10/2019/en

for other languages. The resulted corpora generated for the Bulgarian clinical texts consists of about 370,000 pairs of diagnosis and the corresponding ICD-10 codes and is beyond the usual size that can be generated manually, moreover it was created from scratch and for a relatively short time. Further updates of the corpora are also possible whenever new open resources are available or the current ones are updated.

II. Method

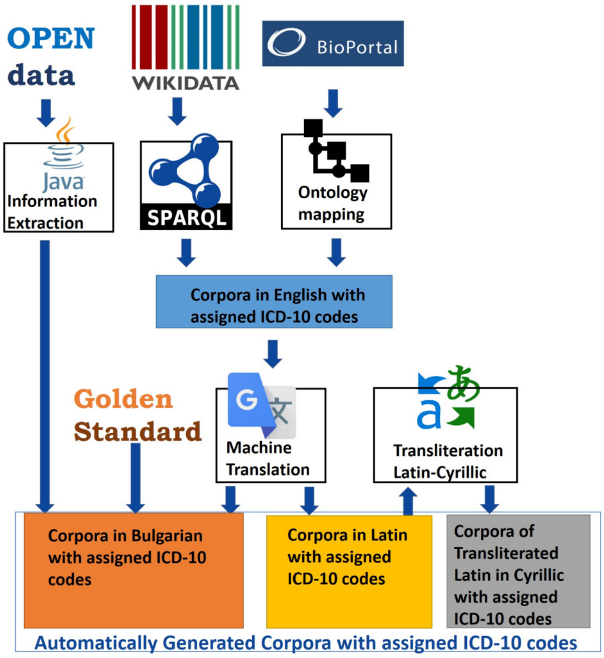

OPEN data
WIKIDATA
BioPortal
Java Information Extraction
SPARQL
Ontology mapping
Corpora in English with assigned ICD-10 codes
Golden Standard
Machine Translation
Transliteration Latin-Cyrillic
Corpora in Bulgarian with assigned ICD-10 codes
Corpora in Latin with assigned ICD-10 codes
Corpora of Transliterated Latin in Cyrillic with assigned ICD-10 codes
Automatically Generated Corpora with assigned ICD-10 codes

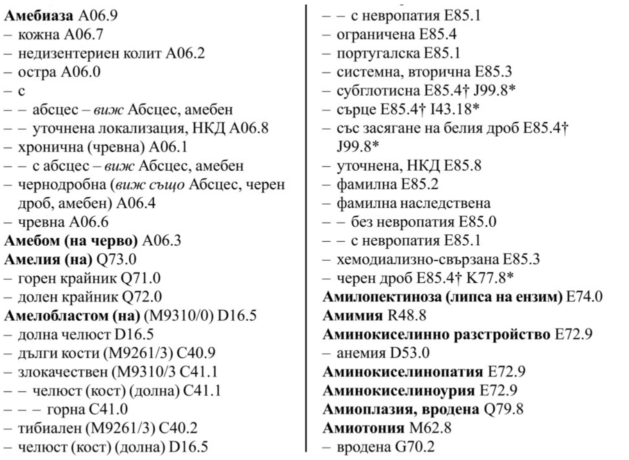
Figure 1. Method

The proposed method for automatic corpora generation is language independent and relies mainly on open data and linked open data (LOD[2]). The following components are used:

- Automatic extraction of Annotations from Open Documents - for this module are used publicly available documents in Bulgarian language as an input, and are developed information extraction algorithms that convert the textual data into dataset of structured pairs of diagnosis and associated ICD-10 codes;
- Automatic extraction of Annotations from LOD - for this module are used SPARQL queries for extraction of diagnosis in English language and corresponding codes to some of the widely used standard classifications. All available mappings between these classifications are used to produce dataset with associated codes to ICD-10.
- Machine translation - this module is used for diagnosis translation from English to Bulgarian and Latin.
- Transliteration tool - this module is responsible for transliteration of the diagnosis from Latin to Cyrillic.
- Other resources - Golden Standard (GS) for some diagnoses with associated ICD-10 codes [3].

A. *Automatic Extraction of Annotations from Open Documents*

The main resource used in this module is the official document of the International Classification of Diseases: "ICD-10-CM Alphabetical Index" (ICD10-Index[3]). This document is translated by health organizations or ministries into the relevant language. A translated version[4] from the website of the Ministry of Health of the Republic of Bulgaria was used to generate the current dataset. It is in the form of two PDF documents. First they are converted to Microsoft Word (".doc") format using the PDF reader "Nitro Pro (7.5.0.18)"[5]. The resulting format is converted in addition to the ".docx" format using Microsoft Word functions. The Apache POI[6] library with the Java programming language are used to read the received documents. The library provides an easy way to read the individual paragraphs. It is important to note that in the process of conversion some paragraphs can be damaged, thus some minor manual cleaning and formatting of the result file is needed. For the rest of the (automatic) part of the processing[7], the structure of the index is very important for the rule based information extraction. A screenshot of the document can be seen at Figure 2.



Амебиаза A06.9
– кожна A06.7
– недизентериен колит A06.2
– остра A06.0
– с
– – абсцес – *виж* Абсцес, амебен
– – уточнена локализация, НКД A06.8
– хронична (чревна) A06.1
– – с абсцес – *виж* Абсцес, амебен
– чернодробна (*виж също* Абсцес, черен дроб, амебен) A06.4
– чревна A06.6
Амебом (на черво) A06.3
Амелия (на) Q73.0
– горен крайник Q71.0
– долен крайник Q72.0
Амелобластом (на) (M9310/0) D16.5
– долна челюст D16.5
– дълги кости (M9261/3) C40.9
– злокачествен (M9310/3 C41.1
– – челюст (кост) (долна) C41.1
– – – горна C41.0
– тибиален (M9261/3) C40.2
– челюст (кост) (долна) D16.5

– – с невропатия E85.1
– ограничена E85.4
– португалска E85.1
– системна, вторична E85.3
– субглотисна E85.4† J99.8*
– сърце E85.4† I43.18*
– със засягане на белия дроб E85.4† J99.8*
– уточнена, НКД E85.8
– фамилна E85.2
– фамилна наследствена
– – без невропатия E85.0
– – с невропатия E85.1
– хемодиализно-свързана E85.3
– черен дроб E85.4† K77.8*
Амилопектиноза (липса на ензим) E74.0
Амимия R48.8
Аминокиселинно разстройство E72.9
– анемия D53.0
Аминокиселинопатия E72.9
Аминокиселиноурия E72.9
Амиоплазия, вродена Q79.8
Амиотония M62.8
– вродена G70.2

Figure 2. ICD-10 Alphabetical Index

To avoid unnecessary repetition, the index is organized in the form of a tree structure: leading terms, which are located in the leftmost column, and other paragraphs, which start on the right. For this reason, the full term consists of several lines, sometimes giving too broad description.

When traversing each node in such tree structure, a number of regular expressions are used in order to be able to determine both the level of the text in the tree and to recognize the individual text items. Some of the main text processing transformations are the following:

- Convert all references to pre-specified categories: "виж също -> виж" (see also -> see).

[2] https://lod-cloud.net/

[3] https://icd.codes/icd10cm/alphabetical-index

[4] https://ncpha.government.bg/bg/2019-02-19-23-22-18/icd-10

[5] https://www.gonitro.com/nps/pro/pdf-software

[6] https://poi.apache.org/

[7] https://github.com/BorisVelichkov/ICD10-Medical-Data

- Remove noun inflexion forms as number and case: "(-а)", "(-та)", etc.
- Remove parentheses, other special characters and redundant white spaces.
- Merge words that have been transferred to a new line.
- Merge erroneously separated paragraphs.
- Combine the different levels in document structure order to form correct sentences for diagnosis.
- Recognize references and remove them after concatenation with the next level text.
- Recognize ICD-10 codes and create valid examples for each type of ICD-10 code (the codes are written in a different format).

Some examples of the diagnosis descriptions with the corresponding ICD-10 codes generated from the tree structure (Figure 2) are displayed in Table I.

Table I
ICD-10 INSTANCES CREATED FROM THE SHOWN TREE STRUCTURE

ICD10	Text
A06.9	Амебиаза
A06.7	Амебиаза кожна
A06.2	Амебиаза недизентериен колит
A06.0	Амебиаза остра
A06.8	Амебиаза с уточнена локализация
A06.1	Амебиаза хронична чревна
A06.4	Амебиаза чернодробна виж Абсцес черен дроб амебен
A06.6	Амебиаза чревна

B. Automatic Extraction of Annotations from LOD

The main resource are translation of the ICD-10-CM translated in Bulgarian[8] (ICD10-BG). It contains (see Fig. 1) about 11,000 classes organized in 4 levels hierarchy - 22 groups at level 1, 211 subgroups at level 2, 2025 are level 3 (3-sign codes) and 8946 at level 4 (4-sign codes). They are not presenting single diagnose, but statistical classification of groups of diagnoses. Thus they can serve only partially as a resource for the generated corpora. The ICD-10 is one of the widely used classification of diseases and translations[9] on several languages are available.

Table II
WIKIDATA ONTOLOGIES

Wikidata code	Q12136	Q179630	Q169872	Q639907
P4229	39,743	107	20	10
P699	47,092	60	36	13
P486	27,478	127	207	39
P3841	5,952	2	100	68
P604	6,326	37	85	20
P5270	50,292	101	55	37
P1550	31,179	103	15	13

As primary resource was used Wikidata[10], which provides encyclopedic data in structured format. Unfortunately only for small subsets of diagnosis are available labels in Bulgarian language, thus our primary focus will be to collect data for English language. We collect from Wikidata results of several SPARQL queries investigating for labels in English language, the availability of the concepts disease (Q12136), illness (Q814207), syndrome (Q179630), symptom (Q169872), medical finding (Q639907) and their associations with medical classifications: ICD-10-CM (P4229), Human Disease Ontology[11] (P699), MeSH[12] (P486), The Human Phenotype Ontology[13](P3841), MedlinePlus (P604), MonDO[14] (P5270), Orphanet[15] (P1550).

The results from different combinations of concepts and ontologies are shown in Table (Table II), where also mapping of the ontologies ID to ICD-10 code is applied (if any). The SPARQL queries[16] were run in Wikidata Query Service and the generated results are stored in CSV format including the following properties *<Item URI to Wikidata, Item Label, Item Alternative Label, Ontology ID, ICD10 code>*. All result CSV tables are merged, and are removed duplicates. For some labels, that contain disjunction ("or") a separate instance is create for each element. The total collected datasets contains 57,142 pairs of 4-sign code in ICD-10 and text label for diagnosis. Further automatic cleaning was applied to remove abbreviations. For example, "ID" caused in normalization some ambiguities and was misinterpreted as "Identification document", instead of "Infectious disease". The final result cleaned dataset (WD-ENG) contain 55,292 pairs of data with ICD-10 codes (4-sign) and diagnosis in English Language.

C. Latin-Cyrillic Transliteration

One of the most important parts of building a usable dataset is data augmentation - the process of generating new data as a variation of already known. In medical text such variations include different ways of writing diagnosis names: using Bulgarian terms, Latin terms, or using Latin terms written using Cyrillic letters. Such variants we call Cyrillic transliterations of Latin terms. There exist a plenty of rules for transliteration of Latin medical terms in Cyrillic representations, described in a Latin-Bulgarian dictionary [4]. We categorise them in three groups depending on number of consecutive Latin letters (1, 2 or 3) they are replacing with string in Cyrillic (type 1, type 2, or type 3 respective). There are 22 rules of type 1, 11 rules of type 2 and 9 from of type 3. Some of the rules are direct replacements of a string with another string. Other include wildcard positions - positions which could be replaced with a set of symbols (e.g. vowels). For instance, rules of:

- type 1 - "u" ⇒ "у", "x" ⇒ "кс".

[8] http://www.zdrave.bg/normativi/MKB10.pdf

[9] https://www.who.int/classifications/icd/ICD-10\%20languages.pdf

[10] https://www.wikidata.org

[11] https://www.ebi.ac.uk/ols/ontologies/doid

[12] https://www.nlm.nih.gov/mesh/meshhome.html

[13] https://hpo.jax.org/app/

[14] https://mondo.monarchinitiative.org/

[15] http://www.orphadata.org/cgi-bin/index.php

[16] https://w.wiki/ZZo

- type 2 - "ci" ⇒ "ци", "ch" ⇒ "х", "qu" ⇒ "кв".
- type 3 - "sua" ⇒ "сва", "sui" ⇒ "суи", "sm" + vowel ⇒ "зм", "rs" + vowel ⇒ "рз"

According to the rules, "basis" transliterates to "базис", "trapez" to "трапец", "xantos" to "кзантос", "sensibilis" to "сензибилис", "neoplasma" to "неоплазма", "suillus" to "суилус", "xiphos" to "кзифос", etc.

The algorithm of transliteration[17] consists of parsing the input string, according to the rules, recognizing groups from the left parts of the rules and generating the output string replacing the left parts with their corresponding right parts. The order in which we applied the rules is from the ones with longest context to the ones with shortest context because the longest ones are more specific and some of the shortest could be their subset, so we give priority to the specificity.

Some of the rules depend on the origin of the word, Greek or Latin, and they are applied only to words with specific origin[18]. For instance, rule only for words with Greek origin is "ph" ⇒ "ф", and rule only for words in Latin origin is "z" ⇒ "ц". So we should create a naive origin extractor. For this purpose we use a corpus of typical prefixes, suffixes and roots of words with a Greek origin and words with a Latin origin . Typical prefixes of words with Greek origin are, e.g., "rhe-", "xanth-", "zon-" and typical prefixes of words with Latin origin are "sub-", "form-", "celer-". In order to extract the origin of a word, we should count the number of prefixes, number of suffixes and number of roots which it contains, respectively for the Greek and Latin ones, and then define the origin to be the one which is more often contained.

A plenty of diseases contain in their notation the name of their founder. For the sake of simplicity, we apply the same rules to the transliteration of names. This is a downside because they should be transliterated according to the transliteration rules of the language they originate from.

Letter "w" does not exist in Latin. It could be transliterated in either German ("в") or English ("у") style. E.g., "Kwashiorkor" should be transliterated to "Квашиоркор", but "Williams" to "Уилиямс". In order not to make the algorithm more complicated using origin extractor of names and applying different rules to name transliteration, we use the German style by default because it occurs more often.

Numbers in Roman notation (they consist of Latin letters) should not be transliterated.

III. Corpora Generation

Further was applied Machine translation using Google Translation of the data from English→Bulgarian (WD-BG); English→Latin (WD-LAT); and transliteration of the result data set in Latin to Cyrillic (WD-TRANS) applying methods described in the next section.

As a result of the creation of datasets, 6 datasets have been generated (see Figure III. There are two options for each of them: one with 4-sign codes and one with 3-sign codes. All have format: <*(ICD-10, Text*>. In the process of merging each of the datasets, the following main transformations are applied:

Table III
ICD10 3 sign and 4 sign datasets statistics

Dataset	**Total Instances**		**Unique Codes**	
	3-sign	**4-sign**	**3-sign**	**4-sign**
ICD10-BG	2025	8946	2025	8946
GS	409	4212	42	408
ICD10-Index	2176	42811	310	8420
WD-BG	3879	46686	434	3499
WD-LAT	3879	46686	434	3499
WD-TRANS	3879	46686	434	3499
Corpus	**189756**	**383042**	**2035**	**10971**

- Transformation of all homoglyphs.
- Remove and convert all obviously incorrectly written codes so that they become valid codes (for example, unnecessary blanks are removed).
- Remove all duplicates.
- Removal of all codes that are not in compliance with the official list of codes used in Bulgaria to date (codes from ICD10-BG).

More detailed statistics can be seen in the Table III and the contribution of each dataset to the generated corpus[19] (see Figure III) and the distribution of instances per classes (see Figure II-C). It is important to mention that “Corpus-4Sign” contains all examples from “Corpus-3Sign”, because all valid 3 sign codes are valid 4 sign codes too.

We can state that the resulting corpus is valid because only official or trusted sources were used to create it and there are no personally associated codes. The open data (ICD10-Index) is an official document from the website of the Ministry of Health in Bulgaria. The Linked Open Data (ICD10-BG) is also 100% reliable as it represents official classifications and ontologies. The gold standard (GS) is made by doctors, which in itself ensures that it is a reliable source. The data taken from Wikidata (WD-LAT and WD-BG) are not an official document, but we can say that they are trusted, as they have been checked through several types of classifications before being approved for publication online. The data generated by transliteration (WD-TRANS) are valid because the rules described in the necessary literature for this are strictly applied for their generation.

IV. Conclusion and Further Work

The paper presents a method for automatic generation of corpora that contains descriptions of diagnoses in Bulgarian and their associated codes in ICD-10-CM. The proposed approach is based on the available open data and Linked Open Data. The proposed approach employs methods for

[17] https://github.com/BorisVelichkov/latin-transliterator

[18] https://www.oakton.edu/user/3/gherrera/Greek\%20and\%20Latin\%20Roots\%20in\%20English/greek_and_latin_roots.pdf

[19] https://github.com/BorisVelichkov/ICD10-Medical-Data/tree/master/datasets

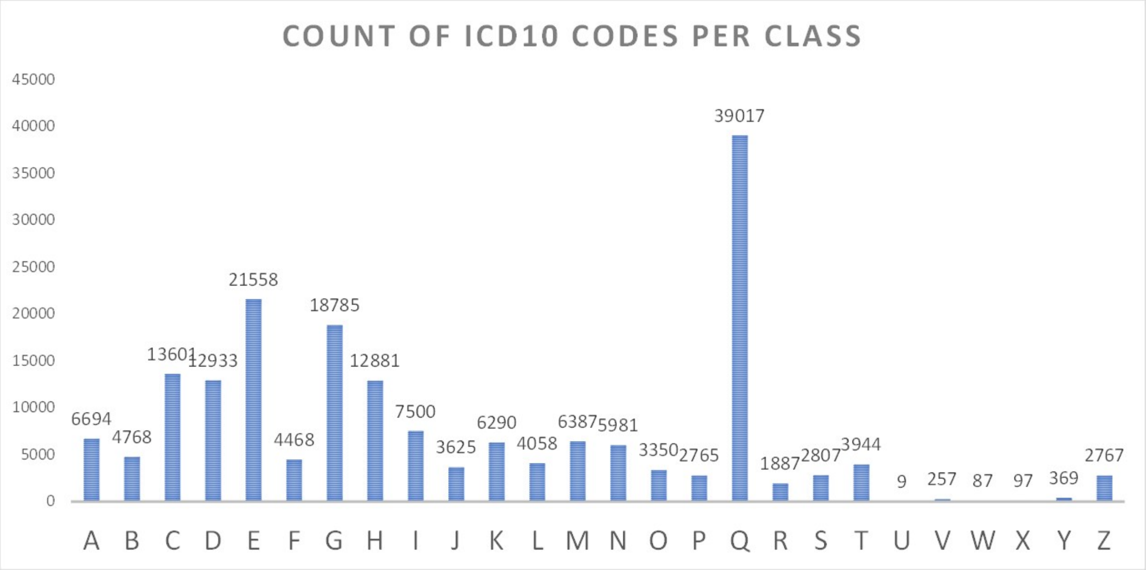


Figure 3. Diagnosis descriptions per ICD-10 code class in the generated corpora

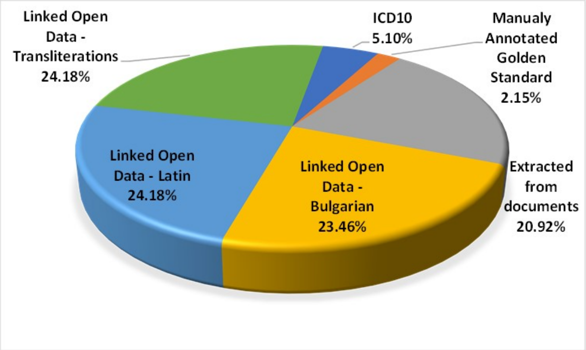


Figure 4. Contribution of each resources to the generated corpora

automatic terms and relations extraction from semi-structured documents; methods for automatic terms extraction from Linked Open Data Cloud and suggests techniques for Latin-Cyrillic transliteration. The resulted corpora generated for the Bulgarian clinical texts consists of about 370,000 pairs and is beyond the usual size that can be generated manually, moreover it was created from scratch and for a relatively short time. Up to our knowledge this is the largest dataset of this type. Further updates of the corpora are also possible whenever new open resources are available or the current ones are updated. The proposed approach is relatively language independent and can be easily adapted for other languages.

Since the generated corpus is highly unbalanced, it is good to do a Data Augmentation [10] in order to reduce the more drastic differences in the number of individual classes. One possible option that would be applicable in the current dataset is through the use of synonyms [11].

REFERENCES

[1] A. Névéol, H. Dalianis, S. Velupillai, G. Savova, P. Zweigenbaum. "Clinical natural language processing in languages other than english: opportunities and challenges." Journal of biomedical semantics, 2018 Dec 1;9(1):12.

[2] S. Boytcheva, "Multilingual aspects of information extraction from medical texts in Bulgarian." Multilingual Processing in Eastern and Southern EU Languages: Less-resourced Technologies and Translation, Cambridge Scholars Publishing. 2012 Apr 25:308-29.

[3] S. Boytcheva, "Automatic matching of ICD-10 codes to diagnoses in discharge letters."*In Proceedings of the second workshop on biomedical natural language processing, RANLP 2011*, pp. 11-18, September 2011.

[4] M. Voinov et al. *Latin-Bulgarian Dictionary*. Planeta-3, pp. 792, 1999. (in Bulgarian)

[5] Q. Wang et al. "A study of entity-linking methods for normalizing Chinese diagnosis and procedure terms to ICD codes". Journal of Biomedical Informatics. 2020 Apr 13:103418. https://doi.org/10.1016/j.jbi.2020.103418

[6] U. Marovac, A. Avdić, D. Janković, and S. Marovac. "Creating Resources for Marking Diagnoses in Electronic Health Reports in Serbian". International Journal of Electrical Engineering and Computing, 2020. 4(1), pp. 18-23.

[7] M. Almagro, R. M. Unanue, V. Fresno and S. Montalvo, "ICD-10 Coding of Spanish Electronic Discharge Summaries: An Extreme Classification Problem", IEEE Access, 2020, vol. 8, pp. 100073-100083, 2020, doi: 10.1109/ACCESS.2020.2997241.

[8] A. Bagheri, A. Sammani, PGM Van der Heijden, FW Asselbergs, and DL Oberski. "Automatic ICD-10 classification of diseases from Dutch discharge letters". In: Proceedings of the 13th International Joint Conference on Biomedical Engineering Systems and Technologies - Volume 3: C2C. 2020, pp. 281-289.

[9] H. Dalianis. "Clinical text retrieval-an overview of basic building blocks and applications". In Professional Search in the Modern World, 2014, pp. 147-165. Springer, Cham.

[10] J. Wei, and K. Zou. "Eda: Easy data augmentation techniques for boosting performance on text classification tasks". arXiv preprint arXiv:1901.11196. 2019 Jan 31.

[11] N. Khairova, S. Petrasova, W. Lewoniewski, O. Mamyrbayev, and K. Mukhsina. "Automatic extraction of synonymous collocation pairs from a text corpus". In 2018 Federated Conference on Computer Science and Information Systems (FedCSIS)". 2018 Sep 9, pp. 485-488, IEEE.

DOI: 10.15439/2020F36
ISSN 2300-5963 ACSIS, Vol. 21

Knowledge Detection and Discovery using Semantic Graph Embeddings on Large Knowledge Graphs generated on Text Mining Results

Jens Dörpinghaus*†, Marc Jacobs†
* German Center for Neurodegenerative Diseases (DZNE), Bonn, Germany, Email: jens.doerpinghaus@dzne.de
† Fraunhofer Institute for Algorithms and Scientific Computing, Schloss Birlinghoven, Sankt Augustin, Germany

***Abstract*—Knowledge graphs play a central role in big data integration, especially for connecting data from different domains. Bringing unstructured texts, e.g. from scientific literature, into a structured, comparable format is one of the key assets. Here, we use knowledge graphs in the biomedical domain working together with text mining based document data for knowledge extraction and retrieval from text and natural language structures. For example cause and effect models, can potentially facilitate clinical decision making or help to drive research towards precision medicine. However, the power of knowledge graphs critically depends on context information. Here we provide a novel semantic approach towards a context enriched biomedical knowledge graph utilizing data integration with linked data applied to language technologies and text mining. This graph concept can be used for graph embedding applied in different approaches, e.g with focus on topic detection, document clustering and knowledge discovery. We discuss algorithmic approaches to tackle these challenges and show results for several applications like search query finding and knowledge discovery. The presented remarkable approaches lead to valuable results on large knowledge graphs.**

I. Introduction

IN THIS paper we will present a novel approach towards knowledge detection and discovery using semantic graph embeddings on large knowledge graphs. The idea of semantic graph embeddings was initially introduced in [1], the theoretical background in [2] and the algorithms which are used as a basis for our approach were introduced in [3]. Combining these results, we will present a novel heuristic approach and present experimental results on a large scale knowledge graph from the biomedical field, see [4]. This graph is build upon text mining results on biomedical literature databases. The real-world use cases were collected from scientific projects.

A knowledge graph has a comprehensible topological representation given by nodes and edges, but this is usually not a very precise representation of the real world. A more generic approach can be constructed by using classes. Thus the basic idea is to divide a knowledge graph in different knowledge layers either directly given by the data (like documents, authors) or manually defined. For example biological relations might be associated with an ontology (ontology layer), they can be annotated to a document with named entity recognition (NER, annotation layer) and they might belong to a domain specific language layer (for example BEL, biological expression language, layer). See figure 1 for an illustration. This approach is similar to the idea of molecular information layers described in [5]. To sum up, we build linked data from different data sources and ontologies. We use text mining and natural language processing approaches to make these linked data information interoperable, findable and re-usable. Thus, every data type from a data source implies a different layer and those layers are either linked with relations given in the data source or by text mining.

The testing system is based on Neo4j and holds a dense large scale labeled property graph with more then 75M nodes and 960M edges. This graph is based on biomedical knowledge graphs as described in [6] and [7].

This paper is divided into six sections. The first section gives a brief overview of the state of the art and related work. The second section describes the theoretical background and the methods used for our novel approach. We will introduce knowledge graphs, semantic graph embeddings and algorithms. In the third section, we present applications from real world use cases like search query finding and generating and optimisation of cluster labels. The fourth section is dedicated to experimental results on artificial and real-world scenarios. Our conclusions are drawn in the final section.

We will propose two novel algorithmic approaches which present promising performance. The results show a significant improvement over the existing engine without using context information.

II. Related Work

In recent decades the field of natural language processing (NLP) and knowledge discovery as well as the related fields data mining and the management of information systems is emerging. Several authors like Manning et al. [8] or Clarc et al. [9] give an overview about the algorithmic part of computational linguistics and NLP. In addition there is a constant interest in using graphs for these problems, see [10].

In scientific research, expert systems provide users with several methods for knowledge discovery. They are widely used to find relevant or novel information. For example, medical and biological researchers try to find molecular pathways, mechanisms within living organisms or special occurrences of drugs or diseases. Using expert knowledge as an input, researches usually consider an initial idea and some content like papers or other documents. The most common approach

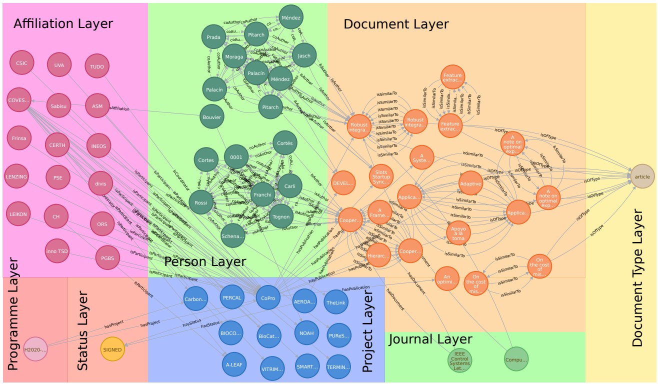


Fig. 1: (Illustration of some knowledge graph layers found in the testing environment. Here, we can see some document-specific layers which are combined from several data sources (PubMed, DBLP, H2020): Document Type Layer, Journal Layer, Person (Author) Layer. Other layers are specific to the H2020 project data obtained from EU Open Data Portal: Project Layer, Status Layer, Programme Layer and the Affiliation Layer. We notice several intersections, for example `Quentin_Bouvier` is no Author, but has both an affiliation and is associated with the project `NOAH`.

is inquiring a search engine to find closely related information. Thus two question are most frequently asked: "How can I find these documents?" to adjust the search query for knowledge discovery or "What are these documents all about?" to find the topic. Both questions are heavily related to the context of documents. Meta-data like authors, keywords and text are used to retrieve results of a query using a search engine. Current research in NLP and text mining usually does not directly focus on finding a search query from a given corpus, although a lot of research has been done on the analyses of a given search query, see [11] or the analyses of queries on different databases, see for example [12] for PubMed data. Topic labeling – or cluster labeling – is under constant research in several research areas.

There is a considerable amount of literature on both problems. Many studies have been published on probabilistic or machine-learning-approaches, see [13], [14] or [15]. In addition, in recent years there has been growing interest in providing users with suggestions for more specific or related search queries, see [16]. We already mentioned [17] but most research focuses on artificial intelligence (AI), machine learning (ML) or deep learning (DL) approaches, see [18] or [19]. Our aim is a precise solution without a prior learning step giving a deeper insight in the data and the context of this data.

Here, knowledge graphs are becoming a key instrument for knowledge discovery and modeling. These approaches rely on structured data, e.g. about related proteins or genes, and form cause-and-effect networks or – if enriched with literature data and other linked datasources – knowledge graphs. A key aspect of analysis on these graphs is the missing context.

III. Method

A. Knowledge Graph

Knowledge graphs play in general an important role in recent knowledge mining and discovery. A *knowledge graph* (sometimes also called a *semantic network*) is a systematic way to connect information and data to knowledge on a more abstract level than language graphs. It is thus a crucial concept on the way to generate knowledge and wisdom, to search within data, information and knowledge. The context is a significant topic to generate knowledge or even wisdom. Thus, connecting knowledge graphs with context is a crucial feature.

Many authors tried to give a definition of knowledge graphs, but still a formal definition is missing, see [20]. In [21] the authors compared several definitions, but the only formal definition was related to RDF graphs which does not cover labeled property graphs. Thus, here we propose a very general definition of a knowledge graph using graph theory:

Definition III.1. *(Knowledge Graph) We define a knowledge graph as graph $G = (E, R)$ with entities $e \in E = \{E_1, ..., E_n\}$ coming from a formal structure E_i like ontologies.*

The relations $r \in R$ can be ontology or layer relations (like "is related to" or "is co-Author"), thus in general we can say every formal structure E_i which is part of the data model is a subgraph of G indicating $O \subseteq G$. In addition, we allow inter-structure relations between two nodes e_1, e_2 with $e_1 \in E_1$, $e_2 \in E_2$ and $O_1 \neq E_2$. In more general terms, we define $R = \{R_1, ..., R_n\}$ as a list of either inter-structure or inner-structure relations. Both E as well as R are finite discrete spaces. See figure 3 for an example.

Every entity $e \in E$ may have some additional meta information which needs to be defined with respect to the application of the knowledge graph. For instance, there may be several node sets (some ontologies, some document spaces (patents, research data, ...), author sets, journal sets, ...) $E_1, ..., E_n$ so that $E_i \subset E$ and $E = \cup_{i=1,...,n} E_i$. The same holds for R when several context relations come together such as "is cited by", "has annotation", "has author", "is published in", etc.

The basis for generating our large-scale Knowledge Graph representation is biomedical literature (e.g.from PubMed and PMC). We also integrated bibliographic data and metadata from DBLP, monthly snapshot release of December 2019, see https://dblp.uni-trier.de/ and [22]. Since the basic data coming from SCAIView is already annotated with different biomedical ontologies, we decided to use the CSO classifier (see [23]) to annotate CSO to DBLP data.

We enriched our graph with data from the EU Open Data Portal (CORDIS - EU research projects under Horizon 2020, see https://data.europa.eu/euodp/en/data/dataset/cordisH2020projects). This data set is free to reuse for both commercial or non-commercial purpose. Here, we integrated projects, their status, affiliations, persons and authors of publications mentioned in their data set.

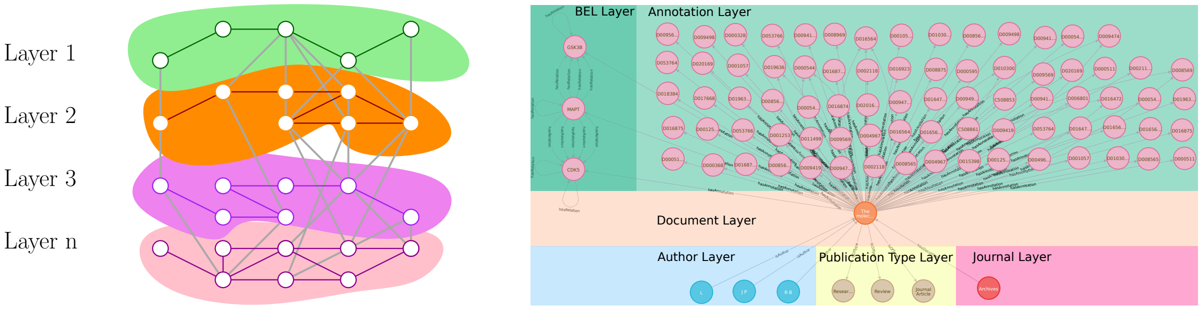


Fig. 2: (left) Illustration of the *knowledge graph embedding* between different layers. Here, every layer corresponds to a context defining of new contexts on several other layers. Thus layers and contexts are flexible and can be defined in a feasible way for every application. Data within the Knowledge Graph can be ordered according to context and information to data layers (e.g. a molecular or mechanism layer). This helps to examine novel causal connections and context. Layer 1 defines *Macro-Context* as Information Highway. The ordering of layers is based on the questions asked. It may also be used to allow an easy and FAIR access to the data and benefit from semantic graph-queries. Date integration, adding more data will increase the Knowledge-Foundation and gives a more precise view on the micro-context and helps to unveil new context and insights. This is a method from top to bottom, the other direction is dedicated to Data Mining.
(right) Examle illustration of different layers obtained by document *The molecular bases of Alzheimer's disease and other neurodegenerative disorders* (PMID:11578751).

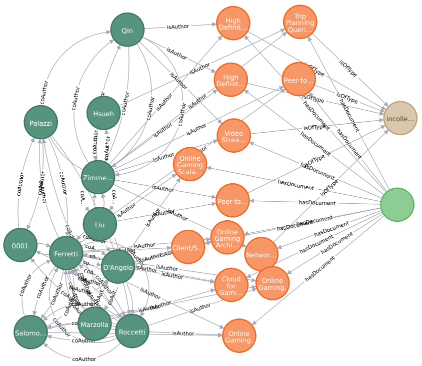


Fig. 3: Illustrations of inter-structure and inner-structure relations. Here, E_1 (left) is a structure containing authors, having an inner relation indicating co-authors. E_2 (orange) is a structure containing documents with an inner relation indicating for example citations. We can see inter-structure relations between both structures indicating authorship.

The articles or abstracts are the source for biological relations. In addition, meta information like authors, journals, keywords, etc. are available. Ontologies can be used to contextualize entities in the knowledge graph providing biological or medical relations. Every ontology will form another knowledge (sub-)graph. Using methods of natural language processing (NLP) and text mining, we can combine and link these knowledge graphs to a giant and very dense new knowledge graph. This will meet a very general definition of context. We can see every knowledge (sub-)graph as context to another. Biological expressions are context of the corresponding literature, authors are context of a text, named entities from ontologies found in a text are context to it or to the corresponding biological expression.

Several ontologies and terminologies were added to the knowledge graph, for example Computer Science Ontology (CSO, see http://cso.kmi.open.ac.uk/home), HUGO Gene Nomenclature Committee (HGNC, see [24]), Gene Ontology (GO, see [25] and [26]) or Disease Ontology (DO), see [27]). These ontologies can be used to annotate context with methods from text mining to data entities within the graph, see [6].

B. Semantic Graph Embeddings

Semantic graph embeddings are closely related to the concept of context. Here, we use a quite general definition of context data. We assume that every information entity can also be a context information for other entities. For example a document can also be a context for other documents (e.g. by citing or referring to the other publication). An author is both a meta information to a document, but also itself context (by other publications, affiliations, co-author networks, ...). Other data is more obvious a context: named entities, topic maps, keywords, etc. extracted with text mining from documents. But already relations extracted from a text may stand for themselves, occurring in multiple documents and still valuable without the original textual information.

Definition III.2. *(Context) We define context C as a set with context subsets $C = \{c_1, ..., c_m\}$. This is a finite, discrete set. Every node $v \in G$ and every edge $r \in R$ may have one or more contexts $c \in C$ denoted by $con(v) \subset G$ or $con(r) \subset G$.*

It is also possible to set $con(v) = \emptyset$. Thus we have a mapping $con : E \cup R \to \mathcal{P}(C)$. If we use a quite general approach towards context, we may set $C = E$. Therefore, every inter-ontology relation defines context of two entities, but also the relations within an ontology can be seen as context,

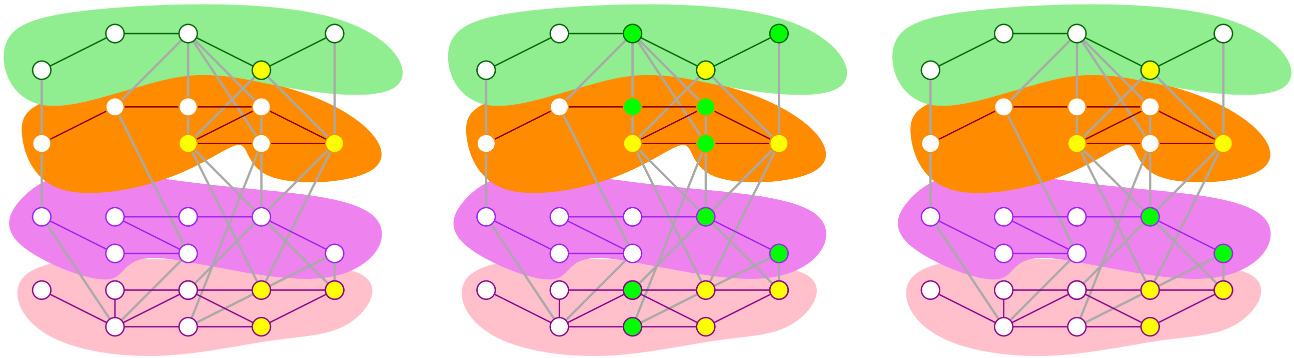


Fig. 4: Illustration of the steps for generating a semantic graph embedding $\mathcal{E}_L$ where N is given by the yellow nodes and L is given by the pink layer (1), see example III.5. Subfigure (2) depicts the output of algorithm 1, $\mathcal{E}(N) = N \cup con(N)$. Limiting this to L returns $\mathcal{E}_L = \mathcal{E}(N) \cap L$, see subfigure (3).

With the neighborhood $N(E_i)$ every node set $E_i \in \{E_1, ..., E_n\}$ induces a subgraph $G[E_i] \subset G$:

Definition III.3. *(Semantic Graph Embeddings) With $G^c[E_i] = G[E_i] \cup N(E_i)$ we denote the extended context subgraph or semantic graph embedding which also contains the neighbors of each node in G, which is context of that node. With $G^c_L[E_i] = G^c[E_i] \cap L$ we denote the graph embedding on layer $L \subset G$.*

To make the notation easier, we set $\mathcal{E}(N) = G^c[N]$ and $\mathcal{E}_L(N) = G^c_L[N]$.

For a graph drawing perspective, if $G^c[E_i]$ defines a proper surface, we can think about a graph embedding of another subgraph $G^c[E_j]$ on $G^c[E_i]$. This concept was introduced in [1]. Here, semantic knowledge graph embeddings were displayed between different layers. Every layer (for example: molecular layer, document layer, mechanism layer) corresponds to another context defining new contexts on other layers.

Example III.4. *Consider the illustration in figure 2: Here we can see, that every subgraph L' of a layer $L_1, ..., L_n$ has an extended context subgraph $G^c[L'] = G[L'] \cup N(L')$ in multiple layers. In addition, if we have a set of nodes L'' in multiple layers L_i, L_j the same holds. Thus to see the embedding on just one layer L_i we can limit this set using $G^c_{L_i}[L'] = G^c[L'] \cap L_i$.*

If the mapping con is well defined for the domain set, then Graph H can be generated in polynomial time. Since this is generally not the case, this step usually contains data or text mining task to generate other contexts from free texts or knowledge graph entities. With respect to the notation described in [2] this problem p can be formulated as

$$p = \mathbb{D}|R|\mathbf{f} : \mathbb{D} \to \mathbb{X}|err|\emptyset \qquad (1)$$

Here, the domain set $\mathbb{D}$ is explicitly given by $\mathbb{D} = G$ or – if additional full-texts $\hat{D}$ supporting the knowledge Graph G exist – $\mathbb{D} = \{G, \hat{D}\}$, which in our case is the domain subset $R = \mathbb{D}$. Therefore, we need to find a description function $f : \mathbb{D} \to \mathbb{X}$ with a description set $\mathbb{X} = C$ which holds all contexts. To find relevant contexts, we also need to measure the error as defined by $err : \mathbb{D} \to [0, 1]$.

C. Heuristic

To solve the knowledge graph embedding problem, we will use an extended version of algorithm 1 introduced in [3] within the field of document set cover. In our case, the input documents $\{d_1, ..., d_n\} \subset \mathbb{D}$ can be seen as any elements or nodes $\{n_1, ..., n_n\} \subset V$. The descriptive elements $f(d_i) = \{x_1, ..., x_m\} \subset \mathbb{X}$ are now given by the context $con(n_i) = \{c_1, ..., c_m\} \subset V$. See algorithm 1 for pseudocode.

Algorithm 1 s-GRAPH-EMBEDDING

```
Require: $N = \{n_1, ..., n_n\} \subset V$ and descriptive elements
    $con(n_i) = \{c_1, ..., c_m\} \subset V$, maxiter as maximum of
    iterations, $s$ as sensitivity
Ensure: A semantic graph embedding $\mathcal{E}(N) = (V', E')$ of
    $N$ with elements in $V$.
    $con' = con$
    for every $v \in N$ do
        while iteration<maxiter AND $con'(v) > (s \cdot con(d))$ do
            remove $c \in con'(v)$ with maximum weight
        end while
    end for
    return $\mathcal{E}(N) = (\{c, \forall c \in con'(n)\} \cup \{n, \forall c \in con'(n) \forall n \in N\}, \{(c, n), \forall c \in con'(n) \forall n \in N\})$
```

Example III.5. *See the example in figure 4. Here, we use algorithm 1 to compute a semantic graph embedding $\mathcal{E}_L$ where N is given by the yellow nodes and L is given by the pink layer. We set $s = 1$ and $maxiter = 1$. The context in this example is defined as neighborhood in the graph, thus $con(v) = N(v)$. Algorithm 1 outputs both yellow and green nodes, which is $N \cup con(N)$. In this simplified example algorithm 1 returns $\mathcal{E}(N) = N \cup con(N)$. Limiting the graph embedding to the pink layer leads to $\mathcal{E}_L = \mathcal{E}(N) \cap L$.*

If we use documents for the input N and only keywords as descriptive elements, algorithm 1 works exactly the same as described in [3]. Thus, our approach is a generalization of the initial algorithm to all descriptive elements found in any descriptive layer in a knowledge graph. Again we can argue, that that – while not limiting to a distinct layer – the algorithm outputs at least the initial nodes given in N. If the sensitivity is decreased to $s < 1$ we can see that less and less descriptive elements are chosen. In the next section, we will explain how to use this semantic graph embedding to knowledge discovery within the knowledge graph.

IV. Application

The initial research question was how to apply a general context added to biomedical knowledge graphs to answer several generic questions dedicated to knowledge discovery. As described above we have integrated several sources of publication data (PubMed, DBLP, H2020), several ontologies like GO, HGNC and mappings, BEL networks from Parkinson's and Alzheimer's disease as well as other structured data.

A. Search Query Finding and Knowledge Discovery

In [2] we proposed a very generic definition of search engines and search queries. Here, we will show, how this generic approach can be used to create real world search queries. A search engine is a function $q : \mathbb{X} \rightarrow \mathbb{D}$ which outputs a set of documents or any other content of the domain set if the input is a subset of a description set $\mathbb{X}$ which we call search query. With this, it follows that the problem of finding a search query is given by

$$p = \mathbb{D}|R|\mathbf{X}|err|R$$

Given a knowledge graph $G = (V, E)$ with layers $L_1, ..., L_n$. We denote L_D with the document layer. Let $D' \subset L_D$ be an initial set of documents, and let $\mathcal{E}_{L_D}(N) = D'$ be the semantic graph embedding on L_D. Thus, $\mathcal{E}(N) \cap L_D$ holds all descriptive elements of all documents in D' in other layers. If all layers can be used to search for documents, this returns a search query for D'. In [3] we proved this concept for one single layer containing keywords.

In order to get a feasible search query, we need to modify algorithm 1. In algorithm 2 we propose a generic approach not limited to a distinct layer returning a logical concatenation of nodes that are related to the semantic graph embedding. We call this a *semantic graph description* of D'.

Changing the value of s makes the search query more or less precise which helps with respect to knowledge discovery. For example, given a set of documents we may use them as seed to discovery more related documents. Here, choosing the right description layers is quite important.

Algorithm 2 s-Graph-Description

```
Require: $N = \{n_1, ..., n_n\} \subset L$ and descriptive elements
    $con(n_i) = \{c_1, ..., c_m\} \subset V$, maxiter as maximum of
    iterations, $s$ as sensitivity
Ensure: A semantic graph description $\mathcal{E}(N) = (V', E')$ of
    $N$ with elements in $V \cap L$.
  $con' = con$
  for every $v \in N$ do
    while iteration<maxiter AND $con'(v) > (s \cdot con(v))$
    do
      remove $c \in con'(v)$ with maximum weight
    end while
  end for
  return $Z = \vee_{v \in N}(\wedge_{x \in con'(v)})$
```

B. Generating and optimisation of Cluster Labels

In [2] we proposed a very generic approach towards cluster labeling. Given a knowledge graph $G = (V, E)$ finding cluster labels for clusters $C_1, ..., C_n$ is the task of assigning a subset of a description set $\mathbb{X}$, in our case on or more layers, with the description function $f : \mathbb{V} \rightarrow \mathbb{X}$ to a cluster $C \in \{C_1, ..., C_n\}$. Thus, this problem is given by

$$p = \mathbb{D}|C|\mathbf{X}|err|R$$

where the resulting label set is the image $f(C) \subset X$. Depending on the choices of different layers to be included in $\mathbb{X}$ this either leads to a set of metadata, terms from ontologies, sentences or any subset of natural language.

Once again we can apply the modified algorithm 2. As input, we use a set of nodes forming a cluster $C \subset G$. The return value needs to be filtered according to our choice of $\mathbb{X}$. As suggested in [3] we can either transform the logical operators to language (term x and term y or term z) or use a very low threshold which will lead to very small return value and return a ranked list of terms.

C. Document or Data Clustering

Document or data clustering is a specific application of text or data mining and a sub-problem of cluster analyses. Without any clusters pre-defined the goal is to cluster documents or data points to clusters sharing common features. Limiting the layers to documents will result in document clustering. If the knowledge graph layers contain any data points, this will result in data clustering. The application of clustering is a wide and open field and in terms of complexity it is still under heavy research, see for example [28] and [29].

Clustering is usually not perceived as a graph problem, although several attempts have been made (e.g. [30]) and here we will show how to generalize it on knowledge graphs. Usually the problem can be formulated in the following way: Given a similarity function for the document or data space D as $sim : D \times D \rightarrow \mathbb{R}^+$ and an $\epsilon \in \mathbb{R}^+$. We search for a minimal number of clusters, so that every two documents x, y in one cluster have $sim(x, y) \geq \epsilon$. For technical terms we refer to [8].

One common problem is to find sim. Here, the inverse problem helps: Given two data points d_1, d_2 they can be interpreted as an embedding of different layers. Thus by

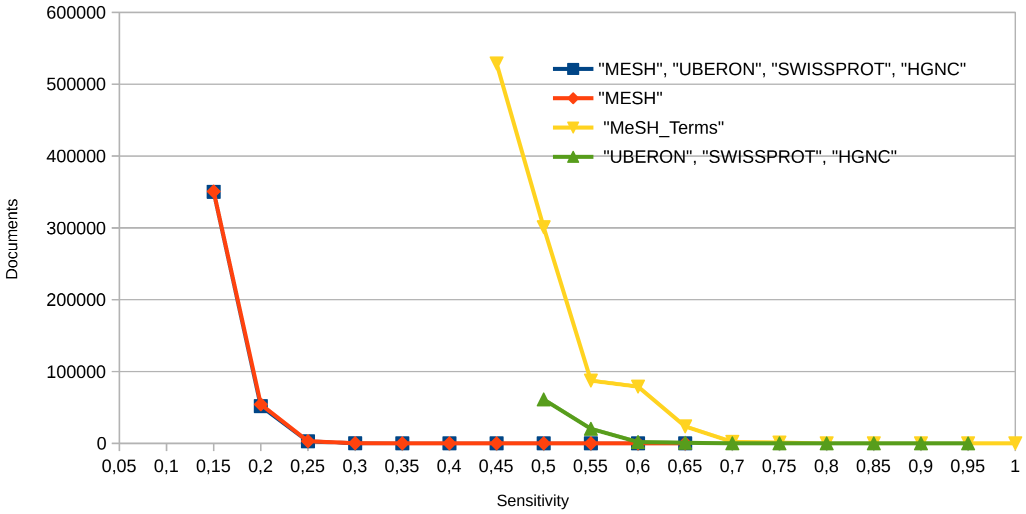


Fig. 5: Example outputs of heuristic for Corpus "Alzheimer Disease" with different layers. We used "MeSH_Terms" (manually annotated keywords from MeSH), MeSH (NER using terms in MeSH), SWISSPROT, HGNC and UBERON (NER). As we can see, the precision varies and depends on which layers are used. The text mining based MeSH has a great impact on the results, whereas the manually annotated expert knowledge from "MeSH_Terms" lead to a totally different result. For knowledge discovery, it is very important to choose the right value for s and to choose the correct layers.

changing algorithm 2 we can compute the distance between any two reverse embeddings or descriptions, see algorithm 3.

Algorithm 3 s-GRAPH-DISTANCE

```
Require: $d_1, d_2 \subset L$ and descriptive elements $con(d_i) = \{c_1, ..., c_m\} \subset V$, maxiter as maximum of iterations, $s$ as sensitivity
Ensure: A semantic graph distance $sim(d_1, d_2)$ of $d_1, d_2$ with elements in $V \cap L$.
    $con' = con(d_1)$
2:  while iteration<maxiter AND $con' > (s \cdot con(d_2))$ do
        remove $c \in con'$ with maximum weight
4:  end while
    $con1 = con'$
6:  $con' = con(d_2)$
    while iteration<maxiter AND $con' > (s \cdot con(d_2))$ do
8:      remove $c \in con'$ with maximum weight
    end while
10: $con2 = con'$
    return $\frac{|con1 \cap con2|}{|con1 \cup con2|}$
```

In line 11 we compute the Jaccard similarity but any other distance measure is also possible. This describes two benefits of the knowledge graph approach: First, data clustering is a generalization of document clustering. Second, the similarity measures can be computed by using any other data layers and can be setup to fit the applications needs.

D. *Knowledge Discovery on custom Layers*

Combining both algorithm 1 and a custom layer in the knowledge graph we can use this for quite general knowledge discovery. Given a knowledge graph $G = (V, E)$ with layers $L_1, ..., L_n$. Let N be a set of nodes which form a subgraph $N \subset G$ of the knowledge graph G. These nodes can be seen as input data. If we generate a new custom layer L' which consists of data from different layers we can use algorithm 1 to embed the input data in the new layer.

We can generate several examples from NLP and text mining for this. For example, we can use this for text classification. If N contains only textual data (e.g. scientific literature from DBLP or PubMed) we can use several subsets of connected data to obtain the classes of any document. For text recognition we may also use subsets of layers which are not directly connected to documents. Given figure 1 we may use H2020 programmes or affiliations to recognize or classify whether a text belongs to a class or not.

V. EXPERIMENTAL RESULTS

The validity and correctness of the proposed algorithm in general was shown in [3]. Here, we will present some experimental results to show the correctness of the proposed algorithms on a multi-layer knowledge graph comprising multiple terminologies and the results of one specific knowledge discovery on custom layers within the context of dementia research.

A. *Search Query Finding and Knowledge Discovery*

Here, we will describe some results using algorithm 2. By design, the heuristic returns the original set of documents and a set of novel documents. Thus, the precision starting with a large value of sensitivity is in general 1.

The testing was done on a set of small literature corpora collected by scientists. Here, we present results using a corpus

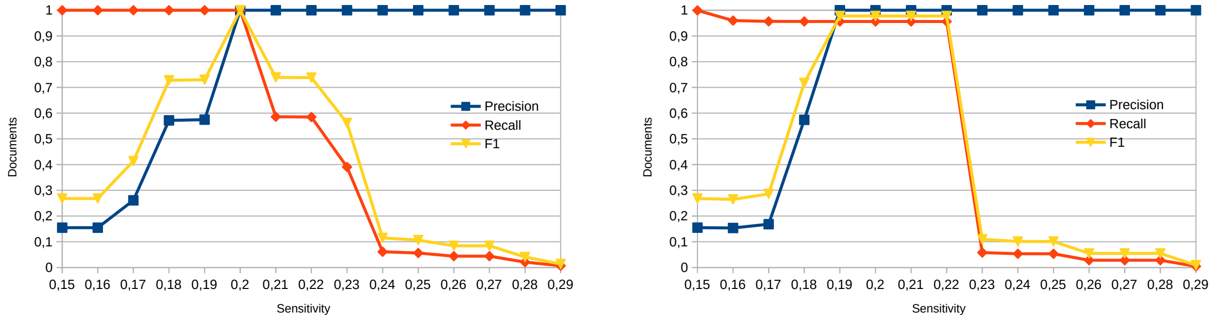


Fig. 6: Curves describing both precision and recall as well as the F1 score for the "Alzheimer Disease" corpus and a gold standard containing 54251 documents. The results were computed using "MESH" and "HGNC" layers (left) and "MESH", "UBERON", "SWISSPROT" and "HGNC" layers (right) in the knowledge graph. It is obvious, that the gold standard was generated using MeSH-Terms. Changing layers has a great impact on the results.

of documents dedicated to alzheimers disease. First of all, we tested the algorithms with a layer of manually annotated keywords, the so called MeSH terms obtained from PubMed. We repeated the testing with several sensitivity values, see figure 5. Starting with the initial 8 documents, the amount of documents increases to 52 when using a sensitivity of 0.95 and rapidly increases to 1078 documents at 0.75. Using MeSH as a terminology used by named entity recognition the number of documents increases to 15 when the sensitivity is less than 0.45. Using all terminologies ("MESH", "UBERON", "SWISSPROT", "HGNC") the result only changes by a few documents, whereas the only usage of "UBERON", "SWISSPROT", "HGNC" changes the picture very much. We can see that different layers in the knowledge graph give a different view on the document layer and return different results.

To analyse the results, we used a manually generated gold standard for Alzheimers disease containing 54251 documents which was generated using the MeSH-Terms. We computed results using "MESH" and "HGNC" layers and "MESH", "UBERON", "SWISSPROT" and "HGNC" layers, see figure 6. We computed both precision, recall and F_1 score which is the harmonic mean of both precision and recall. With true positives (TP) in the gold standard, false positives (FP), false negatives (FN) and true negatives (TN) the precision is given by $p = |TP|/(|TP| + |FP|)$ and recall by $r = |TP|/(|TP| + |FN|)$. With this we can compute F_1-score as $F_1 = \frac{2pr}{p+r}$.

The results in figure 6 show that the quality of results are related to the layers used and whether they were used to manually generate a gold standard. They indicate that the returned documents and their relevance relies on both the used knowledge graph layers as well as the sensitivity used. Thus, the evaluation of the proposed methods needs to consider the use case. Do we need to retrieve just a few more documents closely related to a set of documents or do we want to find all documents within a corpus. Together with the results in figure 5 we would need to discuss how the best value for sensitivity can be found.

B. *Knowledge Discovery on custom Layers*

We have tested the custom layer approach on a biomedical use case in the field of neurodegeneration. Alzheimer's disease (AD), also referred to simply as Alzheimer's, is a chronic neurodegenerative disease that usually starts slowly and gradually worsens over time. It is the cause of 60–70% of cases of dementia. The cause of Alzheimer's disease is poorly understood. There are no medications or supplements that have been shown to decrease risk of acquiring AD and there are no treatments stop or reverse AD progression. The human brain pharmacome project focuses on the design and construction of a dedicated knowledge base for human brain pharmacology. We used the approach discussed in this paper to create this pharmacology knowledge base, referred to as the Human Brain Pharmacome (HBP) as a unique and comprehensive resource that aggregates data and knowledge around current drug treatments that are available for major brain and neurodegenerative disorders. The HBP knowledge base provides data at a single place for building models and supporting hypotheses. Because knowledge-driven approaches to model the relevant biology and chemistry are inherently limited by the completeness and correctness of their associated knowledge assemblies, natural language processing and relation extraction are used to continuously extract biomedical relations from the recent biomedical literature and prioritize for semi-automated curation and update. One application for the HBP is Drug repositioning (also called drug repurposing). It involves the investigation of existing drugs for new therapeutic purposes. One of the main advantages of drug repositioning lies in the reduced number of required clinical trial steps and this could potentially could reduce the time and costs for the medicine to reach market

We used our knowledge graph to search for interesting targets, how these targets are linked to AD and what drugs are known to interact with these targets. As can be seen in figure 8, AD can be linked to the gene CD33 which is altered in some patients suffering from the disease. The gene is coding for a protein also named CD33 which is involved in several biological processes. Microglial activation is one of these processes that can be linked to phagocytosis. In a multicellular organism's immune system, phagocytosis is a major mechanism used to remove pathogens and cell debris. The ingested material is then digested in the phagosome. Phagocytosis is one of the main mechanisms of the innate immune defense. It is one of the first processes responding to infection, and is also one of the initiating branches of an adaptive immune response.

We have integrated H2020 data from EU Open Data Portal which contains several data fields. Persons, affiliations and documents can also be found in DBLP or PubMed data. Thus we get an linked data knowledge graphs combining H2020 data with text mining on documents from other sources.

Carefully considering the H2020 data we found for all projects, their meta data, research institutes, researchers and publications. Not all publications and persons are described. For example only 6 researchers are affiliated with Fraunhofer in this data set. Thus using H2020 as provenance, we get a fare more sparse dataset for Fraunhofer, whilst DBLP or PubMed lists all past and present affiliations in the context of publications. In addition, not all documents are listed. Querying PubMed with project acronyms usually returns more results.

In our knowledge graph the H2020 funded project PHAGO is linked to the topic of phagocytosis. In figure 9 we present a subset of the PHAGO project graph as seen by H2020. Within this project several papers to the role of CD33 and TREM2 in the process of phagocytosis and its context to AD have been published. We can directly identify experts working in the field and the organizations they are working in by switching the context. We can make several observations. First of all, the authors involved in the publications do not intersect with the researchers which are affiliated with the institutes. This is due to the fact that usually only a few researchers are mentioned in projects, thus the researchers illustrated are found in a different project scope. Thus, for knowledge discovery we can use project, documents and authors. Figure 7 illustrates the different layers.

Our goal is to understand the embedding of a H2020 project called Phago in the context of scientific literature and drug databases. Phago is related to Alzheimer's disease and studies TREM2, CD33 and related pathways in this field. Thus, we are interested in overlaps between the knowledge graph embedding towards other Alzheimer's networks, for example [31], and in drug networks, for example [32]. Thus as custom target layers we use PubMed documents, BEL networks and NE coming from the Alzheimer's network, Substances from PubChem, PharmGKB[1] and DrugBank.

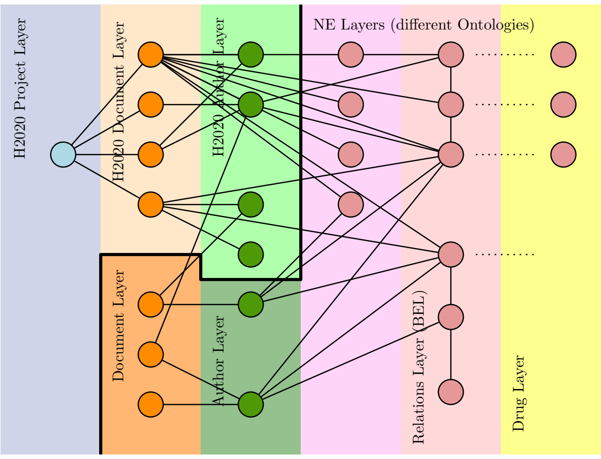


Fig. 7: An illustration of the different layers involved in exploring H2020 data. The first layer – H2020 projects – is just contained in H2020 data. Documents and Authors both contain data from H2020 and other sources. All other layers contain data from different ontologies and terminologies. They are connected using NLP and text mining technologies and also contain intra-ontologie relations like biological or cause-and-effect relations.

Applying the method proposed in section IV-D we obtain a graph containing 126 documents, all from PubMed. We receive 29 substances and descriptive elements from MeSH and MeSH-Terms. In addition, we were able to find biomedical relations from different networks containing more than 133 entites from MeSH, 25 proteins and more than 66 genes, see figure 8 for a subset network illustration.

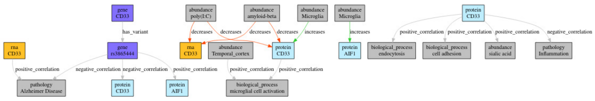


Fig. 8: Biomedical relation subnetwork linked with document `PMID:30037848` entitled "Mycobacterial PknG Targets the Rab7l1 Signaling Pathway To Inhibit Phagosome-Lysosome Fusion".

VI. Conclusion and Outlook

Big Data approaches using NLP technologies on natural language are an emerging topic in all data-driven fields. More and more extensive data is being collected, e.g. in medicine, engineering and also in the humanities (so-called "digital humanities"). To evaluate this data, new methods from the fields of artificial intelligence (AI), big data and high performance computing must be developed. For example, in medical research and digital health the massive data available build the basis for a multitude of predictive medicine Machine Learning (ML) and AI approaches. This includes also the organization of this data (knowledge management) in order to achieve reproducible research and to benchmark and evaluate these methods since both training and validation data are required.

Knowledge graphs play a central role in tackling these challenges. They address central ethical standards of science: reproducibility, transparency and a fair and – if possible –

[1] See https://www.pharmgkb.org/.

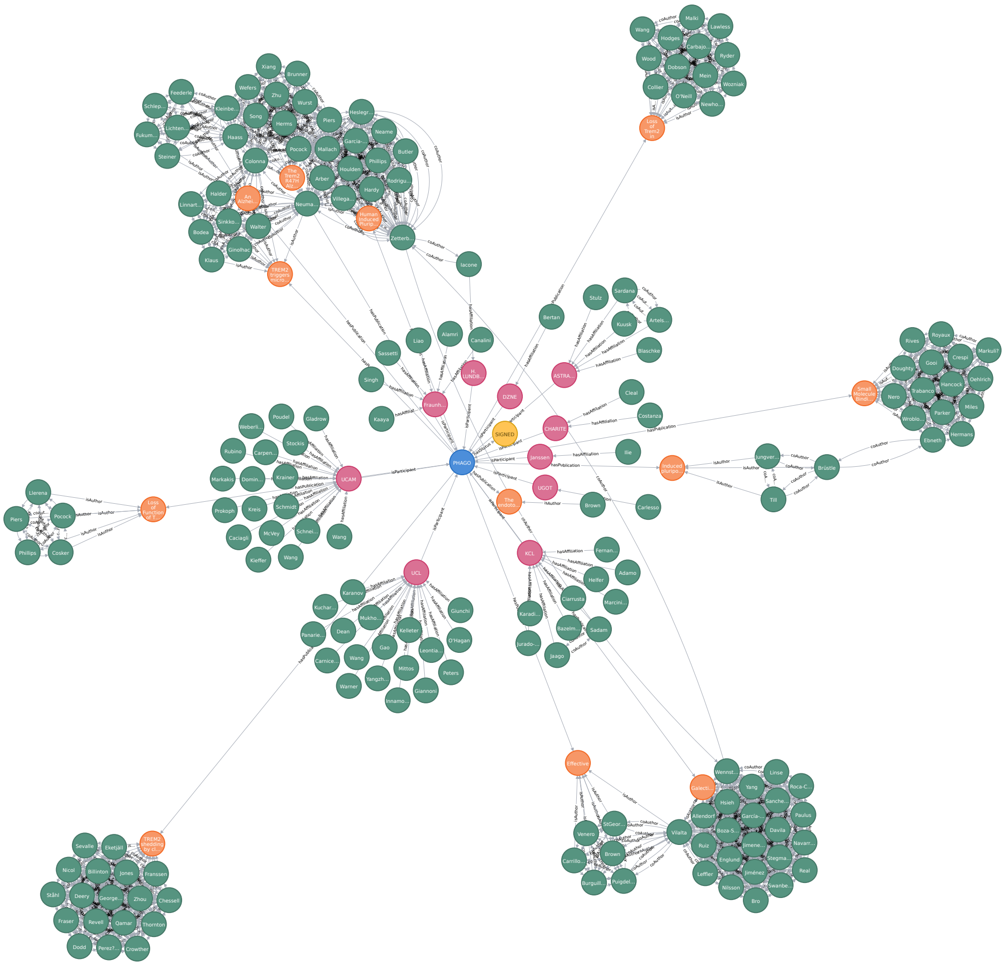


Fig. 9: A subset of the PHAGO project graph as seen by H2020. Blue nodes refer to H2020 projects, red nodes to research institutes, green nodes to persons and orange nodes to documents. Persons, affiliations and documents can be found in DBLP or PubMed data. Thus we get an linked data knowledge graphs combining H2020 data with text mining on documents from other sources.

open, handling of data. These can be summarized with the "FAIR Data" principle, which was published in 2016 by Wilkinson et al. [33]. FAIR as an acronym refers to Findable, Accessible, Interoperable and Re-usable. A central component of FAIR Data is the semantic preparation of knowledge in a format that allows not only the search and retrieval of (meta-)data, but also interoperability and reusability. This provides the central data for the application of AI methods since knowledge graph aim at comparing research data records from different sources as well as the selection of relevant data sets using graph-theoretical algorithms. Making data interoperable and accessible is necessary to develop next-generation services in NLP and text mining.

Here we presented a novel semantic approach towards a context enriched biomedical knowledge graph utilizing data (PubMed, DBLP, H2020, biomedical network) integration with linked data and text mining (NER, relation extraction) which is based on a recent approach that annotates research data with context information. The result is a knowledge graph representation of data, the context graph. It contains computable statement representation (e.g. RDF or BEL). This graph allows to compare research data records from different sources as well as the selection of relevant data sets using graph-theoretical algorithms. It can be used as a reference system for question-answering-processes and it can be a dedicated tool that assists and guides knowledge discovery.

We showed, that this graph concept can be used for graph embedding applied in the described different approaches, e.g with focus on topic detection and knowledge discovery. We discussed several algorithmic approaches to tackle these challenges and show results for three applications: search query finding, generating cluster labels and knowledge discovery. The presented remarkable approaches lead to valuable results on large knowledge graphs. We faced several issues with data integration and missing data, for example because the input data had a bad quality. In addition we have not yet worked on the problem of author and affiliation disambiguation.

We compared the results of different knowledge graph layers on a text corpus. We could show that the graph embeddings itself is only valuable for different use cases when choosing the right layers and sensitivity. Although we have proven that this approach is valid, we might need to evaluate more methods to compute or estimate values for s and the knowledge graph layers. This has thrown up many questions in need of further investigation.

VII. Acknowledgments

We thank Martin Hofmann-Apitius and Vanessa Lage-Rupprecht for valuable suggestions, Bruce Schultz for his technical help and Vanessa Lage-Rupprecht for carefully revising the manuscript.

This manuscript has been supported by Fraunhofer Society under the MAVO Project; Human Brain Pharmacome.

References

[1] J. Dörpinghaus and M. Jacobs, “Semantic knowledge graph embeddings for biomedical research: Data integration using linked open data,” *Posters and Demo Track of the 15th International Conference on Semantic Systems. (Poster and Demo Track at SEMANTiCS 2019)*, no. 2451, pp. 46–50, 2019. [Online]. Available: http://ceur-ws.org/Vol-2451/#paper-10

[2] J. Dörpinghaus, J. Darms, and M. Jacobs, “What was the question? a systematization of information retrieval and nlp problems.” in *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2018.

[3] J. Dörpinghaus, C. Düing, and V. Weil, “A minimum set-cover problem with several constraints,” in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, Sep. 2019, pp. 115–122.

[4] J. Dörpinghaus, A. Stefan, B. Schultz, and M. Jacobs, “Towards context in large scale biomedical knowledge graphs,” *arXiv preprint arXiv:2001.08392*, 2020.

[5] V. Gligorijević and N. Pržulj, “Methods for biological data integration: perspectives and challenges,” *Journal of the Royal Society Interface*, vol. 12, no. 112, p. 20150571, 2015.

[6] J. Dörpinghaus and A. Stefan, “Knowledge extraction and applications utilizing context data in knowledge graphs,” in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019, pp. 265–272.

[7] J. Dörpinghaus, A. Stefan, B. Schultz, and M. Jacobs. (2020) Towards context in large scale biomedical knowledge graphs. [Online]. Available: http://arxiv.org/abs/2001.08392

[8] C. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[9] A. Clark, C. Fox, and S. Lappin, *The handbook of computational linguistics and natural language processing*. John Wiley & Sons, 2013.

[10] H. Mirisaee, E. Gaussier, C. Lagnier, and A. Guerraz, “Terminology-based text embedding for computing document similarities on technical content,” *arXiv preprint arXiv:1906.01874*, 2019.

[11] N. Yarushkina, A. Filippov, and M. Grigoricheva, “Using of linguistic analysis of search query for improving the quality of information retrieval,” in *International Conference on Information Technologies*. Springer, 2019, pp. 215–226.

[12] C. S. Burns, R. M. Shapiro, T. Nix, J. T. Huber *et al.*, “Examining medline search query reproducibility and resulting variation in search results,” *iConference 2019 Proceedings*, 2019.

[13] J. Lin and W. J. Wilbur, “Pubmed related articles: a probabilistic topic-based model for content similarity,” *BMC bioinformatics*, vol. 8, no. 1, p. 423, 2007.

[14] D. Newman, S. Karimi, and L. Cavedon, “Using topic models to interpret medline’s medical subject headings,” in *Australasian Joint Conference on Artificial Intelligence*. Springer, 2009, pp. 270–279.

[15] D. Trieschnigg, P. Pezik, V. Lee, F. De Jong, W. Kraaij, and D. Rebholz-Schuhmann, “Mesh up: effective mesh text classification for improved document retrieval,” *Bioinformatics*, vol. 25, no. 11, pp. 1412–1418, 2009.

[16] Z. Lu, W. J. Wilbur, J. R. McEntyre, A. Iskhakov, and L. Szilagyi, “Finding query suggestions for pubmed,” in *AMIA Annual Symposium Proceedings*, vol. 2009. American Medical Informatics Association, 2009, p. 396.

[17] M. Hagen, M. Michel, and B. Stein, “What was the query? generating queries for document sets with applications in cluster labeling,” in *International Conference on Applications of Natural Language to Information Systems*. Springer, 2015, pp. 124–133.

[18] Y. Yan, X.-C. Yin, C. Yang, S. Li, and B.-W. Zhang, “Biomedical literature classification with a cnns-based hybrid learning network,” *PloS one*, vol. 13, no. 7, p. e0197933, 2018.

[19] A. Varghese, M. Cawley, and T. Hong, “Supervised clustering for automated document classification and prioritization: a case study using toxicological abstracts,” *Environment Systems and Decisions*, vol. 38, no. 3, pp. 398–414, 2018.

[20] D. Fensel, U. Şimşek, K. Angele, E. Huaman, E. Kärle, O. Panasiuk, I. Toma, J. Umbrich, and A. Wahler, *Introduction: What Is a Knowledge Graph?* Cham: Springer International Publishing, 2020, pp. 1–10. [Online]. Available: https://doi.org/10.1007/978-3-030-37439-6_1

[21] L. Ehrlinger and W. Wöß, “Towards a definition of knowledge graphs.” *SEMANTiCS (Posters, Demos, SuCCESS)*, vol. 48, 2016.

[22] M. Ley, “Dblp: some lessons learned,” *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1493–1500, 2009.

[23] A. A. Salatino, F. Osborne, T. Thanapalasingam, and E. Motta, “The cso classifier: Ontology-driven detection of research topics in scholarly articles,” in *International Conference on Theory and Practice of Digital Libraries*. Springer, 2019, pp. 296–311.

[24] B. Yates, B. Braschi, K. A. Gray, R. L. Seal, S. Tweedie, and E. A. Bruford, “Genenames.org: the HGNC and VGNC resources in 2017,” *Nucleic Acids Research*, vol. 45, no. D1, pp. D619–D625, 10 2016. [Online]. Available: https://doi.org/10.1093/nar/gkw1033

[25] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig *et al.*, “Gene ontology: tool for the unification of biology,” *Nature genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[26] G. O. Consortium, “The gene ontology resource: 20 years and still going strong,” *Nucleic acids research*, vol. 47, no. D1, pp. D330–D338, 2019.

[27] L. M. Schriml, E. Mitraka, J. Munro, B. Tauber, M. Schor, L. Nickle, V. Felix, L. Jeng, C. Bearer, R. Lichenstein *et al.*, “Human disease ontology 2018 update: classification, content and workflow expansion,” *Nucleic acids research*, vol. 47, no. D1, pp. D955–D962, 2019.

[28] R. Feldman and J. Sanger, *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2007.

[29] F. França and A. de Souza, *Intelligent Text Categorization and Clustering*, ser. Studies in Computational Intelligence. Springer Berlin Heidelberg, 2008.

[30] J. Dörpinghaus, S. Schaaf, and M. Jacobs, “Soft document clustering using a novel graph covering approach,” *BioData mining*, vol. 11, no. 1, p. 11, 2018.

[31] A. T. Kodamullil, E. Younesi, M. Naz, S. Bagewadi, and M. Hofmann-Apitius, “Computable cause-and-effect models of healthy and alzheimer’s disease states and their mechanistic differential analysis,” *Alzheimer’s & Dementia*, vol. 11, no. 11, pp. 1329–1339, 2015.

[32] D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda *et al.*, “Drugbank 5.0: a major update to the drugbank database for 2018,” *Nucleic acids research*, vol. 46, no. D1, pp. D1074–D1082, 2017.

[33] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne *et al.*, “The fair guiding principles for scientific data management and stewardship,” *Scientific data*, vol. 3, 2016.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 179–183 DOI: 10.15439/2020F20 ISSN 2300-5963 ACSIS, Vol. 21

Overview of the Transformer-based Models for NLP Tasks

Anthony Gillioz
University of Neuchâtel
Neuchâtel, Switzerland
Email: anthony.gillioz@unine.ch

Jacky Casas, Elena Mugellini, Omar Abou Khaled
University of Applied Sciences and Arts Western Switzerland
Fribourg, Switzerland
Email: {firstname.lastname}@hes-so.ch

***Abstract*—In 2017, Vaswani et al. proposed a new neural network architecture named Transformer. That modern architecture quickly revolutionized the natural language processing world. Models like GPT and BERT relying on this Transformer architecture have fully outperformed the previous state-of-the-art networks. It surpassed the earlier approaches by such a wide margin that all the recent cutting edge models seem to rely on these Transformer-based architectures.**

In this paper, we provide an overview and explanations of the latest models. We cover the auto-regressive models such as GPT, GPT-2 and XLNET, as well as the auto-encoder architecture such as BERT and a lot of post-BERT models like RoBERTa, ALBERT, ERNIE 1.0/2.0.

I. Introduction

THE understanding and the treatment of the ubiquitous textual data is a major research challenge. The tremendous amount of data produced by our society through social media and companies has exploded over the past years. All those information are most of the time stored under textual format. The human brain can extract the meaning out of text effortlessly, but this is not the case for a computer. It is then required to have performing and reliable techniques to treat this data.

The Natural Language Processing (NLP) domain aims to provide a set of techniques able to explain a wide variety of Natural Language tasks such as Automatic Translation [1], Text Summarization [2], Text Generation [3]. All those tasks have in common the meaning extraction process to be successful. Undoubtedly, if a technique were able to understand the underlying semantic of texts, this would help to resolve the majority of the modern NLP problems.

A big concern that restricts a general NLP resolver is the single-task training scheme. Gathering data and crafting a specific model to solve a precise problem works successfully. However, it forces us to come up with a solution not only each time a new issue arises but also to apply the model on another domain. A general multi-task solver may be preferable to avoid this time-consuming point.

Recurrent Neural Networks (RNN) were massively used to solve NLP problems. They have been popular for a few years in supervised NLP models for classification and regression. The success of RNNs is due to the Long Short Term Memory (LSTM) [4] and Gated Recurrent Unit (GRU) [5] architectures. Those two units prevent the vanishing gradient issue by providing a more direct way to the backpropagation of the gradient. It helps the computation when the sentences are long.

The high versatility of those networks can solve a wide variety of problems [6]. Unfortunately, those models are not perfect; the inherent recurrent structure made them hard to parallelize on multiple processes, and the treatment of very long clauses is also problematic due to the vanishing gradient.

To counter those two limiting constraints, [7] introduced a new model architecture: the Transformer. The proposed technique get rid of the recurrent architecture to rely on attention mechanism solely. Furthermore, it does not suffer from the gradient vanishing nor the hard parallelization issue. That facilitates and accelerates the training of broader networks.

This work aims to provide a survey and an explanation of the latest Transformer-based models.

II. Background

In this section, we introduce a general NLP background. It gives a broad insight into the unsupervised pre-training and the NLP state-of-the-art pre-Transformers.

A. *Unsupervised Pre-training*

The unsupervised pre-training is a particular case of semi-supervised learning. That is massively used to train the Transformer models. That principle works in two steps; the first one is the pre-training phase. It computes a general representation from raw data in an unsupervised fashion. Second, once it is computed, it can be adapted to a downstream task via fine-tuning techniques.

The principal challenge is to find an unsupervised objective function that generates a good representation. There is no consensus on which task provides the most efficient textual description. [8] propose a language modelling task, [9] introduce a masked language modeling objective, [10] use a multi-tasks language modeling.

B. *Context-free representation*

The recent significant increase in the performance of NLP models is due to the use of word embeddings. It consists of representing a word as a unique vector. The terms with the same meaning are located in a close area of each other. Word2Vec [11] and Glove [12] are the most frequently used word embedding methods. They treat a large corpus of text and produce a unique word representation in a high dimensional space.

Byte Pair Encoding (BPE) [13] is another word embedding technique using subwords units out of character-level and word-level representation. [14] changed the implementation of BPE to be based on bytes instead of Unicode characters. Thus, he could reduce the vocabulary size from 100K+ to approximately 50K tokens. That has the advantage not to introduce [UKN] (unknown) symbols. Besides that, it does not involve a heuristic preprocessing of the input vocabulary. It is used when the amount of corpus to treat is too large and a more efficient technique than Word2Vec or Glove is required.

C. Attention Layer

Primarily proposed by [5], the attention mechanism aims to catch the long-term dependencies of sentences. The relationships between entities in phrases are hard to spot. Furthermore, it is necessary to get a strong understanding of the underlying structure of sentences. Indeed, if we can have a method that can tell us how the units of a sentence are correlated in a phrase, the language understanding tasks would be more straightforward.

The attention mechanism computes a relation mask between the words of a sentence and uses this mask in an encoder-decoder architecture to detect which words are related within each other. Using this process, the NLP tasks such as automatic translation are more flexible because they can have access to the dependencies of the sentence. In a translation context, it is a genuine advantage. Another notable benefit of the attention mechanism is the straightforward human-visualization of the model's outcome.

III. Dataset

The dominant strategy in the creation of deep learning systems is to gather a corpus corresponding to a given problem. The next step is to label this data and build a network that is supposedly able to explain them. This method is not suitable if we want to create a more comprehensive system (i.e. a system that can solve multiple problems without a significant architecture change).

That is then essential to learn on heterogeneous data to create general NLP models. If we want systems that can resolve several tasks at the same time, it is necessary to train this model on a wide variety of subjects. Hopefully, in our ubiquitous data world, a large number of raw texts are available online (e.g. Wikipedia, Web blogs, Reddit).

Table I shows the most commonly used datasets with their size and the number of tokens they contain. The tokenization is done with SentencePiece [15]. In a few cases, for example, in [16], the authors only used a subset of those datasets (e.g. Stories [17] is a subset of CommonCrawl dataset).

TABLE I
Datasets commonly used with Transformer-based models. (†: Tokenization done with SentencePiece, ‡: Uncompressed data)

Dataset	Size	Number of tokens †
BookCorpus [18]		
plus English Wikipedia	13GB	3.87B
Giga5 [19]	16GB	4.75B
ClueWeb09 [20]	19GB	4.3B
OpenWebText [21]	38GB	-
Real-News [22]	120GB ‡	-

IV. Benchmarks

During an extended period, the deep learning models have been trained to resolve one problem at a time. Further, when those models were used in another domain, they struggle to generalize correctly. That is the idea that promotes the creation of GLUE, SQuAD V1.1/V2.0 and RACE to have benchmarks able to check the reliability of models on various tasks.

GLUE: The General Language Understanding Evaluation (GLUE) [23] is a collection of nine tasks created to test the generalization of modern NLP models. It reviews a wide range of NLP problems like Sentiment Analysis, Question Answering and inference tasks. Because of the rapid improvement of the state-of-the-art on GLUE, SuperGLUE [24] is a new proposed benchmark to check general language systems but with more complicated more laborious tasks.

SQuAD: Stanford Question Answering Dataset (SQuAD) V1.1 [25] is a benchmark designed to resolve Reading Comprehension (RC) challenges. There are more than 100,000+ questions in the data set. There is no proposed answer like in the other RD datasets. The task contains a document, and the model has to find the answer directly in the text passage. SQuAD v2.0 [26] is based on the same principle than the V1.1, but this time the answer is not necessarily in the questions.

RACE: Reading Comprehension From Examinations (RACE) [27] is a collection of English questions set aside to Chinese students from middle school up to high school. Each item is divided into two parts, a passage that the student must read and a set of 4 potential answers. Considering that the questions are intended to teenagers, it requires keen reasoning skills to answer correctly to most of the problems. The reasoning subjects present in RACE cover almost all human knowledge.

V. Transformers

The RNNs (LSTM, GRU) have a recurrent underlying structure and are, by definition recurrent. It is then hard to parallelize the learning process because of this fundamental property. To overcome this issue, [7] proposed a new architecture solely based on the attention layers; the Transformer. It has the advantage to catch the long-range dependencies of a sentence and to be parallelizable.

A. Transformer architecture

The Transformer is based on an encoder-decoder structure, where it takes a sequence $X = (x_1, ..., x_N)$ and produce a latent representation $Z = (z_1, ..., z_N)$. Due to the auto-regressive property of this model, the output sequence $Y_M = (y_1, ..., y_M)$ is produced one element at a time. i.e. the word Y_M used the latent representation Z and the previously created sequence $Y_{M-1} = (y_1, ..., y_{M-1})$ to be generated. The Encoder and the Decoder are using the same Multi-Head Attention layer. A single Attention layer maps a query Q and keys K to a weighted sum of the values V. For technical reason there is a scaling factor $\frac{1}{\sqrt{d_k}}$.

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d_k}})V$$

B. Auto-Regressive Models

The auto-regressive models take the previous outputs to produce the next outcome. It has the particularity to be a unidirectional network; it can only reach the left context of the evaluated token. However, despite this flaw, it can learn accurate sentence representations. It relies on the regular Language Modeling (LM) task as an unsupervised pre-training objective:

$$L(X) = \sum_i \log P(x_i | x_{i-k}, ..., x_{i-1}; \Theta)$$

This LM function maximizes the likelihood of the conditional probability P. Where X is the input sequence, k is the context window, and Θ are the parameters of the Neural Network.

Various models are using this property coupled with the Transformer architecture to produce accurate Language Model languages (i.e. it determines the statistical distribution of the learned texts). The first auto-regressive model using the Transformer architecture is GPT [8]. It has a pre-training Language Modeling phase where it learns on raw texts. In the second learning phase, it uses supervised fine-tuning to adjust the network to the downstream tasks.

GPT-2 [14] uses the same pre-training principles than GPT. Though, this time it tries to achieve the same results in a zero-shot fashion (i.e. without fine-tuning the network to the downstream tasks). To accomplish that goal, it must capture the full complexity of textual data. To do so, it needs a wider system with more parameters. The results of this model are competitive to some other supervised tasks on a few subjects (e.g. reading comprehension) but are far from being usable on other jobs such as summarization.

Another auto-regressive network is XLNet [28]. It aims to use the strength of the language modeling of the auto-regressive model and at the same time, use the bidirectionality of BERT [9]. To do so, it relies on transformer-XL [29], the state-of-the-art model for the auto-regressive network.

C. BERT

GPT and GPT-2 use a unidirectional language model; they can only reach the left context of the evaluated token. That property can harm the overall performance of those models in reasoning or question answering tasks. Because, in those topics, both sides of the sentence are crucial to getting an optimal sentence-level understanding.

To counter this unidirectional constraint, [9] introduced the Bidirectional Encoder Representations from Transformers (BERT). This model can fuse the left and the right context of a sentence, providing a bidirectional representation and allow a better context extractor for reasoning tasks. The architecture of BERT is based on the Multi-Head Attention layers encoder like proposed in [7]. Originally [9] proposed two versions of BERT, the base version with 110M of parameters and the large version with 340M parameters.

Like GPT and GPT-2, BERT has an unsupervised pre-training phase where it learns its language representation. Nevertheless, due to its inherent bidirectional architecture, it cannot be trained using the standard Language Model objective. Indeed, the bidirectionality of BERT allows each word to see itself, and therefore it can trivially predict the next token. To overcome this issue and pre-train their model, [9] use two unsupervised objective tasks: the Masked Language Model (MLM) and the Next Sentence Prediction (NSP).

Once the pre-training phase is over, it remains to fine-tune the model to the downstream tasks. Thanks to BERT's Transformer architecture, the downstream can be straightforwardly done because the same structure is used for the pre-training and the fine-tuning. It merely needs to change the final layer to match the requirements of the downstream task.

VI. Post-BERT

Due to the high performance of BERT on 11 NLP tasks, a lot of researchers inspired by BERT's architecture applied it and tweaked it to their needs [30], [31].

A. BERT improvement

Further, studies have been done to improve the pre-training phase of BERT. The post-BERT model RoBERTa [16] proposes three simple modifications of the training procedure. **(I)** Based on their empirical results, [16] shows that BERT is undertrained. To alleviate this problem, they propose to increase the length of the pre-training phase. By learning longer, the outcomes are more accurate. **(II)** As the results of [32] and [14] demonstrate, the accuracy of the end-task performance relies on the wide variety of trained data. Therefore, BERT must be trained on larger datasets. **(III)** In order to improve the optimization of the model, they propose to increase the batch size. There are two advantages to have a bigger batch size; First, the large batch size is easier to parallelize, and second, it increases the perplexity of the MLM objective.

B. Model reduction

Since the Transformer's revolution, state-of-the-art networks have become bigger and bigger. Accordingly, to have a better language representation and better end-task results, the models must grow to catch the high complexity of texts. This expansion of the network's size has a high computational cost. More powerful GPUs and TPUs are required to train those large models. If we take, for example, the Nvidia's GPT-8B [1] with 8 billion parameters, it became infeasible for small tech companies or small labs to train a network as huge as that.

[1] https://nv-adlr.github.io/MegatronLM

It is then necessary to find smaller systems that maintain the high performances of the bigger ones.

Working with smaller models has multiple advantages. If the model size is shrunk, it trains faster, and the inference time will also be reduced. If it is small enough, it can be run on smartphones or IoT devices in real-time.

One technique introduced to reduce the size of those big networks is the knowledge distillation. It is a compression method that consists of a small network (student) trained to reproduce the behaviour of a bigger version of itself (teacher). The teacher is primarily trained as a regular network, and after that, it is distilled to reduce its size. DistilBERT [33] is a distilled version of BERT that reduces the number of layers by a factor of 2. It retains 97% of BERT on the GLUE benchmark while being 40% smaller and 60% faster at the inference time.

Another way to reduce the size of BERT is by changing the architecture itself. AlBERT [34] proposes two ideas to decrease the number of parameters. The first approach factorizes the embedding of the parameters. It separates the large vocabulary embedding matrix into two smaller matrices. The size of the hidden layer is separated from the size of the vocabulary representation. The second method is a cross-layer parameter sharing. This technique prevents the parameters from growing with the depth of the network. With those two tricks, it allows reducing the size of the large BERT version by 18% without a loss of performance. Since this architecture is smaller, the training time is also faster.

C. Multitask Learning

BERT learns several tasks sequentially and increases the overall performance of the downstream end-tasks. The main issue with the continual pre-training method is that it must learn efficiently and quickly newly introduced sub-tasks, and it must remember what has been learned previously. The Multi-task Learning (MTL) principle is based on human consideration. If you learn how to do a first task, then a second related task is going to be more accessible to master. There are two main trends in MTL.

The first one uses an MTL scheme during the fine-tuning phase. MT-DNN [35] based on the backbone of BERT is using the same pre-training procedure, but during the fine-tuning step, it uses four multi-tasks. Training on all the GLUE tasks at the same time makes it gain an efficient generalization ability.

On the opposite [10] proposes an MTL process directly during the pre-training step; ERNIE 2.0 introduces a continual pre-training framework. More specifically, it uses a Sequential Multi-task Learning where it begins to learn a first task. When this first task is mastered, a new task is introduced in the continual learning process. The previously optimized parameters are used to initiate the model, the new task and the previous tasks are trained concurrently. There are three groups of pre-training tasks, and each of them aims to capture a different level of semantic:

Word-Aware Tasks: It captures the lexical information of the text: the Knowledge Masking Task (i.e. it masks phrases and entities), the Capitalization prediction (i.e. it predicts if a word has a capitalized first letter), and the Token-Document Relation Prediction Task (i.e. it predicts if a token of a sentence belongs to a document where the sentence initially appears).

Structure-Aware Tasks: It learns the relationship between sentences: sentence reordering task (i.e. split and shuffle a sentence and must find the correct order), sentence distance task (i.e. it must find if two sentences are adjacent, belong to the same document or if they are entirely unrelated).

Semantic-Aware Tasks: It learns a higher order of knowledge: discourse relation task (i.e. it predicts the semantic or rhetorical relation of sentences), IR relevance task (i.e. find the relevance of information retrieval in texts).

D. Specific language models

In order to tackle specific languages problems, different monolingual versions of BERT were trained in different languages. For example BERTje [36] is a Dutch version, AlBERTo [37] is an Italian version, and CamemBERT [38] and FlauBERT [39] are two different models for French. These models outperform vanilla BERT in different NLP tasks specific to these languages.

E. Cross-language model

XLM [40] aims to build a universal cross-language sentence embedding. The goal is to align sentence representations to improve the translation between languages. To do so, a Transformer architecture with two unsupervised tasks and one supervised is used. The effectiveness of cross-language pre-training in order to improve the multilingual machine translation is shown.

VII. Going Further

Despite the excellent performances of the Transformer architecture, new layers aiming to improve the performance and the complexity have been released.

The Transformer uses a gradient-based optimization procedure. Thus, it needs to save the activation value of all the neurons to be used during the back-propagation. Because of the massive size of the Transformer models, the GPU/TPU's memory is rapidly saturated. The Reformer [41] counter the memory problem of the Transformer by recomputing the input of each layer during the back-propagation instead of storing the information. The Reformer can also reduce the number of operations during the forward pass by computing a hash function that pairs similar inputs together. Like that, it does not compute all pairs of vectors to find the related ones. Therefore, it increases the size of the text it can treat at once.

Another way to improve the architecture of a network is by using an evolving algorithm as proposed by [42]. To create a new architecture designed automatically, they evolve a population of Transformers based on their accuracy. Using the Progressive Dynamic Hurdles (PDH), they could reduce the search space and the training time. With this technique and an extensive amount of computational power (around 200 TPUs), they could find a new architecture that outperforms the previous one.

VIII. Conclusion

The Transformer-based networks have pushed the reasoning-skills to human-level abilities. It can even excel the human capabilities on a few tasks of GLUE. Transformer-based networks have changed the face of NLP tasks. They can go far beyond the results obtained with RNNs, and they can do it faster. They have helped solve many problems at the same time by providing a direct and efficient way to combine several downstream tasks. Nevertheless, much work remains before having a system with a human-level comprehension of the underlying meaning of texts, that is also sufficiently small to run on devices with low computational power.

References

[1] F. J. Och and H. Ney, "The Alignment Template Approach to Statistical Machine Translation," *Computational Linguistics*, vol. 30, pp. 417–449, Dec. 2004.

[2] A. M. Rush, S. Chopra, and J. Weston, "A Neural Attention Model for Abstractive Sentence Summarization," *arXiv:1509.00685 [cs]*, Sept. 2015. arXiv: 1509.00685.

[3] L. Yu, W. Zhang, J. Wang, and Y. Yu, "SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient," *arXiv:1609.05473 [cs]*, Aug. 2017. arXiv: 1609.05473.

[4] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, pp. 1735–1780, Nov. 1997.

[5] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *arXiv:1406.1078 [cs, stat]*, Sept. 2014. arXiv: 1406.1078.

[6] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, pp. 2222–2232, Oct. 2017. arXiv: 1503.04069.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *arXiv:1706.03762 [cs]*, Dec. 2017. arXiv: 1706.03762.

[8] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving Language Understanding by Generative Pre-Training," p. 12, Nov. 2018.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv:1810.04805 [cs]*, May 2019. arXiv: 1810.04805.

[10] Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, and H. Wang, "ERNIE 2.0: A Continual Pre-training Framework for Language Understanding," *arXiv:1907.12412 [cs]*, Nov. 2019. arXiv: 1907.12412.

[11] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *arXiv:1301.3781 [cs]*, Sept. 2013. arXiv: 1301.3781.

[12] J. Pennington, R. Socher, and C. Manning, "Glove: Global Vectors for Word Representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (Doha, Qatar), pp. 1532–1543, Association for Computational Linguistics, 2014.

[13] R. Sennrich, B. Haddow, and A. Birch, "Neural Machine Translation of Rare Words with Subword Units," *arXiv:1508.07909 [cs]*, June 2016. arXiv: 1508.07909.

[14] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language Models are Unsupervised Multitask Learners," p. 24, Nov. 2019.

[15] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing," *arXiv:1808.06226 [cs]*, Aug. 2018. arXiv: 1808.06226.

[16] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv:1907.11692 [cs]*, July 2019. arXiv: 1907.11692 version: 1.

[17] T. H. Trinh and Q. V. Le, "A Simple Method for Commonsense Reasoning," *arXiv:1806.02847 [cs]*, Sept. 2019. arXiv: 1806.02847.

[18] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler, "Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books," *arXiv:1506.06724 [cs]*, June 2015. arXiv: 1506.06724.

[19] R. Parker, D. Graff, and J. Kong, "English gigaword," *Linguistic Data Consortium*, Jan. 2011.

[20] J. Callan, M. Hoy, C. Yoo, and L. Zhao, "The ClueWeb09 Dataset - Dataset Information and Sample Files," Jan. 2009.

[21] A. Gokaslan and V. Cohen, *OpenWebText Corpus*. Jan. 2019.

[22] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, and Y. Choi, "Defending Against Neural Fake News," *arXiv:1905.12616 [cs]*, Oct. 2019. arXiv: 1905.12616.

[23] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding," *arXiv:1804.07461 [cs]*, Feb. 2019. arXiv: 1804.07461.

[24] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems," *arXiv:1905.00537 [cs]*, July 2019. arXiv: 1905.00537.

[25] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ Questions for Machine Comprehension of Text," *arXiv:1606.05250 [cs]*, Oct. 2016. arXiv: 1606.05250.

[26] P. Rajpurkar, R. Jia, and P. Liang, "Know What You Don't Know: Unanswerable Questions for SQuAD," *arXiv:1806.03822 [cs]*, June 2018. arXiv: 1806.03822.

[27] G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy, "RACE: Large-scale ReAding Comprehension Dataset From Examinations," *arXiv:1704.04683 [cs]*, Dec. 2017. arXiv: 1704.04683.

[28] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized Autoregressive Pretraining for Language Understanding," *arXiv:1906.08237 [cs]*, June 2019. arXiv: 1906.08237.

[29] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context," *arXiv:1901.02860 [cs, stat]*, June 2019. arXiv: 1901.02860.

[30] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "VideoBERT: A Joint Model for Video and Language Representation Learning," *arXiv:1904.01766 [cs]*, Sept. 2019. arXiv: 1904.01766.

[31] A. Wang and K. Cho, "BERT has a Mouth, and It Must Speak: BERT as a Markov Random Field Language Model," *arXiv:1902.04094 [cs]*, Apr. 2019. arXiv: 1902.04094 version: 2.

[32] A. Baevski, S. Edunov, Y. Liu, L. Zettlemoyer, and M. Auli, "Cloze-driven Pretraining of Self-attention Networks," *arXiv:1903.07785 [cs]*, Mar. 2019. arXiv: 1903.07785.

[33] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv:1910.01108 [cs]*, Oct. 2019. arXiv: 1910.01108.

[34] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," *arXiv:1909.11942 [cs]*, Oct. 2019. arXiv: 1909.11942 version: 3.

[35] X. Liu, P. He, W. Chen, and J. Gao, "Multi-Task Deep Neural Networks for Natural Language Understanding," *arXiv:1901.11504 [cs]*, May 2019. arXiv: 1901.11504.

[36] W. de Vries, A. van Cranenburgh, A. Bisazza, T. Caselli, G. van Noord, and M. Nissim, "BERTje: A Dutch BERT Model," *arXiv:1912.09582 [cs]*, Dec. 2019. arXiv: 1912.09582.

[37] M. Polignano, P. Basile, and M. de Gemmis, "ALBERTO: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets," p. 6, 2019.

[38] L. Martin, B. Muller, P. J. O. Suárez, Y. Dupont, L. Romary, E. V. de la Clergerie, D. Seddah, and B. Sagot, "CamemBERT: a Tasty French Language Model," *arXiv:1911.03894 [cs]*, May 2020. arXiv: 1911.03894.

[39] H. Le, L. Vial, J. Frej, V. Segonne, M. Coavoux, B. Lecouteux, A. Allauzen, B. Crabbé, L. Besacier, and D. Schwab, "FlauBERT: Unsupervised Language Model Pre-training for French," *arXiv:1912.05372 [cs]*, Mar. 2020. arXiv: 1912.05372.

[40] G. Lample and A. Conneau, "Cross-lingual Language Model Pretraining," *arXiv:1901.07291 [cs]*, Jan. 2019. arXiv: 1901.07291.

[41] N. Kitaev, L. Kaiser, and A. Levskaya, "Reformer: The Efficient Transformer," *arXiv:2001.04451 [cs, stat]*, Jan. 2020. arXiv: 2001.04451.

[42] D. R. So, C. Liang, and Q. V. Le, "The Evolved Transformer," *arXiv:1901.11117 [cs, stat]*, May 2019. arXiv: 1901.11117.

DOI: 10.15439/2020F119
ISSN 2300-5963 ACSIS, Vol. 21

Czech parliament meeting recordings as ASR training data

Jan Oldřich Krůza
Institute of Formal and Applied Linguistics,
Faculty of Mathematics and Physics,
Charles University
Email: kruza@ufal.mff.cuni.cz

***Abstract*—I present a way to leverage the stenographed recordings of the Czech parliament meetings for purposes of training a speech-to-text system. The article presents a method for scraping the data, acquiring word-level alignment and selecting reliable parts of the imprecise transcript. Finally, I present an ASR system trained on these and other data.**

I. Introduction

TRAINING data for speech recognition is always a demanded commodity, especially if it is free. There are for sure already some free Czech corpora fit for speech recognition training:

- Vystadial[1] with its 77 hours of VoIP calls[2],
- The Prague Database of Spoken Czech[3] with its 122 hours of richly annotated spontaneous dialogues[4],
- The Czech Senior COMPANION Expressive Speech Corpus with its 5 hours of professionally spoken utterances by a single speaker[5],
- Otázky Václava Moravce: 35 hours of transcribed recordings of the Czech TV talk show[6],
- STAZKA, a set of speech recording from vehicles with its 35 hours of background noise and utterances[7],
- Spoken Corpus of Karel Makoň[8] with its 100 hours of manually transcribed spontaneous speech by a single speaker[9],
- and possibly others that I am not aware of.

The Czech parliament meeting recordings represent a publicly available dataset of high-quality audio recordings of contemporary Czech in consistent low-noise audio quality worth almost 4000 hours of downloadable material, about 2800 hours after subtraction of the overlaps. Extracting training data for speech recognition systems would provide a corpus at least one order greater in length than those so far publicly available.

Verily, I am not the first person to attempt using these recordings for speech recognition. The Department of Cybernetics of University of West Bohemia developed an automatic online subtitling system for the meetings in 2006[10] and as a result, an 88-hour subset annotated by high-quality automatic transcript has been released for speech recognition training purposes[11].

I attempt to use the official stenographic transcripts available for all the talks so that it can be a new entry in the above list, on par in quality and excelling in size.

II. Data Preparation

Since the source data is publicly available and in the public domain, I merely provide the scripts for downloading and building the corpus. The algorithms and parameters used are described in this section.

A. Scraping

Regrettably, the data are to my best knowledge only available in human-readable form. The transcript is not clearly distinguished in the markup and is interlaced with metainformation. My method of isolating the transcript is quite crude but it covers the vast majority of cases. The criterion is to extract the subtree of all nodes with HTML attribute `[align=justify]`, except HTML elements `<b>`, which contain speaker identification.

The known shortcomings of this method are that 1) it discards the speaker annotations, although it is valuable metainformation and 2) it skips some short passages, e.g. references to other meetings, as can be seen in the meeting from Feb. 12th 2020 10:10 - 10:20[1]. Both can be corrected by devising a smarter scraper and neither has any significance for speech recognition: speaker annotation fundamentally and neglecting the links for their infrequency.

B. Alignment

One of the obstacles in using the stenographic transcripts for training an ASR system is the very loose alignment available. The recordings are all 14 minutes long and have a 4-minute overlap. The corresponding transcript is thus aligned in 10-minute blocks with a roughly 2-minute padding on each side of the audio. Figure 1 schematically shows the alignment of the stenographic transcript to the audio and the overlap of the recordings.

Systems for aligning long audio segments to their transcripts already exist, like that of Moreno et al.[12] or Hazen[13]. They are both based on an already existing automatically acquired transcript. I use this technique as well, though simplified and adapted to the task.

I have used the dataset mentioned above[11] to train a GMM-based ASR system, using the stenographs as training

[1] https://www.psp.cz/eknih/2017ps/stenprot/040schuz/s040372.htm

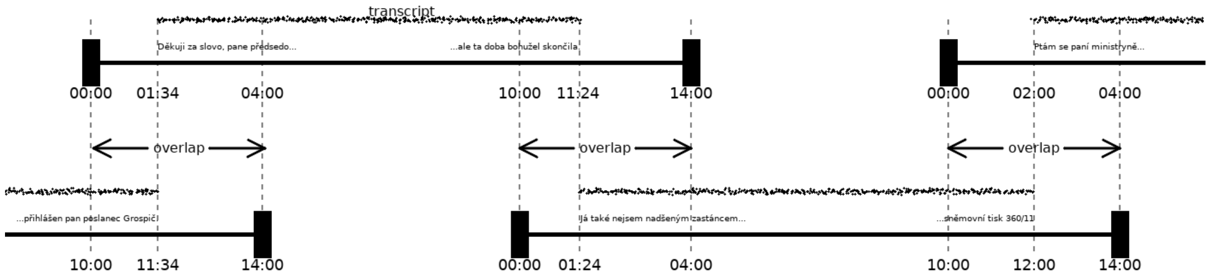

transcript
Děkuji za slovo, pane předsedo...
...ale ta doba bohužel skončila
Ptám se paní ministryně...
00:00
01:34
04:00
10:00
11:24
14:00
00:00
02:00
04:00
overlap
overlap
overlap
...přihlášen pan poslanec Grospič
Já také nejsem nadšeným zastáncem...
...sněmovní tisk 360/11
10:00
11:34
14:00
00:00
01:24
04:00
10:00
12:00
14:00

Fig. 1. Alignment and overlap of audio files and transcript. The examples are from Feb. 12th 2020 around 10 o'clock. The transcript corresponding to the recording in the upper left covers audio positions 01:34 - 11:24. The one in the lower right from 01:24 to 12:00.

data for a language model. Using these models, a word-level-aligned transcript of the whole set of recordings has been acquired.

The predicted transcript and the stenographic one have then been compared for Levenshtein distance, determining the edit operations needed to transform one into the other. For each predicted word, a reliability score is then computed as 1 - unreliability where unreliability is the number of edit operations taken on it divided by its length. Figure 2 shows how the stenographic transcript is aligned with the audio on word level.

Nota bene, a GMM-based system was chosen for the initial transcript instead of a DNN-based for three reasons: 1) Foremost, it is straightforward to obtain precise alignment from a GMM-based system. 2) The training doesn't require so much computational resources and data. 3) It isn't crucial to have maximum possible accuracy in this stage.

C. Audio Segmentation

To create a usable dataset for training a speech-to-text system, it is not necessary to perfectly align the whole transcript. On the contrary, it is desirable to align what is reliably precise and discard the rest.

The criteria for good training samples are:

1) 100% precise transcript,
2) roughly sentence-level length,
3) consistent length.

To ensure precise transcript, it is good to have the samples padded by some silence, since the alignment obtained from the initial ASR may be a bit imprecise. We thus want to split at pauses, the longer the better, up to a certain limit (about 1 second). The need to split at longer pauses goes against the need to split at consistent, none-too-great lengths.

So the problem is to select an optimal set of silences so that the longest ones are used and so that they split the recording into chunks of length in a given range. This looks like a problem for dynamic programming but a simpler approach is also possible: Start with a set of all silences predicted by the forced alignment. Iterate over the silences shortest-first and remove each if it doesn't break the constraints.

I have experimentally set the length boundaries to 12 - 30 seconds. The maximum length could be decreased at the cost of available pauses to choose from, which would lead to more frequent splits in the middle of a word.

D. Training Samples Selection

With the audio segmented and corresponding manual transcripts extracted, the last step remaining is selecting which segments to include in the traning data. Indeed, since the recordings have a 2-minute padding on each side for 10 middle minutes, we must discard at the very least 40% of the segments. I use the following criteria for including a segment in the data:

1) The first and last token have reliability at least 70%,
2) The mean reliability of all tokens is at least 70%,
3) The number of words is no less than five.

Minimum reliability of border tokens is considered to minimize the danger of shifted alignment boundaries. Mean reliability is considered because it is OK for some words to have very low reliability: there are enough errors in the prediction, that's why we use the manual transcript after all. But if too many tokens have too low reliability, then it is a sign of a suspicious segment. The number of words has a minimum because with only a few words, the probability of misalignment with good score is much greater than when there are enough words.

Why use mean reliability and not median? The way the reliability is computed considers the number of edit operations on one line in the automatic transcript. In the case where there are many insertions, the reliability of one line can go arbitrarily deep sub zero. So it can happen that there are several inserted words in a (mis)aligned chunk that only affect the reliability score of a single word. The mean taps these while the median doesn't.

E. Data Extraction Summary

All the constants and criteria are to be considered a baseline solution. They all could be tweaked much more rigorously and solved much more soundly. However, this simple solution readily yields a high-quality training dataset of 1058 hours. Of the total 539,057 segments, 142,530 (26%) have been accepted

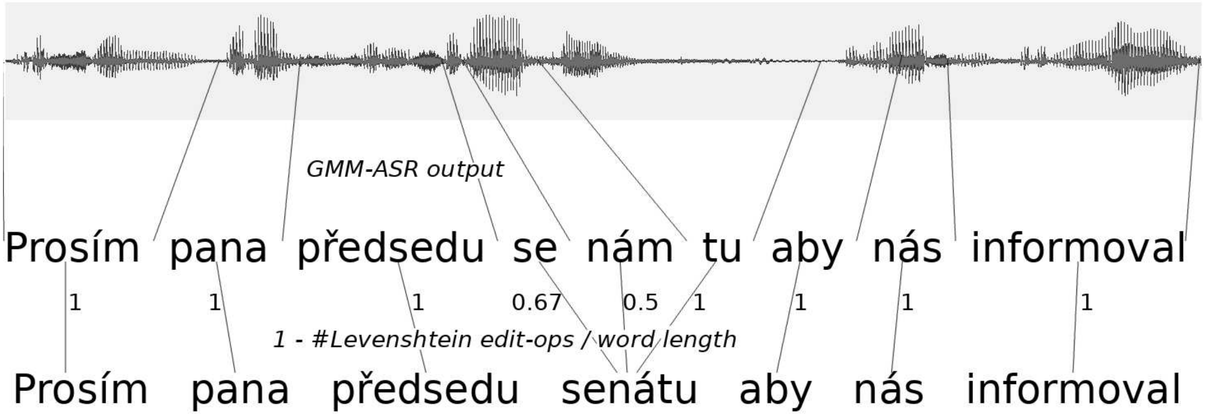

GMM-ASR output
Prosím pana předsedu se nám tu aby nás informoval
1 1 1 0.67 0.5 1 1 1 1
1 - #Levenshtein edit-ops / word length
Prosím pana předsedu senátu aby nás informoval

Fig. 2. Schema of aligning the audio to the stenographic transcript on word level.

to the training dataset. Of the total 396,527 discarded segments, 350,258 (88%) were discarded because of the criterion of unreliable start or end. It should be noted however, that the start / end reliability criterion is applied first, so it catches segments that would be discarded for other reasons also.

Reducing the minimum reliability of the boundary words from 70% to 50% increases the number of accepted chunks by 17%. It adds 5% segments of the total number to the dataset. But if we consider that 40% of the total number of segments must be discarded because of audio padding, the gain is acually 9%. It is an option to increase the training data volume at the cost of matching precision.

III. Numerals and Abbreviations

There are many numeral expressions in the transcripts. They amount to 489,880 out of 25,010,269 tokens in the complete stenographic transcript, which is almost two percent. In the training dataset, 24% of the samples contain one or more numerals.

Originally, I have included the digits into the alphabet for speech recognition, thus attempting to train the system to transcribe numeral expressions directly into digits. The speech recognition system described in the following section would however transcribe numeral expressions as empty strings.

There are four ways to deal with the problem:

1) ignore it,
2) remove digits from the training data,
3) manually expand digits to words,
4) automatically expand digits to words.

The first option needs no elaboration. The second one, removing samples with digits, is an easy and viable option but it is a waste of a quarter of the dataset and of the vast majority of samples with numerals in them. Manual expansion would surely be ideal but very costly. It remains to attempt the fourth variant of automated expansion.

For automated expansion of digits into words, we can use the available initial transcript and the algorithm for alignment with the stenographic transcript.

The expansion is done in two steps:

1) generation of verbal variants,
2) selection of the most likely variant.

I have used the Perl module `Lingua::CS::Num2Word` as a base for the expansion. I modified the module in the following way: 1) I added support for the order of billions, which is very common in the corpus. 2) A number is no longer expanded into a single phrase but instead into all possible phrases expressing the given number. 3) I added support for genitive and accusative cases, decimal numerals, ordinals, dates and times.

All tokens in the stenographs that include digits are expanded into their verbalization variants before further processing. Upon alignment, the variant with least edit distance from the initial transcript is selected.

Common abbreviations and symbols are expanded together with the digits. For example, the very common character *"§" (paragraph)* is expanded into the forms *paragraf, paragrafu, paragrafů, paragrafem, paragrafech* that represent common inflections of the word. Some common abbreviations that undergo inflection include *"čl." (article)*, *"odst." (also paragraph)* and *"tzv." (co-called).*

After incorporating the expansion into the pipeline, the similarity of the stenographic transcript and the initial one raised, which also raised the number of accepted segments from 26% to 35%. The amount of training data grew by 86 hours to 1144.

IV. ASR Based on the Dataset

I have trained a standard DeepSpeech[14] model on the 1058 hours with training : development : test ratio of 18 : 1 : 1; batch size 50; learning rate 0.0001; dropout rate 0.2. The training took 12 epochs to reach optimal dev fit and the final word error rate on testing data from the corpus itself is 8.40% before digit expansion and 7.89% afterwards.

The language model used was a pentagram model with pruned singleton trigrams, tetragrams and pentagrams. The bulk of scraped transcriptions, including those with no downloadable corresponding audio, was used as training data for the language model.

I have also tried training a speech recognition system with other datasets and the combination of them all. Of the datasets listed in section I, only Vystadial, Otázky Václava Moravce (ovm) and the corpus of Karel Makoň (makon) proved useful without much effort.

Apart from them, I used the publicly not available corpora of Charles University Corpus of Financial News (CUCFN, 65 hours)[15], the Balanced corpus of informal spoken Czech (Oral2013, 293 hours)[16] and the spoken Bible (100 hours) available with no license terms from poslouchamebibli.cz. Table I shows the speech recognition results for each corpus on test data from itself and on a common test set from all the corpora.

TABLE I
WORD ERROR RATE OF SPEECH RECOGNITION ON THE INDIVIDUAL CORPORA AND ON THEIR CONCATENATION.

source	WER on self	WER on all
bible	9.20%	94.7%
cucfn	31.6%	72.8%
makon	30.4%	77.3%
oral2013	78.4%	60.7%
ovm	21.6%	72.9%
parliament w/digits	8.74%	39.7%
parliament expanded	7.89%	**36.0%**
vystadial	51.0%	74.0%
all w/digits	28.4%	28.4%
all expanded	26.0%	26.0%

All speech recognition systems were trained with the same hyperparameters as described above.

V. Conclusion

I have presented a new corpus of spoken Czech suitable for training speech recognition systems based on data in the public domain. The corpus size exceeds by an order the size of other freely available such corpora. A speech recognition system with competitive performance was made to show the fitness of the dataset to the purpose.

Among the compared corpora, the Czech parliament corpus performs by far best even in speech recognition outside its domain.

Source code for scraping and building the corpus is in the public domain and available on GitHub.com/Sixtease/cz-parliament-speech-corpus.

Acknowledgments

This work has been using language resources developed, stored and distributed by the LINDAT/CLARIAH project of the Ministry of Education, Youth and Sports of the Czech Republic (project LM2018101).

This research was supported by SVV project number 260 575.

References

[1] M. Korvas, O. Plátek, O. Dušek, L. Žilka, and F. Jurčíček, “Free english and czech telephone speech corpus,” 2014.

[2] O. Plátek, O. Dušek, and F. Jurčíček, “Vystadial 2016 – czech data,” 2016, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. [Online]. Available: http://hdl.handle.net/11234/1-1740

[3] M. Mikulová, J. Mírovský, A. Nedoluzhko, P. Pajas, J. Štěpánek, and J. Hajič, “Pdtsc 2.0-spoken corpus with rich multi-layer structural annotation,” in *International Conference on Text, Speech, and Dialogue*. Springer, 2017, pp. 129–137.

[4] J. Hajič, P. Pajas, P. Ircing, J. Romportl, N. Peterek, M. Spousta, M. Mikulová, M. Grůber, and M. Legát, “Prague DaTabase of spoken czech 1.0,” 2017, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. [Online]. Available: http://hdl.handle.net/11234/1-2375

[5] M. Grůber, “Czech senior COMPANION expressive speech corpus,” 2014, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. [Online]. Available: http://hdl.handle.net/11858/00-097C-0000-0023-1D76-9

[6] L. Šmídl and A. Pražák, “OVM – otázky václava moravce,” 2013, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. [Online]. Available: http://hdl.handle.net/11858/00-097C-0000-000D-EC98-3

[7] L. Šmídl, P. Stanislav, and V. Radová, “STAZKA – speech recordings from vehicles,” 2015, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. [Online]. Available: http://hdl.handle.net/11234/1-1510

[8] O. Krůza and N. Peterek, “Making community and asr join forces in web environment,” in *International Conference on Text, Speech and Dialogue*. Springer, 2012, pp. 415–421.

[9] O. Krůza, “Spoken corpus of karel makoň,” 2012, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. [Online]. Available: http://hdl.handle.net/11372/LRT-1455

[10] A. Pražák, J. V. Psutka, J. Hoidekr, J. Kanis, L. Müller, and J. Psutka, “Automatic online subtitling of the czech parliament meetings,” in *International Conference on Text, Speech and Dialogue*. Springer, 2006, pp. 501–508.

[11] A. Pražák and L. Šmídl, “Czech parliament meetings,” 2012, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. [Online]. Available: http://hdl.handle.net/11858/00-097C-0000-0005-CF9C-4

[12] P. J. Moreno, C. Joerg, J.-M. V. Thong, and O. Glickman, “A recursive algorithm for the forced alignment of very long audio segments,” in *Fifth International Conference on Spoken Language Processing*, 1998.

[13] T. J. Hazen, “Automatic alignment and error correction of human generated transcripts for long speech recordings,” in *Ninth International Conference on Spoken Language Processing*, 2006.

[14] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, “Deep speech: Scaling up end-to-end speech recognition,” *arXiv preprint arXiv:1412.5567*, 2014.

[15] W. Byrne, J. Hajič, P. Ircing, F. Jelinek, S. Khudanpur, J. McDonough, N. Peterek, and J. Psutka, “Large vocabulary speech recognition for read and broadcast czech,” in *International Workshop on Text, Speech and Dialogue*. Springer, 1999, pp. 235–240.

[16] L. Benešová, M. Křen, and M. Waclawičová, “Korpus spontánní mluvené češtiny oral2013,” *Časopis pro moderní filologii (Journal for Modern Philology)*, vol. 1, no. 97, pp. 42–50, 2015.

DOI: 10.15439/2020F24
ISSN 2300-5963 ACSIS, Vol. 21

Named Entity Recognition and Named Entity on Esports Contents

Ziyu Liu*, Yifan Leng†, Meiqi Wang‡ and Congzhu Lin§
Michtom School of Computer Science, Brandeis University
Waltham, Massachusetts, USA
Email: *ziyuliu@brandeis.edu, †yifanleng@brandeis.edu, ‡meiqw@brandeis.edu, §linc@brandeis.edu

***Abstract*—We built a named entity recognition system on Esports News. We established an ontology for Esports-related entities, collected and annotated corpus from 80 articles on four different Esports titles. We also trained a CRF and a BERT-based entity recognizer, built a basic DOTA2 knowledge base, and an entity linker that links mentions of entities to articles in Liquipedia (the Esports Wikipedia), and a naive web app which serves as a demo of this entire proof-of-concept system. We achieved an over 61% overall entity-level F1-score on the test set for the NER task.**

I. Introduction and Related Works

NAMED entity recognition (NER) has been a popular topic within the NLP area. As defined in the MUC7[1] definition, the goal of the task is to find unique identifiers of entities (organizations, persons, locations), times and quantities, and to identify all of those expressions in each text in the test set, and to categorize them.

Entity linking[2](EL) is a new task on the computation linguistics field. The goal of it is, besides identifying mentions of identities within the given text, linking them to the most suitable entry within a reference knowledge base.

NER and EL both address the lexical ambiguity of language and play important roles towards the broader goal of the NLP research: the automatic understanding of natural languages. While the problem of NER/EL tasks on formal text, like news, there is almost no study about NER/EL on texts about new emerging topics, such as Esports news. We are aware that some existing NER works[3], [4], [5] covered the corpus in the sports domain. However, arguably, Esports news are generally more informal, shorter and having a broader types of entities(e.g. virtual characters, users' online ids, etc.). And obviously we would need a different ontology to address these differences.

Another fact motivates our work is that recent years have witnessed a booming Esports industry. It had an estimated market worth of 138 billion US dollars in 2018, according to market research firm Newzoo. Esports news websites such as JoinDota.com, dotesports.com, liquipedia.net, have created a significant amount of high-quality news content covering matches results, transferring, and commercial insights on a variety of popular Esports titles.

Reliable NER/EL system on Esports contents could serve as essential parts in larger real-world NLP systems like automatic Esports news taggers, Esports match result prediction systems, or players/teams popularity analyzers. Moreover, there is no doubt that those systems have great potential in both academic and economic values.

In this paper, we established an ontology for Esports-related entities, collected and annotated corpus from 80 articles on 4 different Esports titles, trained CRF[6] and BERT-based[7] entity recognizer, built a basic DOTA2 knowledge base, an entity linker that links mentions to articles in Liquipedia, and an end-to-end web app which serves as a demo of this entire proof-of-conecpt system.

The rest of the paper is organized as follows. Section II describes the process of collecting corpus. In section III we introduce the ontology we set for the system and explain its rationale; section IV discusses the models and feature sets we used for the NER task; section V shows how we built the DOTA2 knowledge base; section VI explains how does the entity linking system works; and section VIII illustrates how the web app is built. Section VII reports the setting and results of our experiments on NER task and shows perceptive results of the entity linker. And finally, we conclude our project and propose valuable future works on the topic in section IX.

II. Corpus Collection

We limited our scope to four popular games: DOTA2[1], League of Legends[2], CS:GO[3], and Overwatch[4].

Our first attempt was collecting Esports data from Twitter by searching game names. Twitter has sufficient text data, and it is easy to retrieve tweets with Twitter APIs. However, there were two significant issues that discouraged us from using Twitter as the primary data source: 1. Although Twitter has an abundance of data, Esports-related entities are relatively sparse in tweets. 2. Twitter poses rate limits on accessing tweets and other information, e.g., searching tweets is limited to 180 requests per window, where each window is 15 minutes in length[5]. Crawling a large amount of data would be inefficient.

We then decided to utilize Esports news websites (e.g., dotesports.com). These websites are frequently updated by professional editors and contain more condensed information about tournaments, player transfers, and more.

[1] http://blog.dota2.com/
[2] https://signup.na.leagueoflegends.com
[3] https://blog.counter-strike.net/
[4] https://playoverwatch.com
[5] https://developer.twitter.com/en/docs/basics/rate-limits

We handpicked 25 articles for each game, where 20 were used for training/development set, five were held out as the test set. Each article contains 300 - 800 words and has at least 5 Esports entities.

III. Ontology

A. The First Attempt

Our original ontology contained six tags: GAME (game), TOURN (tournament), ORG (organization), PLAYER (player), PERF (performance), and SPONS (sponsor), defined as below:

- GAME: The Esports title.
- TOURN: An Esports event or league.
- ORG: The team in which name players play for.
- PLAYER: Individuals who play and compete on the game as a career (in other words, “pro player”).
- PERF: Any comments on the player/team’s performance on a certain game, a series(set of games).
- SPONS: Third-party sponsor of the event/organization.

B. Refined Ontology

After annotated all articles, we ran our baseline CRF averaged perceptron model and reached over a 0.50 F1 score on all entities except PERF and SPONS. We had zeroes on PERF. PERF contained long text spans (e.g. “He [dominated the DOTA Summit 11 Minor] with iG”). Besides, PERF was relatively difficult to define: it can be any comments on players or teams on a certain game or a series. The annotator agreement was low and might have impacted the performance of PERF. SPONS was absent in training articles, and therefore our baseline model did not tag any SPONS entity in the test set.

We later dropped PERF and SPONS, and added another entity called “AVATAR”. AVATAR represents a player’s role in the game, and it is an essential part of the gameplay. In DOTA2, League of Legends, and Overwatch, players each control their own characters. Each character has different abilities and functions. CS:GO does not have explicitly defined characters, but items in the game can define the roles and functions. For example, a support is generally the person carrying the flashbangs, molotovs, grenades, etc.[6].

Our refined ontology contained five kinds of entities: GAME, TOURN, ORG, PLAYER, and AVATAR, listed below:

- GAME: The Esports title.
- TOURN: An Esports event or league.
- ORG: The team in which name players play for.
- PLAYER: Individuals who play and compete on the game as a career (in other words, “pro players”).
- AVATAR: The character that a player controls. In CS:GO, it is the weapon/items a player uses.

IV. NER Models

We tried two different NER models on this task.

As for the CRF model, we used the averaged perceptron in `CRFSuite` package. We did some ablation tests to determine the best feature set to use and at last the feature set we used are `Bias`, `Token`, `Uppercase`, `Titlecase`, `Digit`, `Punctuation`, and `WordShape`. `BrownCluster` and `WordVector` are discarded as they turned out to hinder the model’s performance. We believe that they should be useful if those representations are trained on Esports-related corpus.

For the BERT model, we used the open source software on https://github.com/kyzhouhzau/BERT-NER with some modification to make it work with our ontology. All of the parameters are remained as default.

On the web app backend, we choose to use the CRF model, as it requires much less computation resources.

V. Knowledge Base Building

Entity requires a well-structured knowledge base as target. Under common scenarios, the target knowledge base is usually built based on Wikipedia[7]. However, although there are surely some articles on Esports entities, Wikipedia is far from comprehensive. Instead we would use Liquipedia[8], one of the biggest Esports wiki sites as the source of our knowledge base.

Undoubtedly, Liquipedia is a comprehensive and reliable source of information, but by choosing it as our target, it also introduces several challenges:

- Poorly-documented-and-implemented APIs. The MediaWiki APIs that Liquipedia provided are not well-documented. And most importantly, many critical actions, like dumping or parsing are not supported or implemented. To address this, we have to write our own crawler to retrieve and parse the document tree in order to extract useful, structured information.
- Inconsistency across sub-sites. Liquipedia is formed of several subsites, e.g. https://liquipedia.net/dota2/, https://liquipedia.net/starcraft2 and https://liquipedia.net/overwatch/. These sub-sites, although looks similar, seem to have slightly different front-end coding. And, as these are different Esports games, these sub-sites are organized differently, inherently. Therefore, it is hard to write a crawler which can easily build a knowledge base that contains all information for all Esports titles. For this reason, we currently only built a knowledge base for DOTA2.
- Access frequency limitation. This is a common practice for most modern websites, that an IP will be banned for a certain period of time, if it is sending requests to the server too frequently. It turns our that this issue is relatively easy to tackle, by simply putting `sleep(2)` on each request.

[6] https://www.pinnacle.com/en/esports-hub/betting-articles/cs-go/a-guide-to-csgo-role/ml2jx57tyd6bxr7z

[7] https://en.wikipedia.org/

[8] https://liquipedia.net/

A. Crawler

To build the crawler, we used `beautifulsoup` as our HTML parser, and we collected all information on teams(organizations), players, tournaments and heroes, then organized and saved them into json files. We also considered putting them into SQL-based database to enable more query functions. However, considering we will only mostly doing key-value searching/ranking operations, and the total data size is only about 400kB, we decided to store them just as json file.

An example tournament entry would look like:

```
"tier": "Major",
"name": "China DOTA2 Professional League
Season 1",
"dates": "Oct 17, 2019 - Mar 1, 2020",
"prize_pool": 212690,
"teams": "10",
"host_location": "China",
"event_location": "Online"
```

VI. Entity Linking

The actual entity linking process is initiated after the entities in the given text are recognized. To determine which entry in the knowledge base should be returned, we query the knowledge base using the text as key, under the recognized entity type. If successful, an entry containing all related information will be returned and used in the next step (in our system, being rendered on the web page).

A. Query Handling

When a query string is passed to the knowledge base, the actual key is returned based on Algorithm1. Inside which, the candidate key set ξ is built when the system is initialized, by combining all `names` and `aliases` in the JSON files.

Algorithm 1 Get matching key

```
Require: $s$: query string, $\xi$ : candidate key set
  if $s \in \xi$ then
    return $s$
  end if
  for $k \in \xi$ do
    if $s$ is substring of $k$ then
      return $k$
    else
      get close match of $s$, $s' \in \xi$
    end if
  end for
```

VII. Evaluation

This section reports our experiment results with different NER models and web app demo screenshots.

For NER tasks, we use entity-level precision/recall/F1 as our metrics, which are calculated based on the whether the prediction for an entity matches perfectly with the true entity start/end labels.

A. CRF Averaged Perceptron

TABLE I
AP with all features

Type	Prec	Rec	F1
ALL	54.83%	52.16%	53.46%
AVATAR	59.57%	35.44%	44.44%
GAME	64.29%	100.00%	78.26%
ORG	69.73%	53.75%	60.71%
PLAYER	44.44%	53.01%	48.35%
TOURN	38.71%	61.54%	47.52%

TABLE II
Remove brown cluster and word vector

Type	Prec	Rec	F1
ALL	59.37%	52.91%	55.95%
AVATAR	47.76%	40.51%	43.84%
GAME	66.67%	88.89%	76.19%
ORG	79.55%	58.33%	67.31%
PLAYER	44.71%	45.78%	45.24%
TOURN	52%	66.67%	58.43%

TABLE III
Best feature set*

Type	Precision	Recall	F1
ALL	62.24%	55.35%	58.59%
AVATAR	57.81%	46.84%	51.75%
GAME	88.89%	88.89%	88.89%
ORG	80.00%	60.00%	68.57
PLAYER	46.71%	46.99%	46.85%
TOURN	52.83%	71.79%	60.87%

*Best feature set: Bias, Token, UpperCase, Titlecase, Digit, Punctuation, WordShape

B. BERT NER

TABLE IV
Best result

Type	Precision	Recall	F1
ALL	62.35%	69.05%	61.22%
AVATAR	50.00%	2.56%	4.88%
GAME	30.00%	37.50%	33.33%
ORG	64.73%	87.91%	74.56%
PLAYER	71.52%	81.38%	76.13%
TOURN	41.03%	55.17%	47.06%

We can see that although the BERT model outperforms CRF-AP in terms of overall F1-score and on ORG and PLAYER. However, it falls short on GAME, TOURN and especially, AVATAR. We believed the much lower recall/F1 score on AVATAR, compared to CRF model, is caused by the lack of model fine-tuning for the task.

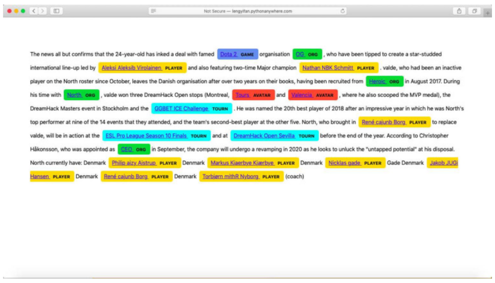
VIII. Web Application

To more conveniently assess the performance of the trained model, we built and deployed a web application [9] on

[9] http://lengyifan.pythonanywhere.com/

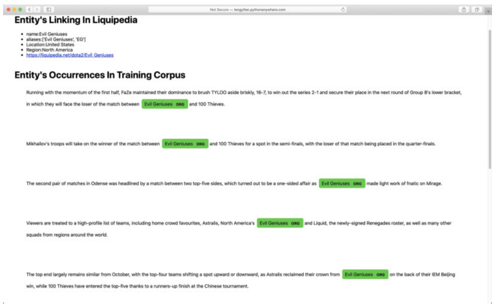
`PythonAnywhere`. The web application uses the trained CRF model to perform tagging on the text snippet. Five test documents were selected from the database that shows the named entities predicted by the model once clicked. The user can also input a text snippet or a URL in the search bar, and the text body will be extracted to perform named entity tagging on.

The tagged named entity is re-directed to a page that shows its related information in Liquipedia with a URL. Different mentions will have the same URL in this information section if they are the same entity (e.g "Invictus Gaming" and its alias "IG" or "iG"), suggesting successful entity . To inspect the mechanism the model uses to predict the label, we displayed the sentences in the training docs where the tagged entity is annotated. It facilitates understanding and selecting the features in the training and tagging stage.



The news all but confirms that the 24-year-old has inked a deal with famed Dota 2 GAME organisation OG ORG , who have been tipped to create a star-studded international line-up led by Aleksi Aleksib Virolainen PLAYER and also featuring two-time Major champion Nathan NBK Schmitt PLAYER . valde, who had been an inactive player on the North roster since October, leaves the Danish organisation after over two years on their books, having been recruited from Heroic ORG in August 2017. During his time with North ORG , valde won three DreamHack Open stops (Montreal, Tours AVATAR and Valencia AVATAR , where he also scooped the MVP medal), the DreamHack Masters event in Stockholm and the GGBET ICE Challenge TOURN . He was named the 20th best player of 2018 after an impressive year in which he was North's top performer at nine of the 14 events that they attended, and the team's second-best player at the other five. North, who brought in René cajunb Borg PLAYER to replace valde, will be in action at the ESL Pro League Season 10 Finals TOURN and at DreamHack Open Sevilla TOURN before the end of the year. According to Christopher Håkonsson, who was appointed as CEO ORG in September, the company will undergo a revamping in 2020 as he looks to unluck the "untapped potential" at his disposal. North currently have: Denmark Philip aizy Aistrup PLAYER Denmark Markus Kjaerbye Kjærbye PLAYER Denmark Nicklas gade PLAYER Gade Denmark Jakob JUGi Hansen PLAYER Denmark René cajunb Borg PLAYER Denmark Torbjørn mithR Nyborg PLAYER (coach)

Fig. 1. Named-Entity Tagging Page



Entity's Linking In Liquipedia

- name:Evil Geniuses
- aliases:['Evil Geniuses', 'EG']
- Location:United States
- Region:North America
- https://liquipedia.net/dota2/Evil_Geniuses

Entity's Occurrences In Training Corpus

Running with the momentum of the first half, FaZe maintained their dominance to brush TYLOO aside briskly, 16-7, to win out the series 2-1 and secure their place in the next round of Group B's lower bracket, in which they will face the loser of the match between Evil Geniuses ORG and 100 Thieves.

Mikhailov's troops will take on the winner of the match between Evil Geniuses ORG and 100 Thieves for a spot in the semi-finals, with the loser of that match being placed in the quarter-finals.

The second pair of matches in Odense was headlined by a match between two top-five sides, which turned out to be a one-sided affair as Evil Geniuses ORG made light work of fnatic on Mirage.

Viewers are treated to a high-profile list of teams, including home crowd favourites, Astralis, North America's Evil Geniuses ORG and Liquid, the newly-signed Renegades roster, as well as many other squads from regions around the world.

The top end largely remains similar from October, with the top-four teams shifting a spot upward or downward, as Astralis reclaimed their crown from Evil Geniuses ORG on the back of their IEM Beijing win, while 100 Thieves have entered the top-five thanks to a runners-up finish at the Chinese tournament.

Fig. 2. Named-Entity Detail Page

Figure 1 shows a piece of sample marked input text produced by the entity tagger with each named entity colored separately. Figure 2 shows the detail page of the recognized entity in Figure 1. With our linking algorithm, different aliases (such as "Evil Geniuses" and "EG") are mapped to their unique identity which is an entry in Liquipedia. We show this Liquipedia entry as its identity along with all of the entity's occurrences in the training corpus on this detail page.

IX. Conclusion and Future Works

In this project, we collected and annotated corpus from Esports news using the ontology set by ourselves, conducted NER experiment using CRF and BERT models on the test data set, and built an end-to-end Esports entity Liquipedia system which is capable of recognizing Esports players, teams and tournaments from texts. Although the system did not yield a satisfying result for AVATAR and GAME entities, we still managed to achieve 61.22% overall F1 score for the NER task using the BERT model, 58.59% overall F1 score using the CRF model.

This paper should serve as a starting point to combine NLP techniques and new emerging fields like Esports. As for future work, we consider these directions as the most meaningful ones:

- Better, finer-tuned NER model. Our CRF and BERT models are not fine-tuned; and as the the Recall and F1 score on AVATAR is abnormally low for BERT, we think there should be a large room of future improvement.
- Refining the knowledge base query algorithm to be ranking-based. Current system would be very likely to return false results when two teams has the same aliases. This could be undermined if we could let the system do ranking based on other information in the given text.
- Building a knowledge base contains more Esports titles so that our system can work on more games. This would be done easily if Liquipedia could help to provide an article dump api.
- Building a corpus on non-formal sources like social media.
- Supporting Esports contents in languages other than English. We found that there are some higher-quality Chinese and Russian corpus and would recommend navigating towards this direction.

References

[1] N. Chinchor and P. Robinson, "Muc-7 named entity task definition," in *Proceedings of the 7th Conference on Message Understanding*, vol. 29, 1997, pp. 1–21.

[2] D. Rao, P. McNamee, and M. Dredze, "Entity linking: Finding extracted entities in a knowledge base," in *Multi-source, multilingual information extraction and summarization*. Springer, 2013, pp. 93–115.

[3] T. Yao, W. Ding, and G. Erbach, "Chiners: a chinese named entity recognition system for the sports domain," in *Proceedings of the second SIGHAN workshop on Chinese language processing*, 2003, pp. 55–62.

[4] C.-K. Lee and M.-G. Jang, "Named entity recognition with structural svms and pegasos algorithm," *Korean Journal of Cognitive Science*, vol. 21, no. 4, pp. 655–667, 2010.

[5] X. Seti, A. Wumaier, T. Yibulayin, D. Paerhati, L. Wang, and A. Saimaiti, "Named-entity recognition in sports field based on a character-level graph convolutional network," *Information*, vol. 11, no. 1, p. 30, 2020.

[6] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

DOI: 10.15439/2020F63
ISSN 2300-5963 ACSIS, Vol. 21

Measuring the Polarity of Conversations between Chatbots and Humans: A Use Case in the Banking Sector

Guillaume Le Noé-Bienvenu
OrangeBank
67 Rue Robespierre
93100 Montreuil, France
Email: guillaume.lenoe.bienvenu@gmail.com

Damien Nouvel
Inalco ERTIM
2 Rue de Lille
75007 Paris, France
Email: damien.nouvel@inalco.fr

Djamel Mostefa
OrangeBank
67 Rue Robespierre
93100 Montreuil, France
Email: djamel.mostefa@orangebank.com

***Abstract*—This paper describes a study on opinion analysis applied to both human to chatbot conversations, but also to human to human conversations using data coming from the banking sector. A polarity classifier SVM model applied to conversations provides insights and visualisations of the satisfaction of users at a given time and its evolution. We conducted a study on the evolution of the opinion on the conversations started with the chatbot and then transferred to a human agent. This work illustrates how opinion analysis techniques can be applied to improve the user experience of the customers but also detect topics that generate frustrations with a chatbot or with human experts.**

I. Introduction

A. Scope and Aim

ORANGE Bank is a mobile bank launched in late 2017 and for which the main channel of communication with its customers is Djingo, a text chatbot. Available 24/7 by chat, Djingo, is the customers first point of contact. Since the launch of Orange Bank in November 2017, more than 2,5 million conversations have been initiated by our clients with Djingo (an average of 100,000 conversations per month), 50% of which are handled entirely by the virtual advisor (without any redirection to the Customer Relationship Centre). Since the chatbot is the first point of contact of Orange Bank clients, all chat conversations with a human agent started with Djingo. We are hence able to measure the evolution of the polarity within the same conversation between a customer and Djingo and then between the customer and the human operator.

In this context, opinion mining may be used to deliver in real time an understanding of the customer relationship for a given service. It could also be used to detect annoyance, irritation or angriness at an early stage of the conversation with Djingo in order to quickly redirect the user to a human expert. In this situation, opinion mining is also useful to detect topics and to provide insights about customer's satisfaction.

Our work focuses on the evolution of customer's opinion, both on conversations or messages within conversation. We implemented an opinion detector that has been evaluated, and plugged into the history of online conversations between customers and chatbot or human support desk. This work provides the customer support service visualisations of the evolution of customer's satisfaction depending on themes. The novelty of this paper relies on a comparison of how much the bot vs humans give satisfaction to the customers.

B. State of the Art

1) Opinion Analysis: Whereas a lot of work has been done in the opinion analysis field, most of it was directed towards product reviews, e.g. identifying the sentiment linked to the aspects of an object or its entities [1], but a few work was done towards written conversations, especially with a chatbot. Reference [2] used the estimation of user satisfaction to improve the learning process of the chatbot. Tools to work on polarity and emotions based on rules such as VADER [3] or SentiWordNet [4] are freely usable, but remain only for the English language. For French, resources are also available, such as the CANÉPHORE Corpus [5], but remain mostly specific to tweets. In this paper, we present a few cases (mostly graphs) in which opinion analysis could help giving valuable information with written talks. We focus on the polarity, defined by [6] as the property of a text being positive, negative or neutral.

2) Text Classification: Text classification is a well known task in NLP, and a reasonably efficient technique to perform it consists of using a TF-IDF [7] representation of the data combined with a support vector machine classifier (SVM) on it. This approach has since be giving satisfactory results. [8], [9], [10]. Deep learning methods can also be used for text classification. In particular, convolutional neural networks obtain very high scores for this task [11], but require more time and examples for training. Also, the winners of many challenges in NLP for the French language used TF-IDF+SVM models as the one used for DEFT 2015 [12] or during the Hackatal 2018[1]).

[1] https://hackatal.github.io/2018/

TABLE I
MOST COMMON ERROR TYPES

Error type	Example (errors in bold)
Diacritics	Je viens **deja** de vous expliquer mon **probleme**
Case	Comment **Recharger** son compte ?
Punctuation	Ma demande de résiliation **n est** toujours pas faite
Contraction	**Bjr** ou envoyer mon RIB ?
Typo	Ok je **vaiq** essayer. Merci
Spelling	je **n'arive** pas a faire **foncioné** ma carte bancaire

C. The Djingo Chatbot

Djingo is Orange Bank's conversational agent, available 24/7 for its 3,000 daily users. It is able to understand 390 intentions and has more than 1,000 answers adapted to the user's needs. Djingo is used both as a Frequently Asked Questions (FAQs) system (products marketed e.g. withdrawal fees, time to deliver a cheque book, etc.) and as an assistant to perform actions related to the customer account (ordering a cheque book, blocking the card, etc.). FAQ-oriented answers are usually the same for all customers, whereas requests performing an action trigger an operation that depends on the account.

For example, if a user wishes to order a checkbook, Djingo will check if the user is identified, if there is currently no checkbook order, if the user can order it, and so on. At each step, depending on the elements received through a programmatic interface (APIs), Djingo provides the user with an appropriate answer. During the conversation, themes and intentions are detected by the IBM Watson module. To date, there are about 60 themes: Orange-Bank, app-site-info, app-site-problem, insurance-info, termination insurance, etc. Conversations can include several themes. If the user asks a question that Djingo does not have the answer to, or detects that the user is unable to make himself understood, he suggests that the user should be redirected to an advisor.

II. OPINIONS FOR MESSAGES AND CONVERSATIONS

A. Chatbot Corpus

The corpus used in this article consists of 1,566,060 unique conversations from November 2017 to March 2019, containing 5,775,227 messages. Most of the messages sent by the users contain a small number of words (around 4.6 words per message) and are often describing the question using simple words. The size of the lexicon is quite important with around 144k entries due to important number of misspellings and typos.

Table I gives some examples of misspellings errors.

B. Annotation

As we focus on the polarity of messages, we built a gold-standard, by manually annotating 3,053 randomly picked user messages from the corpus. Each message is considered as positive, negative or neutral, following the 2015 DEFT annotation guide[2].

The annotation was made by two different annotators, giving a Cohen's kappa coefficient of 0.72. One particular issue during the annotation process was the case of greeting messages. We notice that in our data set, the user uses greetings for 83.96% of the conversations with a human agent, and only 18.99% of those with the chatbot. This gives us a clear indication of the behaviour of the user depending on the interlocutor. From an opinion perspective, we then assumed those greetings were positive and annotated them accordingly.

Table II gives examples of annotated data.

TABLE II
EXAMPLE OF ANNOTATED MESSAGES

Message (*translated*)	**Annotation**
Merci orange pour les 80 euros *Thank you orange for the 80 euros*	positive
Merci, bonne soirée *Thank you, have a nice evening*	positive
OK, super ! *Okay, great!*	positive
Je souhaiterai ouvrir un compte *I'd like you register an account*	neutral
Savoir si ma demande a été traitée *Find out if my request has been processed*	neutral
Quelles sont vos offres pour les étudiants ? *What are your offers for students?*	neutral
Cela ne repond pas a la question *This doesn't anwser the question*	negative
Non merci je suis très contrariée *No, thanks, I'm very upset.*	negative
Vous servez à rien *You're useless.*	negative

Unsurprisingly, our manual annotations dataset is not balanced: 5.01% of the messages are positive, 73.96% of them neutral and 21.03% negative. This was expected as users usually come with problems and questions regarding bank services and operations. Indeed, the company wants to maximise the satisfaction of users at the end of the interaction, while limiting the number of agents hired for this task.

C. Classification

This annotated data set was then divided over a train (4/5) and test parts (1/5). The train data was then pre-processed by computing a TF-IDF transformation. We tested several classical machine learning models using the sklearn API [13]. Results are reported in Table III.

TABLE III
PERFORMANCE OF OPINION CLASSIFIER (MACRO)

ML classifier	**Precision**	**Recall**	**F1**
SVM	0.90	**0.81**	**0.85**
MaxEnt	**0.92**	0.75	0.82
MNB	**0.92**	0.63	0.70
SGDClassifier	0.91	0.79	0.84

[2] https://deft.limsi.fr/2015/guideAnnotation.fr.php

TABLE IV
PROPORTION OF MESSAGES AND CONVERSATIONS IN THE CORPUS

	Number of messages	%	Number of conversations	%
Positive	460,744	3.98	190,057	7.30
Neutral	9,903,323	85.50	1,746,296	67.07
Negative	1,218,890	10.52	541,549	20.80
Mixed	_	_	125,641	4.83
Total	1,1582,957	100	2,603,543	100

As the SVM classifier provides the best F1 score, we ran a grid search on several parameters to optimize this model configuration. We obtained an average 0.85 F1 macro score (0.91 F1 micro). The neutral class obtains the best score (0.95 F1), while positive and negative classes have much lower F1 scores (0.82 and 0.76, respectively). Those results were obtained using the NLTK TweetTokenizer [14], without any other preprocessing (no lemmatization, case is kept as it is) and linear kernel for the SVM. Finally, the model was used to classify all messages of the corpus.

III. Conversation Polarity by Themes

A. Rules to Predict Conversations Polarity

To have a global view of user experience, one needs to compute an opinion score for each conversation. As the data was annotated by messages, simple rules were implemented to predict the polarity of an entire conversation based on the opinion of its messages. A conversation is then:

- **neutral** when all messages are such,
- **positive** when at least one of its messages is such and the remaining is neutral or positive,
- **negative** when at least one of its messages is such and the remaining is neutral or negative,
- **mixed** otherwise.

Using these simple rules, table IV shows the proportion of messages and conversations by polarity, automatically tagged without manual revision. The rules also allowed us incidentally to get strongly oriented conversations (e.g. a conversation where nearly all of its messages are negative would be very negative).

B. Histogram

The first representation we get from this labelling is the proportions of the conversation classes (positive, negative, neutral and mixed) depending of the detected themes. Figure 1 shows those proportions for December 2018. For instance, *the app_site* theme (related to the behaviour of the Bank's application) has more than 50% of its conversations being negative where the *cheque* theme remains globally neutral, this can be explained by the fact that this operation is rarely problematic. The representation of polarity gives us a rough idea of where to improve the user's experience. This type of plot can also be drawn for a different time scale (year, day, etc.).

C. Heatmap

In the previous section, we presented a way of drawing the proportions of the conversation classes for a particular time-lapse. However, this type of plot does not give us information about the evolution of this proportions across a time scale. E.g. on Figure 1, the *app_site* theme has a strong part of negative conversations but one can wonder if those proportions were similar through the year, whether it was due to a temporary failure, or if it was a general trend.

In order to represent a potential evolution of those proportions, we proposed a heatmap showing this evolution of the opinion by theme. To get a polarity score as a single numerical value for each case, a rule was implemented, consisting of adding the neutral and positive proportions of conversation and subtracting the negative. This was given by the following formula:

$$PS(th,t) = \frac{N(neu,th,t) + N(pos,th,t) - N(neg,th,t)}{NTotalConversations(th,t)}$$

Where

- *th*: the theme of the conversation
- *t*: a date
- *N(pol, th, t)*: the number of conversations of the theme *th* at time *t* having the polarity *pol* (negative, positive or neutral)
- *NTotalConversations(th, t)*: the total number of conversations of the theme *th* at time *t*

Figure 2 reports the heat map from November 2017 to March 2019. The bluer the case is the higher proportion of positive conversations the corresponding theme has. Conversely the red cases indicate negative conversations. One can then watch the changes in the proportions of cases throughout the months. For instance, we clearly see that the *Bonus* theme in March 2018 had its lowest polarity score, but its polarity score increased in the next few months. As in the previous section, this plot can also be drawn for a different time scale.

D. Graph of Polarity

We have then studied the way polarity of messages changes for a single conversation, especially when the user switches from a chatbot to an agent. In order to have a visual output, we converted the polarity (negative, neutral, positive) of each message of the conversation to an integer (0 for negative, 1 for neutral, 2 for positive). Table V shows an example of this conversion. This rule provides us with a list of integers that we can plot on a basic polarity graph, as reported in Figure 3 for a single conversation where each message has its detected polarity mapped on a graph.

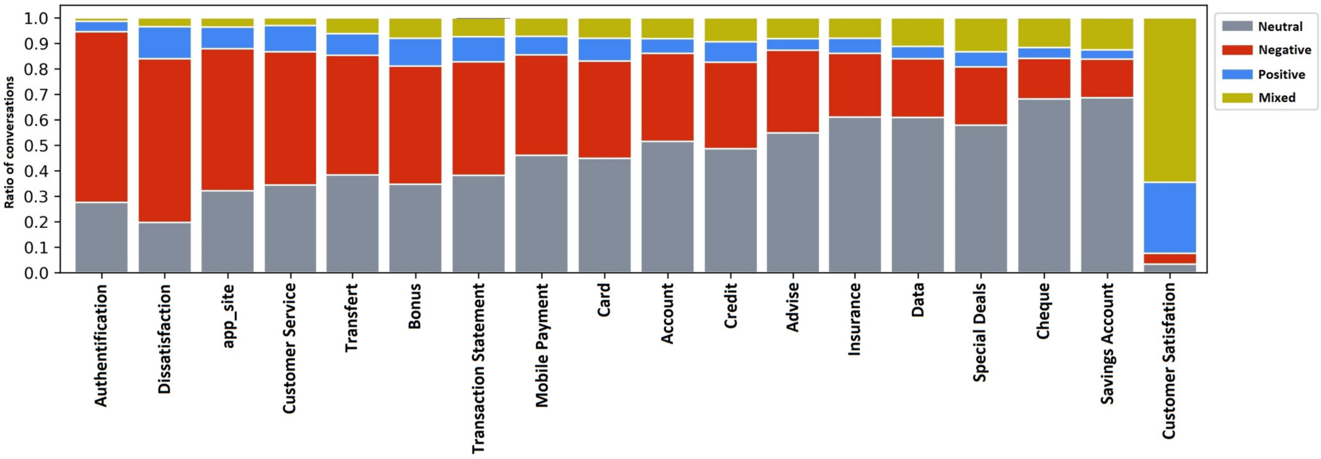


Fig. 1. Basic polarity histogram

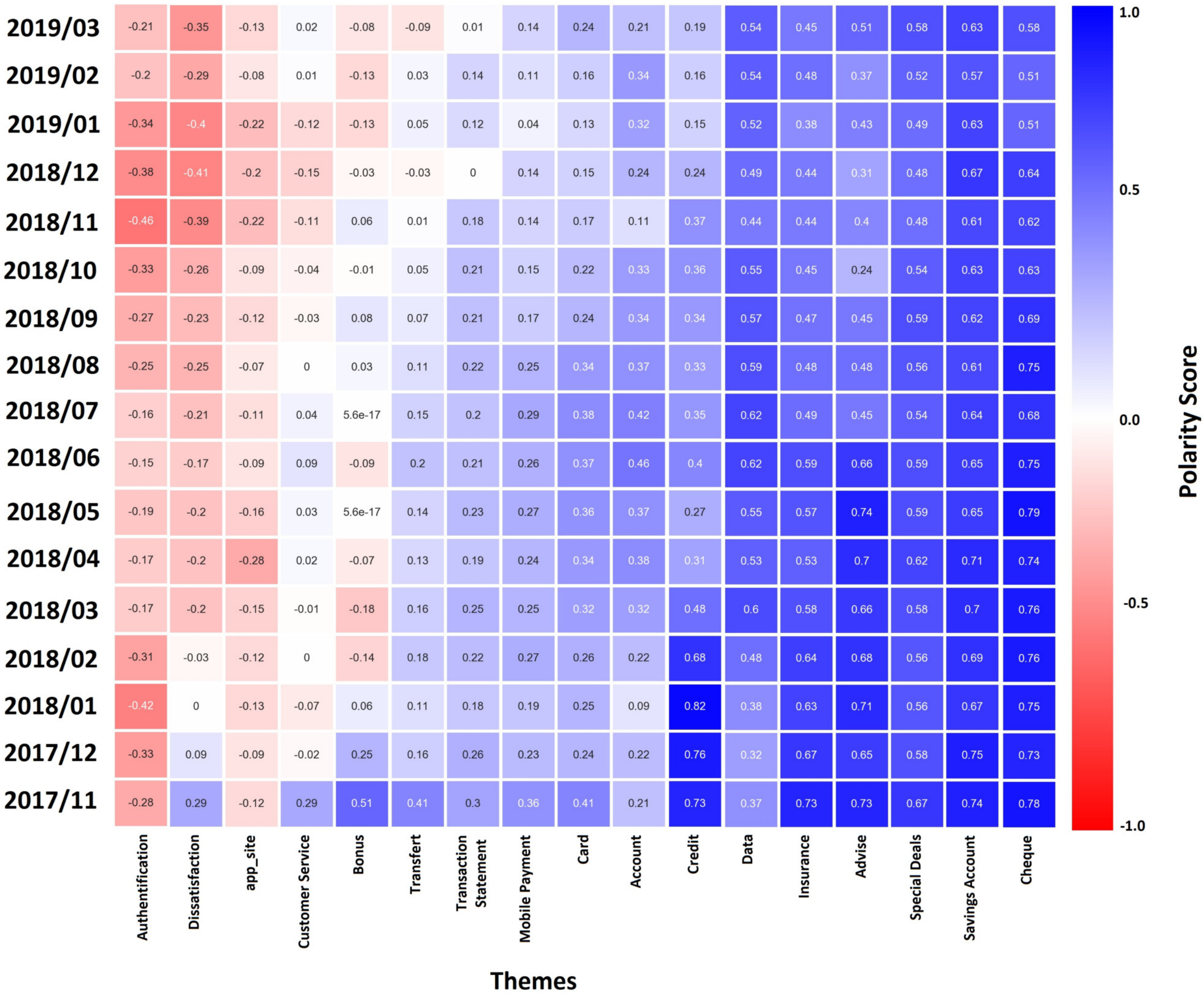


Fig. 2. Heatmap of polarity

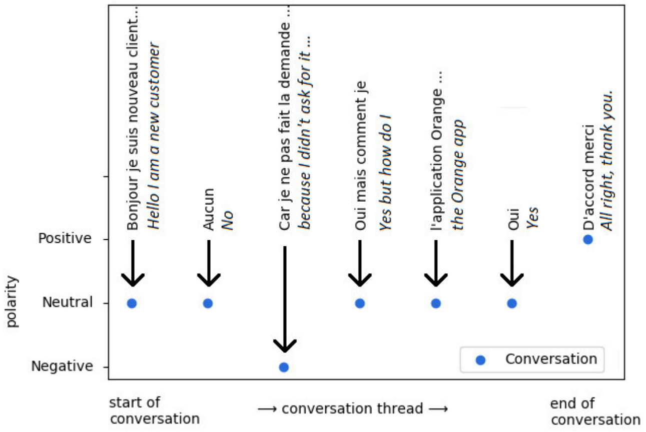


Fig. 3. Single Conversation Polarity Graph

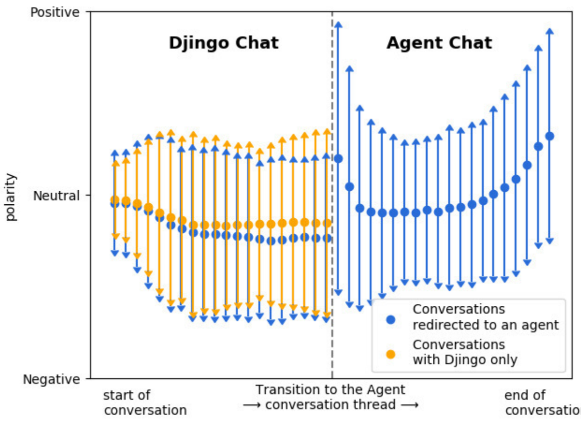


Fig. 4. Polarity graph

TABLE V
EXAMPLE OF A CONVERSATION CONVERTED TO A GRAPH

Message *(translated)*	**Predicted Polarity**	**Converted score**
Bonjour je suis nouveau client mais je n'ai pas fait la premier conexion *Hello I'm a new customer but I haven't made the first connection*	negative	0
Aucun *None*	neutral	1
Car je ne pas fait la demande de carte bancaire car je ne pas fait la demande de carte bancaire *Because I don't apply for a bank card because I don't apply for a bank card*	negative	0
Oui mais comment je fait pour me connecter *Yes, but how do I connect*	neutral	1
l'application Orange Bank *the Orange Bank App*	neutral	1
Oui *Yes*	neutral	1
D'accord merci *All right, thanks.*	positive	2

Since the conversations do not have the same length (different number of user messages), we converted the lists of integers representing the polarity of the user messages into lists of floats of fixed size. The size of the output lists can be modified as an optional parameter[3]. We then compute the average of each point of the list. Figure 4 show the result of the output with a padding of dimension 20.

On Figure 4, we first notice that for both types of users (redirected and non-redirected or full IA), the conversation starts with the same polarity (neutral) on average. After the first third of the conversation, people who are not redirected see the polarity of their conversation stagnate around a value slightly below neutral, while people who will be redirected see the polarity of their conversation decrease until an agent takes over. As soon as people are cared for by a counsellor, the polarity of the conversation takes a more positive trend (signs of politeness such as "hello" are labelled as positive and are more present in conversations with a human being). This is followed by a more neutral phase, which generally corresponds to the advisor's information gathering. At the end of the conversation, the trend is clearly becoming positive, we hypothetize that satisfying solutions are being proposed by the human agent.

IV. Discussion

There are however some limitations to the approaches discussed in this paper. First of all, the classification is based on annotation, and it is quite difficult to annotate into only three polarity classes. In the example: "*Mon épouse est décédé et je souhaite réaliser une demande de succession / My wife has died and I want to make a succession request*", the user of the conversational agent reports a past event as well as the willingness to take action. However, the part "*Mon épouse est décédé / My wife died*" would have been annotated as negative, while the part "*je souhaite réaliser une demande de succession / I wish to make an estate application*" would have been annotated neutral. A new class "positive-negative mix" could have been used as in DEFT 2018[4], but would have required a much more subtle and fine-grained annotation work.

Secondly, polarity is useful information, but does not indicate the subjectivity of the message. There is a significant difference between a user complaining about a particular Orange Bank service (e.g. *Ma carte bancaire ne marche pas / My credit card doesn't work*, negative polarity) and a dissatisfied user without a specific reason being stated (e.g. *Orange c'est vraiment de plus en plus pourri ! / Orange is really getting crap!*, negative polarity).

Thirdly, the transition from the polarity of the messages to the polarity of the conversation was carried out with a rule-

[3] Code available at https://github.com/GuillaumeLNB/perso/blob/master/rounding.py

[4] https://perso.limsi.fr/pap/DEFT2018/annotation_guidelines/index.html

based approach, creating a mixed class. This class does not take into account the intensity of certain messages. In the example in Table VI, the conversation has a mixed polarity (presence of positive and negative), but remains very negative by the presence of the last message. An annotation at the level of the conversation would probably have classified this conversation as negative, but would not have made a difference between this very negative and a less negative conversation.

TABLE VI
EXAMPLE OF A CONVERSATION CLASSIFIED AS MIXED WHERE IT SHOULD HAVE BEEN NEGATIVE

Message (*translated*)	**Predicted Polarity**
bonjour, *hello,*	positive
association loi 1901 peut elle ouvrir un compte chez vous? *Can a nonprofit association* *open an account with you?*	neutral
compte + association oi 1901 *account + aossociation 1901 [l]aw*	neutral
je ne parle pas aux robots, connards *I don't talk to robots, assholes.*	negative

Finally, the heatmap display gives us an overview of the evolution of the polarity, but does not detail the reasons of this variation. In addition, we did not find a correlation for all themes between their monthly polarity scores and their redirection rates. We are wondering if this metric is suitable for comparing these data.

V. CONCLUSION

In this paper, we have presented several applications of opinion analysis on chatbot conversations. By developing a model for polarity analysis (positive, negative, neutral) using standard machine learning algorithms, we were able to use the data to highlight trends. A real corpus of more than 1.5 million of conversations between Orange bank customers and Djingo was used for this study.

For privacy and confidential reasons, this corpus can not be shared at that time but it may be released in the future after anonymization of all personal data.

This analysis allowed to have a deeper insight of the evolution of the customer satisfaction or dissatisfaction, by topics on a time scale. Polarity mean show the sentiment are generaly more negative for conversation which will be handled by a human agent, what is nice since the human agent raises this polarity to positive values.

This tool makes it possible to obtain a quantification of the customers' opinions on the spot. We foresee that this kind of analysis, merging human and bot answers to a client, will be useful to improve customer relationship management. The key point is to detect when the bot has unsificient capacity to deliver an adequate answer and should pass the dialog to a human agent. It also provides our bank the opportunity to bring out very focused conversations (very positive or negative) from the corpus, to train customer relationship human agents for a better service, therefore this work raises opportunities to improve both the bot and the human agent.

REFERENCES

[1] B. Liu, "Sentiment Analysis and Opinion Mining", 2012, pp. 11-19.

[2] B. Hancock, A. Bordes, P.-E. Mazaré and J. Weston , "Learning from Dialogue after Deployment: Feed Yourself, Chatbot!," *CoRR abs/1901.05415,* Madison, WI, 2019,

[3] C. J. Hutto and E. Gilbert, "VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text.," 2014

[4] E. Andrea and S. Fabrizio, "SENTIWORDNET: A publicly available lexical resource for opinion mining," *in Proceedings of the 5th Conference on Language Resources and Evaluation (LREC),* 2006

[5] L. Joseph, E. Morin and S. Peña Saldarriaga, "CANÉPHORE : un corpus français pour la fouille d'opinion ciblée," *in Actes de la 22e conférence sur le Traitement Automatique des Langues Naturelles,* Caen, France, 2015, pp. 418–424.

[6] L. Zhang and S. Ferrari, "Intensité et polarité : un modèle opératoire articulant plusieurs travaux linguistiques," *in Langue française, (num 184)*, 2014, pp. 35–54.

[7] G. Salton and C. Buckley, "Term-weighting Approaches in Automatic Text Retrieval," *in Inf. Process. Manage. vol. 24 num. 5* , Tarrytown, NY, 1988, pp. 513–523.

[8] J. Thorsten, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," 1998

[9] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs Up: Sentiment Classification Using Machine Learning Techniques," *in Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing vol. 10,* Stroudsburg, PA, 2002, pp. 79–86

[10] J. Lilleberg, Y. Zhu, and Y. Zhang, "Support vector machines and Word2vec for text classification with semantic features," *in IEEE,* 2015/07, pp. 136-140

[11] Y. Kim, "Convolutional Neural Networks for Sentence Classification," *in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP,* Doha, Qatar, 2014, pp. 1746–1751

[12] T. Hamon, A. Fraisse, P. Paroubek, P. Zweigenbaum and C. Grouin, "Analyse des émotions, sentiments et opinions exprimés dans les tweets: présentation et résultats de l'édition 2015 du défi fouille de texte (DEFT)," *in Actes de la 22e conférence sur le Traitement Automatique des Langues Naturelles (TALN 2015),* 2015, pp. A20.

[13] L. Buitinck, et al., "API design for machine learning software: experiences from the scikit-learn project," *in ECML PKDD Workshop: Languages for Data Mining and Machine Learning,* Madison, WI, 2013, pp. 108–122.

[14] S. Bird, E. Klein and E. Loper, "Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit," 2009

Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 199–204
DOI: 10.15439/2020F78
ISSN 2300-5963 ACSIS, Vol. 21

Open IE-Triples Inference – Corpora Development and DNN Architectures

Martin Víta
NLP Centre
Faculty of Informatics, Masaryk University
Botanická 68a, 602 00 Brno
Czech Republic
Email: info@martinvita.eu

Petr Škoda
Department of Software Engineering
Faculty of Mathematics and Physics, Charles University
Malostranské nám. 2/25, 118 25 Prague
Czech Republic
Email: skoda@ksi.ms.mff.cuni.cz

Abstract—**Natural language inference (NLI) is a well established part of natural language understanding (NLU). This task is usually stated as a 3-way classification of sentence pairs with respect to entailment relation (entailment, neutral, contradiction). In this work, we focus on a derived task of relation inference: we propose a method of transforming a general NLI corpus to an annotated corpus for relation inference that utilizes existing NLI annotations. We subsequently introduce a novel relation inference corpus obtained from a well known SNLI corpus and provide its brief characterization. We investigate several DNN siamese architectures for this task and this particular corresponding corpus. We set several baselines including hypothesis only baseline. Our best architecture achieved 96.92% accuracy.**

I. INTRODUCTION

NATURAL language inference (NLI), formerly known as recognizing textual entailment (RTE), belongs to the most prominent tasks of natural language understanding (NLU). The importance of NLI arises not only from a number of downstream applications (including question answering, multi-document summarization, plagiarism detection etc.), but also from the suitability of NLI for learning universal sentence representations (INFERSENT in particular: sentence embeddings are obtained from siamese architecture-based DNNs for NLI task [1]). Moreover, there are also problems that can be transformed into NLI task, like relation classification [2].

The original RTE task was formulated as a binary (2-way) classification task for sentence pairs (premise-hypothesis) – whether a given hypothesis can be inferred from a given premise (TRUE/FALSE). This approach was used mainly in the early years of PASCAL/SemEval challenges [3]. The comprehensive overview of these challenges and, mainly of the corpora involved, is provided in [4]. Later, a 3-way classification became a more commonly used setting (with ENTAILMENT, NEUTRAL, CONTRADICTION labels) and the task started to be presented more often "under the NLI title".

Starting in 2015, we can observe a great development in the field of NLI that was allowed mainly by releasing the first large volume annotated corpus for NLI – Stanford NLI corpus [5], later followed by MultiNLI corpus [6] – as well as by exploiting deep learning approaches in NLP in general. An example of SNLI corpus items is provided in Table I.

TABLE I
EXAMPLE OF SNLI CORPUS ITEMS

Premise: *A soccer game with multiple males playing.*
Hypothesis: *Some men are playing a sport.*
Label: ENTAILMENT

Premise: *An older and younger man smiling.*
Hypothesis: *Two men are smiling and laughing at the cats playing on the floor.*
Label: NEUTRAL

Premise: *A man inspects the uniform of a figure in some East Asian country.*
Hypothesis: *The man is sleeping.*
Label: CONTRADICION

Recent state-of-the-art approaches based on ensemble and BERT-derived architectures provide very impressive results on SNLI/MultiNLI data. Up-to-date results are available on a dashboard on SNLI site[1].

In contrast to NLI, other related and/or derived tasks are strongly neglected, e.g. multiple premise entailment task [7], recognizing partial entailment [8], relation inference task [9], recognizing question entailment [10] etc.

In this work, we focus on the inference on the sets of open information extraction-triples (open IE-triples).[2] We state the task, introduce a method for transforming a general annotated NLI corpus into a corpus for open IE-triples inference and apply this method on SNLI corpus. We also provide a basic hypothesis-only baseline.

The motivation for these investigations arises from the issues related to canonicalizing open knowledge bases [11] and, generally, reasoning over assertions contained in open KB. To illustrate the issue, let us consider two open IE-triples `(Barack Obama; was born in; Honolulu)` and `(Former president Obama; has birthplace; Honolulu)`. If these two triples can be inferred one from the other, than it is redundant to store them

[1] https://nlp.stanford.edu/projects/snli/

[2] Open information extraction approaches typically extract textual triples of a form *(noun_phrase; relation_phrase; noun_phrase)* from an unstructured text, sets of these triples form open knowledge bases (open KBs), these triples usually correspond with subject-predicate-object triples.

both in the same open KB. The open IE-triples inference can provide us a straightforward and useful approach for identifying a redundant content in open KBs.

II. Preliminaries

This paper is located on the intersection of two domains: open information extraction (open IE) and NLI. In this section, we are going to recall some basic notions of (open) information extraction and relevant NLI concepts.

A. Elements of Open Information Extraction

Open information extraction systems extract textual n-tuples that represent basic propositions asserted by a sentence [12]. Generally, open IE systems produce textual tuples of different arity, however, in this work, we focus only on triples. Unlike to the task of ("traditional") information extraction, in open IE we do not require a fixed, predefined vocabulary of relations [13]. An open knowledge base (OKB) is a collection of assertions (textual tuples) obtained from an unstructured text(s) [14].

B. Classification of NLI Corpora

Annotated corpora for standard NLI task have basically the "premise-hypothesis-label" form, in some cases also enriched by additional auxiliary information – such as dependency parsing of premise and hypothesis sentences.

NLI corpus items can be produced by different processes. In [15], the authors present a classification of NLI corpora with respect to the process of creation:

- **Human elicited:** in this setting, given a premise, annotators are asked to create hypotheses for each label on their own. The result labels can be the checked by other annotators. Examples: SNLI and MultiNLI corpora.
- **Human judged:** in this case, hypotheses and premises are automatically paired but the labeling is done by a human. Example: SciTail corpus [16].
- **Automatically recast:** corpora in this class are automatically generated and labeled from an existing dataset (even for a different NLP task) with a minimal human intervention. Example: SICK corpus [17].

According to this classification, our annotated corpus for open-IE triples inference proposed in this work can be considered as an *automatically recast* (based on human elicited corpora SNLI and MultiNLI).

C. Annotation Artifacts in NLI Corpora

Annotation artifacts are certain patterns that appear in the data during annotation process. Especially human elicited corpora are prone to occurrence of annotation artifacts. This arises from the fact that crowd workers adopt several strategies when creating hypotheses for each label including lexical choice, sentence length etc. [18]. For example, the hypotheses with ENTAILMENT label often contain generic words such as *sport, animal, outdoors, instrument* etc., exact words are often replaced by approximations like *some, at least*.

To estimate the degree to which the artifacts appear in the NLI dataset, the authors in [18] trained a classifier that uses only hypotheses without seeing the premises. It has been shown that more than a half in a case of MultiNLI corpus and more than two thirds in a case of SNLI of the instances can be classified correctly using only the information contained in the hypotheses.

TABLE II
EXAMPLE OF RELATION INFERENCE CORPUS ITEMS

Premise: `(animal; has; fur)`
Hypothesis: `(the gazelle; will have; fur)`
Label: Y (ENTAILMENT)

Premise: `(Hypothesis animal; has; fur)`
Hypothesis: `(baboon; cleans; the fur)`
Label: N (NON-ENTAILMENT)

Since our proposed corpus is based on SNLI, we should take the annotation artifacts into account and focus also on this phenomenon.

III. Related Tasks and Definitions

A. Relation Inference in Context

Recognizing entailment between predicates (natural language relations) is a keystone task for several downstream applications. Let us consider a following example also used in [9]:

Aspirin *eliminates* headaches → Aspirin *treats* headaches

In this context, the relation *eliminate* entails/implies *treat.*

Several ***lexical*** *entailment in context* datasets implicitly capture this phenomenon. Nevertheless, these datasets are not primarily intended for relation inference and are focused only on a single word substitution, see [19] for instance.

In [20] Berant et al. focused on annotation between typed relations

[DRUG] *eliminates* [SYMPTOM] → [DRUG] *treats* [SYMPTOM],

redefining the notion of context. Levy et al. [21] annotated inference between instantiated relations sharing at least one argument

aspirin *eliminates* headaches → drugs *treat* headaches

Zeichner et al. [22] annotated inference between instantiated relations sharing both arguments:

aspirin *eliminates* headaches → aspirin *treats* headaches,
aspirin *eliminates* headaches ↛ aspirin *murders* headaches.

In all cases, the annotation was performed by experts.

In [9], the authors proposed a method for collecting data for relation inference in context corpus. They converted the inference task to a simple factoid question answering task and annotated more than 16000 high quality items. Examples for each entailment label from their corpus can be found in Table II.

Our work can be considered as a complement to this work. Our procedure of creating items is based on different assumption and approaches, however, it produces the output in the same form (subject-predicate-object triples). As an automatically recast corpus, the creation does not require any manual annotation work – in contrast to annotation in QA task.

B. Exploiting Open IE for Tasks Derived from NLI

Open information extraction systems has already been utilized within NLI environment.

In [23] and later in [24], the authors proposed a new task called *recognizing relational entailment (RRE)*. This task connects sentences and general textual n-tuples (expressing certain assertions): the premise is in a form of a sentence, the hypothesis has a form of a textual tuple. It is motivated by the issue of checking or approving facts in OKBs with respect to given unstructured texts.

The task is formulated as follows: Given a text T (premise) and a textual n-tuple t, the task of *recognizing relational entailment* is to classify the relation between the text T and the n-tuple expressed by t as:

- ENTAILMENT: if the meaning of t can be inferred from T,
- NEUTRAL: if the assertion expressed by t might be true in case of T is true and, moreover, the entailment does not hold,
- CONTRADICTION: if the meaning of t is contradictory to the meaning of T.

To illustrate this notion, we recall an example taken from [24]: given a sentence T = *Patrick flew from Boston to Los Angeles with Delta Airlines with one stopover.* and textual quadruples t_1 = `(Patrick; flew, from East Coast; with Delta Airlines)`, t_2 = `(Patrick, flew; from East Coast; to Los Angeles; via Atlanta)` and t_3 = `(Patrick; flew; from Los Angeles; to East Coast; via Chicago)`. Obviously, (T, t_1) should be labeled as ENTAILMENT, (T, t_2) as NEUTRAL and the last example (T, t_3) as CONTRADICTION.

Another usage of open IE systems within NLI environment, is transforming annotated NLI corpora – having single sentences as premises – into a multiple premise setting, i.e., building annotated multiple premises inference corpus [25].

IV. CORE WORK: OPEN IE-TRIPLES INFERENCE TASK

In this section, we state a new task of *open IE-triples inference*. After that we also propose potential applications and describe a transformation method for building annotated corpora for this task from a given annotated NLI corpus.

Although open IE systems can generally produce textual tuples of an arbitrary arity, we restricted tuples to size 3, i.e. triples. This make our method compatible with knowledge graphs [26].

A. Task Definition

By the *meaning of an open IE-triple* t we mean just the assertion expressed by t.

Given a pair of open IE-triples p (premise triple) and h (hypothesis triple), the task of *open IE-triples inference* is to classify the relation between the p and h as:

- ENTAILMENT: if the meaning of h can be inferred from the meaning of p,
- NEUTRAL: if the assertion expressed by h might be true in case of assertion expressed p is true and, moreover, the case of entailment does not hold,
- CONTRADICTION: if the meaning of h is contradictory to the meaning of p.

Particular examples will be provided in the next section. This approach is compatible to the previously mentioned RRE task definition.

Potential Applications: Classifiers trained on these corpora can be exploited generally for reasoning in OKBs, such as discovering redundant open IE-triples in OKBs and filtering new incoming triples that carry the same information which is already contained in the OKB. Another field for applications of this approach is knowledge graph completion [26].

Note that relation inference in context can be easily transformed into task of open IE-triples inference: rel_1 entails rel_2 in context given by arg_1 and arg_2 if and only if there is an entailment (open IE-triples inference) between a triple `(arg_1; rel_1; arg_2)` as a premise and a triple `(arg_1; rel_2; arg_2)` as a hypothesis.

B. Description of the Transforming Method for Corpora

Before we describe the method for transforming a general annotated NLI corpus into a corpus for open IE-triples inference corpus, we introduce a simple notation convention. A set of word types contained in a sentence or a textual triple s by a symbol $||s||$. Let $t(s)$ be a set of open IE-triples extracted by an open IE system from a sentence s.

Let us assume that we have an annotated corpus for NLI, i.e., set of items in the following form: a pair of sentences – premise P, hypothesis H – accompanied with a label L, where

$$L \in \{\text{ENTAILMENT, NEUTRAL, CONTRADICTION}\}.$$

The corpus for open IE-triples inference contains an item $(p; h; L)$ if and only if in the "input" NLI corpus there exists an item (P, H, L) such that at least of the following conditions hold:

1) $p \in t(P)$, $h \in t(H)$, $||p|| = ||P||$ and $||h|| = ||H||$,
2) $p \in t(P)$, $h \in t(H)$, $||p|| = ||P||$ and $L \in \{\text{ENTAILMENT}\}$.

The first condition covers a simple situation when, roughly said, the extracted open IE-triple contain just the same words as the source sentence (in both cases – premise and hypothesis). Obviously, in this situation, the sentence and the tuple express the same fact. Thus, the entailment label for a pair of original sentences as well as the label for a pair of tuples is identical. We implicitly assume that the open extraction tool works correctly: it extracts textual tuples with respect to dependencies in the original sentence – for instance,

TABLE III
EXAMPLE OF OUR OPEN IE-TRIPLES CORPUS ITEMS OBTAINED FROM SNLI

Premise: `(A little girl; holding; a baby)`
Hypothesis: `(A girl; is carrying; an infant)`
Label: ENTAILMENT

Premise: `(The trend; is; clear)`
Hypothesis: `(The trend; is; foggy)`
Label: CONTRADICTION

Premise: `(A small girl; is painting; a picture)`
Hypothesis: `(A small girl; is painting; her cat)`
Label: NEUTRAL

from a sentence *"Small company has a big revenue."* the tool does not extract `(big company; has; a small revenue)`. For incorrectly working extraction tools "equal words does not ensure equal meaning".

The second condition may not be so obvious: if the hypothesis H is entailed by the premise P in the original NLI corpus, then every assertion expressed by a triple h extracted from the hypothesis H is entailed by the premise P. Moreover, if for $p \in t(P)$, the equation $||p|| = ||P||$ holds, then P and p express the same fact, therefore we can straightforwardly put p, any tuple h extracted from H with ENTAILMENT label into the corpus being created.

In practice, we perform a loop over all instances in NLI corpus, extract all tuples p, h from premise-hypothesis pair P, H being processed and check whether the previous conditions hold.

V. DATA: OPEN IE-TRIPLES INFERENCE CORPUS OBTAINED FROM SNLI

To obtain an experimental open IE-triples inference corpus, we applied the method from the previous section on data from SNLI corpus, more preciously, on its *training* data split. Training dataset of SNLI contains 550152 labeled items.

For the open information extraction process, we used OPENIE 5.0 system[3]. As already mentioned, we restrict ourselves only on triples, other tuples are not taken into account. In order to avoid longer phrases as arguments, we also restrict the number of words in any part of an extracted triple *up to three.*

After removing duplicate items and items with label "-" (approx. 2% of instances of SNLI has label "-" indicating a lack of consensus among annotators), we obtained a final corpus containing 25234 items.

For illustration, we provide examples for each output label in Table III.

As a development set we have randomly chosen 2500 items, for test set 2500 items as well. The corpus is publicly available.[4]. There is no overlap between TRAIN and TEST set, i.e., TEST set contains only instances unseen during training.

The distribution of labels in this final corpus splits is summarized in the Table IV. We can straightforwardly see that majority-vote classifier would achieve accuracy of 0.408 at the test set.

TABLE IV
DISTRIBUTION OF LABELS IN EACH SPLIT

	ENTAILMENT	NEUTRAL	CONTRADICTION
TRAIN	8134	5170	6930
DEV	1000	653	847
TEST	1020	646	834

VI. MODELS

In this section we are going to present several architectures for our newly proposed corpus.

We investigated the following approaches for representing (open IE) triples:

1) SUM: sum of embeddings of all words contained in the triple (regardless if they are contained in the subject or predicate or object) – this approach serves as a baseline,
2) AVG: analogous to SUM, but average of all words' embeddings was taken,
3) SPO: concatentation of subject, predicate and object representations obtained by feed-forward architecture (described below),
4) USE: embeddings obtained by universal sentence encoders [27] applied on corresponding sentences (i.e., triple is considered as a one textual object).

In the first three approaches, GLOVE embeddings [28] are used. SPO representation is constructed as follows: let $subj_{in}$, $pred_{in}$ and obj_{in} are a simple sum of embeddings of words that form the subject, predicate and object, respectively.

$$subj = W_w * subj_{in} + b_w \tag{1}$$

$$pred = W_r * pred_{in} + b_p \tag{2}$$

$$obj = W_w * obj_{in} + b_w, \tag{3}$$

where W_w, W_r are weight matrices, b_w, b_r bias vectors to be learnt (these correspond to dense layers), and $subj$, $pred$ and obj are representations of subject, predicate and object, respectively. The final representation of a triple has a form: $[subj, pred, obj]$. Notice that subject and object representations are based on shared weights.

The shared dense layer used for subject and object has a dimension 64, the dense layer used for predicate encoding has a dimension 20. Similar encoding is used for representing of dependency triples in [29].

In this approach, premise triples and hypothesis triples are encoded by the same networks, i.e., we use siamese architectures. These representations are concatenated (concatenation of a premise and a hypothesis has a dimension 296, since both triples are encoded by vectors of dimension $64+20+64 = 148$). This concatenation is subsequently fed into dense layers, the final decision is obtained subsequently by a standard softmax layer – the layers are depicted on Figure 1.

[3] https://github.com/dair-iitd/OpenIE-standalone
[4] https://github.com/martinvita/openIEtriplesInference

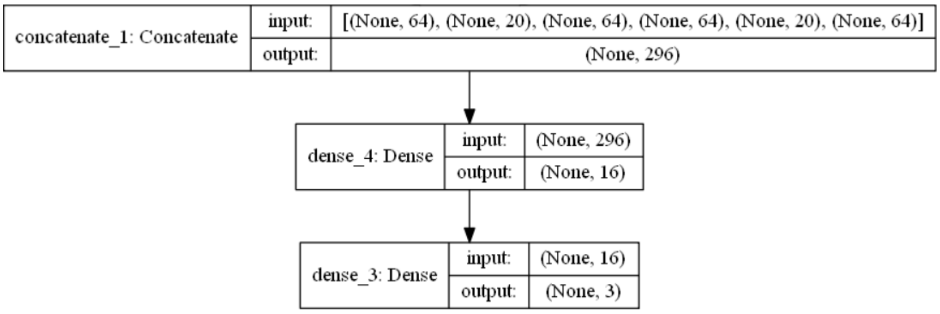


Fig. 1. Top layers of the network

TABLE V
RESULTS – ACCURACY OVER TEST SET

Model	Accuracy
SUM	0.4708
AVG	0.4772
SPO	**0.9692**
USE	0.6596
HYP-ONLY	**0.6896**

TABLE VI
NUMBER OF EPOCHS AND TEST ACCURACY

No. of epochs	Accuracy
256	0.7172
512	0.7952
1024	0.8160
2048	0.9064
4096	**0.9692**

Along with these approaches, we provide also a simple hypothesis only (HYP-ONLY) baseline based on SPO encoding of triples in order to model the presence of annotation artifacts. (The triple embeddings are fed again to dense layers and the result is obtained by a standard softmax layer).

Table V summarizes results of considered models.

For the SPO model, we provide a brief description of its training: we used RMSprop optimizer, batches of size 64. The effect of different number of training epochs on the accuracy on the TEST set is summarized in Table VI.

Accuracy w.r.t. the ENTAILMENT label is 0.9853, NEUTRAL label: 0.9412 and CONTRADICTION label: 0.9772.

There is a 95% likelihood that the confidence interval $[0.0222, 0.0354]$ covers the true classification error of the model on unseen data.

VII. CONCLUSION

In this work we presented a method for transforming a general annotated NLI corpus into a corpus for open IE-triples inference. The main advantage compared to existing resources focused on inference with relations/predicates is that this approach requires a very little manual effort. Then we applied this approach to a well known SNLI corpus, creating a new publicly available corpus (containing TRAIN/DEV/TEST split).

These approach can be generally used on any NLI corpus in the language where open IE tools are available.

In such a setting, the quality of obtained corpus depends naturally on the quality of the input NLI corpus, since the annotation artifacts may transfer from the source to target corpus. In case of our corpus obtained from SNLI, this fact was indicated by a relatively high accuracy of hypothesis only classifier: 0.6896 vs. 0.402 of majority vote classifier – this result of a hypothesis only classifier is roughly comparable with hypothesis only classifier over the entire SNLI corpus (0.69, see [15] that is based on INFERSENT architecture).

Finally, we have investigated several approaches, including universal sentence encoders. Our best (siamese) architecture based on dense layers for encoding each part of the triple achieves 96.92% accuracy.

A. Further Work

A natural part of further work is training models for this task that will be based most likely on contextual word embeddings like ELMo [30] as well as exploiting BERT-based [31] approaches over this corpus. These classifiers should also be evaluated on other relation inference corpora mentioned in the Preliminaries section.

Another part of investigations contains deriving related tasks such as shifting open IE triples-inference task to multilingual level and stating an analogy of multiple premises entailment task also for open IE-triples inference nature, i.e., dealing with premises in a form of sets of open IE-triples instead of single triples. There is also an issue to apply principles currently presented in [32].

Remark. This paper is partially based on the results achieved during the work on the PhD thesis of the first author, the thesis is currently under review.

REFERENCES

[1] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, “Supervised learning of universal sentence representations from natural language inference data,” *arXiv preprint arXiv:1705.02364*, 2017.

[2] A. Obamuyide and A. Vlachos, “Zero-shot relation classification as textual entailment,” in *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, 2018, pp. 72–78.

[3] I. Dagan, O. Glickman, and B. Magnini, “The pascal recognising textual entailment challenge,” in *Machine Learning Challenges Workshop*. Springer, 2005, pp. 177–190.

[4] L. Bentivogli, I. Dagan, and B. Magnini, "The recognizing textual entailment challenges: Datasets and methodologies," in *Handbook of Linguistic Annotation*. Springer, 2017, pp. 1119–1147.

[5] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," *arXiv preprint arXiv:1508.05326*, 2015.

[6] A. Williams, N. Nangia, and S. R. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," *arXiv preprint arXiv:1704.05426*, 2017.

[7] A. Lai, Y. Bisk, and J. Hockenmaier, "Natural language inference from multiple premises," *arXiv preprint arXiv:1710.02925*, 2017.

[8] O. Levy, T. Zesch, I. Dagan, and I. Gurevych, "Recognizing partial textual entailment," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2013, pp. 451–455.

[9] O. Levy and I. Dagan, "Annotating relation inference in context via question answering," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2016, pp. 249–255.

[10] A. B. Abacha and D. Demner-Fushman, "Recognizing question entailment for medical question answering," in *AMIA Annual Symposium Proceedings*, vol. 2016. American Medical Informatics Association, 2016, p. 310.

[11] L. Galárraga, G. Heitz, K. Murphy, and F. M. Suchanek, "Canonicalizing open knowledge bases," in *Proceedings of the 23rd acm international conference on conference on information and knowledge management*. ACM, 2014, pp. 1679–1688.

[12] G. Stanovsky, J. Michael, L. Zettlemoyer, and I. Dagan, "Supervised open information extraction," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 885–895.

[13] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni, "Open information extraction from the web." in *Ijcai*, vol. 7, 2007, pp. 2670–2676.

[14] T.-H. Wu, Z. Wu, B. Kao, and P. Yin, "Towards practical open knowledge base canonicalization," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, 2018, pp. 883–892.

[15] A. Poliak, J. Naradowsky, A. Haldar, R. Rudinger, and B. Van Durme, "Hypothesis only baselines in natural language inference," in *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, 2018, pp. 180–191.

[16] T. Khot, A. Sabharwal, and P. Clark, "SciTail: A textual entailment dataset from science question answering," in *AAAI*, 2018.

[17] M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. Bernardi, and R. Zamparelli, "A SICK cure for the evaluation of compositional distributional semantic models," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014*, 2014, pp. 216–223.

[18] S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. Bowman, and N. A. Smith, "Annotation artifacts in natural language inference data," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018, pp. 107–112.

[19] C. Biemann, "Creating a system for lexical substitutions from scratch using crowdsourcing," *Language Resources and Evaluation*, vol. 47, no. 1, pp. 97–122, 2013.

[20] J. Berant, I. Dagan, and J. Goldberger, "Global learning of typed entailment rules," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 610–619.

[21] O. Levy, I. Dagan, and J. Goldberger, "Focused entailment graphs for open ie propositions," in *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, 2014, pp. 87–97.

[22] N. Zeichner, J. Berant, and I. Dagan, "Crowdsourcing inference-rule evaluation," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*. Association for Computational Linguistics, 2012, pp. 156–160.

[23] M. Víta, "From building corpora for recognizing faceted entailment to recognizing relational entailment," in *Position Papers of the 2018 Federated Conference on Computer Science and Information Systems*, 2018, p. 33.

[24] M. Víta and J. Klímek, "First steps in recognizing relational entailment – experimental corpus and baselines," in *Human Language Technologies as a Challenge for Computer Science and Linguistics - 2019*, P. P. Zygmunt Vetulani, Ed. Wydawnictwo Nauka i Innowacje, 2019, pp. 143–147.

[25] ——, "Exploiting open ie for deriving multiple premises entailment corpus," in *Proceedings of Recent Advances in Natural Language Processing*, 2019, pp. 1257–1264.

[26] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[27] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar *et al.*, "Universal sentence encoder," *arXiv preprint arXiv:1803.11175*, 2018.

[28] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[29] Q. Du, C. Zong, and K.-Y. Su, "Adopting the word-pair-dependency-triplets with individual comparison for natural language inference," in *Proceedings of the 27th International Conference on Computational Linguistics*, 2018, pp. 414–425.

[30] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proceedings of NAACL-HLT*, 2018, pp. 2227–2237.

[31] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[32] V. Žitkus, R. Butkienė, R. Butleris, R. Maskeliūnas, R. Damaševičius, and M. Woźniak, "Minimalistic approach to coreference resolution in lithuanian medical records," *Computational and mathematical methods in medicine*, vol. 2019, 2019.

13th International Workshop on Computational Optimization

MANY real world problems arising in engineering, economics, medicine and other domains can be formulated as optimization tasks. These problems are frequently characterized by non-convex, non-differentiable, discontinuous, noisy or dynamic objective functions and constraints which ask for adequate computational methods.

The aim of this workshop is to stimulate the communication between researchers working on different fields of optimization and practitioners who need reliable and efficient computational optimization methods.

Topics

The list of topics includes, but is not limited to:

- combinatorial and continuous global optimization
- unconstrained and constrained optimization
- multiobjective and robust optimization
- optimization in dynamic and/or noisy environments
- optimization on graphs
- large-scale optimization, in parallel and distributed computational environments
- meta-heuristics for optimization, nature-inspired approaches and any other derivative-free methods
- exact/heuristic hybrid methods, involving natural computing techniques and other global and local optimization methods
- numerical and heuristic methods for modeling

The applications of interest are included in the list below, but are not limited to:

- classical operational research problems (knapsack, traveling salesman, etc)
- computational biology and distance geometry
- data mining and knowledge discovery
- human motion simulations; crowd simulations
- industrial applications
- optimization in statistics, econometrics, finance, physics, chemistry, biology, medicine, and engineering
- environment modeling and optimization

Best Paper Award

The best WCO'20 paper will be awarded during the social dinner of FedCSIS 2020.

The best paper will be selected by WCO'20 co-Chairs by taking into consideration the scores suggested by the reviewers, as well as the quality of the given oral presentation.

Technical Session Chairs

- **Fidanova, Stefka,** Bulgarian Academy of Sciences, Bulgaria
- **Mucherino, Antonio,** INRIA, France
- **Zaharie, Daniela,** West University of Timisoara, Romania

Program Committee

- **Abud, Germano,** Universidade Federal de Uberlândia, Brazil
- **Bonates, Tibérius,** Universidade Federal do Ceará, Brazil
- **Breaban, Mihaela**
- **Gruber, Aritanan**
- **Hadj Salem, khadija,** University of Tours - LIFAT Laboratory, France
- **Hosobe, Hiroshi,** Hosei University, Japan
- **Lavor, Carlile,** IMECC-UNICAMP, Brazil
- **Micota, Flavia,** West University of Timisora, Romania
- **Muscalagiu, Ionel,** Politehnica University Timisoara, Romania
- **Stoean, Catalin,** University of Craiova, Romania
- **Wang, Yifei**
- **Zilinskas, Antanas,** Vilnius University, Lithuania

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 207–214 DOI: 10.15439/2020F81 ISSN 2300-5963 ACSIS, Vol. 21

An Effective Integrated Metaheuristic Algorithm For Solving Engineering Problems

Adis Alihodzic
University of Sarajevo, BiH
Department of Mathematics
ul. Zmaja od Bosne, 33-35, Sarajevo
Email: adis.alihodzic@pmf.unsa.ba

Sead Delalic
University of Sarajevo, BiH
Department of Mathematics
ul. Zmaja od Bosne, 33-35, Sarajevo
Email: delalic.sead@pmf.unsa.ba

Dzenan Gusic
University of Sarajevo, BiH
Department of Mathematics
ul. Zmaja od Bosne, 33-35, Sarajevo
Email: dzenang@pmf.unsa.ba

***Abstract*—To tackle a specific class of engineering problems, in this paper, we propose an effectively integrated bat algorithm with simulated annealing for solving constrained optimization problems. Our proposed method (I-BASA) involves simulated annealing, Gaussian distribution, and a new mutation operator into the simple Bat algorithm to accelerate the search performance as well as to additionally improve the diversification of the whole space. The proposed method performs balancing between the grave exploitation of the Bat algorithm and global exploration of the Simulated annealing. The standard engineering benchmark problems from the literature were considered in the competition between our integrated method and the latest swarm intelligence algorithms in the area of design optimization. The simulations results show that I-BASA produces high-quality solutions as well as a low number of function evaluations.**

I. Introduction

IN THE last fifteen years, it was shown that most design nonlinear constrained optimization problems are an essential class of issues in real-world applications, and almost all are characterized as NP-hard problems. For such design optimization problems, the finding of the best solution may require centuries, even with a supercomputer. These highly nonlinear and multimodal optimization problems are based on the optimization of objective functions with complex constraints which usually involve thousands of or even millions of elements, and they were written in the form of simple bounds or more often as nonlinear inequalities. Nonlinearly constrained optimization problems contain continuous and discrete design variables, nonlinear objective functions, and constraints, some of which may be active at the global optima. Due to the complex nature of an objective function, as well as the constraints that need to be met, it is challenging how to effectively and robustly explore overall search space. Therefore, practically solving engineering problems are come down to some efficient methods which are problem-specific [1]. Since optimization methods can not escape falling in into some of the local optima, metaheuristics as very modern and efficient global techniques are considered to overcome these type of problems [2]. Besides, those are capable of generating quality solutions in a reasonable amount of time. The creating of quality solutions is related to the establishment of the right balance between exploration and exploitation. [3]. Since a magic formula does not exist, which works for all types of problems [4], in this paper, several swarm intelligence algorithms [5] have been adopted for solving nonlinear engineering problems. Some of the most popular swarm intelligence optimization techniques are artificial bee colony(ABC) [6][7][8][9], firefly algorithm (FA) [10][1][11][12], cuckoo search (CS) [13][14], bat algorithm (BA) [15][16][17][18][19], flower pollination algorithm [20], and etc. In this article, we have combined the bat algorithm as a representative of swarm intelligent multi-agent algorithm with one agent simulated annealing method to produce as much as possible suboptimal solutions.

The Bat meta-heuristic algorithm (BA) has proposed by Xin-She Yang 2010 [15]. The primary mechanism of this swarm intelligence technique propagates echolocation of bats as agents. The agents seek for prey and avoid obstacles by using echolocation. In the paper [19], it has been shown that the BA very well performs local search, but at times it deviated into some local optima, and it can not reach the optimal solution while solving a hard problem. The original version of bat algorithm, as well as the other metaheuristic algorithms, were designed to address unconstrained problems. To tackle the constrained problems, bat algorithm (BA) uses a penalty approach as a constraint handling technique [16]. From the experiments presented in [16], it can be seen that the BA is almost always superior to other metaheuristics.

To promote the results obtained by the simple bat algorithm, in this article, we propose an integrated I-BASA approach to take on engineering problems. Unlike the original bat algorithm which is not capable to found satisfying balance between diversification and intensification, the proposed I-BASA approach based on simulated annealing (SA) [21], a new mutation operator, and Gaussian distribution achieves a right balance and raises overall search performance. The integrated I-BASA method was tested on the eight well-chosen benchmark problems, and the simulation results report that our approach almost always wins the state-of-the-art algorithms regarding the convergence and accuracy. In this paper, we have decided to exploit Deb's rules as a constraint handling process instead of a standard penalty method. Throughout the simulation results, it can be seen that introduced rules significantly improve the quality of the solutions.

The basic structure of the article looks like this. The basic definitions related to constrained optimization are described in

Section II, while the detailed description of the Bat Algorithm (BA) and Simulated Annealing (SA) is presented in Section III and Section IV, respectively. Details of our I-BASA approach are in Section V. The brief review of eight engineering optimization problems is there in Section VI. Parameter settings and comparative results of applying state-of-the-art algorithms for solving engineers problems are presented in Section VII. Ultimately, the article is concluded in the last Section VIII.

II. Constrained Optimization

The general form of most engineering problems is expressed over objective functions and constraints, which are usually nonlinear manner. These problems are considered as constrained optimization problems containing inequality and equality constraints. They become increasingly difficult or even impossible when the traditional techniques are employed for their solving. Generally, their solving can be reduced on the next nonlinear programming problem

$$\min_{\mathbf{x}\in\mathbf{F}\subset\mathbb{R}^n} f(\mathbf{x}), \tag{1}$$

where **x** is a decision vector composed of n decision variables

$$\mathbf{x} = (x_1, x_2, \dots, x_n)^T \tag{2}$$

The decision variables x_i may have continuous or discrete values, where each of them is limited by its lower bound L_i and upper bound U_i $(i = 1, \dots n)$. The objective function f is defined on an n-dimensional hypercube **S** such that $S \subset \mathbb{R}^n$. It is used as a measure of effectiveness of a decision. The sets $\mathbf{F} \subseteq \mathbf{S}$ and $\mathbf{U} = \mathbf{S} \setminus \mathbf{F}$ denote feasible and infeasible search space, respectively. The feasible region can being presented as follows

$$\begin{aligned} \psi_k(\mathbf{x}) \leq 0 \quad (k = 1 \dots K) \\ \phi_j(\mathbf{x}) = 0 \quad (j = 1 \dots J), \end{aligned} \tag{3}$$

where letters K and J denote the number of inequality and equality, respectively. If a solution $\mathbf{x} \in F$, then all constraints defined by Eq. 3 must be satisfied. Otherwise, some of the constraints does not hold. For optimization algorithms, the participation of the equality constraints poses a problem in the sense of reducing available space F, so inequality ones usually replace those in this way

$$|\psi_k(\mathbf{x})| \leq \epsilon \quad (\forall k), \tag{4}$$

where $\epsilon \geq 0$ is a small violation tolerance.

It is well-known that swarm intelligence algorithms can not directly solve constrained engineering problems because they were designed for unconstrained ones. Therefore, the mapping of constrained problems into unconstrained ones is achieved by either using a penalty function or utilizing the fly-back mechanism. By using the penalty functions, a constrained issue is being addressed as an unconstrained in such way that infeasible solutions are punished or "penalized" so that the selection process favours feasible solutions. In this way, in the latest phases executing of algorithms, the search is directed towards the feasible regions of search space. The advantage of penalty functions lies in their simplicity and easy implementation, but the most challenging aspect of them lies in the finding appropriate penalty parameters in pursuit of constrained optimum. Their performance is not always satisfactory, and there is a need for more sophisticated penalty functions.

III. Bat Algorithm

In basic Bat algorithm (BA) offered by Yang [15], bats are being moved in a specific area thanks to the time delay between emission and reflection of the signal. Other words, bats produce a booming, but not long stroke and then monitor for the answers returned from the nearby objects. They have various rates of pulse emission and frequency. For experimental purposes, the pulse can be taken from $[0, 1]$, where 0 means that the emission does not exist and 1 means that the bats perform their maximum emitting. In the conventional bat algorithm, Yang has been proposed three idealized rules. The first rule states that each bat can determine distance by using echolocation, as well as it knows the background in some mysterious way. The second rule says that each bat flies entirely arbitrary when it hunts for prey. Also, any bat can adjust the wavelength of own emitted pulses and modify the vibration emission depending on the closeness of the victim. The last rule declares that loudness ranges of high positive value to some small constant value.

It is essential to highlight that the original Bat algorithm, besides standard control parameters, has a few relevant parameters. Those parameters are frequency tuning, climbing within a promising neighbourhood, shifting between exploration and exploitation. As we mentioned earlier, in order to deal with constrained design problems, Bat algorithm has to be modified. The use of penalty functions for reducing of constrained optimization to an unconstrained one, to which the pure Bat algorithm can be later applied, does not yield reliable results. Namely, the mentioned transformation demands much fine-tuning of the penalty elements that predict the quantity of penalization to be engaged. Since the shortage of punishment strategy does not commonly deliver satisfactory outcomes, we decided to employ the following three of Deb's rules in our I-BASA approach. The first Deb's rule tells that an algorithm chooses among two feasible solutions, the one with the better objective function value. Based on the second Deb's rule, a feasible solution beats infeasible one. In the last Deb's rule, if both solutions are infeasible, the one with the weakest amount of constraint violation was favoured. Some difficulties can appear in issues in which the global optima lies on frontier within feasible and infeasible parts.

Techniques for solving constrained design problems mainly begin with solutions which are not within the feasible area. Our proposed Bat algorithm for solving engineering problems also does not begin with the feasible initial population. During the running process, Deb's feasibility rules direct the solutions to the feasible region. Hence, slightly infeasible solutions are not discarded but kept in the population. They are utilized in the generation in the next iteration with the hope of giving feasible solutions. In this strategy, initially larger error values are used, and this value is gradually reduced with each iteration until it reaches to whatever acceptable error value. The pseudocode of the BA strategy for solving constrained engineering problems can be summarized in this way:

Step 1. *The building of beginning agents*: Build an initial group of N agents (bats) $(i = 1, 2, \ldots, N)$ which are randomly dispersed. Before beginning an iterative process, evaluate them, and by utilising Deb's rule, determine the fittest solution as $\mathbf{x}_{best}$.

Step 2. *Investigating of novel solutions*: Querying for a new promising solution $\mathbf{x}_i^t$ is done by Eq. 5 and Eq. 6.

$$\mathbf{x}_i^t = \mathbf{x}_i^{t-1} + \mathbf{v}_i^t, \tag{5}$$

$$\mathbf{v}_i^t = \mathbf{v}_i^{t-1} + (\mathbf{x}_{best} - \mathbf{x}_i^{t-1}) f_i, \tag{6}$$

In Eq. 5 and Eq. 6 letters $\mathbf{v}_i^t$ and $\mathbf{x}_{best}$ present agent quickness of change and current global most suitable solution, sequentially, while the alphabet f_i in Eq. 6 is the frequency which is being yielded as

$$f_i^t = f_{min} + (f_{max} - f_{min})\beta, \tag{7}$$

where β is a random quantity uniformly extracted from $[0, 1]$, while the letters f_{min} and f_{max} are constants which are usually initialized to 0 and 2, respectively.

It is worth noting here when a new vector $\mathbf{x}_i^t$ has been built by Eq. 5, then if its arbitrary element x_i^j is not inside the interval $[L_i,\ U_i]$, it will be substituted by the element $L_i + |x_i^j| \% (U_i - L_i)$.

Step 3. *Intensification and diversification*: In this step depending on the values $rand_1$ and pulse rate r_i^t, it is performed intensification and diversification by applying the innovative operator

$$\mathbf{x}_{new} = \begin{cases} \mathbf{x}_{best} + \epsilon A_t, & \text{if } rand_1 > r_i^t \\ \mathbf{x}_i^t, & \text{else} \end{cases} \tag{8}$$

where $A_t =< A_i^t >$ denotes noisiness on average in the iteration t of all agents $\mathbf{x}_i^t$, while the parameters ϵ and $rand_1$ are random numbers uniformly chosen from the intervals $[0, 1]$ and $[-1, 1]$, respectively. In Eq. 8, the pulse rate r_i^t was defined by

$$r_i^t = r_i^0 (1 - e^{-\beta t}), \tag{9}$$

where $r_i^0 \in$ is an initial pulse rate of the ith agent, and β is a fixed number. Additionally, in this step, edge conditions have to be controlled as in Step 2.

Step 4. *The election of a different candidate in fly*: In this step, if the solution $\mathbf{x}_{new}$ in the sense of Deb's rules has better cost value than the past one $\mathbf{x}_i^{t-1}$ or holds $A_i^t > rand_2$, then the solution $\mathbf{x}_i^t$ and the cost value $f(\mathbf{x}_i^t)$ are modified to $\mathbf{x}_{new}$ and $f(\mathbf{x}_{new})$, respectively. Here, the letter $rand_2$ is a random number from $[0, 1]$, while the loudness A_i^t can be expressed as

$$A_i^t = \alpha A_i^{t-1}, \tag{10}$$

where the changeless factor α behaves likewise to the cooling constant in the SA algorithm.

Step 5. *Record the fittest solution*: According to Deb's rules, write down the fittest solution as $\mathbf{x}_{best}$.

Step 6. *The end criteria*: The algorithm is over if the finish criteria are reached or all iterations of the algorithm are consumed. Otherwise, revert to Step 2.

IV. Simulated Annealing

In this section, we explain the simulated annealing (SA) algorithm as one of the fundamental and often picked heuristic technique [21].Simulated annealing is a well-known heuristic algorithm, whose mechanism is relied on the annealing procedure through metal adaptation. If the convergence period is prolonged, this algorithm can almost always achieve a global convergence. The primary preference of simulated annealing is that it can control its transition probability by controlling temperature, which further implies that the algorithm principally escapes to being caught into some local optima. Since simulated annealing is one kind of Markov chain, the fundamental steps of this method for solving constrained design problems are:

Step 1. Select temperature T_0, generate randomly drawn components of a solution $\mathbf{x}_0$ and the counter of iterations t sets to one.

Step 2. Determine the stopping temperature T_{stop}, set n as a maximum number of iterations and define the cooling table as follows

$$T_t = \alpha T_{t-1}, \quad \alpha \in (0, 1), \tag{11}$$

where α is a cooling schedule factor.

Step 3. Randomly select a new solution $\mathbf{x}_{t+1}$ as follow

$$\mathbf{x}_{t+1} = \mathbf{x}_t + r_1, \tag{12}$$

where $r_1 \in (0,\ 1)$ denotes an uniform random number.

Step 4. Calculate the difference Δf between the fitness values of the solutions $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$ as follow

$$\Delta f = f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t), \tag{13}$$

where $f(\mathbf{x})$ is the cost value of vector $\mathbf{x}$.

Step 5. According to Deb's rules, the new solution $\mathbf{x}_{t+1}$ generated by Eq. 12 is being accepted if it has more useful fitness value than $\mathbf{x}_t$. Otherwise, the solution $\mathbf{x}_{t+1}$ will be selected if the following condition is satisfied

$$p = e^{\frac{-\Delta f}{T}} > r, \tag{14}$$

where p denotes the transition probability, and $r \in (0, 1)$ is randomly chosen number.

Step 6. Memorize the current optimal solution $\mathbf{x}_*$ and the best cost value $f(\mathbf{x}_*)$. Reduce the temperature T due to the Eq. 11.

Step 6. If termination criterion $T > T_{stop}$ is not valid or holds $t \geq n$, then the iteration procedure is over. Oppositely, set $t \leftarrow t + 1$, and repeat above steps from Step 3 to Step 6.

There are many types of research on how to merge the simulated annealing and other optimization techniques to obtain hybrid methods [22]. In this paper for engineering optimization, for the first time, we integrate simulated annealing, Gaussian distribution, and a new mutation operator with the original BA to extra improve the searchability and also accelerate overall convergency.

V. An Integrated Bat Approach For Solving Constrained Engineering Problems: I-BASA

By analysing preliminary outcomes shown in the paper [16], we can infer that standard bat algorithm has succeeded at least once to produce near-optimal solutions during 30 independent runs. However, although it was able to generate acceptable solutions using a small number of evaluations, it can be perceived based on statistical results, how it is less stable contrast to other algorithms. The main disadvantages can be classified as a short seeking of search space and not well established an equilibrium between exploitation and exploration. To overcome discussed drawbacks, we incorporate some parts of the SA algorithm, a new mutation operator, and Gaussian perturbations into the original bat algorithm to improve its performance. As a result, we provide the I-BASA approach in solving engineering optimization problems. By applying this method, the overall stability will be increased because a better exploration disables algorithm being trapped in some local optimum. Also, as another consequence of that, the enhanced integrated bat algorithm will not iterate until all iterations are exhausted, and it only will require a few iterations for obtaining high-quality solutions. Hence, the proposed I-BASA consists of two significant parts similarly as it was done in the case of unconstrained optimization [23]. In the first part, as soon as the algorithm builds the first group of agents, fittest solutions are changed by novel solutions produced by employing SA, accompanied by the original updating formulas of the bat algorithm. In the second part of the mentioned approach, Gaussian distribution is utilised to scatter locations as much as possible. Also, a new mutation operator was introduced in order to raise the convergency of approach and establish an acceptable ratio between intensification and diversification. Also, the I-BASA includes Deb's rules to manage constraints instead of a penalty approach shown in paper [16]. Experimental analysis will show that our proposed I-BASA can efficiently perform intensification as well as diversification of the space compared to the rest algorithms. The details of our proposed I-BASA approach are given as follows:

Step 1. Our I-BASA method begins by randomly generating population P containing n agents $\mathbf{x}_i = (x_{i,j})_{j=1}^{d}$ $(i = 1, \cdots, n)$ of dimension d, where each vector $\mathbf{x}_i$ can be solution of an engineering problem. Also, in this step are initialized initial loudness A_i, pulse rates r_i and r_i^0 $(\forall i = 1, \cdots, n)$ as well as the annealing constant in SA algorithm. Before starting the iterative search process, for each solution, $\mathbf{x}_i$ fitness value is evaluated, and according to Deb's rules, the algorithm identifies both fittest solution $\mathbf{x}_{best}$ and the smallest violation g_{min}. After that, it determines the starting temeperature T_0 and the cycle counter t is reset to 0.

Step 2. Adaptation value of any agent $\mathbf{x}_i$ $(i = 1, \cdots, n)$ in the current temperature t can be depicted as:

$$Av(\mathbf{x}_i) = \frac{e^{-\frac{f(\mathbf{x}_i) - f(\mathbf{x}_{best})}{t}}}{\sum_{i=1}^{n} e^{-\frac{f(\mathbf{x}_i) - f(\mathbf{x}_{best})}{t}}} \tag{15}$$

According to the roulette selection strategy, the alternative solution $\mathbf{x}_{best}^{'}$ was picked up among all bats, while the new formula calculates the new velocity of movement v_i^t

$$\mathbf{v}_i^t = \mathbf{v}_i^{t-1} + (\mathbf{x}_{best}^{'} - \mathbf{x}_i^{t-1}) f_i, \tag{16}$$

where frequency f_i is selected by Eq. 7. To additionally boost the heterogeneity of agents into space, we introduced Gaussian operator δ by Eq. 5. Hence, the estimation of the solution $\mathbf{x}_i^t$ is accomplished by driving virtual agents $\mathbf{x}_i^{t-1}$ by the following equation

$$\mathbf{x}_i^t = \delta \mathbf{x}_i^{t-1} + \mathbf{v}_i^t, \tag{17}$$

where $\delta \in N(0, 1)$. In this step, it is necessary to scan the side conditions of the computed new solutions $\mathbf{x}_i^t$.

Step 3. For each solution $\mathbf{x}_i^t$, it should be checked the condition $r_i < rand_i$. If it is satisfied, then the local search is performed around the solution $\mathbf{x}_{best}$ as follows

$$\mathbf{x}_l^t = \mathbf{x}_{best} + a_1 \epsilon, \tag{18}$$

where $a_1 \in (0, 1)$ is a scaling factor, while $\epsilon \in (-1, 1)$ is a random number. As a result, the new solution $\mathbf{x}_l^t$ was generated. Then, as in Step 2, the boundary conditions have to be controlled for each coordinate of the vector $\mathbf{x}_l$. For the experimental purposes, we have fixed the parameter a_1 to value 0.1.

Step 4. In this step, the algorithm performs both computation sum of the violations and fitness value of the selected solution in Step 3. The generated solution from this stage will be accepted as a new one if it is better than the previous one according to Deb's rules or it holds the condition $A_i^t > rand_i$. If one of these two conditions is satisfied, then it does perform the update process. It is based on the modifications of old solutions, fitness values, and violations with the new ones. Also, in this step, the rate r_i^t is increased by Eq. 9, while the loudness of signal A_i^t is decreased by Eq. 10. It will be demonstrated throughout the simulation that the most reliable results were found for $r_i^0 = 0.5$, $A_0 = 0.99$, and $\beta = 0.9$.

Step 5. In this step, we apply the new mutation operator $\mathbf{x}_{mut}$ to the previously calculated solutions $\mathbf{x}_i$ to additionally increase the search of the entire scope. The operator $\mathbf{x}_{mut}$ is defined by

$$\mathbf{x}_{mut} = \mathbf{x}_{r_3} + a_2(\mathbf{x}_{r_1} - \mathbf{x}_{r_2}), \tag{19}$$

where r_1, r_2, r_3 are three various randomly chosen numbers in the interval $(0, n)$, and $a_2 \in (0, 2)$ is a scaling factor. Then, we compare the quality of solutions before and after introducing the $\mathbf{x}_{mut}$ operator to attain the optimal solution $\mathbf{x}_{best}$ as well as fitness value $f(\mathbf{x}_{best})$.

Step 6. In this step we memorize the solution $\mathbf{x}_{best}$ and the highest fitness value according to Deb's rules. Also, the smallest violation g_{min} was determined. The value of temperature parameter T is updated from the cooling schedule, which is defined by Eq. 11, where $\alpha = 0.97$.

Step 7. The I-BASA method stops if the end criterion is reached or the counter t is equal max_no_cycles. In contrast, increment t by one and go to Step 2.

VI. Benchmark Problems

In this part, we will quickly outline eight non-linear design problems in order to assess the performance of our proposed I-BASA approach. Each of the eight problems P_i $(i = 1, 2, \cdots, 8)$ has discrete and continuous variables. Table I summarizes basic characteristics of mentioned problems, such as dimension d, number of linear L_e and non-linear N_e inequalities. Complete mathematical formulation and their description in detail can be found in papers [1][13].

TABLE I
The main properties for the eight benchmark problems

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
d	3	4	3	7	4	2	5	11
L_e	2	2	1	4	0	0	0	0
N_e	1	5	3	7	0	3	1	10

P1. Pressure Vessel Design Problem

The basic task of the pressure vessel design problem is to develop a compressed air room with a pressure of 3×10^3 psi and a minimum volume of 750 ft^3. It is determined as a mixed discrete-continuos constrained problem because it has two discrete variables x_1 and x_2 and two continuous variables x_3 and x_4. These variables have the following meaning: x_1 is a shell thickness, x_2 is a thickness of the spherical head, x_3 is a radius of the cylindrical shell and x_4 is a shell length. The first two variables x_1, x_2 take the values inside interval [0.0625, 6.1875], while values of the remaining two variables x_3, x_4 belong to interval [10, 200]. The primary goal is to reduce the total charge of the pressure vessel.

P2. Welded Beam Design Problem

The primary objective of the welded beam design problem is to reduce the construction cost of the welded beam subject to restrictions on shear stress τ, bending stress σ in the beam, end deflection δ of the beam and buckling load P_c on the bar. The length of the beam is equal to 14 in, while the force of size 6000 lb is enforced at the end of the shaft. The design variables related to this problem have the following meaning: x_1 is weld thickness h, x_2 present the clamping rod length l, x_3 denotes rod height t, and x_4 is rod thickness b. These variables are bounded by the following limits: $x_1 \in [0.125,\ 5]$, x_2, x_3, $x_4 \in [0.1,\ 10]$.

P3. Tension/compression spring design problem

The aim of this problem is to reduce the construction cost of the spring, which is limited by four nonlinear constraints. It can be described by three variables x_1, x_2 and x_3, where x_1 is a wire diameter d, x_2 is a mean diameter of the spring D and x_3 is a number of effective coils N. The ranges of those variables are: $x_1 \in [0.05,\ 1.0]$, $x_2 \in [0.25,\ 1.3]$, $x_3 \in [2,\ 15]$.

P4. Speed Reducer Design Problem

The speed deducer design problem is a mixed discrete-continuous optimization problem which describes how to design a simple gearbox. Its application can be exploited between the engine and a propeller of a light aeroplane to achieve a maximum speed of rotation. The primary goal is to perform reducing of the weight for speed reducer subject to restrictions on bending stress of the gear teeth, surface stress, transverse deflections of the shifts, and stress in the shafts. The variables participating in the construction of speed reducer have the following meaning: x_1 is a face width, x_2 is a module of teeth, x_3 is a number of teeth on the pinion, x_4 and x_5 respectively represent the length of the first and second shaft between the bearings. In contrast, x_6 and x_7 are the diameters of the first and the second shaft, respectively. For these seven variables hold: $2.6 \leq x_1 \leq 3.6,\ 0.7 \leq x_2 \leq 0.8,\ 17 \leq x_3 \leq 28,\ 7.3 \leq x_4,\ x_5 \leq 8.3,\ 2.9 \leq x_6 \leq 3.9,\ 5.0 \leq x_7 \leq 5.5$.

P5. Gear Train Design Problem

The gear train design problem is a discrete optimization problem. It represents a complex issue involving a highly non-linear design space. The determination of volume or centre-to-centre distance of gear is an important subject in the design of power transmission systems. The gear ratio for a reduction gear train can be defined as the ratio of the angular velocity between input and output shafts. The total gear train ratio can be defined as follows

$$Gear\ ratio = \frac{w_0}{w_i} = \frac{x_2 x_3}{x_1 x_4} \tag{20}$$

where the variables w_o and w_i present the angular velocities of the output and input shafts, respectively. At the same time, variables x_1, x_2, x_3 and x_4 denote the numbers of teeth of the gears A, B, C and D, respectively. Those variables take values in the interval $[12,\ 60]$.

P6. Truss design problem with three-bar

The three-bar truss design problem is a continuous optimization problem in civil engineering first proposed by Nowicki in 1974. The purpose of this problem is to seek the optimum cross-section that decreases the weight of the truss. Two design variables x_1 and x_2 are used for its modelling which describe cross-sectional area. The values of mentioned variables are taken from the interval $[0,\ 1]$.

P7. Cantilever Beam Design Problem

The cantilever beam design problem presents a continuous optimization problem proposed by Fleury and Braibant. It can be described by using five connected square hollow blocks in order to make a beam. The beams are strictly braced at the one end, while a vertical force operates on the free end of the cantilever. The main objective of this problem was to minimize the weight of the cantilever. The design space includes five continuous variables x_j and one constraint g_1, where the range of variables x_j $(j = 1, 2, \cdots, 5)$ is the closed interval $[0.01,\ 100.0]$.

P8. Car Side Impact Design Problem

The car side impact design problem formulated by Gu is a mixed-continuous optimization problem. The overall number of elements in the model is approximately 90000, while the total number of nodes is close to 96000. For side-impact protection, two basic side-impact procedures are NHTSA and EEVC [1]. Based on these procedures, a car was exhibited to a side-impact. The prime goal is to reduce the weight using 11 design variables x_j $(j = 1, \cdots, 11)$ and 10 nonlinear constraints g_k $(k = 1, \cdots, 10)$. The bound conditions for these variables x_j are defined with $0.5 \leq x_j \leq 1.5 \;\; (j = 1, \cdots, 7)$, $x_8,\ x_9 \in \{0.192,\ 0.345\}$, $-30 \leq x_j \leq 30$ $(j = 10,\ 11)$.

VII. Experimental Analysis

In this experimental analysis,we have chosen 8 well-known constrained engineering problems to carry out a straightforward competition between our approach I-BASA and valuable algorithms such as SA, ABC, FA, CS, and BA. All algorithms participating in the simulation were carried out on the local machine which has the following performance:

- **Operating System:** Windows 10x64;
- **Type of processor:** Intel Core i7 3770K processor with a speed of 3.5 GHz;
- **Memory (RAM)**: 16GB;
- **Programming language:** C#;
- **Software:** Visual Studio 2019.

A. Parameter Settings

As metaheuristics have stochastic properties, each experiment was done in 30 series for each of the problems P_1, $P_2, \cdots, P_8$. The run of each algorithm is over when all its iterations are being consumed. For the experimental purposes, each algorithm allocates 2000 iterations. In this case analysis, except standard control parameters, each of the algorithms has extra control parameters which have a direct influence on their execution. The adjustments of algorithm parameters are given below:

- **SA** - The temperature T_0 at the beginning is set to 1.0, the stopping temperature T_{stop} was initialized to 1.0E-10, the beginning search period was set to 500, the annealing constant is equal to 0.5, the maximum number of rejections, acceptance and runs are set to 250, 150, and 50, respectively.
- **ABC** - The max. size of population $SP = 40$, the constant 'limit' is initialized to $SP \times d \times 5$, where d denotes the number of variables of the problem, while the modification rate MR and scout production period SPP were set to 0.9 and 400, respectively.
- **FA** - The max. size of firefly population is 40, the initial value of attractiveness β was set to 0.05, the randomization parameter α takes value from [0, 1]. Other parameters were set as $\beta_0 = 1$ and $\gamma = 1.0$;
- **CS** - The maximum population size SP is equal 40 for all benchmark problems. The parameter p_a of catching a cuckoo egg was set to 0.99;
- **BA** - The max. number of agents is 40, the initial values of the pulse rates and loudness are set to 0.5 and 0.99, respectively, the frequencies f_{min} and f_{max} respectively are set to 0 and 2.0, while both constants α and γ are initialized to 0.9;
- **I-BASA** - The size of bat population is 40, $f_{min} = 0$, $f_{max} = 2.0$, $\alpha = 0.9$, $\gamma = 0.99$, the values of parameters r_i^0 and loudness A_i^0 were initialized to 0.5 and 0.99, respectively. The annealing constant is fixed to 0.5.

B. Discussion of Experimental Results

The experimental results of the algorithms which participate in the competition are reported in Table II. The best feasible solutions demonstrate the capability of an algorithm to discover the optimal solution. At the same time, the statistical quantities such as mean and standard deviation determine the robustness of the algorithm. Also, the maximum number of iterations is closely related to the convergence of the algorithm. Best results are in bold, and those do not violate any of the constraints.

For the problem P_1, only our I-BASA approach obtained fittest solution in each run. On the other hand, the remaining algorithms are not equipped to gain the optimal solution except the ABC algorithm, which generated a slightly worse best result compared to the I-BASA method. Based on the statistical measures, it can be seen that our I-BASA is superior to other algorithms. For the problem P_2, only algorithms such as the I-BASA, CS and ABC have achieved the best optimum. The CS and I-BASA have utilized the smallest number of evaluations, wherein because of the more straightforward structure of the proposed I-BASA, the CS has consumed more than twice CPU time compared to the I-BASA algorithm as it can bee seen in Table III. Also, from the results shown in Table II, it can be observed that the I-BASA was slightly stable compared to the

TABLE II

COMPARISON OF RESULTS BETWEEN THE IBASA METHOD AND OTHER VALUABLE METAHEURISTICS FOR EIGHT DESIGN PROBLEMS OVER 30 INDEPENDENT RUNS

Problem	**Statistics**	**SA**	**ABC**	**FA**	**CS**	**BA**	**I-BASA**
P_1	*Best*	6099.738697241	6059.712773680	6059.712977959	6059.718470374	6059.796804678	**6059.712773616**
	Mean	75604.673833747	6059.713833747	6059.713518710	6059.711971580	6079.702933294	**6059.712773616**
	SD	4.54E+02	2.66E-06	3.24E-08	7.07E-09	1.18E+01	**6.27E-12**
	ANI	92649.3	1000	1000	2000	600	**1000**
	SP	1	25	40	13	40	**24**
P_2	*Best*	1.728322970	1.724852309	1.724852338	1.724852309	1.725381600	**1.704852309**
	Mean	1.739173713	1.724874566	1.724863780	1.724852309	2.255381896	**1.704852309**
	SD	5.92E-03	6.05E-06	4.02E-05	3.76E-12	5.85E-01	**3.65E-13**
	ANI	78545.8	1000	500	2000	1000	**800**
	SP	1	30	30	10	40	**26**
P_3	*Best*	0.012708232	0.012666879	0.012665383	0.012666450	0.012668443	**0.011215198**
	Mean	0.013009247	0.012797428	0.012694825	0.012695146	0.019492306	**0.011215198**
	SD	2.81E-04	9.96E-05	2.17E-05	2.89E-05	6.55E-03	**1.20E-12**
	ANI	252033.93	1000	600	1000	2000	**684.90**
	SP	1	25	30	20	40	**22**
P_4	*Best*	2994.842065360	**2993.542819303**	2993.542821348	**2993.542819303**	2993.668899790	2993.542819550
	Mean	2998.837062287	**2993.542819303**	2993.544598620	**2993.542819303**	3007.132712792	2993.542825755
	SD	2.53E+00	**2.48E-12**	5.50E-03	**2.65E-12**	6.53E+00	8.16E-06
	ANI	82264.43	**1000**	1000	**2000**	2000	1000
	SP	1	**12**	40	**7**	40	38
P_5	*Best*	3.38E-14	2.79E-13	2.55E-20	1.51E-15	5.71E-15	**1.11E-31**
	Mean	7.19E-10	1.60E-09	2.38E-13	2.79E-09	7.94E-09	**2.03E-16**
	SD	8.00E-10	3.38E-09	1.20E-12	3.24E-09	4.27E-08	**8.58E-16**
	ANI	163655.17	60	60	60	50	**60**
	SP	1	10	10	10	30	**10**
P_6	*Best*	263.896818396	263.895844535	263.895843378	263.895844333	263.895891445	**263.852843376**
	Mean	263.918269245	263.895913071	263.895843384	263.895875913	263.907303271	**263.852843376**
	SD	1.95E-02	7.28E-05	5.31E-09	2.85E-05	1.63E-02	**5.189E-14**
	ANI	87690.93	1000	500	2000	1000	**926.63**
	SP	1	40	35	10	40	**17**
P_7	*Best*	1.343702597	1.339912015	1.339911699	1.339912807	1.339919926	**1.339911698**
	Mean	1.349069456	1.339914165	1.339911722	1.339916864	1.433529924	**1.339911698**
	SD	3.65E-03	1.26E-06	2.73E-08	2.18E-06	2.80E-01	**5.44E-16**
	ANI	74087.9	2000	900	2000	1000	**811.07**
	SP	1	19	40	19	40	**24**
P_8	*Best*	22.851120887	22.843581559	22.842969358	**22.842824093**	22.846273985	**22.842824207**
	Mean	22.880749817	22.855576650	22.843086577	**22.844898883**	24.010514696	**22.843767285**
	SD	7.74E-02	1.57E-02	2.20E-04	**1.20E-03**	5.92E-01	**1.47E-03**
	ANI	70471.57	1000	600	**2000**	1500	**842.63**
	SP	1	40	40	**14**	40	**35**

ANI: *Average number of iterations*, **SP**: *Size of population*

CS algorithm. Further, the I-BASA method has achieved the best result for the problem P_3 as well as the best statistical values, such as mean value and standard deviation. Also, for this problem, the proposed I-BASA has consumed the least number of evaluations to generate the best optimum solution. By analyzing the outcomes in Table II, it can be seen that only CS and ABC have delivered the best optimum for the problem P_4, while the I-BASA and FA have generated slightly worse best results. Moreover, the ABC algorithm has used up the smallest number of evaluations as well as the least required CPU time as it reported in Table III. Since the problem P_5 is not a very hard problem, all algorithms were generated the best optimum. The least number of evaluations have consumed the algorithms ABC, FA, CS, and I-BASA. Our proposed I-BASA has produced the most precise and more stable solutions. For the problem P_6, the I-BASA method required both the smallest number of evaluations and least CPU time to build the robust solution as it can be seen in Table II and Table III. Further, the FA has generated slightly worse best solution than I-BASA, but much better optimal solution than other algorithms using only 17.500 evaluations. Therefore, for this problem, regarding convergence speed as well as robustness, the remaining algorithms are considerably inferior to the I-BASA algorithm. Considering the results of the problem P_7, we can observe that the I-BASA and FA algorithms were made the best results, where the I-BASA has a slight advantage in terms of precision, and drastically better statistical values such as mean value and standard deviation than the original FA. Also, for this problem, equally good results are obtained by ABC and CS algorithms. As it can be seen from Table II, the achieving global optima has cost 19465.68 evaluations by the proposed I-BASA, which is almost twice fewer evaluations than other methods. Finally, by analyzing the simulation results for the last problem P_8, we can conclude that I-BASA and CS algorithms achieve similar best solutions in each run. As it is shown in Table II, both FA and ABC algorithms were produced the acceptable solutions as well. The summary results confirm that the proposed I-BASA

TABLE III
Average time (in sec.) consumed by algorithms SA, ABC, FA, CS, BA and I-BASA over 30 independent series

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
SA	0.60	0.90	1.47	1.25	0.65	0.29	0.81	0.57
ABC	0.42	0.79	0.48	0.25	0.01	0.35	1.01	0.98
FA	5.91	3.96	2.80	13.58	0.03	1.47	8.92	5.24
CS	1.77	1.42	1.63	1.15	0.04	0.99	3.20	2.47
BA	0.22	0.54	0.65	0.95	0.01	0.19	0.53	0.82
I-BASA	0.41	0.64	0.28	1.00	0.01	0.18	0.59	1.01

approach can accomplish the best solutions from literature for engineering problems $P_1, P_2, \cdots, P_8$. Also, the proposed I-BASA works better than the other algorithms concerning the quality and robustness with noticeably enhanced convergence rate for the bulk of design problems.

VIII. Conclusion

The main job of the article was to design intelligent hybridization called I-BASA based on simulated annealing (SA), Bat algorithm (BA), Gaussian perturbations, novel mutation operator, which regulates the diversity of solutions in the population. It has been shown that I-BASA technique very successfully tackles design constrained problems while preserving a low convergence rate and generating accurate results. Also, it was demonstrated that the proposed I-BASA algorithm retains the standard BA's characteristics as well as improves its accuracy. Besides, the proposed I-BASA employs Deb's rules instead of a penalty approach which has been used in [16]. Additionally, our proposed I-BASA uses the geometric scheme as in the case of the SA to further improve the quality of solutions and speeds up the global convergence. Conducted experimental analysis of accuracy and performance on the eight benchmark problems state that our I-BASA model is robust, most accurate and stable as well as it has a rapid convergence rate. By examining the stated facts, it can be concluded that the I-BASA method in the future can be applied for practical solving of large-scale real-world engineering problems.

References

[1] A. H. Gandomi, X. S. Yang, and A. H. Alavi, "Mixed variable structural optimization using Firefly Algorithm," *Computers and Structures*, vol. 89, no. 23-24, pp. 2325–2336, December 2011. doi: https://doi.org/10.1016/j.compstruc.2011.08.002

[2] X.-S. Yang, "Review of meta-heuristics and generalised evolutionary walk algorithm," *International Journal of Bio-Inspired Computation*, vol. 3, no. 2, pp. 77–84,, 2011. doi: https://doi.org/10.1504/IJBIC.2011.039907

[3] M. Črepinšek, S.-H. Liu, and M. Mernik, "Exploration and exploitation in evolutionary algorithms: A survey," *ACM Comput. Surv.*, vol. 45, no. 3, pp. 35:1–35:33, July 2013. doi: https://doi.org/10.1145/2480741.2480752

[4] X.-S. Yang, "Free lunch or no free lunch: That is not just a question?" *International Journal on Artificial Intelligence Tools*, vol. 21, no. 3, pp. 5360–5366, 2012. doi: https://doi.org/10.1142/S0218213012400106

[5] ——, "Efficiency analysis of swarm intelligence and randomization techniques," *Journal of Computational and Theoretical Nanoscience*, vol. 9, no. 2, pp. 189–198, 2012. doi: https://doi.org/10.1166/jctn.2012.2012

[6] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," *Technical Report - TR06*, pp. 1–10, 2005.

[7] M. Tuba and R. Jovanovic, "Improved ant colony optimization algorithm with pheromone correction strategy for the traveling salesman problem," *International Journal of Computers, Communications & Control*, vol. 8, no. 3, pp. 477–485, June 2013. doi: https://doi.org/10.15837/ijccc.2013.3.7

[8] N. Bacanin and M. Tuba, "Artificial bee colony (ABC) algorithm for constrained optimization improved with genetic operators," *Studies in Informatics and Control*, vol. 21, no. 2, pp. 137–146, June 2012. doi: https://doi.org/10.24846/v21i2y201203

[9] I. Brajevic and M. Tuba, "An upgraded artificial bee colony algorithm (abc) for constrained optimization problems," *Journal of Intelligent Manufacturing*, vol. 24, no. 4, pp. 729–740, August 2013. doi: https://doi.org/10.1007/s10845-011-0621-6

[10] I. Fister, J. Fister, X. Yang, and J. Brest, "A comprehensive review of firefly algorithms," *Swarm and Evolutionary Computation*, vol. 13, no. 1, pp. 34–46, 2013. doi: https://doi.org/10.1016/j.swevo.2013.06.001

[11] N. Bacanin and M. Tuba, "Firefly Algorithm for Cardinality Constrained Mean-Variance Portfolio Optimization Problem with Entropy Diversity Constraint," *The Scientific World Journal*, vol. 2014, pp. 115–139, April 2014. doi: https://doi.org/10.1155/2014/721521

[12] M. Tuba, N. Bacanin, and A. Alihodzic, "Firefly algorithm for multi-objective RFID network planning problem," *Telecommunications Forum Telfor (TELFOR)*, pp. 95–98, September 2014. doi: https://doi.org/10.1109/TELFOR.2014.7034365

[13] A. H. Gandomi, X. S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Engineering with Computers*, vol. 29, no. 1, pp. 17–35, January 2013. doi: https://doi.org/10.1007/s00366-011-0241-y

[14] W. Long, X. Liang, Y. Huang, and Y. Chen, "An effective hybrid cuckoo search algorithm for constrained global optimization," *Neural Computing and Applications*, vol. 25, no. 3-4, pp. 911–926, September 2014. doi: https://doi.org/10.1007/s00521-014-1577-1

[15] X.-S. Yang, "A new metaheurisitic bat-inspired algorithm," *Studies in Computational Intelligence*, vol. 284, pp. 65–74, 2010. doi: https://doi.org/10.1007/978-3-642-12538-6%5F6

[16] A. H. Gandomi, Yang, A. H. Alavi, and S. Talatahari, "Bat algorithm for constrained optimization tasks," *Neural Computing and Applications*, vol. 22, no. 6, pp. 1239–1255, May 2013. doi: https://doi.org/10.1007/s00521-012-1028-9

[17] A. Alihodzic and M. Tuba, "Improved bat algorithm applied to multilevel image thresholding," *The Scientific World Journal*, vol. 2014, no. Article ID 176718, p. 16, July 2014. doi: https://doi.org/10.1155/2014/176718

[18] M. Tuba, A. Alihodzic, and N. Bacanin, "Cuckoo Search and Bat Algorithm Applied to Training Feed-Forward Neural Networks," vol. 585, pp. 139–162, 2014. doi: https://doi.org/10.1007/978-3-319-13826-8%5F8

[19] A. Alihodzic and M. Tuba, "Improved hybridized bat algorithm for global numerical optimization," *16th IEEE International Conference on Computer Modelling and Simulation, UKSim-AMSS 2014*, pp. 57–62, March 2014. doi: https://doi.org/10.1109/UKSim.2014.97

[20] S. M. Nigdeli, G. Bekdaş, and X.-S. Yang, "Application of the Flower Pollination Algorithm in Structural Engineering," *Modeling and Optimization in Science and Technologies*, vol. 7, pp. 25–42, December 2015. doi: https://doi.org/10.1007/978-3-319-26245-1%5F2

[21] J. M. P. V. S. Kirkpatrick, C. D. Gelatt, "Optimization by Simulated Annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983. doi: https://doi.org/10.1126/science.220.4598.671

[22] H. Yu, H. Fang, P. Yao, and Y. Yuan, "A combined genetic algorithm/simulated annealing algorithm for large scale system energy integration," *Computers & Chemical Engineering*, vol. 24, no. 8, pp. 2023–2035, September 2000. doi: https://doi.org/10.1016/S0098-1354(00)00601-3

[23] X. shi He, W.-J. Ding, and X.-S. Yang, "Bat algorithm based on simulated annealing and Gaussian perturbations," *Neural Computing & Applications*, vol. 25, no. 2, pp. 459–468, September 2013. doi: https://doi.org/10.1007/s00521-013-1518-4

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 215–222 DOI: 10.15439/2020F79 ISSN 2300-5963 ACSIS, Vol. 21

An Exact Two-Phase Method For Optimal Camera Placement In Art Gallery Problem

Adis Alihodzic, Sead Delalic, Damir Hasic
University of Sarajevo, BiH
Department of Mathematics
ul. Zmaja od Bosne, 33-35, Sarajevo
Email: {adis.alihodzic, delalic.sead, damir.hasic}@pmf.unsa.ba

***Abstract*—It is well-known that determining the optimal number of guards which can cover the interior of a simple nonconvex polygon presents an NP-hard problem. The optimal guard placement can be described as a problem which seeks for the smallest number of guards required to cover every point in a complex environment. In this paper, we propose an exact two-phase method as well as an approximate method for tackling the mentioned issue. The proposed exact approach in the first phase maps camera placement problem to the set covering problem, while in the second phase it uses famous state-of-the-art CPLEX solver to address set covering problem. The performance of our combined exact algorithm was compared to the performance of the approximate one. According to the results presented in the experimental analysis, it can be seen that the exact approach outperforms the approximate method for all instances.**

I. Introduction

THE ART gallery problem (AGP) dates back to the 1970s, and it was one of the earliest and most significant problems in sensor placement [1][2]. The calculation of optimal solutions for AGP is not only relevant from a theoretical aspect, but it also has practical importance in architecture, placement of radio antennas, urban planning, ultrasonography, sensors, mobile robotics, and other branches of science and industry [3]. In computational geometry, it presents a visibility problem of placing at least one security guard to cover every area of a museum or gallery [4]. Since the optimal camera placement (OCP) problem represents the process of finding the minimal number of cameras that are sufficient to cover every point in the environment, we can say the both AGP and OCP are very similar to each other. Art gallery problem in the original form was based on determining smallest number of security guards sufficient to see every point in an n-sided two-dimensional polygon P with or without holes. The scientists such as O'Rourke and Supowit, Lee and Lin, Katz and Rpoisman, Schuchardt and Hecker have shown that the process of looking for the smallest number of guards who can surveillance any polygons (ordinary or orthogonal) still presents an intractable NP-hard problem [5][6][7][8]. In 1975, Chvátal proved that only $\lfloor \frac{n}{3} \rfloor$ cameras are sometimes necessary and always sufficient to being covered the simple polygons composed of n vertices [9]. For n-sided polygon P with h holes, O'Rourke showed that it is necessary at most $\lfloor \frac{n+2h}{3} \rfloor$ vertex guards. On the other hand, Bjorling-Sachs, Souvaine, Hoffmann and others have shown that it is quite enough $\lfloor \frac{n+h}{3} \rfloor$ guards to being covered polygons with n vertices on the outer boundary which contain h holes inside them [10][11]. The researchers Györi, Hoffmann, Kriegel and Shermer have been shown that for the orthogonal polygons with h holes always is sufficient $\lfloor \frac{3n+4h+4}{16} \rfloor$ guards to being covered [12]. By using the colouring technique which has been used by Fisk [13] to prove the Chvátal result, the authors Avis and Toussaint have developed $O(n \log n)$ time complexity algorithm for camera placement in a simple polygon. However, the number of cameras is not minimal. Also, Bjorling-Sachs and Souvaine [10] proposed an $O(n^2)$ time algorithm for non-optimal guards positioning in a polygon P with h holes.

The placement of visual sensors in two-dimensional space can be modelled as AGP. Tasks such as surveillance require observing the interior of a polygon with a minimum number of sensors or cameras. This watching we call the interior covering (IC). For other tasks, such as inspection and image-based rendering, observing the boundaries of the environment is sufficient. In this case, keeping the boundaries, we call the edge covering (EC). Both interior covering and edge covering present NP-hard problem, and no deterministic (finite) algorithms are known for tackling this type of issue. In this paper, we propose two variants of algorithms such as an exact two-phase algorithm as well as the approximate method which participate in solving camera placement problem. For the experimental study, we developed two versions of the mentioned algorithms, to be able to process them in parallel for both edges covering and interior covering. The first phase of the exact two-phase approach serves for translating the art gallery problem into the famous set covering problem (SCP). In this phase, we will consider coverage of edges as well as surveillance of convex components, i.e. triangles, from which a polygon is composed. Also, in this phase, we exploit preprocessing algorithms, i.e. algorithms for creating a list of sets of vertices (components) that cover all vertices (parts) of a polygon P when take into consideration edge covering problem (interior covering problem). At the second phase of the algorithm, for a list of sets obtained in the previous step, we define a modified version of set covering problem and solve it as a linear optimization problem (LOP) by using a standard mathematical tool to find the exact optimal solution. The standard solving tool for such LOP is ILOG IBM CPLEX Solver [14]. The set covering problem is an NP-hard problem in the strong sense, and many algorithms have been developed for its solving [15]. The SCP is vital in practice, as it has been used to model a broad range of problems arising from scheduling, manufacturing, delivery and routing, service planning, information retrieval, etc. [16][17]. The exact algorithms are almost all based on branch-and-bound and branch-and-cut [18][19]. Caprara et al. compared different exact and heuristic algorithms for solving the SCP [20]. They demonstrated that in practice, IBM ILOG CPLEX Solver is the best exact algorithm for tackling set covering issue [21]. ILOG CPLEX delivers high-performance, robust, flexible optimizers for solving linear, mixed-integer and quadratic programming problems (including mixed-integer quadratic constrained issues). ILOG CPLEX optimizer has a modelling layer that provides interfaces to C++, C#, Java, Python, Matlab, etc. In this paper, we have used the layer Concert Technology to integrate C# into the ILOG Optimization Studio, since all routines for the first part of the exact approach were written in C#. In order to show the power of proposed techniques, the two-phase algorithm has been tested on 268 various randomly generated simple nonconvex polygons. The results produced in the experimental analysis were compared with the one reached by our sub-optimal approximate algorithm, which we also have been developed for comparison purposes. For both observings, the experimental results show that tho-phase method is better technique and yields the optimal solutions in a reasonable amount of time.

The rest of the paper is organized as follows. For art gallery problem (AGP), an approximate method for both edges covering and interior covering of a simple nonconvex polygon is described in Sect. 2. The details of our exact two-phase algorithm are presented in Sect. 3. Experimental and comparative results of applying different versions of the algorithms for AGP are presented in Sect. 4. Finally, conclusions and suggestion for future work are discussed in the last section of the paper, Sect. 5.

II. An approximate method for AGP

In this section, before we describe the approximate algorithm for solving the AGP, we will briefly introduce some additional notation to facilitate the exposure. For any two distinct points v_1 and v_2 in the plane, we denote by $\overline{v_1 v_2}$ the segment whose two endpoints are v_1 and v_2. A planar polygon P presents a closed plane figure whose boundary is composed of segments $\overline{v_i v_{i+1}}$ $(i = 0, 1, \cdots, n-1)$, where $v_n = v_0$. Also, a polygon P is simple if it is not self-crossing and has no holes. A planar polygon P is convex if it contains all the segments connecting any pair of its points. A nonconvex (concave) polygon P is a polygon that is not convex. In other words, a polygon P is nonconvex if there are two points u and w inside of P such that the segment $\overline{uw}$ is not entirely contained in the P. Also, a concave polygon must have at least four sides, and it always has at least one reflex interior angle, that is, an angle with a measure that is between 180 degrees and 360 degrees exclusive. Any point u in P is said to be visible from any other point w in P if and only if the segment $\overline{uw}$ does not intersect the exterior of P as well it is entirely contained in P. For any point $u \in P$, the set of all points in P which are visible from a vertex u is called the visibility region of u.

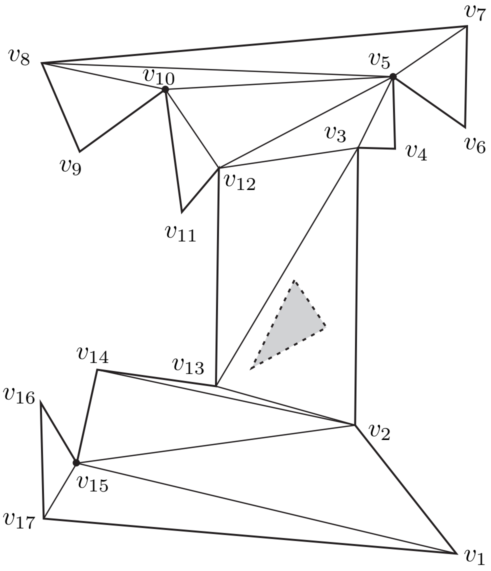


Fig. 1. The whole hull of the polygon P has covered by vertices v_5, v_{10} and v_{15}, but the inner coloured part is not visible by them.

In the following of this section, we will describe the approximate method. Let set V denotes the vertices of a simple non-convex polygon P which contains n vertices, i.e. $|V| = n$. Assume that vertices of a given polygon P are labeled by $v_1, v_2, \cdots, v_n$. Also, let $F(P, u)$ denotes the set of all points of P which can be observed from a point u. If the point u is a vertex of the polygon P, i.e. exists some index $k \in \{1, 2, \cdots, n\}$ such that $u = v_k$, then we call the subset $F(P, u)$ of P fan F_k, where the vertex v_k denotes the fan vertex of the set F_k. On the other hand, let u is not a vertex of the polygon P. Then, the set $F(P, u)$ is called a region under surveillance from the point u.

By taking into account these definitions, the main idea of our approximate method we will describe below is to being maximized fans. At the beginning of the method, we determine such fan F_{i_1} that covers the most vertices of the polygon P and set the number i_1 as the index of the first guard (camera). After that, we update the remaining sets F_j by removing from them all the elements which appear in the set F_{i_1}, i.e. we make difference $F_j \leftarrow F_j \setminus F_{i_1}$ for all fans. After this, the set F_{i_1} becomes empty, so it is no longer considered. For non-empty updated fans F_j, we repeat the same procedure as at the beginning of the algorithm, i.e. we select the fan F_{i_2} which has the most elements, and then take that index i_2 be the index of the second guard. It is clear now that the guard with index i_1 covers more vertices than the guard with the index i_2. By repeating the mentioned procedure, we can note that after a certain number of iterations, all fans F_i will be empty, which is an indicator for the end of the algorithm. Also, generated

TABLE I
THE FAN'S CALCULATION BY USING THE INDEXES OF COMPONENTS WERE COVERED BY THE VERTICES OF THE POLYGONS.

	$\mathbf{C}_1$	$\mathbf{C}_2$	$\mathbf{C}_3$	$\mathbf{C}_4$	$\mathbf{C}_5$	$\mathbf{C}_6$	$\mathbf{C}_7$	$\mathbf{C}_8$	$\mathbf{C}_9$	$\mathbf{C}_{10}$	$\mathbf{C}_{11}$	$\mathbf{C}_{12}$	$\mathbf{C}_{13}$	$\mathbf{C}_{14}$	$\mathbf{C}_{15}$
v_1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0
v_2	1	1	1	0	0	0	0	0	1	0	0	0	1	1	0
v_3	0	0	1	1	0	0	0	0	1	1	0	1	0	0	0
v_4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
$\mathbf{v}_5$	0	0	0	**1**	**1**	**1**	0	0	**1**	**1**	**1**	**1**	0	0	0
v_6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
v_7	0	0	0	0	1	1	0	0	0	0	1	1	0	0	0
$\mathbf{v}_8$	0	0	0	0	0	**1**	**1**	0	0	0	**1**	0	0	0	0
v_9	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
v_{10}	0	0	0	1	0	1	1	1	0	1	1	1	0	0	0
v_{11}	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
$\mathbf{v}_{12}$	0	0	**1**	0	0	0	0	**1**	**1**	**1**	0	**1**	0	0	0
v_{13}	1	1	1	0	0	0	0	0	1	1	0	0	1	1	0
v_{14}	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0
$\mathbf{v}_{15}$	**1**	**1**	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**
v_{16}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
v_{17}	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1

numbers $i_1, i_2, \cdots, i_k$ were sorted in descending order, where k denotes the number of guards required to cover the boundary of the polygon P. Based above described procedure for a seeking a smallest number of guards to cover vertices of a simple nonconvex polygon P, the necessary steps of the vertex observing problem has summarized by Algorithm 1. From the pseudo-code presented in Algorithm 1, we can see that the method stops when all fans become empty. In other words, since the union of fans F_i $(\forall i = 0, 1, \cdots, n-1)$ denotes the vertices indexes of a polygon P, it is easy to conclude that the algorithm terminates as soon each vertex v_i has covered. Since a simple polygon P has been compounded of the segments $\overline{v_i v_{i+1}}$, and the mentioned algorithm can cover all vertices v_i of P, i.e. all endpoints of the segments $\overline{v_i v_{i+1}}$, immediately follows that proposed method can being exploited for edge covering (EC).

Algorithm 1 Approximate Method For Vertex Covering (VC)

```
Set $n_g \leftarrow 0$, $G \leftarrow \emptyset$, where $n_g$ is a number of guards and
  $G$ is their list. Initialize the list of all fan's indexes with
  $F \leftarrow \{0, 1, \cdots, n-1\}$.
For each vertex $v_i$ $(i = 0, 1, \cdots, n-1)$ determine fan $F_i$
  by adding indexes of vertices $v_j \in P$ $(v_j \neq v_i)$ into $F_i$
  which are completely visible from the vertex $v_i$.
while $n_g \neq n$ do
    From the list $F$, find the fan that has most elements and
      denotes its index by $i$.
    Put to the list $G$ the vertex (guard) $v_i \in P$ which was
      referred to the biggest founded fan $F_i$ from the previous
      step.
    From all fans $F_j$ remove the elements which were
      appeared to the set $F_i$, i.e. $F_j \leftarrow F_j \setminus F_i$ $(\forall j \neq i)$.
    Set $n_g \leftarrow n_g + |F_i|$ and remove the index $i$ from the
      list $F$.
end while
```

From the pseudo-code presented in Algorithm 1, we can see that time complexity of our approximate method is proportional with $O(n^3)$ since the determination of fans F_i $(i = 0, 1, \cdots, n-1)$ is the most expensive step and it costs $O(n^3)$. More precisely, to examine which vertices are covered by an arbitrary vertex v_i of a polygon P, we first connect vertex v_i with the nonadjacent vertices v_k $(k \neq i-1, k \neq i, k \neq i+1)$, thus $n-3$ segments were obtained. Then in time $O(n^2)$ we check whether the generated segments intersect $n-2$ segments (segments $\overline{v_{i-1} v_i}$ and $\overline{v_i v_{i+1}}$ are not examined) which lie on the boundary of a polygon P. Since a polygon P has n vertices, then a total number of checks in the worst case is equal to $(n-3)(n-2)n$, which is proportional to $O(n^3)$.

Although the edge covering of a polygon is essential in image processing as well as in other applications, in this paper, we investigate the interior covering of a simple polygon. It is especially important to highlight here that there is a difference between the edge covering of a polygon and its interior covering. Namely, the number of guards necessary to cover the boundary of a polygon is not always sufficient to cover its overall interior, as can be seen in Figure 1. Conversely, it is always valid. For example, for the polygon has been shown in Figure 1, to perform its interior covering it was required exactly four guards such as v_5, v_8, v_{12}, and v_{15}, which represents the optimal number of guards. On the other hand, guards such as v_5, v_{10} and v_{15} can only cover the boundary of the polygon P, because it remains uncovered shaded triangle.

Before we perform interior covering of a nonconvex simple polygon P, we will address a polygon decomposition into a set of convex components C_k such that their union is the entire region of P. Now, interior covering (IC) can be modelled as a seeking the smallest number of guards required to cover the building components C_k such that their union is a whole polygon P. As earlier, in terms of fans, we define that fan F_i contains the indexes of components that can be covered by the vertex v_i. It is easy to note that each component belongs at least one of the fan so that the entire region of P is covered. The method shown in Algorithm 1 can also be exploited for interior covering of a polygon by modifying its step 2 as follows. Instead of vertex covering, we will now visit the components, i.e. at the fan F_k, we will add those indexes of components that the vertex v_k can visit. In this way, we get the algorithm for interior covering (IC) of a simple nonconvex polygon.

Algorithm 2 The Exact Two-Phase Approach For Interior Covering (IC)

```
1: Determine a triangulation of of $n$ sided nonconvex simple polygon $P$. Let us denote the obtained triangles as components $C_1, C_2, \cdots, C_{n-2}$.
2: For each vertex $v_i$ $(i = 0, 1, \cdots, n-1)$, determine the fan $F_i$ by adding indexes of components $C_j$ into $F_i$ which are completely visible from the vertex $v_i$.
3: Create the matrix $A$ from the content of fans $F_i$.
4: Make preprocessing and reduce the number of rows for the matrix $A$.
5: Apply CPLEX solver to generate the smallest number of guards necessary for interior covering of a polygon.
6: Visualize the founded guards.
```

III. An Exact Two-Phase Approach For AGP

In this section, we will describe in detail our exact two-stage algorithm designed to solve the Art Gallery Problem (AGP). First of all, in the first stage, we will transform the art gallery problem to the well-known Set Covering Problem (SCP). After that, in the second stage, we will apply prominent CPLEX solver to address the adjusted set covering problem obtained in the first stage.

A. The mapping of the AGP on the SCP

In order to map the art gallery problem to the set covering one, we will first divide the interior of a polygon P into a set of nonoverlapping convex parts. There are several ways how to perform dividing a simple closed nonconvex polygon into nonoverlapping convex sub-polygons or pieces [22]. In this paper, partitioning a polygon into convex parts has obtained by exploiting triangulation. To efficiently perform triangulation, we have implemented a very efficient algorithm whose time complexity is proportional to the $O(n \log n)$ [23]. This algorithm consists of two steps. In the first step, we made a partition of a simple polygon with n vertices into monotone pieces in $O(n \log n)$ time, while in the second step, we triangulated monotone pieces (polygons) in linear time $O(n)$. The above steps together imply that any simple nonconvex polygon P can be triangulated in $O(n \log n)$ time.

The application of triangulation on any simple nonconvex polygon P composed of n vertices produces $n-2$ triangles, i.e. $n-2$ convex components $C_1, C_2, \cdots, C_{n-2}$. By introducing these components, optimal coverage of a polygon interior has reduced to seeking the smallest number of guards which can see all components. In order to determine those guards, we will first create fans F_j $(j \in \{1, 2, \cdots, n\})$ for each vertex v_j. In the context of components, arbitrary fan F_j contains the indexes of components which are visible from the vertex v_j. In the following, we consider the creating of 15 fans for a simple non-convex polygon shown in Figure 1. It is easy to see that for vertex v_1 fan F_1 has indexes 1, 2, 13, 14, since the vertex v_1 covers the components (triangles) C_1, C_2, C_{13}, C_{14}. In Table I, for each vertex v_j $(j = 1, 2, \cdots, n)$, we have presented calculated fans F_j in the form of rows. For example, with respect to the vertex v_1, the fan F_1 has indexes 1, 2, 13, and 14.

From the structure of data shown in Table I, it is easy to notice that an optimal covering of a polygon can be made by exploiting the adjusted version of the set-covering problem (SCP). The adjusted version of the SCP can be defined as follows. Let $A = (a_{i,j})$ be an zero-one matrix of $n \times n - 2$ size. We say that a row i covers a column j if holds $a_{ij} = 1$. Let $I = \{1, 2, \cdots, n\}$ and $J = \{1, 2, \cdots, n-2\}$ be the row set and column set, respectively. The SCP needs determining the minimum subset $I' \subset I$ such that each column $j \in J$ is covered by at least one row $i \in I'$. A mathematical model for the adjusted SCP is defined as follows

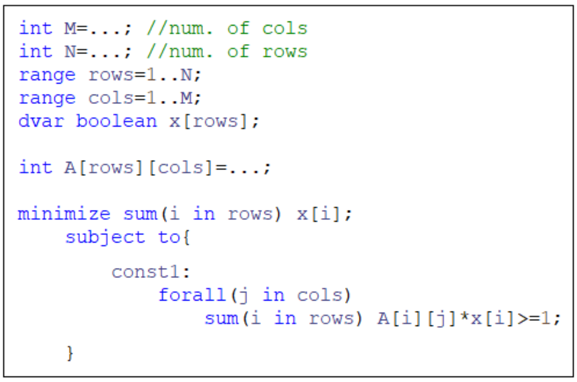
$$Minimize\ f(x) = \sum_{i=1}^{n} x_i \tag{1}$$

subject to

$$\sum_{i=1}^{n} a_{ij} x_i \geq 1,\ \forall j \in J \tag{2}$$

$$x_i \in \{0, 1\},\ \forall i \in I \tag{3}$$

From Eq. 1 immediately implies that we have to minimize the number of guards (number of the selected rows), where $x_i = 1$ if the row i is in the solution and $x_i = 0$ otherwise. Each column j is covered at least by one row i. The SCP constraints guarantee this.



```
int M=...; //num. of cols
int N=...; //num. of rows
range rows=1..N;
range cols=1..M;
dvar boolean x[rows];

int A[rows][cols]=...;

minimize sum(i in rows) x[i];
    subject to{
        const1:
            forall(j in cols)
                sum(i in rows) A[i][j]*x[i]>=1;
    }
```

Fig. 2. The OPL Code For Solving Simplified SCP.

After we have described the mechanism for optimal interior covering of a simple nonconvex polygon, its realization is carried out by the method whose necessary steps were summarized in the pseudo-code of Algorithm 2. From the pseudo-code, we can see that the overall time complexity of this approach is proportional to $O(n^3)$. Although this approach for both interior covering (IC) and edge covering (EC) has the same time complexity, there is a slight difference in the specified number of evaluations during their execution, which will be seen in the experimental analysis. Also, for both types of coverage, the preprocessing remained more costly in time compared to the finding of an optimal solution by using CPLEX solver.

B. The preporcessing of the adjusted SCP

It is well-known that preprocessing is a superior technique to accelerate the solving process by reducing the instance sizes additionally. There are a lot of preprocessing methods in the literature for the SCP [18]. In this paper, to reduce the instance size, i.e. the size of the matrix A, we apply methods such as *row domination* and *row inclusion*. The row domination can be defined as follows. We say that row i is dominated and can be removed from the matrix A if its columns J_i can be covered by other rows. In Table I, we can see that row 14 is dominated by rows 1 and 15, and it can be removed. Also, rows 1 and 17 are dominated by row 15, so they should be dropped. After the elimination of the rows from the matrix A, it was reduced on only six rows. Therefore, it consists of rows with indices 3, 5, 8, 12, 13 and 15. Row inclusion means that if a column j is covered by precisely one row after the above domination, then it row is included in an optimal solution. It can be noted that matrix A whose data has been shown in Table I before processing had 17 rows, and after processing it has only six rows.

C. The application of CPLEX solver on the adjusted SCP

The second phase of our proposed approach was based on solving of the adjusted set covering problem (ASCP). The adjusted set covering problem in IBM ILOG CPLEX Optimization Solver [14] is presented in the form of a binary integer programming problem, as we can see in Figure 2. From the code shown in Figure 2, we can note that the matrix A is being generated in the first phase of our algorithm. At the same time, the vector x is a decision binary vector whose components are being determined by CPLEX solver. On the beginning of the algorithm, CPLEX solver set the content of the vector x to zero. CPLEX solver by using techniques such as branch-and-bound as well as branch-and-cut, it is capable of determining an optimal solution just over several seconds, as we will see in the experimental analysis. For the polygon drawn in Figure 1, after the execution of CPLEX solver, the content of the vector x becomes this one $x =$[0 0 0 0 1 0 0 1 0 0 0 1 0 0 1 0 0], where ones (1) indicate that the vertices v_5, v_8, v_{12}, v_{15} are covered in case of interior covering. If we first make the elimination of rows, and after that, we apply CPLEX solver on the remaining rows of the matrix A, we will see that the size of vector x is reduced from 17 to 6, so the final solution x now has this form: $x =$[0 1 1 1 0 1].

IV. Experimental Results

In this experimental study, we compare two groups of deterministic algorithms for solving both edges covering (EC) and interior covering (IC) of a simple nonconvex polygon. The goal of our proposed methods was to minimize the number of cameras required for polygon coverage. The proposed techniques have been thoroughly tested to assess the quality of the results. In the first group of deterministic algorithms, there are exact methods, while the second group encompasses the approximated techniques. The algorithms have been applied to 268 various randomly generated simple nonconvex polygons. The simple nonconvex polygons have been produced with our random polygon generator developed for purposes of this paper, whose implementation details we omit. The interested reader can refer to similar random polygon generator (RPG) [24][25]. The examples of randomly generated simple nonconvex polygons with 50 edges are shown in Figure 3. Each instance is called RI-k-i, where k denotes the size of the ith instance. The coordinates of points (x, y) are chosen from the interval $[0, 500]$. Through the experimental evaluation, we assess the applicability of four algorithms for the AGP. The proposed approaches have been implemented in C# programming language, where two exact two-phase methods in their second phase use IBM ILOG CPLEX Optimizer 12.10 to address set covering problem expressed in the form of binary integer programming (BIP). This optimizer is being called from C# in conjunction with the Concert Technology. In order to evaluate time efficiency and the coverage rate of the proposed algorithms, a comparison test was performed using a PC with an Intel Core i7-3770K @3.5GHz with 64GB of RAM running under the Windows 10 x64 operating system.

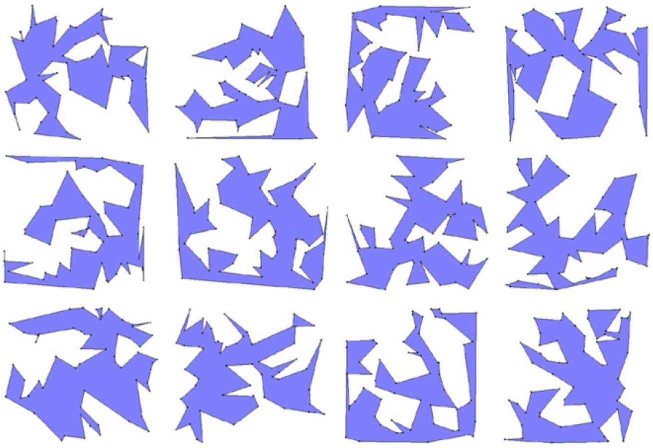


Fig. 3. Randomly generated simple nonconvex polygons with 50 vertices.

In Table II, for the purpose of checking the quality of obtained solutions as well as computational times, four algorithms were selected and tested through 8 groups of randomly generated polygons, where each group was composed of five randomly generated polygons containing 20, 40, 60, 100, 300, 500, 1000, and 2000 vertices, respectively. From obtained simulation results shown in Table II, we can see for edge covering (EC) problem that approximate method is slightly faster than the exact two-phase method for almost all groups except for the next to last group which contains 1000 vertices.

TABLE II
THE AVERAGE NUMBER OF GUARDS AND MEAN TIME PROCESSING PROVIDED BY THE EXACT AND APPROXIMATE ALGORITHMS FOR 40 RANDOMLY DISTRIBUTED INSTANCES.

	INTERIOR COVERING (IC)				**EDGE COVERING (EC)**			
	Two-phase approach		Approximate approach		Two-phase approach		Approximate approach	
Random instances	Number of guards	Time (sec.)	Number of guards	Time (sec.)	Number of guards	Time (sec.)	Number of guards	Time (sec.)
RI- 20-1	4	0.19	5	0.01	3	0.01	4	0.00
RI- 20-2	3	0.19	4	0.01	3	0.19	4	0.00
RI- 20-3	2	0.02	3	0.01	2	0.17	3	0.01
RI- 20-4	3	0.02	4	0.01	3	0.18	4	0.00
RI- 20-5	3	0.19	4	0.01	3	0.01	4	0.00
RI- 40-1	5	0.06	7	0.05	5	0.15	6	0.02
RI- 40-2	6	0.19	7	0.04	6	0.03	7	0.02
RI- 40-3	6	0.19	7	0.05	5	0.16	7	0.02
RI- 40-4	7	0.06	8	0.04	7	0.03	8	0.02
RI- 40-5	6	0.14	7	0.05	6	0.16	7	0.02
RI- 60-1	9	0.12	11	0.15	8	0.13	9	0.06
RI- 60-2	9	0.19	11	0.12	8	0.08	10	0.07
RI- 60-3	11	0.13	12	0.11	9	0.08	10	0.05
RI- 60-4	10	0.13	12	0.10	10	0.12	11	0.06
RI- 60-5	9	0.14	11	0.13	8	0.12	11	0.07
RI- 100-1	14	0.37	16	0.36	12	0.38	14	0.24
RI- 100-2	15	0.40	17	0.31	14	0.23	15	0.19
RI- 100-3	16	0.37	18	0.27	14	0.21	16	0.19
RI- 100-4	14	0.38	19	0.36	12	0.24	15	0.19
RI- 100-5	15	0.37	17	0.36	13	0.21	16	0.14
RI- 300-1	40	5.48	44	4.86	35	3.65	38	3.82
RI- 300-2	46	4.44	52	4.28	42	2.99	46	3.15
RI- 300-3	44	4.46	50	4.10	37	3.19	39	2.96
RI- 300-4	43	5.03	50	4.22	36	3.64	43	3.57
RI- 300-5	42	4.44	50	4.83	37	3.20	42	3.26
RI- 500-1	77	16.31	87	13.99	66	12.08	75	10.57
RI- 500-2	70	15.20	85	12.80	63	11.02	72	9.81
RI- 500-3	71	17.29	77	14.77	62	13.20	68	11.43
RI- 500-4	73	16.11	87	13.73	62	12.37	77	10.75
RI- 500-5	72	12.87	83	11.48	63	8.58	74	7.90
RI- 1000-1	148	74.09	162	73.92	129	60.92	147	66.20
RI- 1000-2	143	77.38	155	76.96	125	61.06	138	70.30
RI- 1000-3	141	82.21	154	83.04	120	69.34	137	80.68
RI- 1000-4	145	85.95	161	77.50	133	55.98	154	59.68
RI- 1000-5	140	93.73	155	92.02	124	75.53	141	81.98
RI- 2000-1	296	639.77	322	643.25	260	579.70	293	570.81
RI- 2000-2	303	618.68	335	621.49	265	561.08	305	548.73
RI- 2000-3	292	682.25	318	687.29	253	624.10	280	602.25
RI- 2000-4	285	455.22	318	445.97	254	398.18	291	385.15
RI- 2000-5	291	567.14	316	547.54	249	511.58	290	478.43

More precisely, the approximate method requires 3012.26 seconds to process all instances of polygons from all groups. On the other hand, the exact two-phase method costs 3072.93 seconds, which is 60.67 seconds more in comparison with the approximate method. In the sense of quality solutions, i.e. in the sense of the smallest number of guards, the exact two-phase method outperformed approximate method for any groups of instances. Namely, the exact method requires only 2566 guards (cameras) to cover all polygons from all eight groups, while the approximate method allocates 1558 cameras more, i.e. it needs 4124 cameras. Particularly superiority comes to the fore with an increase in the size of vertices. In practice for the case of observing a polygon consisting of 2000 vertices, e.g. in this paper if we take the randomly

TABLE III
THE AVERAGE NUMBER OF GUARDS AND MEAN TIME PROCESSING PROVIDED BY THE EXACT AND APPROXIMATE ALGORITHMS FOR 228 RANDOMLY GENERATED INSTANCES.

		INTERIOR COVERING (IC)				**EDGE COVERING (EC)**			
		Two-phase approach		Approximate approach		Two-phase approach		Approximate approach	
No. rand. instances	Size (n)	Mean no. of guards	Mean time (s)	Mean no. of guards	Mean time (s)	Mean no. of guards	Mean time (s)	Mean no. of guards	Mean time (s)
30	20	3.43	0.13	3.77	0.01	3.07	0.11	3.53	0.00
30	40	6.23	0.13	7.03	0.05	5.77	0.13	6.40	0.02
28	60	9.07	0.16	10.04	0.12	8.11	0.13	9.21	0.06
23	80	12.22	0.21	13.43	0.15	10.52	0.15	12.09	0.10
25	100	14.88	0.36	16.28	0.32	13.44	0.24	14.88	0.18
24	200	29.50	1.80	33.04	1.60	25.54	1.08	29.38	0.84
21	300	44.00	4.37	48.90	4.28	39.05	3.16	43.00	3.14
24	400	58.83	8.65	66.08	7.89	51.46	6.17	57.67	6.25
23	500	72.35	15.41	81.17	13.56	63.78	11.68	72.13	9.97

generated polygon such as RI-2000-5, we make earnings of 41 cameras. These earnings are not only referred to money for purchasing of additional cameras as well as on the savings of other resources. For example, each object covered by cameras requires electricity to power them, as well as specific hardware resources, such as external memory, which is used to store images (usually 30 frames per second) obtained via cameras daily. Based on the number of guards necessary to cover the boundary of a polygon, we can conclude that the approximate method is not able to find the global optimum. In contrast, the exact two-phase approach is capable of doing it in a short period. We have earlier shown on the example of the polygon shown in Figure 1 that edge covering (EC) is not the same as interior covering (IC). Namely, more guards are needed to perform interior covering compared to edge covering as we can see in Table Table II. For instance, in the case of polygon RI-2000-1, the exact two-phase method needs 296 cameras for interior covering, while it costs only 260 cameras for edge covering. Based on the results shown in Table II, we can note that also for interior surveillance of the polygon, the exact method yields a better solution (a smaller number of guards). On the other hand, the approximate approach is usually get trapped in some local optima, and as a consequence of it does not generate the optimal number of cameras.

In order to show the real robustness of the proposed methods, we tested them for a reasonably large dataset, i.e. for a dataset composed of 228 randomly generated simple nonconvex polygons, and the results obtained were saved in Table III. The simulation results show that the exact method gets in average better quality solutions compared with the approximate one for all sizes of the polygon. Other words, for both interior coverage and edge coverage, the mean number of cameras increases linearly concerning the size of vertices (n), so that a growth rate of the cameras being noticeably slower in the exact method compared to the price of growth generated by the approximate comparative approach. Also, by considering produced experimental results in Table III, we can conclude for all versions of polygon coverage, both exact and approximate methods are comparable in terms of CPU execution time, so that the approximate method being negligibly faster than the exact one.

Based on the experimental analysis, it can be concluded the exact method presents an appropriate practical tool that with a minimal number of cameras can cover the interior of the polygon as well as its hull, which has direct applications in security systems, computer graphics, computer vision, and other branches of industry.

V. Conclusion

In this paper, we studied the problem of guarding a simple nonconvex polygon and proposed four versions of algorithms for its solving. Quality of the proposed methods was tested throughout 268 randomly generated instances. Based on the obtained results, it can be concluded that our exact two-phase algorithm is convenient for this task, and it produces excellent overall performance. Also, our two-phase approach proved to be robust, in the sense that it was able to tackle different instances from a broad range of randomly generated. Since the first phase of our two-phase method is computationally expensive, in future work, we will investigate the efficient techniques in order to tackle these drawbacks. This further says that the improvements of the two-phase algorithm can be achieved. Also, we will consider other types of polygons with and without holes, such as orthogonal polygons, Von Koch polygons, and so for.

References

[1] E. Tuba, I. Tuba, D. Dolicanin-Djekic, A. Alihodzic, and M. Tuba, "Efficient drone placement for wireless sensor networks coverage by bare bones fireworks algorithm," in *2018 6th International Symposium on Digital Forensic and Security (ISDFS)*, 2018. doi: https://doi.org/10.1109/ISDFS.2018.8355349 pp. 1–5.

[2] E. Tuba, R. Capor-Hrosik, A. Alihodzic, and M. Tuba, "Drone placement for optimal coverage by brain storm optimization algorithm," in *Hybrid Intelligent Systems*. Cham: Springer International Publishing, 2018. doi: https://doi.org/10.1007/978-3-319-76351-4%5F17 pp. 167–176.

[3] A. Elnagar and L. Lulu, "An art gallery-based approach to autonomous robot motion planning in global environments," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 2079–2084.

[4] J. O'Rourke, *Art Gallery Theorems and Algorithms*. Oxford University Press, 1987.

[5] J. O'Rourke and K. Supowit, "Some np-hard polygon decomposition problems," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 181–190, March 1983. doi: https://10.1109/TIT.1983.1056648

[6] D. Lee and A. Lin, "Computational complexity of art gallery problems," *IEEE Transactions on Information Theory*, vol. 32, no. 2, pp. 276–282, 1986. doi: https://10.1109/TIT.1986.1057165

[7] M. J. Katz and G. S. Roisman, "On guarding the vertices of rectilinear domains," *Computational Geometry*, vol. 39, no. 3, pp. 219 – 228, 2008. doi: https://doi.org/10.1016/j.comgeo.2007.02.002

[8] D. Schuchardt and H. Hecker, "Two np-hard art-gallery problems for ortho-polygons," *Mathematical Logic Quarterly*, vol. 41, no. 2, pp. 261 – 267, 1995. doi: https://

[9] V. Chvátal, "A combinatorial theorem in plane geometry," *Journal of Combinatorial Theory, Series B*, vol. 18, no. 1, pp. 39 – 41, 1975. doi: https://doi.org/10.1016/0095-8956(75)90061-1

[10] I. Bjorling-Sachs and D. L. Souvaine, "An efficient algorithm for guard placement in polygons with holes," *Discrete & Computational Geometry*, vol. 13, p. 77–109, January 1995. doi: https://doi.org/10.1007/BF02574029

[11] F. Hoffmann, M. Kaufmann, and K. Kriegel, "The art gallery theorem for polygons with holes," in *[1991] Proceedings 32nd Annual Symposium of Foundations of Computer Science*, October 1991. doi: https://10.1109/SFCS.1991.185346 pp. 39–48.

[12] E. Györi, F. Hoffmann, K. Kriegel, and T. Shermer, "Generalized guarding and partitioning for rectilinear polygons," *Computational Geometry*, vol. 6, no. 1, pp. 21 – 44, 1996. doi: https://doi.org/10.1016/0925-7721(96)00014-4

[13] S. Fisk, "A short proof of chvátal's watchman theorem," *Journal of Combinatorial Theory, Series B*, vol. 24, no. 3, p. 374, 1978. doi: https://doi.org/10.1016/0095-8956(78)90059-X

[14] CPLEX, *IBM ILOG CPLEX Optimization Studio CPLEX User's Manual V 12.7*. International Business Machines Corporation, 2017. [Online]. Available: https://www.ibm.com/support/knowledgecenter/SSSA5P_12.7.1/ilog.odms.studio.help/pdf/usrcplex.pdf

[15] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. USA: W. H. Freeman & Co., 1990. ISBN 0716710455

[16] E. Balas and A. Ho, *Set covering algorithms using cutting planes, heuristics, and subgradient optimization: A computational study*. "Springer Berlin Heidelberg, 1980, pp. 37–60. ISBN 978-3-642-00802-3

[17] T.-P. Shuai and X.-D. Hu, "Connected set cover problem and its applications," in *Algorithmic Aspects in Information and Management*. Springer Berlin Heidelberg, 2006. doi: https://doi.org/10.1007/11775096%5F23. ISBN 978-3-540-35158-0 pp. 243–254.

[18] M. L. Fisher and P. Kedia, "Optimal solution of set covering/partitioning problems using dual heuristics," *Management Science*, vol. 36, no. 6, pp. 674–688, 1990. doi: https://doi.org/10.1287/mnsc.36.6.674

[19] E. Balas and M. C. Carrera, "A dynamic subgradient-based branch-and-bound procedure for set covering," *Operations Research*, vol. 44, pp. 875–890, December 1996. doi: https://doi.org/10.1287/opre.44.6.875

[20] A. Caprara, P. Toth, and M. Fischetti, "Algorithms for the set covering problem," *Annals of Operations Research*, vol. 98, no. 1-4, p. 353–371, December 2000. doi: https://doi.org/10.1023/A:1019225027893

[21] P. Laborie, J. Rogerie, P. Shaw, and P. Vilím, "Ibm ilog cp optimizer for scheduling," *Constraints*, vol. 23, pp. 210–250, April 2018. doi: https://doi.org/10.1007/s10601-018-9281-x

[22] J. O'Rourke, *Computational Geometry in C*. Cambridge University Press, September 1998.

[23] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational geometry: algorithms and applications*, 3rd ed. Springer, Berlin, Heidelberg, 2008. ISBN 978-3-540-77973-5

[24] T. Auer and M. Held, "Heuristics for the generation of random polygons," August 1996, pp. 38–43.

[25] S. Sadhu, S. Hazarika, K. K. Jain, S. Basu, and T. De, "Grp_ch heuristic for generating random simple polygon," in *Combinatorial Algorithms*. Springer Berlin Heidelberg, 2012. doi: https://doi.org/10.1007/978-3-642-35926-2978-3-642-35926-2 pp. 293–302.

DOI: 10.15439/2020F75
ISSN 2300-5963 ACSIS, Vol. 21

Conceptual Optimization of a Generalized Net Model of a Queuing System

Velin Andonov
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev Str, block 8
1113 Sofia, Bulgaria
Email: velin_andonov@math.bas.bg

Stoyan Poryazov
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev Str, block 8
1113 Sofia, Bulgaria
Email: stoyan@math.bas.bg

Emiliya Saranova
Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev Str, block 8
1113 Sofia, Bulgaria
Email: emiliya@math.bas.bg

***Abstract*—The problem of conceptual optimization of Generalized Nets (GNs) models is discussed. An overview of some operators for complexity of GNs and relations with respect to them is presented. Some new operators and relations are defined. A GN model of a queuing system with finite capacity of the buffer and server, and FIFO discipline of service of the requests, is optimized with respect to some of the operators for complexity.**

I. Introduction

ONE of the first attempts to define a set of quantifiable characteristics of a conceptual model; a measurement of the characteristics together with a fixed measurement of the decision-maker's preferences are done in Oren [8]. The proposed characteristics are: 1) size, 2) change pr. month, 3) data description inaccuracy, 4) semantic relevance, 5) semantic inaccuracy and 6) l/0-model size. Oren mentions that the conceptual model characteristics may be quantified absolutely or relatively and discusses the accuracy of quantifying the characteristics. Examples are not given.

Another approach can be found in [10]: Conceptual modelling is about abstracting a model that is fit-for-purpose and by this we mean a model that is 1) valid, 2) credible, 3) feasible and 4) useful. Some important features of the conceptual models are:

- the model is designed for a specific *purpose* and without knowing this purpose it is impossible to create an appropriate simplification;
- *simplifications* are incorporated in the model to enable more rapid model development and use, and to improve transparency;
- *assumptions* are made either when there are uncertainties or beliefs about the real world being modelled.

The characteristics in [10] are not quantifying.

Eric [2] considers the Universe of Discourse (UoD), describing which classes of entities and propositions are important for an application area. UoD consists of: functional and existence dependencies, attributes, subtype-connections, classes, labels. Let S be a concrete conceptual schema (diagram, model) which is to be evaluated by a proposed evaluation function of the following quality measures [2]:

- number of functional dependencies that hold in the UoD, but which are not expressed in S;
- number of existence dependencies that hold in the UoD, but which are not expressed in S;
- number of attributes and subtype connections in S;
- number of classes in S;
- number of labels of S.

In [6], considered Metrics for Structural Complexity are:

- number of associations – total number of associations in a model;
- number of dependencies – this metrics is used to calculate the total number of dependency relationships within the class diagram;
- number of aggregations – it calculates the number of aggregation relationships within a class diagram;
- depth inheritance tree – it calculates the longest path from the class to the root of the hierarchy in a generalization hierarchy.

Metrics for Modularity are [6]:

- Cohesion – this metric calculates the cohesion of different modules;
- Coupling – it calculates the coupling between different modules.

Many of the metrics above are difficult to be evaluated automatically. In the present paper, we use a more formal approach to metrics of structural complexity.

Generalized Nets (GNs, see [5]) are extensions of Petri Nets ([1]). For many types of Petri Nets and their extensions, it is proven that the functioning and the results of their work can be represented by an ordinary GN [4]. An important property of the GNs is that one and the same process can be modeled by more than one GN. As a result, a problem arises of choosing

The work of Velin Andonov is supported by the 2020 National Scientific Program for post doctoral and young scientists of the Ministry of Education and Science of Bulgaria. The work of Emiliya Saranova is partially supported by the joint research project "Symbolic-Numerical Decision Methods for Algebraic Systems of Equations in Perspective Telecommunication Tasks" of IMI-BAS, Bulgaria and JINR, Dubna, Russia. The work of Stoyan Poryazov was supported by the Task 1.2.5. "Prediction and Guaranteeing of the Quality of Service in Human-Cyber-Physical Systems" of National Scientific Program "Information and Communication Technologies for a Single Digital Market in Science, Education and Security (ICT in SES)" financed by the Bulgarian Ministry of Education and Science.

the most suitable GN model of a particular process among the many possible.

In the present paper, we study the problem for conceptual optimization of GN models. It is based on operators for complexity of GNs some of which are defined in [4], while some others are defined here for the first time. Relations of inclusion with regard to the results of the work of GNs about the operators are defined which allow a comparison of the GN models to be made. The optimization is demonstrated for a GN model of a queuing system with finite capacities of the server and buffer, and FIFO (First-In, First-Out) discipline of service of the requests. The choice of the model is justified by the fact that many GN models of queuing systems exist (see [9], [14], [15]).

II. On the Concepts in Generalized Net Models

The GN is a relatively complex object. Detailed definition of a *transition of a GN*, *GN* and the algorithms for transition and net functioning can be found in [5]. The concepts of a GN model can be divided into model description concepts and graphical representation concepts.

First, we shall describe non-formally the elements used in the graphical representation of a GN. GN's *places* are represented by ○.

Every transition of a GN contains *transition's conditions* which are graphically represented by ⊤.

Like Petri nets, GNs contain tokens which are transferred from place to place through the *arcs* of the net. The arcs are denoted by arrows in Fig. 1.

The names of the transitions and the places are also included in the graphical representation of the GN model. They can be very important for the understanding of the model by non-specialists in the area of GNs and for the users in general.

To summarize, the concepts of a GN model which are represented graphically are: *transition, place, arc* and the names of the transitions and the places.

III. Operators for Complexity of GNs Models

Some operators for complexity of GNs are defined in [4]. Below, we briefly present some of them and propose new ones. For arbitrary transition Z and arbitrary GN E (see [4]):

- $\phi_1(E) = |pr_1 pr_1 E|$ is the number of the transitions of the net;
- $\phi_2(E) = |pr_1 pr_1 pr_1 E \cup pr_2 pr_1 pr_1 E|$ is the number of places of the net;
- $\phi_3(E) = |pr_1 pr_2 E|$ is the number of tokens of the net;
- $\phi_4(E) = |pr_3 pr_3 E|$ is the duration of the GN functioning;
- $$\phi_5(E) = \sum_{Z \in pr_1 pr_1 E} \phi_5'(Z) = \sum_{k=1}^{|L'|}\sum_{l=1}^{|L''|} k \cdot l \binom{|L'|}{k}\binom{|L''|}{l}$$ is operator for the complexity of the transitions of the net;
- $\phi_6(E) = \max\limits_{\alpha \in pr_1 pr_2 E} b(\alpha)$ is the maximum number of characteristics that the tokens can keep during the functioning of the net;
- $\phi_7(E) = |pr_1 pr_4 E|$ is the number of initial characteristics of the tokens;
- $\phi_8(E) = \sum\limits_{Z \in pr_1 pr_1 E} pr_1 Z pr_2 Z$ is the number of arcs of the net;
- $\phi_9(E) = |\cup_{Z \in pr_1 pr_1 E} \{l | l \in pr_1 Z \& l \in pr_2 Z\}|$ is the number of places which are both intput and output for a given transition, i.e., the number of loops. This operator gives us information about the graphical representation of the net.
- $$\phi_{10}(E) = \frac{\sum\limits_{r \in pr_5 pr_1 pr_1 E} |\{r_{i,j} | r_{i,j} \in r \& (r_{i,j} = false \vee r_{i,j} = true)\}|}{\sum\limits_{r \in pr_5 pr_1 pr_1 E} |\{r_{i,j} | r_{i,j} \in r\}|}$$ is operator of determinacy, i.e., a ratio of the number of elements of the IMs of the predicates of the transitions with truth values "true" or "false" to the total number of predicates.
- $\phi_{11}(E) = |\Omega_E|$ is the number of concepts used to describe the GN E;
- $\phi_{12}(E)$ is the number of concepts used in the graphical representation of the GN E.

Above, we denote by $pr_i A$ the i-th projection of the set A.

If ϕ is some operator for complexity, then using the relations $\approx$ and $\sqsubset$ defined in [4], we can define relations of inclusion and equivalence between GNs with respect to the operator in the following way:

Definition 1: $E_1 \vdash_\phi E_2 \equiv (E_1 \approx E_2 \& \phi(E_1) \geq \phi(E_2)) \vee (E_2 \sqsubset E_1)$.

Definition 2: $E_1 \approx_\phi E_2 \equiv (E_1 \approx E_2) \& (\phi(E_1) \geq \phi(E_2))$.

IV. Optimization of a GN model of a queuing system

Since different GNs can be used to model one and the same process, it is important to determine which one is the best with regard to the purpose of the modelling. As shown in [4], a given GN can be modified through the operators over GNs. As a result of the application of some of the operators, the resulting net can have less (or more) transitions, places, tokens, etc. Suppose we have a GN with high value of some operator for complexity ϕ_i which we want to simplify. By applying the operators to it, we can obtain a sequence of GNs $E, E_1, ..., E_n$ such that

$$E \vdash_{\phi_i} E_1 \vdash_{\phi_i} ... \vdash_{\phi_i} E_n \,. \tag{1}$$

This process can continue until we obtain a GN with one transition and two places, which would be minimal with respect to the operator. However, such GN is not very useful. Therefore, this process must be terminated at some point when the last obtained GN is the most optimal one. Specifying the

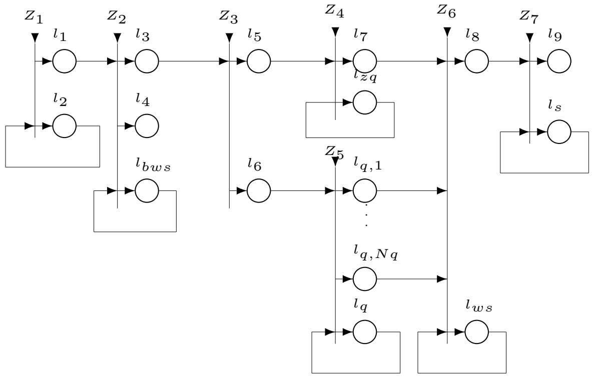


Fig. 1. First GN model of a queuing system.

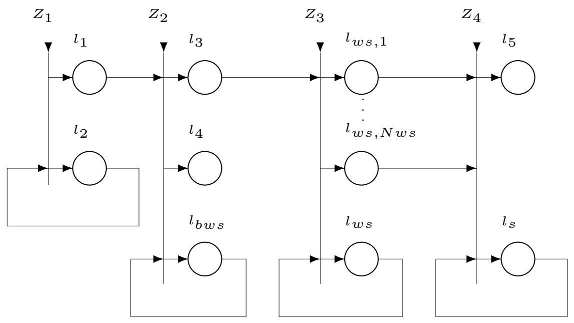


Fig. 2. Second GN model of a queuing system.

optimization criteria, for example the threshold value of the corresponding operator for complexity, is a problem from the methodological aspect of the theory of the GNs. The modeller should determine the optimal number of transitions, places, tokens, etc, of the GN which give the optimal values of the complexity operator (or collection of operators).

A. *First GN model of a queuing system*

To illustrate the optimization of a GN, we consider a queuing system [3] consisting of buffer and server with finite capacities and FIFO (First-In, First-Out) discipline of service of the requests by the server. GN models of queuing systems with various disciplines of service of the requests are described in [14], [15]. A comparison between the GN approach to the conceptual modelling of queuing systems and the Service Systems theory approach is made in [9]. Below we use some of the results presented in these papers.

First, we propose a detailed GN model of a queuing system with graphical representation shown in Fig. 1. It corresponds to a detailed conceptual model of queuing system proposed in [9], which uses elements of Service Systems Theory.

The GN consists of 7 transitions and $14+Nq$ places where Nq is the capacity of the buffer. The transitions represent the following functions of the queuing system:

- Z_1 represents the process of generating of requests.
- Z_2 represents the blocking of the requests when the buffer has reached its capacity.
- Z_3 determines the way of service of the requests, i.e., with waiting or without waiting.
- Z_4 represents the service of the requests without delay, when the server has not reached its capacity.
- Z_5 represents the service of the requests with waiting, when the server has reached its capacity.
- Z_6 represents the function of the buffer of the queuing system.
- Z_7 represents the function of the server of the queuing system.

A special naming system of the important places in which tokens of the GN collect the values of the parameters of the queuing system is used. Six types of tokens are used in the model.

Let E_1 be the GN shown in Fig. 1. Then $\phi_1(E_1) = 7$, $\phi_2(E_1) = 14 + Nq$, $\phi_3(E_1) = 6$, $\phi_7(E_1) = 6$, $\phi_8(E_1) = 24 + 4Nq$, $\phi_9(E_1) = 6$,

$$\phi_{10}(E_1) = \frac{17 + Nq}{24 + 4Nq},$$

$\phi_{11}(E) = 6$, $\phi_{12}(E) = 5$.

B. *Second GN model of a queuing system*

The GN E_1 represents the most detailed representation of a queuing system with FIFO discipline of service of the requests. Here, we modify this model by substituting the three transitions Z_3, Z_4 and Z_5 with a single transition which represents the function of the buffer. The new GN is shown in Fig. 2.

Transitions Z_1 and Z_2 are the same as in the first GN model. Transitions Z_3 and Z_4 are different compared to the first GN model. Transition Z_3 represents the function of the buffer. Transition Z_4 represents the function of the server. The waiting places of the buffer are represented by places $l_{ws,1}, l_{ws,2}, ..., l_{ws,Nws}$, where Nws is the buffer capacity and place l_{ws} is used to store the values of the parameters of the buffer device. Four types of tokens are used in the model.

Let E_2 be the GN described above. Then we have: $\phi_1(E_2) = 4$, $\phi_2(E_2) = 8 + Nws$, $\phi_3(E_2) = 4$, $\phi_7(E_2) = 4$, $\phi_8(E_2) = 12 + 4Nws$, $\phi_9(E_2) = 4$,

$$\phi_{10}(E_2) = \frac{8 + 2Nws}{10 + 4Nws},$$

$\phi_{11}(E_2) = 6$, $\phi_{12}(E_2) = 5$. Since $Nq = Nws$, because the buffer capacity is the same in both GNs, we obtain $E_1 \vdash_{\phi_i} E_2$ for $i = 1, 2, 3, 7, 8, 9, 11, 12$.

C. *Third GN model of a queuing system*

In the previous two GN models, the transitions representing the function of the buffer have one place for every waiting place of the buffer. For queuing systems with low buffer capacity this is a convenient representation, especially with regard to the graphical representation. For queuing systems with large buffer capacities (or infinite) that is not optimal

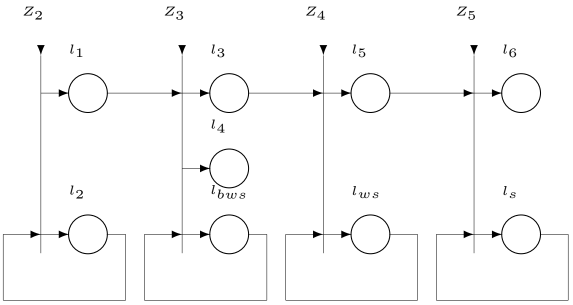


Fig. 3. Third GN model of a queuing system.

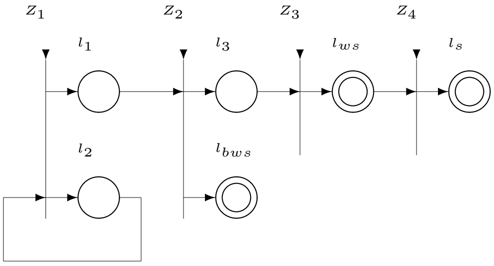


Fig. 4. Fourth GN model of a queuing system.

representation. It is possible to substitute the waiting places with one place, as in the third GN model shown in Fig. 3.

Let E_3 be the GN shown in Fig.3. The operators for complexity have the following values: $\phi_1(E_3) = 4$, $\phi_2(E_3) = 9$, $\phi_3(E_3) = 4$, $\phi_7(E_3) = 4$, $\phi_8(E_3) = 16$, $\phi_9(E_3) = 4$, $\phi_{10}(E_3) = 5/8$, $\phi_{11}(E_3) = 6$, $\phi_{12}(E_3) = 5$. Therefore, we have $E_2 \vdash_{\phi_i} E_3$ for $i = 1, 2, 3, 7, 8, 9, 11, 12$.

D. Fourth GN model of a queuing system

The third GN model can be further optimized if we use the extension of the ordinary GNs – Generalized Nets with Characteristics of the Places (GNCP, [11]). The places which can obtain characteristics are presented with two concentric circles in the graphical representation of the net in Fig. 4. Now, only one type of tokens is used.

Let E_4 be the GN shown in Fig. 4. Then, we have: $\phi_1(E_4) = 4$, $\phi_2(E_4) = 6$, $\phi_3(E_4) = 1$, $\phi_7(E_4) = 1$, $\phi_8(E_4) = 6$,$\phi_9(E_4) = 1$, $\phi_{10}(E_4) = 1/2$, $\phi_{11}(E_4) = 6$, $\phi_{12}(E_4) = 6$. In this case, the relations $E_3 \vdash_{\phi_i} E_4$ for $i = 1, 2, 3, 7, 8, 9, 10, 11$.

The last GN is the optimal representation of a queuing system, in the sense that it has the least acceptable number of transitions and places. It is possible to further reduce the number of transitions and places but some of the concepts of the queuing system will not be presented. The text conceptual description of a queuing system has 4 concepts which must be presented in the graphical representation of the GN model. These are: generator, blocked waiting requests branch, buffer and server. All of them are presented in the fourth GN.

V. Conclusion

The conceptual model optimization needs appropriate indicators of quality. They have to be objective and evaluated predominantly automatically, if we want to design optimization algorithms performed by computer. Most of the existing indicators are subjective. In [7], all of the 10 proposed conceptual modeling evaluation criteria are subjective. The proposed indicators here are suitable for computer evaluation.

The operators for complexity and the relations defined over GNs with respect to these operators are a base for conceptual optimization of GN models. The operators for complexity and the relations over GNs should be generalized to allow comparison of conceptual models based on the GNs theory with conceptual models in Service Systems Theory.

The comparison of conceptual models presented in different languages, e.g., comparison of conceptual models based on the GNs theory with conceptual models in Service Systems Theory is an extremely challenging task.

References

[1] C.-A. Petri, *Kommunication mit Automaten.* Ph.D. Thesis, Univ. of Bonn, 1962.; Schriften des Inst. fur Instrument. Math., No. 2, Bonn, 1962.

[2] C. F. Eick, A Methodology for the Design and Transformation of Conceptual Schemas, *Proceedings of the 17th International Conference on Very Large Databases.*Barcelona, September, 1991, 25-34.

[3] G. Giambene, *Queuing Theory and Telecommunications*, Springer US, 2nd Edition, 2014, https://dx.doi.org/10.5555/1205907.

[4] K. Atanassov, *Generalized Nets*, World Scientific, Singapore, London, 1991, http://dx.doi.org/10.1142/1357.

[5] K. Atanassov, *On Generalized Nets Theory*, Prof. M. Drinov Academic Publ. House, Sofia, 2007.

[6] K. Mehmood, S. Cherfi, I. Comyn-Wattiau, *Data quality through conceptual model quality - reconciling researchers and practitioners through a customizable quality model.* Published in ICIQ 2009 (http://mitiq.mit.edu/ICIQ/Documents/IQ%20Conference%202009/Papers/2-C.pdf)

[7] M. L. Loper, L. G. Birta, G. Arbez, Lessons from a conceptual modeling exercise. *Proceedings of the 2012 Winter Simulation Conference.*C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A.M. Uhrmacher, eds. 978-1-4673-4780-8/12/ ľ2012 IEEE, http://dx.doi.org/10.1109/WSC.2012.6465215

[8] O. Oren, *A method for optimization of a conceptual model,*1984 IEEE First International Conference on Data Engineering, Los Angeles, CA, USA, 1984, 126–132, http://dx.doi.org/10.1109/ICDE.1984.7271264

[9] S. Poryazov, V. Andonov, E. Saranova, *Comparison of Four Conceptual Models of a Queuing System in Service Networks*, Proc. of the 26th National conference with international participation TELECOM 2018, Sofia, 25-26 October, 2018, 71-77.

[10] S. Robinson, *Conceptual Modelling: Who Needs It?*, SCS M&S Magazine - 2010 / n2 (April)

[11] V. Andonov, K. Atanassov, Generalized nets with characteristics of the places, *Compt. rend. Acad. bulg. Sci.*, vol 66, 12, 2013, 1673–1680.

[12] V. Andonov, Reduced generalized nets with characteristics of the arcs,*Issues in Intuitionistic Fuzzy Sets and Generalized Nets*, vol 14, 2018/19, 25–35.

[13] V. M. Vishnevskiy, *Theoretical foundations of computer networks planning*, Tehnosfera, Moscow, 2003. (in Russian)

[14] Z. Tomov, M. Krawczak, V. Andonov, E. Dimitrov, K. Atanassov, *Generalized net models of queueing disciplines in finite buffer queueing systems*, Proceedings of the 16th International Workshop on Generalized Nets, Sofia, 10 February, 2018, 1-9.

[15] Z. Tomov, M. Krawczak, V. Andonov, K. Atanassov, S. Simeonov, *Generalized net models of queueing disciplines in finite buffer queueing systems with intuitionistic fuzzy evaluations of the tasks*, Notes on Intuitionistic Fuzzy Sets, Vol 25, 2019, No 2, 115-122, https://doi.org/10.7546/nifs.2019.25.2.115-122.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 227–236 DOI: 10.15439/2020F2 ISSN 2300-5963 ACSIS, Vol. 21

Optimization of Retrieval Algorithms on Large Scale Knowledge Graphs

Jens Dörpinghaus*†, Andreas Stefan†
* German Center for Neurodegenerative Diseases (DZNE), Bonn, Germany, Email: jens.doerpinghaus@dzne.de
† Fraunhofer Institute for Algorithms and Scientific Computing, Schloss Birlinghoven, Sankt Augustin, Germany

***Abstract*—Knowledge graphs have been shown to play an important role in recent knowledge mining and discovery, for example in the field of life sciences or bioinformatics. Although a lot of research has been done on the field of query optimization, query transformation and of course in storing and retrieving large scale knowledge graphs the field of algorithmic optimization is still a major challenge and a vital factor in using graph databases. Few researchers have addressed the problem of optimizing algorithms on large scale labeled property graphs. Here, we present two optimization approaches and compare them with a naive approach of directly querying the graph database. The aim of our work is to determine limiting factors of graph databases like Neo4j and we describe a novel solution to tackle these challenges. For this, we suggest a classification schema to differ between the complexity of a problem on a graph database. We evaluate our optimization approaches on a test system containing a knowledge graph derived biomedical publication data enriched with text mining data. This dense graph has more than 71M nodes and 850M relationships. The results are very encouraging and – depending on the problem – we were able to show a speedup of a factor between 44 and 3839.**

ALTHOUGH graph databases are a new field with constantly emerging technologies often missing common standards (like query languages) a lot of research has been done on the field of query optimization, query transformation and of course in storing and retrieving large scale knowledge graphs. While current state of the art systems often use RDF data models which are a collection of nested graphs and SPARQL queries the field is now driven by labeled property graphs to overcome their serious limitations. For example nodes and edges have no internal structure which does not allow complex queries like subgraph matchings or traversals and it is not possible to uniquely identify instances of relationships which have the same type, see [1].

Here, we will present research on a more general topic related to large-scale optimization in parallel and distributed computational environments: Optimization of graph algorithms using queries to communicate with a graph database backend. We present two optimization approaches and compare them with a naive approach of directly querying the graph database.

The topic of graph algorithms and their applications is widely studied in computer science and discrete mathematics. Using a graph database as data backend, graph algorithms rely on the robustness and velocity of the underlying system. This is according to our knowledge a still unconsidered topic. We will focus on a particular graph database system (Neo4j) and consider the optimization of graph algorithms an dense large scale labeled property graphs with more then 71M nodes and 850M edges. They are based on biomedical knowledge graphs, see [2].

Communication with the database system might either be a complex query involving heuristics (like "give me all paths from node a to b) or a simple query asking for a data set (like "give me all neighbors of node a) which are usually considered to take $\mathcal{O}(1)$ time.

As a naive approach, we might expect that the runtime will not change using a graph database backend. If we want to find shortest paths between two nodes a and b, we can rely on a build in function. We found, that for some nodes the database backend crashed due to insufficient memory. As a second try, we can use more simple queries. For example Dijkstra's algorithm is well known to have a time complexity of $\mathcal{O}(m + n \cdot \log(n))$ given a graph $G = (V, E)$ with $|V| = n$ and $|E| = m$, see [3]. Here, we only need to retrieve the whole set of nodes and regularly the neighborhood of nodes and the weight of edges. Although these retrievals are considered to take $\mathcal{O}(1)$ time we have serious time problems to retrieve a dataset of 71M nodes using the Neo4j API. Using the graph database adds a factor based on a complex clew containing database efficiency, memory and computing power, connection speed and much more.

This little example above illustrates, that the usage of graph databases has serious algorithmic challenges not covered by computing complexity. The underlying challenges are related and not limited to query optimization, scaling and sharding technologies for databases and parallel algorithms. We will give an overview about this and other related work as a state of the art in the first section. After that, the second section will give a brief overview about the background, infrastructure, data and research questions to solve. A novel, generic schema to categorize algorithms on graphs is presented in the third section. Here, we point at those candidates, where we need optimization approaches. The next section introduces three approaches to optimize graph queries. The fifth section presents an evaluation of these optimization strategies. After that, we will discuss the results and finish with conclusion and outlook.

I. Related work

It is obvious, that graph databases show different query times on different situations and there is a considerable amount of literature on that topic. For example an analyses of

Neo4j and the performance of queries was done by [4]. They show that there are difference in performance under different scenarios and they suggest query performance optimization for business applications. A review on storing big graphs in graph databases and their comparison is published by [1]. They conclude: "Graph data management has attracted immense research attention though it has escaped strong foundations of designing paradigms for storage and retrieval. With growth and change in data with time, the need to identify patterns and semantics becomes difficult." We will present some recent related work which underlines this statement.

A lot of research has been done with respect to analyses and optimization of graph queries, especially with focus on Cypher and Neo4j. Hölsch and Grossniklaus [5] focus on an algebraic query transformation without the usage of a relational database system to process graph data. [6] conclude, that there is a very confusing situation, it is "an unforeseen race of developing new task specific graph systems, query languages and data models, such as property graphs, key-value, wide column, resource description framework (RDF)". They focus on Gremlin, which is a graph traversal language and machine to support multiple graph systems. They suggest a graph pattern matching for Gremlin queries supporting multiple graph data models. [7] discuss issues of interoperability and optimization of queries between RDF and property graphs. They conclude, that more standards need to be developed. [8] questions about the general problems of knowledge graphs: "Although graph databases are conceived schema-less, additional knowledge about the data's structure and/or semantics is beneficial in many graph database management tasks, from efficient storage, over query optimization, up to data integration." This is very plausible and we will highlight this possible pitfall in our work.

A second topic in research is the technical optimization of the database. [9] address the graph query problem on large networks by decomposing shortest paths around vertex neighborhood as basic indexing unit. This was found superior to GraphQL. For Neo4j there are also several approaches. [10] suggest a throughput optimization called in-graph batching which outperform standard Neo4j for large datasets. This is a similar approach to [11] who extended traditional lazy evaluation towards query batching while the application is executed. They noticed, that usually the communication, retrieval and storing of data is a crucial factor reducing the execution time of applications. This is exactly, what we noticed in our introduction example. Other approaches have been proposed by [12] or [13].

Finally, a third way of optimization has to be mentioned. In the context of GIS graph databases [14] try to optimize the heuristic for shortest paths. While also noticing the increasing time complexity for large graphs they tried to solve the problem using filters and adjusting the algorithms. These limitations were also found by [15] while discussing the Frequent Subgraph Mining (FSM) task. Their novel TKG algorithm is also bound in size of the substructures analyzed in the graph database. One of the major drawbacks of these studies is that they focus on single problems in a very specific environment. There is still a considerable uncertainty with regard to algorithms and heuristics from a graph theoretical background when applying to graph databases.

II. Background

Using graph structures to house data has several advantages for knowledge extraction in life sciences and biological or medical research. Here, questions come from the field of exploring the mechanisms of living organisms and gaining a better understanding of underlying fundamental biological processes of life. In addition systems biology approaches, such as integrative knowledge graphs, are important as a holistic approach towards disease mechanism. In addition, pathway databases play an important role. As a basis, biomedical literature and text mining are used to build knowledge graphs, see [2]. In addition relational data from domain specific languages like BEL are widely applied to convert unstructured textual knowledge into a computable form. The BEL statements that form knowledge graphs are semantic triples that consist of concepts, functions and relationships [16]. In addition, several databases and ontologies can implicitly form a knowledge graph. For example Gene Ontology, see [17] or DrugBank, see [18] or [19] cover a large amount of relations and references to which reference other fields.

In [20] we collected 27 real world questions and queries in scientific projects to test the performance and output of the knowledge graph. We could show, that the performance of several queries was very poor and some of them even did not terminate. In order to identify limitations and understand the underlying problems, we carried on our work. The testing system is based an Neo4j and holds a dense large scale labeled property graph with more then 71M nodes and 850M edges. They are based on biomedical knowledge graphs as described in [2].

III. Classification of Problems

There seems to be no generally established procedure for categorizing graph-based queries. What we know about graph queries is largely based on six sources that categorize graph queries or describe them according to different criteria. The contents of this work and the results of the criteria are presented in this section.

[21] examines various theoretical classes of graph query languages with respect to the possible expressions and the complexity of evaluating queries. However, the study is not based on the property graph model, but on a simpler model with a finite directed graph with edge labels. In their analysis they show that for current graph databases, including Neo4j, there is a lack of a language with clear syntax and semantics. They claim that this is a difficulty to evaluate the expressiveness and computational effort of possible queries.

(1) [22] describe graph queries that are considered relevant on the basis of the author's literature research and can be divided into the following four categories:

- Adjacency queries

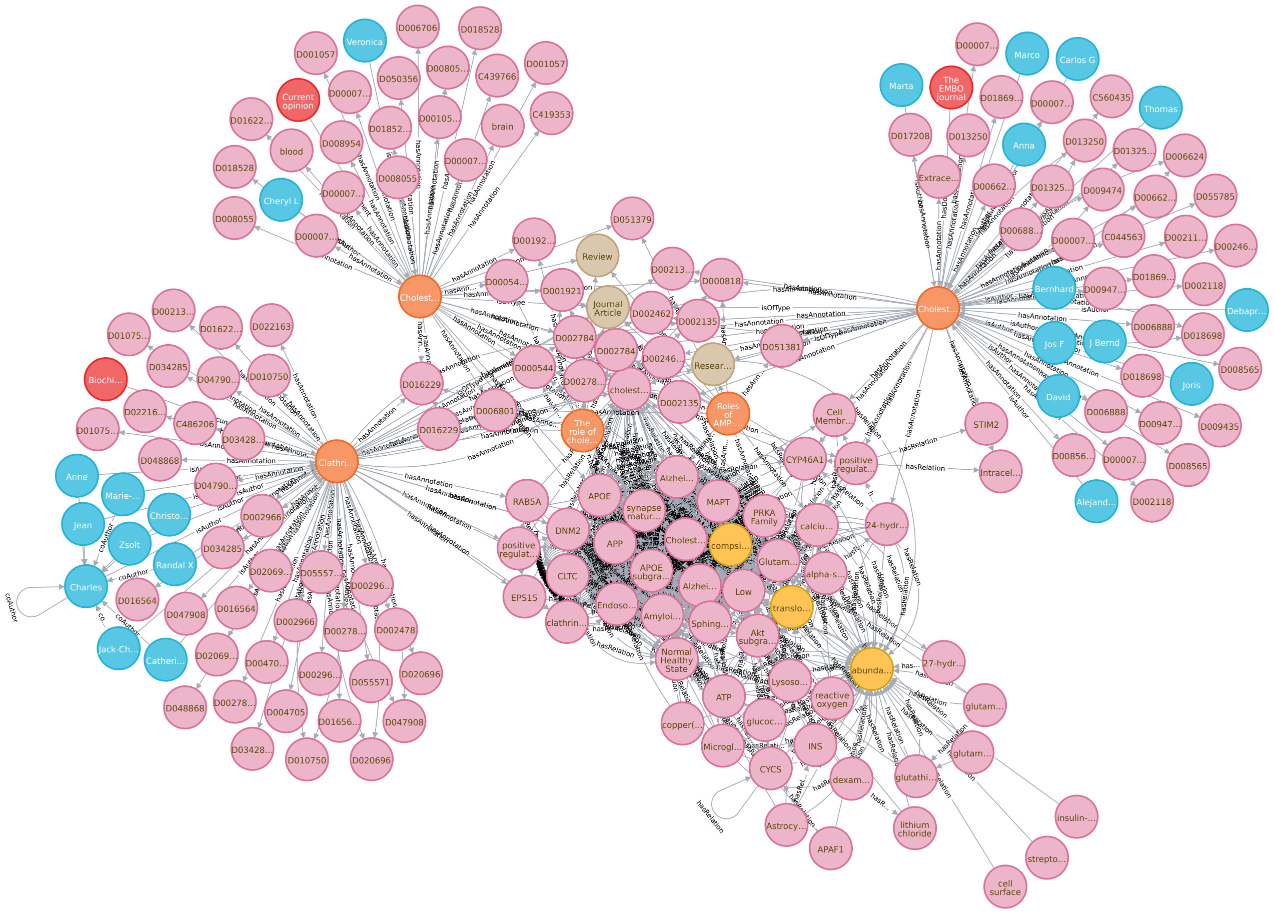


Fig. 1: A subgraph of the large scale biomedical knowledge graph. We can see three orange nodes indicating documents with their context: authors (blue), journal (red) and entities from both keywords and named entity recognition. There are many BEL relations found between single entities which in addition have relations to other documents and biological functions (yellow).

Adjacency queries check whether two nodes are connected or in the *k neighborhood* of each other.

- Reachability queries
  Accessibility queries check whether a node can be reached via a fixed-length path or via a simple regular path and which is the shortest path between the nodes.
- Pattern Matching queries
  Pattern matching consists of finding all subgraphs of a graph that are isomorphic to a pattern graph.
- Summarization queries
  These types of queries are based on functions that allow the results of the queries to be summarized, usually returning a single value. These include functions such as average, number, maximum, etc. They also include functions for calculating properties of the graph and its elements such as the degree of a node, the minimum, maximum and average degrees in the graph, the length of a path, the distance between two nodes, the diameter of the graph, etc.

(2) [23] divide graph queries into two basic functions: *Graph Patterns*, where a pattern structured as a graph is searched in the database, and *Graph Navigation*, which should find paths of any length. The graph pattern queries can be further restricted by projection, union and difference. The result of a Graph Pattern query is a set of all mappings of variables from the query to constants in the database.

The simplest query in the class of Graph Navigation Queries is wether a certain path exists in the graph. This can be extended by additional restrictions, for example, by allowing only certain edge labels. To do this, a path query can be described in general terms as $P = x \longrightarrow \alpha y$, where α specifies the restrictions. The endpoints x and y can be variables or specific nodes. The best known formalism for representing α is *regular expressions*. Regular expressions allow the concatenation of paths and the application of a union or disjunction of paths. Path queries specified with regular expressions are commonly referred to as *Regular Path Queries (RPQ)*. [23] provide information on the complexity of evaluating RPQs to determine whether a path exists. However, the complexity information for RPQs cannot simply be applied to Cypher. In addition, they show that everal open questions regarding complexity or the graph query language Cypher remain. In contrast to SPARQL, the semantics and complexity of Cypher has not yet been investigated due to the lack of theoretical formalization, see [23] and [24].

(3) [24] describe different classes of queries for several graph query languages, as well as several core functionalities supported by the graph query languages. They also discuss the expressiveness and complexity of query evaluation. Unfortunately, Cypher is not described as a graph query language. The author divides the queries into the following categories:

- *CQ* (conjunctive query)
  A sample query of this type looks for documents that have both the *PublicationType* `Journal Article` and `Review`.
- *RPQ* (regular path query)
  A search is made for a node pair (x, y) so that a path exists between x and y, with the sequence of edge labels following a given pattern. The given pattern is described by a regular expression.
- *CRPQ* (conjunctive regular path queries)
  *CQ*s and *RPQ*s can be combined to form the class *CRPQ*. According to the author, this class serves as a basis for several graph query languages. However, this class is not sufficient for problems where relationships between paths need to be specified.
- *ECRPQ* (extended conjunctive regular path query
  This class extends the *CRPQ*s by the possibility to specify path variables or to allow paths as output of a query.

In addition [24] examine functionalities of graph query languages. They are divided into the following categories:

- Subgraph Matching
  It searches for subgraphs in a graph. This is a *CQ*.
- Find connected nodes by path
  Determining accessibility between nodes in a graph is a graph query that is supported in many graph query languages. The *RPQ*s class includes queries that return all node pairs from a graph that are connected by a path that matches a regular expression.
- Compare and return paths
  It specifies relationships between paths and searches for paths that connect two nodes to find connections in linked data. By providing these two functions, the class of extended *CRPQ*s (*ECRPQ*s) is created.
- Aggregation
  Determining different properties of graphs requires a calculation that goes beyond matching and finding paths. Such properties are for example the determination of node degrees.

(4) Both [25] and [26] consider queries with property graphs and name among others Cypher and Gremlin as important graph query languages. These sources name these categories of graph queries:

- k-hop Queries
  According to the authors, these queries are most common in practice. They include queries such as *find node*, *find the node's neighbors (1-hop query)*, *find edges in multiple hops*, and *get attribute values*.
- subgraph and supergraph queries
- width search / depth search
- Seeking and shortcuts
- Search for strongly connected components
- Regular Path Queries

(5) In [27], queries and graph algorithms are described and subdivided according to different properties. On the one hand, the authors subdivide the queries according to *graph pattern*-based queries for local analysis of the data and according to *graph algorithms*, which often analyze globally and iteratively. Local queries only look at a specific section of the graph like a start node and the surrounding subgraph. This type of query is often used for transactions and pattern-based queries. Graph algorithms typically search for global structures. The algorithm takes the entire graph as input and returns an enriched graph or an aggregated value.

The authors divide different graph algorithms into the three categories *Pathfinding*, *Centrality* and *Community Detection*. The book describes several graph algorithms and assigns them to the categories:

- pathfinding
  - Shortest Paths
  - All Pair Shortest Path
  - Minimum Spanning Tree
  - random walk
- Centrality
  - Degree Centrality
  - Closeness Centrality
  - Betweenness Centrality
  - page rank
- Community Detection
  - Triangle Count
  - (Strongly) Connected Components
  - Label Propagation
  - Louvain Modularity

A. New criteria

In order to categorize graph queries, we introduce new criteria, which were found relevant for the use case evaluated for our knowledge graph:

- Accessing attributes
  How many attributes must be considered when executing the query? Accessing attributes requires reading an additional file and therefore requires more processing power and access time. In section V we will proof, that data stored in attributes will significantly slow down queries.
- Data type of attributes
  What data types are accessed in queries? We expect, that this also influences the runtime.
- Node and edge types to be considered
  Which node and edge types must be considered in the query? Is it only a small subset or is the majority of the types required? Is it possible to decide for all queries

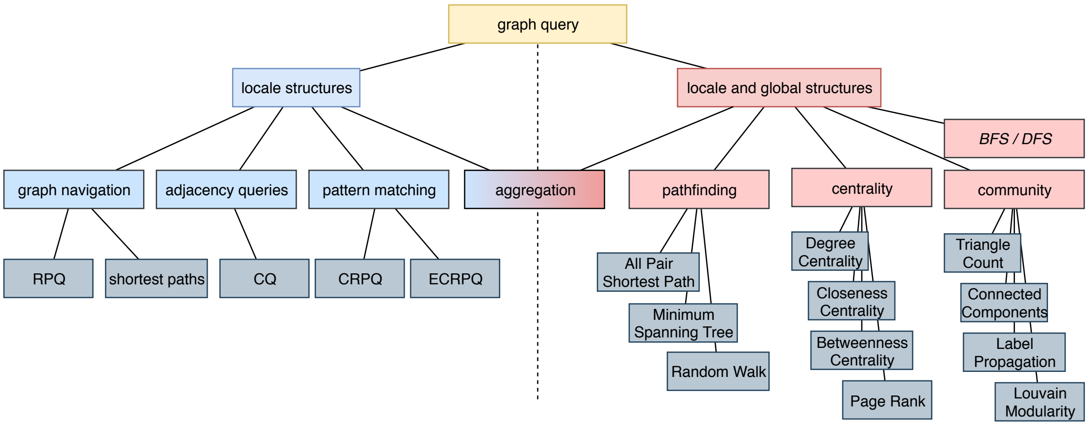


Fig. 2: An overview of the categories for graph queries unified from literature sources. These categories give a first overwiev and a categorization scheme for graph queries and their complexity.

whether and which node and edge types can be exported as subgraphs?

- Entry point
  Does the query rely on a unique node specified for the query (e.g. as a starting point for the search), or is there a general search for pattern between nodes?

Various approaches have been proposed in literature, but we can examine connections and a hierarchy. In the next step, we merge and cluster these approaches in order to create a categorization scheme for graph queries. This is shown in figure 2.

The schema divides the categories for graph queries into *local structures* and *local & global* structures (according to literature source (5)). The second structure category is called *local & global*, because some of the graph algorithms can act locally by specifying a start node or a subgraph. Furthermore, some categories, such as *CRPQ* or *ECRPQ*, were identified as subcategory of other categories. This is illustrated by the hierarchical structure of the categorization scheme. The category *Aggregation* belongs to graph queries that search for both local and global structures. For example, the category *Aggregation* can include questions such as "What is the degree of node A?" or "What is the average of the graph?", the former referring to local and the latter to global structures.

IV. Method

Here, we propose a multi-step optimization approach towards graph queries. Usually, graph queries are executed using a Cypher query. Here, the application or the user directly communicates with the graph database. To optimize this, we suggest that an external algorithm communicates with the graph database and executes only elementary queries. With this, the queries are limited to typical questions like neighborhood, paths and relations. Since all trivial requests (like "give me this node") can usually be handled by common relational or special purpose databases, we suggest a third optimization approach, if necessary. Here, a polyglot persistence approach uses other data sources to execute trivial queries. See figure 3 for an illustration.

A. Pathfinding

In [20] we introduced a large set of queries and categorized them according to the schema discussed in section III. We will start with those problems using in general both locale as well as global structures in the graph. A problem with a very poor performance was graph navigation and pathfinding. These include Regular Path Queries (RPQ, see [23]) (problems 2,11,14,16,17,19,21) and finding shortest paths (problems 4,12). Since the problems of retrieving a single or all shortest paths are quite similar, we will discuss both of them here.

Queries 4 and 12 are both a typical shortest path problem: *What is the shortest way between {Entity1} and {Entity2} and what is on that way?* and *How far apart are {document1} and {document2}?* Thus both problems can be solved using Cypher:

(Q4)
```
match (entity1:Entity {preferredLabel:
"axonal transport"}), (entity2:Entity
{preferredLabel: "LRP3"}) call
algo.shortestPath.stream(entity1, entity2)
yield nodeId return algo.asNode(nodeId)
```

(Q12)
```
match (doc1:Document {documentID:
"PMID:16160056"}), (doc2:Document
{documentID: "PMID:16160050"}) call
algo.shortestPath.stream(doc1,doc2) yield
nodeId return algo.asNode(nodeId)
```

Both queries rely on the function `shortestPath` available in Neo4j. Both Bellman–Ford and Dijkstra's algorithm are known to solve this problem for weighted graphs. For unweighted graphs a modified Breadth-first search will solve this issue in $\mathcal{O}(E+V)$ [28]. Other algorithms like Dijkstra's should be faster, for example using binary heaps the time complexity is $\mathcal{O}(m + n \cdot \log(n))$ given a graph $G = (V, E)$ with $|V| = n$ and $|E| = m$, see [3]. According to Neo4j

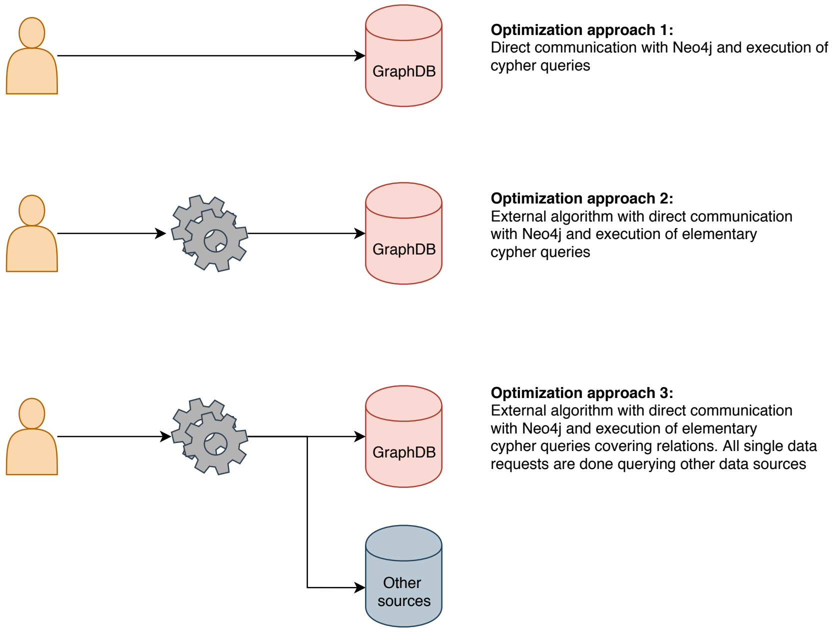

GraphDB
Optimization approach 1:
Direct communication with Neo4j and execution of cypher queries
GraphDB
Optimization approach 2:
External algorithm with direct communication with Neo4j and execution of elementary cypher queries
GraphDB
Optimization approach 3:
External algorithm with direct communication with Neo4j and execution of elementary cypher queries covering relations. All single data requests are done querying other data sources
Other sources

Fig. 3: An overview of the optimization approaches discussed in this paper. The first approach contains the basic Cypher query, the second approach transfers the algorithm to a different system. The third approach relies on a polyglot persistence architecture and excludes all time-consuming queries that can be answered by a key-value store.

documentation, the build in function `shortestPath` uses Dijkstra's algorithm[1].

With algorithm 1 we suggest a BFS-approach to tackle the shortest-path problems. Given both a starting node s and an ending node e, the only communication with the graph database is done in line 18. Here, the neighborhood of a node is retrieved.

This algorithm implements the optimization approach 2. Since no other data sources are needed, optimization approach 3 will not improve this query.

B. CRPQ

Several questions introduced in [20] are conjunctive regular path queries (CRPQ, see [24]). These are pattern matching problems using locale structures within the graph. Some of them are quite simple. For example query 15 – *How many sources are there for the statements of a contradictory BEL statement?* – can be easily translated into Cypher:

(Q15)
```
match (e1:Entity) -[r1:hasRelation
{function:"increases"}]->
(e2:Entity), (e1) -[r2:hasRelation
{function:"decreases"}] -> (e2)
return distinct e1.preferredLabel,
e2.preferredLabel, count(r1) as
`increases`, count(r2) as `decreases`
order by count(r1) desc
```

This query matches two contradicting relations, their numbers and returns a decreasing sorted list. More complex is the example query 1: *Which author was the first to state that {Entity1} has an enhancing effect on {Entity2}?* A Cypher query solving this uses several node attributes, for example the publication date to sort the result set:

(Q1)
```
match (n:Entity preferredLabel:
"APP") -[r:hasRelation function:
"increases"]-> (m:Entity preferredLabel:
"gamma Secretase Complex"), (doc:Document
documentID: r.context) <-[r2:isAuthor]-
(author:Author) return doc, author order
by doc.publicationDate limit 1'
```

As a first optimization approach denoted by opt1 we exclude the sorting functions from the queries and do this manually. This leads to the following two queries:

(Q1-1)
```
match (n:Entity preferredLabel:
"APP")-[r:hasRelation function:
"increases"]->(m:Entity preferredLabel:
"gamma Secretase Complex") return n,r,m
```

(Q15-1)
```
match (e1:Entity) -[r1:hasRelation
function:"increases"]-> (e2:Entity), (e1)
-[r2:hasRelation function:"decreases"]->
(e2) return distinct e1.preferredLabel,
e2.preferredLabel
```

The algorithm for query 1 can be found in 2, the algorithm for query 15 in 3. As we can see, query 1 is more complex, since it includes the retrieval of node attributes, the publication data. Both algorithms include the sorting of lists.

The second optimization approach can only be applied to query 1. Here, we try to retrieve the node attributes from a dedicated information system. This is related to the polyglot

[1] See https://neo4j.com/docs/graph-algorithms/current/labs-algorithms/shortest-path/.

Algorithm 1 GRAPH-BFS

```
Require: two nodes $s, e \in V$
Ensure: shortest path $p = [s, ..., e]$
    $Q$ = []
    $discovered = [s]$
    $Parent =$
    $Q.append(s)$
    while $len(Q) > 0$ do
        $v = Q.pop(0)$
        if $getNode(v) == e$ then
            $x = v$
            $path = [v]$
            while $Parent[x]! = s$ do
                $x = Parent[x]$
                $path.append(x)$
            end while
            $x = Parent[x]$
            $path.append(x)$
            $returnpath$
        end if
        $N = getNeighbours(v)$
        for $winN$ do
            if $w$ not in $discovered$ then
                $discovered.append(w)$
                $Parent[w] = v$
                $Q.append(w)$
            end if
        end for
    end while
    return  $d$ with max ($pd$)
```

Algorithm 2 QUERY1-OPT1

```
Require: Documents $D = \{d_1, ..., d_n\}$ obtained from query
    (Q1-1)
Ensure: Document $d$
    $pd = []$
 2: for every $d \in D$ do
        $pd$.add ($d$,$d$.publicationdate)
 4: end for
    return  $d$ with max ($pd$)
```

Algorithm 3 QUERY15-OPT1

```
Require: Data points $T = \{t_1, ..., t_n\}$ with $t_i =$
    $\{e1_i, e2_i, inc_i, dec_i\}$ obtained from query (Q15-1)
Ensure: Sorted data points $T$
    return  sort($T$)
```

persistence approach introduced in [20]. Here, we suggest to retrieve this value direct from the SCAIView API.

(Q1-2)
```
match (n:Entity preferredLabel:
"APP")-[r:hasRelation function:
"increases"]->(m:Entity preferredLabel:
"gamma Secretase Complex") return n,r,m
```

Here, algorithm QUERY1-OPT2 will use a different function to add the publicationdate in line 3.

V. EVALUATION

We evaluate our optimization approaches on a test system containing a knowledge graph derived biomedical publication data enriched with text mining data and domain specific language data using BEL, see [20]. This dense graph has more than 71M nodes and 850M relationships.

The testing system run Neo4j Community 3.5.8. on a server with 16 Intel Xeon CPUs with 3GHz and 128GB main memory. We applied several approaches described in the chapter "Performance" in the Neo4j Operations Manual[2].

A. Pathfinding

Both queries 4 and 12 are pathfinding problems. To retrieve the shortest path, we suggested the execution of a Cypher query using the build in `shortestPath` algorithm. Applying optimization strategy 1, we suggest the usage of a BFS-approach called GRAPH-BFS.

Contrary to expectations, build in Dijkstra's algorithm performs very poor. The runtime lay between 40 and 60 minutes, the average runtime was 2390.44 seconds. In contrast the BFS-approach had a runtime of 1-2 seconds, the average runtime was 1.65 seconds. This is a speedup factor of 1453, see figure 4.

These results could also be reproduced on Query 12, see figure 5. The average runtime of `shortestPath` is 567.44 seconds, approximating 10 minutes. The average runtime of the BFS-approach is 0.14 seconds. This is a speedup factor of 3838.77, see figure 5.

These results highlighted that the `shortestPath` function cannot be used for large scale knowledge graphs due to the runtime. Unexpectedly, the simple BFS-approach utilizing our first optimization strategy decreases the runtime nearly by the factor 3840. Further analysis showed that the speedup is highly influenced by node degree. Nevertheless, `shortestPath` is unacceptable for information systems with a user frontend.

B. CRPQ

We had a more simple query (15) and a more complex query (1). Regarding QUERY15, we could only implement our first optimization approach QUERY15-OPT1. Figure 6 presents the runtime data. The average runtime of QUERY15 is 8.6 seconds, the average runtime of QUERY15-OPT1 is 8.4 seconds. As we can see, there is no real advantage in applying the optimization approach here. In general both heuristics are competitive, while the simple Cypher query has some situations where it is significantly slower. Although no significant differences were found, the optimization approach shows a rather constant runtime.

The most striking results are obtained with more complex queries. The situation changes significantly when analyzing query 1. Here, the Cypher query QUERY1 usually has an execution time of about 7 or 8 minutes, the average runtime

[2] See https://neo4j.com/docs/operations-manual/current/performance/.

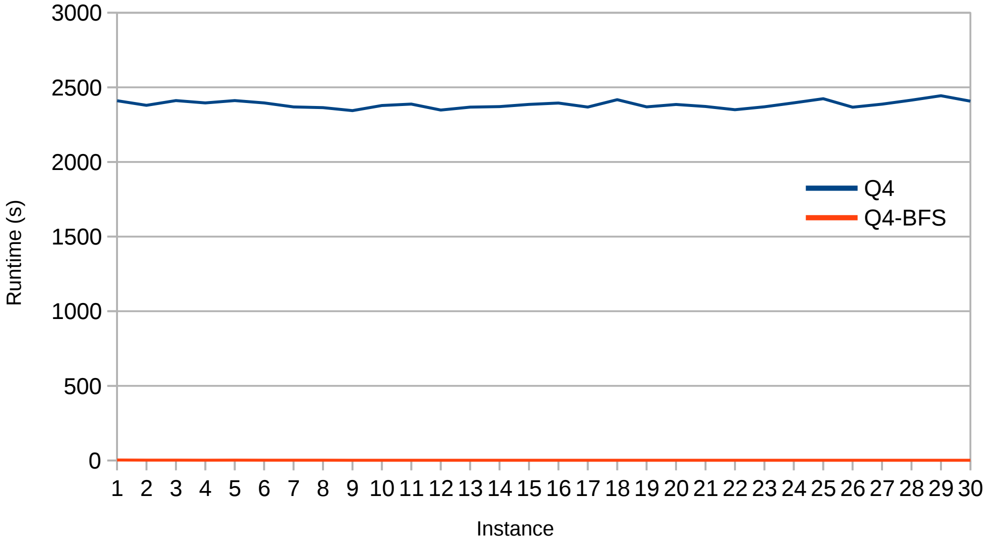


Fig. 4: Results for query 4 QUERY4 (average runtime 2390.44 seconds) and the optimization approach 1 GRAPH-BFS (average runtime 1.65 seconds). The speedup factor is 1453.

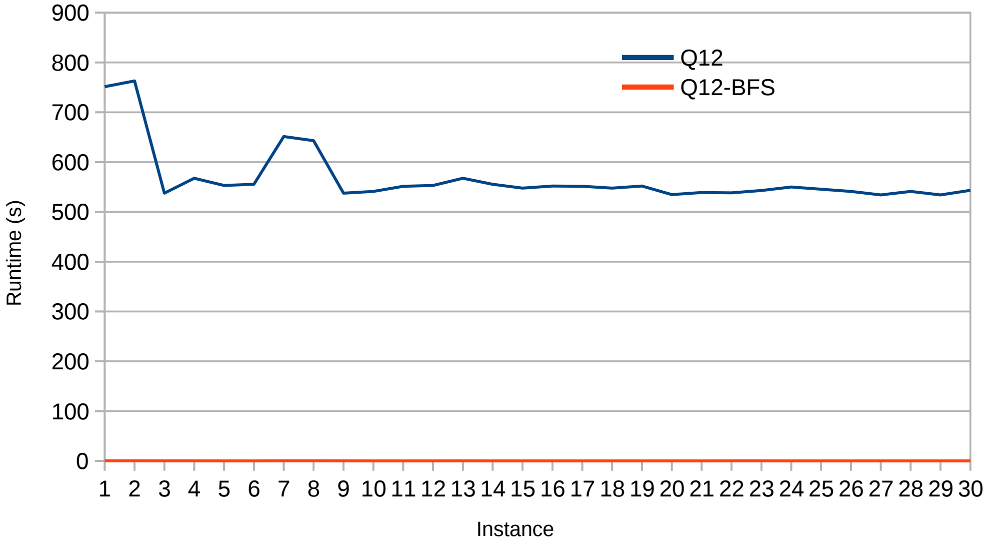


Fig. 5: Results for query 12 QUERY12 (average runtime 567.44 seconds) and the optimization approach 1 GRAPH-BFS (average runtime 0.14 seconds). The speedup factor is 3838.77.

is 364.45 seconds. Using the optimization approach 1, the execution time of QUERY1-OPT1 reduces to 1-2 minutes, the average runtime is 80.2 seconds. Thus, the runtime decreases by the factor 4.5. Using an polyglot persistence approach and querying SCAIView for the metadata, the execution time of QUERY1-OPT2 once again decreases to more or less 10 seconds, in average 9.6 seconds. Here, the runtime decreases by the factor 9,6 compared with QUERY1-OPT1 and by the factor 43.8 compared with QUERY1, see figure img:q1.

It is important to note, that simple queries like Q15 cannot be improved very easy. Graph databases are highly optimized to retrieve relations. But our technique shows a clear advantage over simple Cypher queries when multiple relations are queried, functions for sorting or other purposes are called and especially when single nodes or edges are called to retrieve metadata. Neo4j shows no good performance when used as a key-value store.

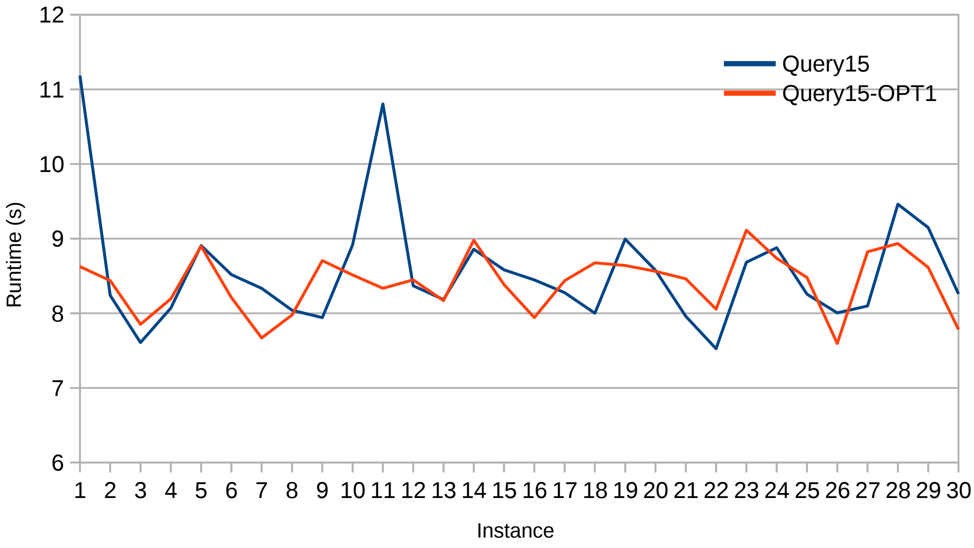


Fig. 6: Results for query 15 QUERY15 (average runtime 8.6 seconds) and the optimization approach 1 QUERY15-OPT1 (average runtime 8.4 seconds).

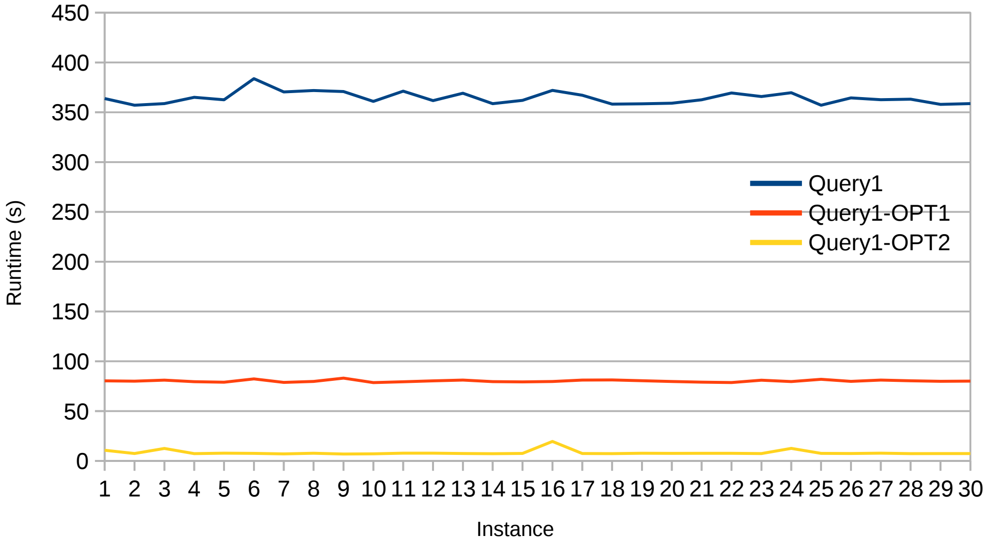


Fig. 7: Results for query 1 QUERY1 (average runtime 364.45 seconds) and the optimization approaches 1 QUERY1-OPT1 (average runtime 80.2 seconds) and 2 QUERY1-OPT2 (average runtime 9.6 seconds). In total the speedup factor is 43.8.

VI. Conclusion and Outlook

In this paper we presented two new approaches for query optimization on large scale knowledge graphs using graph databases. Knowledge graphs have been shown to play an important role in recent knowledge mining and discovery. A *knowledge graph* (sometimes also called a *semantic network*) is a systematic way to connect information and data to knowledge on a more abstract level compared to language graphs.

We used three approaches to compare our optimization strategies to state-of-the-art Cypher queries. Our goal was to reach the best optimization level without changing the underlying graph database. We believe this solution will aid researchers without a technological background to effectively improve their queries.

Our experiments showed that the proposed optimization strategies can effectively improve the performance by excluding those parts of queries with the highest runtime. Especially the retrieval of single entities like nodes and edges, but also the usage of functions like sorting or shortest paths have been detected for decreasing the execution time significantly.

Graph databases are highly efficient and optimized for storing and retrieving relations between data points. Thus, we propose to review graph queries carefully and check, if heuristics can be used to merge those parts of a query that are very fast in graph databases. Thus it is an important step to provide a deeper understanding of the underlying graph structures. We could show that most graph queries categorized as locale structures cannot be executed efficiently out of the box: graph navigation and pattern matching. Only adjacency queries seem to perform very good.

Although this is a good step towards a better understanding of the underlying problem field, it does not help to find a general solutions to optimize graph queries. Improving the runtime of graph queries needs a careful understanding and improving of the heuristics.

Our future work includes optimization approaches for federated queries on multiple data sources and better understanding of those cases, where optimization approaches are feasible and lead to a significant improvement of execution time. In addition we plan to evaluate our results with other graph databases like OrientDB.

VII. Acknowledgments

We thank Marc Jacobs for valuable suggestions, Bruce Schultz for his technical help and both for carefully revising the manuscript.

This manuscript has been supported by Fraunhofer Society under the MAVO Project; Human Brain Pharmacome.

References

[1] M. Desai, R. G Mehta, and D. P Rana, "Issues and challenges in big graph modelling for smart city: An extensive survey," *International Journal of Computational Intelligence & IoT*, vol. 1, no. 1, 2018.

[2] J. Dörpinghaus and A. Stefan, "Knowledge extraction and applications utilizing context data in knowledge graphs," in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019, pp. 265–272.

[3] D. B. Johnson, "Efficient algorithms for shortest paths in sparse networks," *Journal of the ACM (JACM)*, vol. 24, no. 1, pp. 1–13, 1977.

[4] H. Huang and Z. Dong, "Research on architecture and query performance based on distributed graph database neo4j," in *2013 3rd International Conference on Consumer Electronics, Communications and Networks*. IEEE, 2013, pp. 533–536.

[5] J. Hölsch and M. Grossniklaus, "An algebra and equivalences to transform graph patterns in neo4j," in *EDBT/ICDT 2016 Workshops: EDBT Workshop on Querying Graph Structured Data (GraphQ)*, 2016.

[6] H. Thakkar, D. Punjani, S. Auer, and M.-E. Vidal, "Towards an integrated graph algebra for graph pattern matching with gremlin," in *International Conference on Database and Expert Systems Applications*. Springer, 2017, pp. 81–91.

[7] R. Angles, H. Thakkar, and D. Tomaszuk, "Rdf and property graphs interoperability: Status and issues," in *Proceedings of the 13th Alberto Mendelzon International Workshop on Foundations of Data Management, Asunción, Paraguay*, 2019.

[8] S. Mennicke, "Modal schema graphs for graph databases," in *International Conference on Conceptual Modeling*. Springer, 2019, pp. 498–512.

[9] P. Zhao and J. Han, "On graph query optimization in large networks," *Proceedings of the VLDB Endowment*, vol. 3, no. 1-2, pp. 340–351, 2010.

[10] J. Eymer, P. Dexter, and Y. D. Liu, "Toward lazy evaluation in a graph database," *SPLASH 2019*.

[11] A. Cheung, S. Madden, and A. Solar-Lezama, "Sloth: Being lazy is a virtue (when issuing database queries)," *ACM Transactions on Database Systems (ToDS)*, vol. 41, no. 2, p. 8, 2016.

[12] A. B. Mathew, "Efficient query retrieval from social data in neo4j using lindex." *KSII Transactions on Internet & Information Systems*, vol. 12, no. 5, 2018.

[13] W. Cabrera and C. Ordonez, "Scalable parallel graph algorithms with matrix–vector multiplication evaluated with queries," *Distributed and Parallel Databases*, vol. 35, no. 3-4, pp. 335–362, 2017.

[14] X. Wu and S. Deng, "Research on optimizing strategy of database-oriented gis graph database query," in *2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS)*. IEEE, 2018, pp. 305–309.

[15] P. Fournier-Viger, C. Cheng, L. J. chuan Wei, U. Yun, and R. U. Kiran, "Tkg: Efficient mining of top-k frequent subgraphs," in *Big Data Analytics: 7th International Conference, BDA 2019, Ahmedabad, India, December 17–20, 2019, Proceedings*, vol. 11932. Springer Nature, 2019, p. 209.

[16] J. Fluck, A. Klenner, S. Madan, S. Ansari, T. Bobic, J. Hoeng, M. Hofmann-Apitius, and M. Peitsch, "Bel networks derived from qualitative translations of bionlp shared task annotations," in *Proceedings of the 2013 Workshop on Biomedical Natural Language Processing*, 2013, pp. 80–88.

[17] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig *et al.*, "Gene ontology: tool for the unification of biology," *Nature genetics*, vol. 25, no. 1, p. 25, 2000.

[18] D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda *et al.*, "Drugbank 5.0: a major update to the drugbank database for 2018," *Nucleic acids research*, vol. 46, no. D1, pp. D1074–D1082, 2017.

[19] K. Khan, E. Benfenati, and K. Roy, "Consensus qsar modeling of toxicity of pharmaceuticals to different aquatic organisms: Ranking and prioritization of the drugbank database compounds," *Ecotoxicology and environmental safety*, vol. 168, pp. 287–297, 2019.

[20] J. Dörpinghaus, A. Stefan, B. Schultz, and M. Jacobs. (2020) Towards context in large scale biomedical knowledge graphs. [Online]. Available: http://arxiv.org/abs/2001.08392

[21] P. Barceló Baeza, "Querying graph databases," *Proceedings of the ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, 2013.

[22] R. Angles, "A Comparison of Current Graph Database Models," in *2012 IEEE 28th International Conference on Data Engineering Workshops*, apr 2012, pp. 171–177.

[23] R. Angles, M. Arenas, P. Barceló, A. Hogan, J. Reutter, and D. Vrgoč, "Foundations of Modern Query Languages for Graph Databases," *ACM Comput. Surv.*, vol. 50, no. 5, pp. 68:1—-68:40, sep 2017. [Online]. Available: http://doi.acm.org/10.1145/3104031

[24] P. T. Wood, "Query Languages for Graph Databases," *SIGMOD Rec.*, vol. 41, no. 1, pp. 50–60, apr 2012. [Online]. Available: http://doi.acm.org/10.1145/2206869.2206879

[25] J. Pokorný, "Functional querying in graph databases," *Vietnam Journal of Computer Science*, vol. 5, no. 2, pp. 95–105, 2018. [Online]. Available: https://doi.org/10.1007/s40595-017-0104-6

[26] J. Pokorny, "Graph Databases: Their Power and Limitations," in *Computer Information Systems and Industrial Management*, K. Saeed and W. Homenda, Eds. Cham: Springer International Publishing, 2015, pp. 58–69.

[27] M. Needham and A. E. Hodler, *Graph Algorithms*. O'Reilly Media, Inc., 2019.

[28] A. Aziz and A. Prakash, *Algorithms for Interviews: A Problem Solving Approach*. algorithmsforinterviews.com, 2010.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 237–240
DOI: 10.15439/2020F15
ISSN 2300-5963 ACSIS, Vol. 21

Ant Colony Optimization Algorithm for Fuzzy Transport Modelling

Stefka Fidanova
IICT, BAS
Sofia, Bulgaria
E-mail: stefka@parallel.bas.bg

Olympia Roeva
IBPhBME, BAS
Sofia, Bulgaria
E-mail: olympia@biomed.bas.bg

Maria Ganzha
SRI, PAS
Warsaw, Poland
E-mail: maria.ganzha@ibspan.waw.pl

***Abstract*—Public transport plays an important role in our live. It is very important to have a reliable service. Up to 1000 km, trains and buses play the main role in the public transport. The number of the people and which kind of transport they prefer is important information for transport operators. In this paper is proposed algorithm for transport modeling and passenger flow, based on Ant Colony Optimization method. The problem is described as multi-objective optimization problem. There are two optimization purposes: minimal transportation time and minimal price. Some fuzzy element is included. When the price is in a predefined interval it is considered the same. Similar for the starting traveling time. The aim is to show how many passengers will prefer train and how many will prefer buses according their preferences, the price or the time.**

I. Introduction

COMFORTABLE transportation from one town to another one is very important. It exists different ways of transportation. The cheaper transport is a railway (excluding the super-fast with velocity more than 200 km/h), but the trains are slower. Buses and fast trains are more expensive, but faster. All this need to be taken in to account, when a transportation model is prepared. In this paper the transportation problem is defined as an optimization problem. It is a multi-objective problem with two objective functions: total time and total price of all passengers. The goal is to minimize the both objective functions. The two objective functions are antithetic, the faster transportation is expensive and the cheaper transportation is slower. Thus when one of the objective functions decreases, the other increases. The problem is multi-objective, therefore is received set of nondominated solutions instead of one optimal solution. The set of solutions is analyzed and the final decision, which solution is optimal accordingly with some additional constraints. The solutions of our problem shows how many passenger will use the train and how many will use bus and fast train.

The oldest public transport, among those that are still in use, is the railroads. Nowdays the main concurrencies of the trains are buses, especially in the regions with highways. Thus the models, which can analyze the passenger flow and its preferences, are important for transportation planning. In our model we include some fuzzy element, thus we try to make it more realistic and close to human thinking.

Various transportation models can be found in the literature [2]. The importance of every of the models depends of its functions. One of the models are concentrated on scheduling [1]. Other models are focused on simulation to analyse the level of utilization of different types of transportation [13]. The model in [10] aims to optimize the transportation network design. In [5] is modeled freeway traffic flow. When a network of freeway is is given , their model can predict the traffic flow with high accuracy. Our model is focused on modeling the passenger flow according their preferences. The fuzzification of the model makes it more realistic, more close to the human thinking. When the price or the time is in some predetermined interval we accept it as the same. The problem shows the distribution of the passenger flow and how it changes when the timetable or type of the vehicles are changed.

The problem is difficult in computational point of view and cannot be solved with traditional numerical methods with reasonable computational resources. It is more appropriate to apply some metaheuristic method on this kind of problems. We apply ant colony optimization algorithm. The model is tested on real problem, the passenger flow between Sofia and Varna, one of the longest destinations in Bulgaria.

The rest of the paper is organized as follows. In section 2 is given an ant colony optimization algorithm. In section 3 the transportation problem is formulated and an ACO algorithm which solves it is proposed. Experimental results are shown and analyzed in Section 4. In section 5 are drawn some concluding remarks and possibilities for future work.

II. Ant Colony Optimization Method

The considered optimization problem (see Section III) is NP-hard, and therefore we consider the use of a metaheuristic search for its solution. Therefore is impractical to be applied some traditional numerical method. Hereof we apply Ant Colony Optimization (ACO) algorithm, one of the best metaheuristics.

The behavior of ants in nature has inspired the creation of this method. Ants put on the ground chemical substance called pheromone, which help them to return to their nest when they look for a food. The ants smell the pheromone and follow the path with a highest pheromone concentration. Thus they find shorter path between the nest and the source of the food.

The ACO algorithm uses a colony of artificial ants that behave as cooperating agents, like ants in the nature. With the help of the pheromone they try to construct better solutions and to optimize them. The problem is represented by a graph and the solution is represented by a path in the graph or by tree in the graph. The graph representation is crucial for the good algorithm performance.

Ants start from random nodes of the graph and try to construct feasible solutions. When all ants construct their solution the pheromone values are updated. Ants compute a set of feasible moves and select the best one, according to the transition probability rule. The transition probability p_{ij}, to choose the node j when the current node is i, is based on the heuristic information η_{ij} and on the pheromone level τ_{ij} of the move, where $i, j = 1, \ldots, n$. α and β shows the importance of the pheromone and the heuristic information respectively.

$$p_{ij} = \frac{\tau_{ij}^{\alpha}\, \eta_{ij}^{\beta}}{\sum\limits_{k \in \{allowed\}} \tau_{ik}^{\alpha}\, \eta_{ik}^{\beta}} \tag{1}$$

The construction of the heuristic information function depends highly of the solved problem. It is appropriate combination of problem parameters and is very important for ants' management. An ant selects the move with highest probability. The initial pheromone is set to a small positive value τ_0 and then ants update this value after completing the construction stage [3], [6], [7]. The search stops when $p_{ij} = 0$ for all values of i and j, which means that it is impossible to include new node in the current partial solution.

The pheromone trail update rule is given by:

$$\tau_{ij} \leftarrow \rho\tau_{ij} + \Delta\tau_{ij}, \tag{2}$$

where $\Delta\tau_{ij}$ is a new added pheromone and it depends of the quality of achieved solution.

The pheromone is decreased with a parameter $\rho \in [0, 1]$. This parameter models evaporation in the nature and decreases the influence of old information in the search process. After that, a new pheromone is included. It is proportional to the quality of the solution (value of the fitness function). Several variants of ACO algorithm exist. The main difference is the pheromone updating.

Multi-Objective Optimization (MOP) begins in the nineteenth century in the work of Edgeworth and Pareto in economics [11]. The optimal solution for MOP is not a single solution as for mono-objective optimization problems, but a set of solutions defined as Pareto optimal solutions. A solution is Pareto optimal if it is not possible to improve a given objective without deteriorating at least another one. The main goal of the resolution of a multi-objective problem is to obtain the Pareto optimal set and consequently the Pareto front. One solution dominates another if minimum one of its components is better than the same component of other solutions and other components are not worse. The Pareto front is the set of non dominated solutions related to the solved problem. After that, the users decide which solution from the Pareto front to use according additional constraints, related with their specific application. When metaheuristics are applied, the goal becomes to obtain solutions close to the Pareto front.

III. Problem Formulation

Various problems arise in the area of long-distance passenger transport with a different kind of transport. One of the problem is optimal scheduling [9], others concern the optimal management of the passenger flow [12]. In some developments, it is involved only one type of vehicle [4]. The common is that all they are difficult in computational point of view.

Our problem concerns passengers traveling in a same direction, covered with several different types of vehicles, trains and buses and every one of them can have different price and speed. The problem is how passengers will be allocated to different vehicles Let the first stop be station A and the last stop be station B. There are two kinds of vehicles, trains and buses, which travel between station A and station B. Every vehicle has its set of stations where it stops, only the first station and the terminus are common for all vehicles. Some of the stations can be common for some of the vehicles. Let the set of all stations is $S = \{s_1, \ldots, s_n\}$ and on every station s_i, $i = 1, \ldots, n-1$, n is the number of stations, at every time slot there are number of passengers which want to travel to station s_j, $j = i+1, \ldots, n$. Every vehicle travel with different speed and the price to travel from station s_i to station s_j can be different. We fix a parameters k_1 and k_2. They are used for calculation of the time and price intervals respectively. If a passenger have in mind to start his travel at time t he will chose a vehicle in the interval $(t - k_1, t + k_1)$. If a passenger have in mind to pay for his travel price P he can pay price from the interval $(P, P + P * k_2/100)$. Thus, we include in our model some fuzzy element with an aim it to become more realistic.

The input data of our problem are set of stations S, starting time of every vehicle from the first station, time for every vehicle to go from station s_i to station s_j, the capacity of every vehicle, the price for every vehicle to travel from one station to another one, number of passengers which want to travel from one station to another one at every moment. Our algorithm calculates how many passengers will get on every of the vehicles on station s_i to station s_j at every time slot. There are two objectives, the total price of all tickets, Equation 3, and the total travel time, Equation 4. If some vehicle does not stop on some station, we put the travel time and the price to this destination to be 0.

$$TP = \sum_{i=1}^{M} p_i \tag{3}$$

where TP is the total price, M is the number of passengers, p_i is the price, payed by the passenger i.

$$TT = \sum_{i=1}^{M} T_i \tag{4}$$

where TT is the total time, M is the number of passengers, T_i is the traveling time of passenger i.

TABLE I: Algorithm parameters

ρ	0.5
α	1
β	1
τ_0	0.5
number of ants	10
number of iterations	100

The output is the number of passengers in every vehicle in every station and the values of the two objective functions.

It is NP-hard multi-objective optimization problem, therefore we chose a metaheuristic method to solve it, in particular ACO.

The model is prepared to solve the problem for one direction. It can be applied to model and optimize transportation network direction by direction. One of the important points of the ACO algorithm is representation of the problem by graph. In our case the time is divided to time periods, $N \times 24$ time periods correspond to 60/N minutes, thus $2 \times 24 = 48$ time periods, correspond to 30 minutes. Every station is represented by set of $N \times 24$ nodes, showing different time moments in which a vehicle stops on this station. The pheromone is deposited on the nodes of the graph. The ants start to construct their solutions from the first station. If the number of the passengers from this station is P, the ants chose a random number P_1 from the interval $[0, min\{P, C_1\}]$ and assign this number to the first vehicle as a number of passengers. To the next vehicle the interval is decreased with P_1. C_1 is the capacity of the vehicle. The number of all passengers getting vehicle in some time moment is maximal possible. If there is only one vehicle at this moment the maximal possible number of passengers gets on this vehicle. We model the number of the passengers for the next stations by applying probabilistic rule called transition probability. Our heuristic information is a sum of the reciprocal values of the two objective functions.

IV. Experimental Results

We have programmed our ACO algorithm in C programming language. After several experiments the algorithm parameters are set as it is shown in a Table I

We test our algorithm on one real problem, destination Sofia Varna. The starting station is Sofia, Bulgarian capital and the terminus is Varna the maritime capital of the country. The distance between the first and the last station is about 450 km. There are 5 trains and 23 buses which travel from Sofia to Varna, but they move with different speed, the prices are different and they stop on different stations between Sofia and Varna. There are not data available on passenger numbers therefore we approximate them, taking in to account the population of every one of the towns where some of the vehicles stops. 5 trains and 23 buses, with different speed and price travel between them every day. The stations can differ for different vehicles.

TABLE II: Experimental results Sofia Varna, deterministic

No	Price	Time	Train
1	51843	25840	1951
2	51797	25842	1952
3	51579	25862	1978
4	51571	25869	1979
5	51563	25870	1980

TABLE III: Experimental results Sofia Varna, fuzzy

No	Price	Time	Train
1	51821	25856	1961
2	51775	25864	1963
3	51565	25873	1991
4	51560	25880	1995
5	51549	25882	1998

The Table II and Table III shows achieved solutions by two variant of ACO algorithm, deterministic and fuzzy respectively. The results in Table II are from our previous work [8] where we apply the deterministic variant of the algorithm. 10 ants are used and the algorithm is run 100 iterations. In a both cases there are 5 nondominated solutions. In every row are shown the travel price of hall passengers, the travel time of hall passengers and the sum of the passengers used train. In the both tables can be seen that the solutions with more passengers in the train have more traveling time and less price. The number of passengers used train or respectively bus is changed if on the same station on the same time there is more than one transportation possibility for deterministic case. In deterministic case the difference in number of passengers in the train comes from long destinations. In fuzzy variant of the algorithm we observe that the number of the passengers in the train is more than in the buss comparing with deterministic case. When the price between the bus and train is similar in a short destination in the fuzzy case it is perceived as the same, it is the same for the time, and the passengers chose bus or train with the same probability. In deterministic case even the small difference is perceived as a different and the vehicle with less price has high probability to be chosen by the passengers which prefer cheaper transportation. Thus we can explain why in the fuzzy case more passengers chose the train than in deterministic one.

V. Conclusion

Transportation is a very important branch of economics and our everyday life. The different kinds of transportation propose different services. Ones are faster, others are cheaper. The passenger decision depends on his preferences. In this paper we propose a model of the flow of passengers taking into account the two main criteria that guide the passengers in their choice, traveling time and traveling price. Thus the problem is defined as multi-objective optimization problem with two objective functions. A fuzzy variant of the model is proposed. When the prices or times are in a predefined interval, they are considered equal. Thus the model becomes closer to human thinking and, from there, more realistic. The proposed model can help for transport analysis of existing transport. It can predict the change of passenger flow when some vehicle is included or excluded and when the timetable is changed.

Thus the transportation can be optimized and to become close to the people's needs. In a future we can include additional elements in the model like other preferences of the passengers.

Acknowledgment

Work presented here is partially supported by the National Scientific Fund of Bulgaria under grant DFNI DN12/5 "Efficient Stochastic Methods and Algorithms for Large-Scale Problems", Grant No BG05M2OP001-1.001-0003, financed by the Science and Education for Smart Growth Operational Program and by the Bulgarian scientific fund by the grant DFNI DN 02/10.

References

[1] A. El Amaraoui A.,K. Mesghouni, *Train Scheduling Networks under Time Duration Uncertainty*, In proc. of the 19th World Congress of the Int. Federation of Automatic Control, 2014, 8762–8767.

[2] A. A. Assad, *Models for Rail Transportation*, Transportation Research Part A General, **14**3, 1980, 205–220.

[3] E. Bonabeau, M. Dorigo, G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, 1999.

[4] O. Diaz-Parra, J. A. Ruiz-Vanoye, B. B. Loranca, A. Fuentes-Penna, R.A. Barrera-Camara, *A Survey of Transportation Problems* Journal of Applied Mathematics Volume 2014 (2014), Article ID 848129, 17 pages.

[5] Ch. Dong, Zh. Xiong, Ch. Shao, H. Zhang *A spatial–temporal-based state space approach for freeway network traffic flow modelling and prediction* Journal of Transportmetrica A:Transport Science **11**(7) (2015), 574-560.

[6] M. Dorigo, T. Stutzle. *Ant Colony Optimization*, MIT Press, 2004.

[7] S. Fidanova, K. Atanasov *Generalized Net Model for the Process of Hibride Ant Colony Optimization* Comptes Randus de l'Academie Bulgare des Sciences, **62**(3), 2009, 315–322.

[8] Fidanova S.. *Metaheuristic Method for Transport Modelling and Optimization* Studies in Computational Intelligence, 648, Springer, 2016, 295–302.

[9] F. S. Hanseler,N. Molyneaux, M. Bierlaire, and A. Stathopoulos, *Schedule-based estimation of pedestrian demand within a railway station*, Proceedings of the Swiss Transportation Research Conference (STRC) 14-16 May, 2014.

[10] J. G. Jin, J. Zhao, D. H. Lee, *A Column Generation Based Approach for the Train Network Design Optimization Problem*, J. of Transportation Research, **50**(1), 2013, 1–17.

[11] V. K. Mathur, *How Well do we Know Pareto Optimality?* J. of Economic Education **22**(2), 1991, 172–178.

[12] N. Molyneaux, F. Hanseler, M. Bierlaire, *Modelling of train-induced pedestrian flows in rail- way stations*, Proceedings of the Swiss Transportation Research Conference (STRC) 14-16 May, 2014.

[13] C. Woroniuk, M. Marinov, *Simulation Modelling to Analyze the Current Level of Utilization of Sections Along Rail Rout*, J. of Transport Literature, textbf7(2), 2013, 235–252.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 241–249 DOI: 10.15439/2020F10 ISSN 2300-5963 ACSIS, Vol. 21

Fast BF-ICrA Method for the Evaluation of MO-ACO Algorithm for WSN Layout

Jean Dezert
ONERA - DTIS
The French Aerospace Lab
Palaiseau, France.
jean.dezert@onera.fr

Stefka Fidanova
Inst. of I&C Tech.
Bulgarian Academy of Sciences
Sofia, Bulgaria.
stefka@parallel.bas.bg

Albena Tchamova
Inst. of I&C Tech.
Bulgarian Academy of Sciences
Sofia, Bulgaria.
tchamova@bas.bg

***Abstract*—In this paper, we present a fast Belief Function based Inter-Criteria Analysis (BF-ICrA) method based on the canonical decomposition of basic belief assignments defined on a dichotomous frame of discernment. This new method is then applied for evaluating the Multiple-Objective Ant Colony Optimization (MO-ACO) algorithm for Wireless Sensor Networks (WSN) deployment.**

Keywords: Inter-Criteria Analysis, belief functions, information fusion, canonical decomposition, PCR5 rule.

I. Introduction

In OUR previous work [1] we propose a new and improved version of classical Atanassov's InterCriteria Analysis (ICrA) [2] - [4] approach based on Belief Functions (BF-ICrA). This method proposes a better construction of InterCriteria Matrix that fully exploits all the information of the score matrix, and the closeness measure of agreement between criteria based on belief interval distance. In [5], we show how the fusion of many sources of evidences represented by Basic Belief Assignments (BBAs) defined on a same dichotomous frame of discernment can be fast and easily done thanks to the Proportional Conflict Redistribution rule no.5 based canonical decomposition of the BBAs, proposed recently in [6]. In the recent paper we consider BF-ICrA based on this promising technique. Then we show how to apply it for the evaluation of the Multiple-Objective Ant Colony Optimization (MO-ACO) algorithm for Wireless Sensor Networks (WSN) deployment. After a condensed presentation of basics of belief functions in Section II, including the short description of canonical decomposition of dichotomous BBAs approach, and the main steps of fast fusion method of dichotomous BBAs, in Section III the BF-ICrA method is described and analyzed. Section IV is devoted to the multi-objective ACO algorithm. In Section V the results of the fast BF-ICrA method with the MO-ACO algorithm for WSN layout deployment is presented and discussed. Conclusion is given in Section VI.

II. Basics of belief functions

A. Basic definitions

Belief functions (BF) have been introduced by Shafer in [7] to model epistemic uncertainty and to combine distinct sources of evidence thanks to Dempster's rule of combination. In Shafer's framework, we assume that the answer[1] of the problem under concern belongs to a known finite discrete frame of discernment (FoD) $\Theta = \{\theta_1, \theta_2, \dots, \theta_n\}$, with $n > 1$, and where all elements of Θ are mutually exclusive and exhaustive. The set of all subsets of Θ (including empty set $\emptyset$ and Θ) is the power-set of Θ denoted by 2^Θ. A proper Basic Belief Assignment (BBA) associated with a given source of evidence is defined [7] as a mapping $m(\cdot) : 2^\Theta \rightarrow [0, 1]$ satisfying $m(\emptyset) = 0$ and $\sum_{A\in 2^\Theta} m(A) = 1$. The quantity $m(A)$ is called the mass of A committed by the source of evidence. Belief and plausibility functions are respectively defined from a proper BBA $m(\cdot)$ by

$$Bel(A) = \sum_{B\in 2^\Theta | B\subseteq A} m(B) \tag{1}$$

and

$$Pl(A) = \sum_{B\in 2^\Theta | A\cap B\neq\emptyset} m(B) = 1 - \text{Bel}(\bar{A}). \tag{2}$$

where $\bar{A}$ is the complement of A in Θ.

$Bel(A)$ and $Pl(A)$ are usually interpreted respectively as lower and upper bounds of an unknown (subjective) probability measure $P(A)$. The quantities $m(\cdot)$ and $Bel(\cdot)$ are one-to-one and linked by the Möbius inverse formula (see [7], p. 39). A is called a Focal Element (FE) of $m(\cdot)$ if $m(A) > 0$. When all focal elements are singletons, $m(\cdot)$ is called a *Bayesian BBA* [7] and its corresponding $Bel(\cdot)$ function is equal to $Pl(\cdot)$ and they are homogeneous to a (subjective) probability measure $P(\cdot)$. The vacuous BBA, representing a totally ignorant source, is defined as $m_v(\Theta) = 1$. A dichotomous BBA is a BBA defined on a FoD which has only two proper subsets, for instance $\Theta = \{A, \bar{A}\}$ with $A \neq \Theta$ and $A \neq \emptyset$. A dogmatic BBA is a BBA such that $m(\Theta) = 0$. If $m(\Theta) > 0$ the BBA $m(\cdot)$ is nondogmatic. A simple BBA is a BBA that has at most two focal sets and one of them is Θ. A dichotomous non dogmatic mass of belief is a BBA having three focal elements A, $\bar{A}$ and $A \cup \bar{A}$ with A and $\bar{A}$ subsets of Θ.

In his Mathematical Theory of Evidence [7], Shafer proposed to combine $s \geq 2$ distinct sources of evidence repre-

[1] i.e. the solution, or the decision to take.

sented by BBAs with Dempster's rule (i.e. the normalized conjunctive rule), which unfortunately behaves counterintuitively both in high and low conflicting situations as reported in [8]–[11]. In our previous works (see [12], Vol. 2 and Vol. 3 for full justification and examples) we did propose new rules of combination based on different Proportional Conflict Redistribution (PCR) principles, and we have shown the interest of the PCR rule No 5 (PCR5) for combining two BBAs, and PCR rule No 6 (PCR6) for combining more than two BBAs altogether [12], Vol. 2. PCR6 coincides with PCR5 when one combines two sources. The difference between PCR5 and PCR6 lies in the way the proportional conflict redistribution is done as soon as three (or more) sources are involved in the fusion. PCR5 transfers the conflicting mass only to the elements involved in the conflict and proportionally to their individual masses, so that the specificity of the information is entirely preserved in this fusion process.

The general (complicate) formulas for PCR5 and PCR6 rules are given in [12], Vol. 2. The fusion of two BBAs based on PCR5 (or PCR6) rule which will be use for canonical decomposition of a dichotomous BBA is obtained by the formula

$$m_{PCR5}(X) = \sum_{\substack{X_1,X_2\in 2^\Theta\\ X_1\cap X_2=X}} m_1(X_1)m_2(X_2) +$$

$$\sum_{\substack{X_2\in 2^\Theta\\ X_2\cap X=\emptyset}} [\frac{m_1(X)^2 m_2(X_2)}{m_1(X)+m_2(X_2)} + \frac{m_2(X)^2 m_1(X_2)}{m_2(X)+m_1(X_2)}] \quad (3)$$

where all denominators in (3) are different from zero. If a denominator is zero, that fraction is discarded.

From the implementation point of view, PCR6 is simpler to implement than PCR5. For convenience, very basic (not optimized) Matlab™codes of PCR5 and PCR6 fusion rules can be found in [12], [13] and from the toolboxes repository on the web [14]. The main drawback of PCR5 and PCR6 rules is their very high combinatorial complexity when the number of source is big, as well as the cardinality of the FoD. In this case, PCR5 or PCR6 rules cannot be used directly because of memory overflow. Even for combining BBAs defined on a simple dichotomous FoD as those involved in the Inter-Criteria Analysis (ICrA), the computational time for combining more than 10 sources can take several hours[2]. That is why a fast fusion method to combine dichotomous BBAs is necessary, and we present it in the next subsections.

B. Canonical decomposition of dichotomous BBA

A FoD $\Theta = \{A, \bar{A}\}$ is called dichotomous if it consists of only two proper subsets A and $\bar{A}$ with $A \cup \bar{A} = \Theta$ and $A \cap \bar{A} = \emptyset$, where $\bar{A}$ is the complement of A in Θ and A is different from Θ and from Empty-Set. We consider a given proper BBA $m(\cdot) : 2^\Theta \to [0, 1]$ of the general form

$$m(A) = a, \quad m(\bar{A}) = b, \quad m(A \cup \bar{A}) = 1 - a - b \quad (4)$$

The canonical decomposition problem consists in finding the two following simple proper BBAs m_p and m_c of the form

$$m_p(A) = x, \quad m_p(A \cup \bar{A}) = 1 - x \quad (5)$$

$$m_c(\bar{A}) = y, \quad m_c(A \cup \bar{A}) = 1 - y \quad (6)$$

with $(x, y) \in [0, 1] \times [0, 1]$, such that $m = Fusion(m_p, m_c)$, for a chosen rule of combination denoted by $Fusion(\cdot, \cdot)$. The simple BBA $m_p(\cdot)$ is called the *pro-BBA* (or pro-evidence) of A, and the simple BBA $m_c(\cdot)$ the *contra-BBA* (or contra-evidence) of A. The BBA $m_p(\cdot)$ is interpreted as a source of evidence providing an uncertain evidence in favor of A, whereas $m_c(\cdot)$ is interpreted as a source of evidence providing an uncertain contrary evidence about A.

In [6], we have shown that this decomposition is possible with Dempster's rule only if $0 < a < 1$, $0 < b < 1$ and $a + b < 1$, and we have $x = \frac{a}{1-b}$ and $y = \frac{b}{1-a}$. However, any dogmatic BBA $m(A) = a$, $m(\bar{A}) = b$ with $a + b = 1$ is not decomposable from Dempster's rule for the case when $(a, b) \neq (1, 0)$ and $(a, b) \neq (0, 1)$, and the dogmatic BBAs $m(A) = 1$, $m(\bar{A}) = 0$, or $m(A) = 0$, $m(\bar{A}) = 1$ have infinitely many decompositions based on Dempster's rule of combination. We have also proved that this canonical decomposition cannot be done from conjunctive, disjunctive, Yager's [15] or Dubois-Prade [16] rules of combination, neither from the averaging rule. The main result of [6] is that this canonical decomposition is unique and is always possible in all cases using the PCR5 rule of combination. This is very useful to implement a fast efficient approximating fusion method of dichotomous BBAs as presented in details in [5]. We recall the following two important theorems proved in [6].

Theorem 1: Consider a dichotomous FoD $\Theta = \{A, \bar{A}\}$ with $A \neq \Theta$ and $A \neq \emptyset$ and a nondogmatic BBA $m(\cdot) : 2^\Theta \to [0, 1]$ defined on Θ by $m(A) = a$, $m(\bar{A}) = b$, and $m(A \cup \bar{A}) = 1 - a - b$, where $a, b \in [0, 1]$ and $a + b < 1$. Then the BBA $m(\cdot)$ has a unique canonical decomposition using PCR5 rule of combination of the form $m = PCR5(m_p, m_c)$ with pro-evidence $m_p(A) = x$, $m_p(A \cup \bar{A}) = 1 - x$ and contra-evidence $m_c(\bar{A}) = y$, $m_c(A \cup \bar{A}) = 1 - y$ where $x, y \in [0, 1]$.

Theorem 2: Any dogmatic BBA defined by $m(A) = a$ and $m(\bar{A}) = b$, where $a, b \in [0, 1]$ and $a + b = 1$, has a canonical decomposition using PCR5 rule of combination of the form $m = PCR5(m_p, m_c)$ with $m_p(A) = x$, $m_p(A \cup \bar{A}) = 1 - x$ and $m_c(\bar{A}) = y$, $m_c(A \cup \bar{A}) = 1 - y$ where $x, y \in [0, 1]$.

Theorems 1 & 2 prove that the decomposition based on PCR5 always exists and it is unique for any dichotomous (nondogmatic, or dogmatic) BBA.

For the case of dichotomous nondogmatic BBA considered in Theorem 1, one has to find x and y solutions of the system

$$a = x(1 - y) + \frac{x^2 y}{x + y} = \frac{x^2 + xy - xy^2}{x + y} \quad (7)$$

$$b = (1 - x)y + \frac{xy^2}{x + y} = \frac{y^2 + xy - x^2 y}{x + y} \quad (8)$$

[2] with a MacBook Pro 2.8 GHz Intel Core i7 with 16 Go 1600 MHz DDR3 memory running Matlab™R2018a.

under the constraints $(a, b) \in [0, 1]^2$, and $0 < a + b < 1$. The explicit expression of x and y are difficult to obtain analytically (even with modern symbolic computing systems like Mathematica™, or Maple™) because one has a quartic equation to solve whose general analytical expression of its solutions is very complicate. Fortunately, the solutions can be easily calculated numerically by these computing systems, and even with Matlab™system (thanks to the *fsolve* function) as soon as the numerical values are committed to a and to b, and this is what we use in our simulations.

C. Fast Fusion of dichotomous BBAs

The main idea for making the fast fusion of dichotomous BBAs $m_s(.)$, for $s = 1, 2, \dots, S$ defined on the same FoD Θ is based on the three following main steps:

1) In the first step, one decomposes canonically each dichotomous BBA $m_s(\cdot)$ into its pro and contra evidences $m_{p,s} = (m_{p,s}(A), m_{p,s}(\bar{A}), m_{p,s}(A \cup \bar{A})) = (x_s, 0, 1 - x_s)$ and $m_{c,s} = (m_{c,s}(A), m_{c,s}(\bar{A}), m_{c,s}(A \cup \bar{A})) = (0, y_s, 1 - y_s)$,
2) In the second setp, one combines the pro-evidences $m_{p,s}$ for $s = 1, 2, \dots, S$ altogether to get a global pro-evidence m_p, and in parallel one combines all the contra-evidences $m_{c,s}$ for $s = 1, 2, \dots, S$ altogether to get a global contra-evidence m_c. The fusion step of pro and contra evidences is based on conjunctive rule of combination.
3) Once m_p and m_c are calculated, then one combines them with PCR5 fusion rule to get the final result.

Because the PCR5 rule of combination is not associative, the fusion of the canonical BBAs followed by their PCR5 fusion will not provide in general the same result as the direct fusion of the dichotomous BBAs altogether but only an approximate result, which is normal. However, this new fusion approach is interesting because the fusion of the pro-evidence $m_{p,s}$ (resp. contra-evidences $m_{c,s}$) is very simple because there is non conflict between $m_{p,s}$ (resp. between $m_{c,s}$), so that their fusion can be done quite easily and a large number of sources can be combined without a high computational burden. In fact, with this fusion approach, only one PCR5 fusion step of simple (combined) canonical BBAs is needed at the very end of the fusion process. In [5], we have proved with a Monte-Carlo simulation analysis that the approximation obtained by this new fusion method based on the fusion of pro-evidences and contra-evidences with respect to the direct fusion of the BBAs with PCR5 (or PCR6 when considering more than two sources to combine) is effective because the agreement between the decision taken from the direct fusion method, and the indirect (canonical decomposition based) method is very good. This new fusion method based on this canonical decomposition does not suffer of combinatorial complexity limitation which is of great interest in some applications because many (hundreds or even thousands) of dichotomous BBAs could be easily combined very quickly. Actually with this method what takes a bit time is only the canonical decomposition done by the numerical solver. Our analysis [5] has shown that complexity of this fast approach is quasi-linear with the number of sources to combine.

III. The BF-ICrA method

In [1], we did present an improved version of Atanassov's Inter-Criteria Analysis (ICrA) method [2]–[4] based on belief functions. This new method has been named BF-ICrA (Belief Function based Inter-Criteria Analysis) for short. It has already been applied to GPS surveying problems in [17]. We present briefly in this section the principles of BF-ICrA.

BF-ICrA starts with the construction of an $M \times N$ BBA matrix $\mathbf{M} = [m_{ij}(\cdot)]$ from the score matrix $\mathbf{S} = [S_{ij}]$. The BBA matrix $\mathbf{M}$ is obtained as follows - see [18] for details and justification.

$$m_{ij}(A_i) = Bel_{ij}(A_i) \tag{9}$$

$$m_{ij}(\bar{A}_i) = Bel_{ij}(\bar{A}_i) = 1 - Pl_{ij}(A_i) \tag{10}$$

$$m_{ij}(A_i \cup \bar{A}_i) = Pl_{ij}(A_i) - Bel_{ij}(A_i) \tag{11}$$

where[3]

$$Bel_{ij}(A_i) \triangleq Sup_j(A_i)/A^j_{\text{max}} \tag{12}$$

$$Bel_{ij}(\bar{A}_i) \triangleq Inf_j(A_i)/A^j_{\text{min}} \tag{13}$$

with

$$Sup_j(A_i) \triangleq \sum_{k \in \{1,\dots M\} | S_{kj} \leq S_{ij}} |S_{ij} - S_{kj}| \tag{14}$$

$$Inf_j(A_i) \triangleq - \sum_{k \in \{1,\dots M\} | S_{kj} \geq S_{ij}} |S_{ij} - S_{kj}| \tag{15}$$

and

$$A^j_{\text{max}} \triangleq \max_i Sup_j(A_i) \tag{16}$$

$$A^j_{\text{min}} \triangleq \min_i Inf_j(A_i) \tag{17}$$

For another criterion $C_{j'}$ and the j'-th column of the score matrix we will obtain another set of BBA values $m_{ij'}(\cdot)$. Applying this method for each column of the score matrix we are able to compute the BBA matrix $\mathbf{M} = [m_{ij}(\cdot)]$ whose each component is in fact a triplet $(m_{ij}(A_i), m_{ij}(\bar{A}_i), m_{ij}(A_i \cup \bar{A}_i))$ of BBA values in $[0, 1]$ such that $m_{ij}(A_i) + m_{ij}(\bar{A}_i) + m_{ij}(A_i \cup \bar{A}_i)) = 1$ for all $i = 1, \dots, M$ and $j = 1, \dots, N$.

The next step of BF-ICrA approach is the construction of the $N \times N$ Inter-Criteria Matrix $\mathbf{K} = [K_{jj'}]$ from $M \times N$ BBA matrix $\mathbf{M} = [m_{ij}(\cdot)]$ where elements $K_{jj'}$ corresponds to the BBA $(m_{jj'}(\theta), m_{jj'}(\bar{\theta}), m_{jj'}(\theta \cup \bar{\theta}))$ about positive consonance θ, negative consonance $\bar{\theta}$ and uncertainty between criteria C_j and $C_{j'}$ respectively. The construction of the triplet $K_{jj'} = (m_{jj'}(\theta), m_{jj'}(\bar{\theta}), m_{jj'}(\theta \cup \bar{\theta}))$ is based on two steps:

- **Step 1 (BBA construction)**: Getting $m^i_{jj'}(.)$.

  For each alternative A_i for $i = 1, \dots, M$, we first compute the BBA $(m^i_{jj'}(\theta), m^i_{jj'}(\bar{\theta}), m^i_{jj'}(\theta \cup$

[3] assuming that $A^j_{\text{max}} \neq 0$ and $A^j_{\text{min}} \neq 0$. If $A^j_{\text{max}} = 0$ then $Bel_{ij}(A_i) = 0$, and if $A^j_{\text{min}} = 0$ then $Pl_{ij}(A_i) = 1$.

$\bar{\theta}$)) for any two criteria $j, j' \in \{1, 2, \dots, N\}$. For this, we consider two sources of evidences (SoE) indexed by j and j' providing the BBA m_{ij} and $m_{ij'}$ defined on the simple FoD $\{A_i, \bar{A}_i\}$ and denoted $m_{ij} = [m_{ij}(A_i), m_{ij}(\bar{A}_i), m_{ij}(A_i \cup \bar{A}_i)]$ and $m_{ij'} = [m_{ij'}(A_i), m_{ij'}(\bar{A}_i), m_{ij'}(A_i \cup \bar{A}_i)]$. We also denote $\Theta = \{\theta, \bar{\theta}\}$ the FoD about the relative state of the two SoE, where θ means that the two SoE agree, $\bar{\theta}$ means that they disagree and $\theta \cup \bar{\theta}$ means that we don't know. Hence, two SoE are in total agreement if both commit their maximum belief mass to the same element A_i or to the same element $\bar{A}_i$. Similarly, two SoE are in total disagreement if each one commits its maximum mass of belief to one element and the other to its opposite, that is if one has $m_{ij}(A_i) = 1$ and $m_{ij'}(\bar{A}_i) = 1$, or if $m_{ij}(\bar{A}_i) = 1$ and $m_{ij'}(A_i) = 1$. Based on this very simple and natural principle, one can now compute the belief masses as follows:

$$m^i_{jj'}(\theta) = m_{ij}(A_i)m_{ij'}(A_i) + m_{ij}(\bar{A})m_{ij'}(\bar{A}) \quad (18)$$

$$m^i_{jj'}(\bar{\theta}) = m_{ij}(A_i)m_{ij'}(\bar{A}_i) + m_{ij}(\bar{A}_i)m_{ij'}(A_i) \quad (19)$$

$$m^i_{jj'}(\theta \cup \bar{\theta}) = 1 - m^i_{jj'}(\theta) - m^i_{jj'}(\bar{\theta}) \quad (20)$$

$m^i_{jj'}(\theta)$ represents the degree of agreement between the BBA $m_{ij}(\cdot)$ and $m_{ij'}(\cdot)$ for the alternative A_i, $m^i_{jj'}(\bar{\theta})$ represents the degree of disagreement of the two BBAs and $m^i_{jj'}(\theta \cup \bar{\theta})$ the level of uncertainty (i.e. how much we don't know if they agree or disagree). By construction $m^i_{jj'}(\cdot) = m^i_{j'j}(\cdot)$, $m^i_{jj'}(\theta), m^i_{jj'}(\bar{\theta}), m^i_{jj'}(\theta \cup \bar{\theta}) \in [0, 1]$ and $m^i_{jj'}(\theta) + m^i_{jj'}(\bar{\theta}) + m^i_{jj'}(\theta \cup \bar{\theta}) = 1$. This BBA modeling permits to build a set of M symmetrical Inter-Criteria Belief Matrices (ICBM) $\mathbf{K}^i = [K^i_{jj'}]$ of dimension $N \times N$ relative to each alternative A_i whose components $K^i_{jj'}$ correspond to the triplet of BBA values $m^i_{jj'} = (m^i_{jj'}(\theta), m^i_{jj'}(\bar{\theta}), m^i_{jj'}(\theta \cup \bar{\theta}))$ modeling the belief of agreement and of disagreement between C_j and $C_{j'}$ based on A_i.

- **Step 2 (fusion)**: Getting $\mathbf{m_{jj'}}(.)$.

  In this step, one needs to combine the BBAs $\mathbf{m^i_{jj'}}(.)$ for $i = 1, \dots, M$ altogether to get the component $K_{jj'} = (m_{jj'}(\theta), m_{jj'}(\bar{\theta}), m_{jj'}(\theta \cup \bar{\theta}))$ of the Inter-Criteria Belief matrix[4] (ICBM) $\mathbf{K} = [K_{jj'}]$. For this and from the theoretical standpoint, we recommend to use the PCR6 fusion rule [12] (Vol. 3) because of known deficiencies of Dempster's rule.

Once the global Inter-Criteria Belief Matrix (ICBM) $\mathbf{K} = [K_{jj'} = (m_{jj'}(\theta), m_{jj'}(\bar{\theta}), m_{jj'}(\theta \cup \bar{\theta}))]$ is calculated, we can identify the criteria that are in strong agreement, in strong disagreement, and those on which we are uncertain. For identifying the criteria that are in strong agreement, we evaluate the distance of each component of $K_{jj'}$ with the BBA representing the best agreement state and characterized by the specific BBA[5] $m_T(\theta) = 1$. From a similar approach we can also identify, if we want, the criteria that are in very strong disagreement using the distance of $m_{jj'}(\cdot)$ with respect to the BBA representing the best disagreement state characterized by the specific BBA $m_F(\bar{\theta}) = 1$. We use the belief interval distance $d_{BI}(m_1, m_2)$ presented in [19] for measuring the distance between the two BBAs.

A. Fast BF-ICrA method

The computational complexity of BF-ICrA is of course higher than the complexity of ICrA because it makes a more precise evaluation of local and global inter-criteria belief matrices with respect to inter-criteria matrices calculated by Atanassov's ICrA. The overall reduction of the computational burden of the original MCDM problem thanks to BF-ICrA depends highly on the problem under concern, the complexity and cost to evaluate each criteria involved in it, as well as the number of redundant criteria identified by BF-ICrA method.

The main drawback of BF-ICrA method is the PCR6 combination required in its step 2 for combining altogether the dichotomous BBAs $m^i_{jj'}(.)$. Because of combinatorial complexity of PCR6 rule, it cannot work in reasonable computational time as soon as the number of sources to combine altogether is greater than 10, which prevents its use for solving ICrA problems involving more than 10 alternatives (as in the examples 2 and 3 presented in section V). That is why it is necessary to adapt the original BF-ICrA method for working with a large number of alternatives and criteria. For this, we can in step 2 of BF-ICrA exploit the method for the fast fusion of dichotomous BBAs presented in section II-C. More precisely, each dichotomous BBA $m^i_{jj'}(.)$ will be canonically decomposed in its pro-evidence $m^i_{jj',p}(.)$ and its contra-evidence $m^i_{jj',c}(.)$ that will be combined separately to get the global pro-evidence $m_{jj',p}(.)$ and the global contra-evidence $m_{jj',c}(.)$. Then, the BBAs $m_{jj',p}(.)$ and $m_{jj',c}(.)$ are combined with PCR5 rule to get the BBAs $m_{jj'}(.)$ and, finally, the global Inter-Criteria Belief Matrix $\mathbf{K} = [K_{jj'} = (m_{jj'}(\theta), m_{jj'}(\bar{\theta}), m_{jj'}(\theta \cup \bar{\theta}))]$. The principle of this modified step 2 of BF-ICrA is summarized in the Figure 1 for convenience.

Another simpler fusion method to combine the dichotomous BBAs $m^i_{jj'}(.)$ would just consist to average them. In section V, we will show how these two methods behave in the examples chosen for the evaluation of MO-ACO Algorithm for optimal WSN deployment.

IV. Multi-Objective ACO algorithm

Recently Wireless Sensor Networks (WSNs) have attracted the attention of the research scientists community, conditioned by a set of challenges: theoretical and practical. WSNs consists of distributed sensor nodes and their main purpose is to monitor the real-time environmental status, based on gathering available sensor information, processing and transmitting the

[4] For presentation convenience, the ICBM $\mathbf{K} = [K_{jj'} = (m_{jj'}(\theta), m_{jj'}(\bar{\theta}), m_{jj'}(\theta \cup \bar{\theta}))]$ is decomposed into three matrices $\mathbf{K}(\theta) = [K^{\theta}_{jj'} = m_{jj'}(\theta)]$, $\mathbf{K}(\bar{\theta}) = [K^{\bar{\theta}}_{jj'} = m_{jj'}(\bar{\theta})]$ and $\mathbf{K}(\theta \cup \bar{\theta}) = [K^{\theta \cup \bar{\theta}}_{jj'} = 1 - m_{jj'}(\theta) - m_{jj'}(\bar{\theta})]$.

[5] We use the index T in the notation $m_T(\cdot)$ to refer that the agreement is true, and F in $m_F(\cdot)$ to specify that the agreement is false.

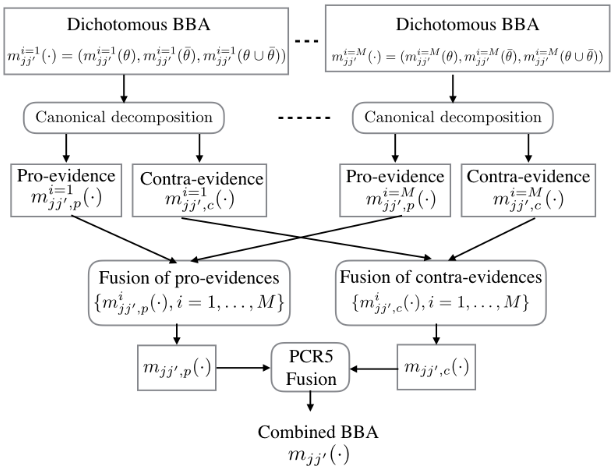

Dichotomous BBA
$m^{i=1}_{jj'}(\cdot) = (m^{i=1}_{jj'}(\theta), m^{i=1}_{jj'}(\bar{\theta}), m^{i=1}_{jj'}(\theta \cup \bar{\theta}))$
Dichotomous BBA
$m^{i=M}_{jj'}(\cdot) = (m^{i=M}_{jj'}(\theta), m^{i=M}_{jj'}(\bar{\theta}), m^{i=M}_{jj'}(\theta \cup \bar{\theta}))$
Canonical decomposition
Canonical decomposition
Pro-evidence $m^{i=1}_{jj',p}(\cdot)$
Contra-evidence $m^{i=1}_{jj',c}(\cdot)$
Pro-evidence $m^{i=M}_{jj',p}(\cdot)$
Contra-evidence $m^{i=M}_{jj',c}(\cdot)$
Fusion of pro-evidences $\{m^{i}_{jj',p}(\cdot), i = 1, \ldots, M\}$
Fusion of contra-evidences $\{m^{i}_{jj',c}(\cdot), i = 1, \ldots, M\}$
$m_{jj',p}(\cdot)$
PCR5 Fusion
$m_{jj',c}(\cdot)$
Combined BBA $m_{jj'}(\cdot)$

Figure 1. Principle of fast fusion of $m^{i}_{jj'}(.)$ of Step 2 of BF-ICrA.

collected data to the specified remote base station. It is a promising technology that is used in a coverage of application requiring minimum human contribution, ranging from civil and military to healthcare and environmental monitoring. One of the key mission of WSN is the full surveillance of the monitoring region with a minimal number of sensors and minimized energy consumption of the network. The lifetime of the sensors is strongly related to the amount of the power loaded in the battery, that is why the control of the energy consumption of sensors is an important active research problem. The small energy storage capacity of sensor nodes intrudes the possibility to gather the information directly to the main base. Because of this they transfer their data to the so called High Energy Communication Node (HECN), which is able to collect the information from across the network and to transmit it to the base computer for processing. The sensors transmit their data to the HECN, either directly or via hops, using closest sensors as communication relays. The WSN can have large numbers of nodes and the problem can be very complex.

In order to solve successfully the key mission of WSNs, in [20], we did apply multi-objective Ant Colony Optimization (ACO) to solve this hard, from the computational point of view, telecommunication problem. The number of ants is one of the key algorithm parameters in the ACO and it is important to find the optimal number of ants needed to achieve good solutions with minimal computational resources. In [20], the optimal solution was obtained by applying the classical Atanassov's ICrA method. In the next section we will present the results obtained by the fast BF-ICrA approach and compare their results.

The problem of designing a WSN is multi-objective, with two objective functions: 1) one wants to minimize the energy consumption of the nodes in the network, and 2) one wants to minimize the number of nodes. The full coverage of the network and connectivity are considered as constraints. For solving this problem, we have proposed to use a M ulti-Objective Ant Colony Optimization (MO-ACO) algorithm in [20] and we have studied the influence of the number of ants on the algorithm performance and quality of the achieved solutions. The computational resources, which the algorithm needs, are not negligible. The computational resources depends on the size of the solved problem and on the number of ants. The aim is to find a minimal number of ants which allow the algorithm to find good solution for WSN deployment.

The ACO algorithm uses a colony of artificial ants that behave as cooperating agents. With the help of the pheromone and the heuristic information they try to construct better solutions and to find the optimal ones. The pheromone corresponds to the global memory of the ants and the heuristic information is a some preliminary knowledge of the problem. The problem is represented by a graph and the solution is represented by a path in the graph or by tree in the graph. Ants start from random nodes and construct feasible solutions. When all ants construct their solution the pheromone is updated. The new, added, pheromone depends to the quality of the solution. The elements of the graph, which belong to better solutions will receive more pheromone and will be more desirable in the next iteration. In our implementation, we use the MAX-MIN Ant System (MMAS) which is one of the most successful ant approaches originally presented in [21]. In our case, the graph of the problem is represented by a square grid. The nodes of the graph are enumerated. The ants will deposit their pheromone on the nodes of the grid. We will deposit the sensors on the nodes of the grid too. The solution is represented by tree. An ant starts to create a solution starting from random node, which communicates with the HECN. Construction of the heuristic information is a crucial point in the ant algorithms. Our heuristic information represented by (21) is a product of three values.

$$\eta_{ij}(t) = s_{ij} l_{ij} (1 - b_{ij}) \tag{21}$$

where s_{ij} is the number of the new points (nodes of the graph) which the new sensor will cover, and which are not covered by other sensors, and

$$l_{ij} = \begin{cases} 1 & \text{if communication exists ;} \\ 0 & \text{if there is no communication.} \end{cases} \tag{22}$$

and where b_{ij} is the solution matrix. The matrix element b_{ij} equals 1 when there is sensor on this position, otherwise $b_{ij} = 0$. With s_{ij}, we try to increase the number of points covered by one sensor and thus to decrease the number of sensors we need. With l_{ij}, we guarantee that all sensors will be connected. With b_{ij} we guarantee that maximum one sensor will be mapped on the same point. The search stops when transition probability $p_{ij} = 0$ for all values of i and j. It means that there are no more free positions, or that all area is fully covered. At the end of every iteration the quantity of the pheromone is updated according to the rule: $\tau_{ij} \leftarrow \rho\tau_{ij} + \Delta\tau_{ij}$, with the increment $\Delta\tau_{ij} = 1/F(k)$ if (i, j) belongs to the non-dominated solution constructed by ant k, or $\Delta\tau_{ij} = 0$ otherwise. The parameter ρ is a pheromone decreasing parameter chosen in $[0, 1]$. This parameter ρ models evaporation in the nature and decreases the influence of old information on the search process. After that, we add the new pheromone, which is proportional to the value of the fitness function constructed as $F(k) = \frac{f_1(k)}{\max_i(f_1(i))} + \frac{f_2(k)}{\max_i(f_2(i))}$, where $f_1(k)$ is the number of sensors proposed by the k-th ant, and $f_2(k)$ is the energy of the solution of the k-th ant. These are also the objective functions of the WSN layout problem. We normalize the values of two objective functions with their maximal achieved values from the first iteration.

V. Application of the fast BF-ICrA method

In this section we present the results of the fast BF-ICrA method with the MO-ACO algorithm for WSN layout deployment. Fidanova and Roeva have developed a software, which realizes the MO-ACO algorithm. This software can solve the problem at any rectangular area, the communication and the coverage radius can be different and can have any positive value. We can have regions in the area. The program was written in C language, and the tests were run on computer with an Intel Pentium 2.8GHz processor. In their tests, they use an example where the area is square. The coverage and communication radii cover 30 points. The HECN is fixed in the centre of the area. In the sequel we consider three examples of areas with three sizes: 350×350 points, 500×500 points, and 700×700 points. The MO-ACO algorithm is based on 30 runs for each number of ants. We extract the Pareto front from the solutions of these 30 runs, and we show the achieved non dominated solutions (approximate Pareto fronts) for each case on which the BF-ICrA will be applied. The score matrices for each case is given in Tables I, II and III [20].

Table I
THE 6×10 SCORE MATRIX $\mathbf{S}$ FOR 350×350 CASE (EXAMPLE 1).

S =	ACO$_1$	ACO$_2$	ACO$_3$	ACO$_4$	ACO$_5$	ACO$_6$	ACO$_7$	ACO$_8$	ACO$_9$	ACO$_{10}$
111	30	36	30	30	30	30	30	30	30	30
112	30	36	30	30	30	30	30	30	30	30
113	28	35	28	30	30	30	28	28	28	28
114	26	26	26	26	26	26	26	26	26	26
115	26	26	26	26	26	26	26	26	26	26
116	26	26	26	26	26	26	25	25	26	25

Each row of $\mathbf{S}$ corresponds to the number of sensors used in WSN to cover the area as indicated in the first column at the left side of the score matrix. Each column of $\mathbf{S}$ corresponds to ACO$_j$ algorithm used with j ants ($j = 1, 2, \ldots, 10$). Each element S_{ij} of $\mathbf{S}$ corresponds to the energy corresponding to this number of sensors and with the number of ants used for Multiple Objective ACO algorithm.

Application of BF-ICrA in example 1 (350×350 *points)*

In this example, one sees from the score matrix of the Table I that ACO$_1$, ACO$_3$ and ACO$_9$ algorithms perform equally for all alternatives (i.e. all rows) and they define a first group/cluster of methods providing exactly the same performances. Similarly, ACO$_4$, ACO$_5$ and ACO$_6$ constitute a second group of algorithms. The third group is made of ACO$_7$, ACO$_8$ and ACO$_{10}$ algorithms. It is worth noting that these three groups $\{\text{ACO}_1, \text{ACO}_3, \text{ACO}_9\}$, $\{\text{ACO}_4, \text{ACO}_5, \text{ACO}_6\}$, and $\{\text{ACO}_7, \text{ACO}_8, \text{ACO}_{10}\}$ differ only very slightly, whereas the ACO$_2$ algorithm (i.e the 2nd column of the score matrix $\mathbf{S}$) differs a bit more from all the three aforementioned groups.

Table II
THE 22×10 SCORE MATRIX $\mathbf{S}$ FOR 500×500 CASE (EXAMPLE 2).

S =	ACO$_1$	ACO$_2$	ACO$_3$	ACO$_4$	ACO$_5$	ACO$_6$	ACO$_7$	ACO$_8$	ACO$_9$	ACO$_{10}$
223	90	96	90	90	89	81	90	90	90	90
224	61	96	89	89	88	65	61	59	57	71
225	61	96	74	58	60	58	57	58	57	57
226	59	95	73	57	59	57	56	58	57	57
227	60	57	57	57	57	56	56	57	57	57
228	60	57	57	57	57	56	56	57	54	57
229	58	57	57	55	57	56	56	56	54	56
230	57	57	57	55	57	52	56	54	54	56
231	57	55	57	55	55	52	56	54	54	56
232	57	55	55	51	54	50	52	51	54	48
233	57	55	55	51	54	50	51	51	54	48
234	57	55	55	51	53	50	51	48	53	48
235	57	55	54	51	53	50	51	48	50	48
236	57	55	54	51	53	50	51	48	50	48
237	57	55	54	51	53	50	51	48	50	48
238	57	55	53	51	53	50	51	48	50	48
239	56	55	53	50	53	50	51	48	50	48
240	53	53	53	50	53	50	51	48	50	48
241	53	53	53	50	53	50	51	48	50	48
242	53	53	53	50	53	50	51	48	50	48
243	53	53	53	50	53	50	51	48	50	48
244	53	53	53	50	52	50	51	48	50	48

Table III
THE 19×10 SCORE MATRIX $\mathbf{S}$ FOR 700×700 CASE (EXAMPLE 3).

S =	ACO$_1$	ACO$_2$	ACO$_3$	ACO$_4$	ACO$_5$	ACO$_6$	ACO$_7$	ACO$_8$	ACO$_9$	ACO$_{10}$
437	173	173	173	173	173	118	168	172	261	172
438	173	173	173	173	173	118	112	117	260	172
439	172	173	173	173	140	93	110	115	131	172
440	172	173	173	173	115	93	110	114	111	162
441	172	173	173	122	111	93	110	114	111	110
442	172	173	173	114	111	93	110	112	111	110
443	172	150	123	114	111	93	110	112	111	110
444	124	112	112	106	107	93	110	102	111	105
445	117	112	112	106	107	93	110	102	108	105
446	117	112	105	105	105	93	107	102	104	105
447	117	112	105	105	105	93	105	102	102	105
448	115	111	105	105	105	93	105	102	102	105
449	115	111	105	105	105	93	102	99	102	105
450	113	111	105	105	105	93	102	99	102	105
451	113	109	105	105	105	93	102	99	97	105
452	113	109	105	105	105	93	99	99	97	104
453	113	109	105	105	105	93	99	99	97	104
454	113	109	105	105	96	93	96	96	96	104
455	106	106	105	105	96	93	96	96	96	97

Example 1 with fast PCR6: If we apply the fast BF-ICrA method using approximate PCR6 fusion rule based on the canonical decomposition of the $M = 6$ dichotomous BBAs $(m^i_{jj'}(\theta), m^i_{jj'}(\bar{\theta}), m^i_{jj'}(\theta \cup \bar{\theta}))$, we get the matrix of mass of belief of agreement between criteria given in Table[6] IV.

Table IV
MATRIX $\mathbf{K}_{\approx PCR6}(\theta)$ FOR EXAMPLE 1.

0.865	0.821	0.865	0.790	0.790	0.790	0.806	0.806	0.865	0.806
0.821	0.928	0.821	0.950	0.950	0.950	0.805	0.805	0.821	0.805
0.865	0.821	0.865	0.790	0.790	0.790	0.806	0.806	0.865	0.806
0.790	0.950	0.790	1.000	1.000	1.000	0.795	0.795	0.790	0.795
0.790	0.950	0.790	1.000	1.000	1.000	0.795	0.795	0.790	0.795
0.790	0.950	0.790	1.000	1.000	1.000	0.795	0.795	0.790	0.795
0.806	0.805	0.806	0.795	0.795	0.795	0.843	0.843	0.806	0.843
0.806	0.805	0.806	0.795	0.795	0.795	0.843	0.843	0.806	0.843
0.865	0.821	0.865	0.790	0.790	0.790	0.806	0.806	0.865	0.806
0.806	0.805	0.806	0.795	0.795	0.795	0.843	0.843	0.806	0.843

The matrix of distances to full agreement based on fast BF-ICrA method, denoted by $\mathbf{D}_{\approx PCR6}(\theta)$, is given in Table V.

[6] All the numerical values presented in the matrices have been truncated at their 3rd digit for typesetting convenience.

Table V
MATRIX $\mathbf{D}_{\approx PCR6}(\theta)$ WITH FAST BF-ICRA FOR EXAMPLE 1.

$$\begin{bmatrix}
0.134 & 0.178 & \mathbf{0.134} & 0.209 & 0.209 & 0.209 & 0.193 & 0.193 & \mathbf{0.134} & 0.193 \\
0.178 & 0.071 & 0.178 & 0.049 & 0.049 & 0.049 & 0.194 & 0.194 & 0.178 & 0.194 \\
\mathbf{0.134} & 0.178 & 0.134 & 0.209 & 0.209 & 0.209 & 0.193 & 0.193 & 0.134 & 0.193 \\
0.209 & 0.049 & 0.209 & 0 & \mathbf{0} & \mathbf{0} & 0.204 & 0.204 & 0.209 & 0.204 \\
0.209 & 0.049 & 0.209 & \mathbf{0} & 0 & \mathbf{0} & 0.204 & 0.204 & 0.209 & 0.204 \\
0.209 & 0.049 & 0.209 & \mathbf{0} & \mathbf{0} & 0 & 0.204 & 0.204 & 0.209 & 0.204 \\
0.193 & 0.194 & 0.193 & 0.204 & 0.204 & 0.204 & 0.156 & \mathbf{0.156} & 0.193 & \mathbf{0.156} \\
0.193 & 0.194 & 0.193 & 0.204 & 0.204 & 0.204 & \mathbf{0.156} & 0.156 & 0.193 & \mathbf{0.156} \\
\mathbf{0.134} & 0.178 & 0.134 & 0.209 & 0.209 & 0.209 & 0.193 & 0.193 & 0.134 & 0.193 \\
0.193 & 0.194 & 0.193 & 0.204 & 0.204 & 0.204 & \mathbf{0.156} & \mathbf{0.156} & 0.193 & 0.156
\end{bmatrix}$$

In examining the table V, one sees that $ACO1$, $ACO3$ and $ACO9$ are at a small distance 0.134, with respect to other algorithms, so that they belong to the same group and behave similarly. Same remarks holds for the group $\{\mathrm{ACO}_4, \mathrm{ACO}_5, \mathrm{ACO}_6\}$ because its inter-distance is zero, and for the group $\{\mathrm{ACO}_7, \mathrm{ACO}_8, \mathrm{ACO}_{10}\}$ because its inter-distance is 0.156. In a relative manner ACO_2 appears closer to $\{\mathrm{ACO}_4, \mathrm{ACO}_5, \mathrm{ACO}_6\}$, than $\{\mathrm{ACO}_1, \mathrm{ACO}_3, \mathrm{ACO}_9\}$ or $\{\mathrm{ACO}_7, \mathrm{ACO}_8, \mathrm{ACO}_{10}\}$, which intuitively makes sense when comparing directly the columns of the matrix of Table I.

Example 1 with averaging fusion: The matrix of distances to full agreement based on BF-ICrA method using average fusion rule, denoted by $\mathbf{D}_{\text{Aver.}}(\theta)$, is given in Table VI.

Table VI
MATRIX $\mathbf{D}_{\text{AVER.}}(\theta)$ WITH BF-ICRA USING AVERAGING RULE FOR EXAMPLE 1.

$$\begin{bmatrix}
0.084 & 0.082 & 0.084 & 0.081 & 0.081 & 0.081 & 0.156 & 0.156 & 0.084 & 0.156 \\
0.082 & 0.030 & 0.082 & 0.016 & 0.016 & 0.016 & 0.142 & 0.142 & 0.082 & 0.142 \\
0.084 & 0.082 & 0.084 & 0.081 & 0.081 & 0.081 & 0.156 & 0.156 & 0.084 & 0.156 \\
0.081 & 0.016 & 0.081 & 0 & 0 & 0 & 0.138 & 0.138 & 0.081 & 0.138 \\
0.081 & 0.016 & 0.081 & 0 & 0 & 0 & 0.138 & 0.138 & 0.081 & 0.138 \\
0.081 & 0.016 & 0.081 & 0 & 0 & 0 & 0.138 & 0.138 & 0.081 & 0.138 \\
0.156 & 0.142 & 0.156 & 0.138 & 0.138 & 0.138 & 0.198 & 0.198 & 0.156 & 0.198 \\
0.156 & 0.142 & 0.156 & 0.138 & 0.138 & 0.138 & 0.198 & 0.198 & 0.156 & 0.198 \\
0.084 & 0.082 & 0.084 & 0.081 & 0.081 & 0.081 & 0.156 & 0.156 & 0.084 & 0.156 \\
0.156 & 0.142 & 0.156 & 0.138 & 0.138 & 0.138 & 0.198 & 0.198 & 0.156 & 0.198
\end{bmatrix}$$

One sees that only the group $\{\mathrm{ACO}_4, \mathrm{ACO}_5, \mathrm{ACO}_6\}$ can be clearly identified based on the averaging fusion rule. The other groups ACO_2 appears also close to $\{\mathrm{ACO}_4, \mathrm{ACO}_5, \mathrm{ACO}_6\}$. But ACO_1, ACO_3 and ACO_9 are closer to $\{\mathrm{ACO}_4, \mathrm{ACO}_5, \mathrm{ACO}_6\}$ also than in-between. Same remarks holds for ACO_7, ACO_8, and ACO_{10}. So one sees that the averaging fusion rule is not recommended for making the BF-ICrA in this example.

Application of BF-ICrA in example 2 (500×500 points)

Example 2 with fast PCR6: If we apply the fast BF-ICrA method using approximate PCR6 fusion rule based on the canonical decomposition of the $M = 22$ dichotomous BBAs $(m^i_{jj'}(\theta), m^i_{jj'}(\bar{\theta}), m^i_{jj'}(\theta \cup \bar{\theta}))$, we get the following matrix of distances to full agreement, denoted by $\mathbf{D}_{\approx PCR6}(\theta)$, given in Table VII.

Based on these results, one sees that no clear group can be identified but we emphasize in boldface in Table VII the minimal value for each row of the distance matrix $\mathbf{D}_{\approx PCR6}(\theta)$ (diagonal elements excluded). We see that ACO_2 is at the farthest distance of ACO_1 because $D_{12}(\theta) = 0.376$, but in the mean time ACO_2 is at closest distance to ACO_1 because $D_{2j}(\theta) > 0.376$ (for $j > 2$) as shown in second line of Table VII. So we can conclude that ACO_2 is not close to any other algorithm in fact. If we choose a ad-hoc distance threshold, say for instance 0.28, then we can identify the group $\{\mathrm{ACO}_1, \mathrm{ACO}_7, \mathrm{ACO}_8, \mathrm{ACO}_9\}$.

Table VII
MATRIX $\mathbf{D}_{\approx PCR6}(\theta)$ WITH FAST BF-ICRA FOR EXAMPLE 2.

$$\begin{bmatrix}
0.158 & 0.376 & 0.338 & 0.300 & 0.286 & 0.279 & 0.247 & 0.251 & \mathbf{0.225} & 0.280 \\
\mathbf{0.376} & 0.324 & 0.426 & 0.456 & 0.437 & 0.453 & 0.457 & 0.433 & 0.435 & 0.449 \\
\mathbf{0.338} & 0.426 & 0.407 & 0.411 & 0.382 & 0.423 & 0.418 & 0.402 & 0.393 & 0.414 \\
\mathbf{0.300} & 0.456 & 0.411 & 0.349 & 0.323 & 0.381 & 0.368 & 0.370 & 0.362 & 0.363 \\
\mathbf{0.286} & 0.437 & 0.382 & 0.323 & 0.284 & 0.348 & 0.334 & 0.334 & 0.328 & 0.333 \\
\mathbf{0.279} & 0.453 & 0.423 & 0.381 & 0.348 & 0.316 & 0.298 & 0.317 & 0.308 & 0.308 \\
\mathbf{0.247} & 0.457 & 0.418 & 0.368 & 0.334 & 0.298 & 0.235 & 0.276 & 0.255 & 0.283 \\
\mathbf{0.251} & 0.433 & 0.402 & 0.370 & 0.334 & 0.317 & 0.276 & 0.265 & 0.260 & 0.303 \\
\mathbf{0.225} & 0.435 & 0.393 & 0.362 & 0.328 & 0.308 & 0.255 & 0.260 & 0.211 & 0.304 \\
\mathbf{0.280} & 0.449 & 0.414 & 0.363 & 0.333 & 0.308 & 0.283 & 0.303 & 0.304 & 0.277
\end{bmatrix}$$

Example 2 with averaging fusion: The matrix of distances to full agreement based on BF-ICrA method using average fusion rule, denoted by $\mathbf{D}_{\text{Aver.}}(\theta)$, is given in Table VIII.

Table VIII
MATRIX $\mathbf{D}_{\text{AVER.}}(\theta)$ WITH BF-ICRA USING AVERAGING RULE FOR EXAMPLE 2.

$$\begin{bmatrix}
0.361 & 0.316 & 0.310 & 0.311 & 0.336 & 0.300 & 0.306 & 0.316 & 0.320 & 0.309 \\
0.316 & 0.125 & 0.158 & 0.198 & 0.225 & 0.187 & 0.216 & 0.225 & 0.240 & 0.206 \\
0.310 & 0.158 & 0.165 & 0.185 & 0.215 & 0.178 & 0.200 & 0.215 & 0.227 & 0.193 \\
0.311 & 0.198 & 0.185 & 0.183 & 0.216 & 0.181 & 0.197 & 0.217 & 0.231 & 0.192 \\
0.336 & 0.225 & 0.215 & 0.216 & 0.243 & 0.214 & 0.231 & 0.249 & 0.261 & 0.226 \\
0.300 & 0.187 & 0.178 & 0.181 & 0.214 & 0.159 & 0.175 & 0.194 & 0.210 & 0.176 \\
0.306 & 0.216 & 0.200 & 0.197 & 0.231 & 0.175 & 0.181 & 0.202 & 0.216 & 0.186 \\
0.316 & 0.225 & 0.215 & 0.217 & 0.249 & 0.194 & 0.202 & 0.215 & 0.229 & 0.204 \\
0.320 & 0.240 & 0.227 & 0.231 & 0.261 & 0.210 & 0.216 & 0.229 & 0.233 & 0.222 \\
0.309 & 0.206 & 0.193 & 0.192 & 0.226 & 0.176 & 0.186 & 0.204 & 0.222 & 0.183
\end{bmatrix}$$

Based on the average fusion rule there is no clear clustering of algorithms. However based on shortest inter-distance we could make the following distinct pairwise groupings $\{\mathrm{ACO}_2, \mathrm{ACO}_3\}$, $\{\mathrm{ACO}_6, \mathrm{ACO}_7\}$, $\{\mathrm{ACO}_4, \mathrm{ACO}_{10}\}$, $\{\mathrm{ACO}_8, \mathrm{ACO}_9\}$ and $\{\mathrm{ACO}_1, \mathrm{ACO}_5\}$ if necessary, but remember that average fusion rule cannot provide the best result as shown in Example 1.

Application of BF-ICrA in example 3 (700×700 points)

Example 3 with fast PCR6: If we apply the fast BF-ICrA method using approximate PCR6 fusion rule based on the canonical decomposition of the $M = 19$ dichotomous BBAs $(m^i_{jj'}(\theta), m^i_{jj'}(\bar{\theta}), m^i_{jj'}(\theta \cup \bar{\theta}))$, we get the matrix of distances to full agreement, denoted by $\mathbf{D}_{\approx PCR6}(\theta)$, given in Table IX.

Table IX
MATRIX $\mathbf{D}_{\approx PCR6}(\theta)$ WITH FAST BF-ICRA FOR EXAMPLE 3.

$$\begin{bmatrix}
0.313 & 0.388 & 0.465 & 0.498 & 0.469 & 0.500 & 0.426 & 0.451 & 0.498 & 0.477 \\
0.388 & 0.339 & 0.403 & 0.496 & 0.461 & 0.500 & 0.421 & 0.440 & 0.497 & 0.464 \\
0.465 & 0.403 & 0.348 & 0.493 & 0.456 & 0.500 & 0.416 & 0.437 & 0.495 & 0.457 \\
0.498 & 0.496 & 0.493 & 0.362 & 0.385 & 0.500 & 0.376 & 0.391 & 0.470 & 0.303 \\
0.469 & 0.461 & 0.456 & 0.385 & 0.230 & 0.380 & 0.256 & 0.288 & 0.300 & 0.324 \\
0.500 & 0.500 & 0.500 & 0.500 & 0.380 & 0 & 0.312 & 0.356 & 0.308 & 0.500 \\
0.426 & 0.421 & 0.416 & 0.376 & 0.256 & 0.312 & 0.137 & 0.185 & 0.272 & 0.330 \\
0.451 & 0.440 & 0.437 & 0.391 & 0.288 & 0.356 & 0.185 & 0.205 & 0.314 & 0.351 \\
0.498 & 0.497 & 0.495 & 0.470 & 0.300 & 0.308 & 0.272 & 0.314 & 0.283 & 0.438 \\
0.477 & 0.464 & 0.457 & 0.303 & 0.324 & 0.500 & 0.330 & 0.351 & 0.438 & 0.228
\end{bmatrix}$$

We observe that the average distance between ACO algorithms is much higher than in Tables V and VII of examples

1 and 2. This shows clearly the difficulty to precisely identify the clusters of similar algorithms because only few ACO algorithms perform actually very well for this third example. Eventually, and based on shortest inter-distance we could make the first pairwise group $\{\mathrm{ACO}_7, \mathrm{ACO}_8\}$ because $D_{78}(\theta) = 0.185$ is the minimal inter-distance we have between the ACO algorithms. Once the rows and columns of Table IX corresponding to ACO$_7$ and ACO$_8$ are eliminated, then the second best group will be $\{\mathrm{ACO}_5, \mathrm{ACO}_9\}$ because $D_{59}(\theta) = 0.300$. Similarly, we will get the group $\{\mathrm{ACO}_4, \mathrm{ACO}_{10}\}$ because $D_{4,10}(\theta) = 0.303$, and then the group $\{\mathrm{ACO}_1, \mathrm{ACO}_2\}$ because $D_{12}(\theta) = 0.388$. Finally we could also cluster ACO$_3$ with ACO$_6$ because $D_{36}(\theta) = 0.500$, although this distance of agreement is quite large to be considered as a trustable cluster.

Example 3 with averaging fusion: The matrix of distances to full agreement based on BF-ICrA method using average fusion rule, denoted by $\mathbf{D}_{\text{Aver.}}(\theta)$, is given in Table X.

Table X
MATRIX $\mathbf{D}_{\text{AVER.}}(\theta)$ WITH BF-ICRA USING AVERAGING RULE FOR EXAMPLE 3.

$$\begin{bmatrix}
0.170 & 0.154 & 0.142 & 0.221 & 0.351 & 0.350 & 0.392 & 0.345 & 0.332 & 0.298 \\
0.154 & 0.120 & 0.092 & 0.167 & 0.321 & 0.295 & 0.369 & 0.313 & 0.290 & 0.261 \\
0.142 & 0.092 & 0.042 & 0.114 & 0.289 & 0.237 & 0.342 & 0.279 & 0.242 & 0.224 \\
0.221 & 0.167 & 0.114 & 0.054 & 0.255 & 0.139 & 0.327 & 0.260 & 0.184 & 0.177 \\
0.351 & 0.321 & 0.289 & 0.255 & 0.339 & 0.245 & 0.391 & 0.355 & 0.287 & 0.324 \\
0.350 & 0.295 & 0.237 & 0.139 & 0.245 & 0 & 0.304 & 0.242 & 0.115 & 0.247 \\
0.392 & 0.369 & 0.342 & 0.327 & 0.391 & 0.304 & 0.390 & 0.368 & 0.336 & 0.387 \\
0.345 & 0.313 & 0.279 & 0.260 & 0.355 & 0.242 & 0.368 & 0.328 & 0.288 & 0.341 \\
0.332 & 0.290 & 0.242 & 0.184 & 0.287 & 0.115 & 0.336 & 0.288 & 0.190 & 0.279 \\
0.298 & 0.261 & 0.224 & 0.177 & 0.324 & 0.247 & 0.387 & 0.341 & 0.279 & 0.261
\end{bmatrix}$$

Surprisingly, the use of averaging rule provides in this example lower distance values on average with respect to values given in Table IX. However no clear clustering of algorithms can be made because only few ACO algorithms perform actually very well for this third example. If we adopt the pairwise strategy to cluster algorithms, we will obtain now as first group $\{\mathrm{ACO}_2, \mathrm{ACO}_3\}$ because $D_{23}(\theta) = 0.092$, as second group $\{\mathrm{ACO}_6, \mathrm{ACO}_9\}$ because $D_{69}(\theta) = 0.115$, as third group $\{\mathrm{ACO}_4, \mathrm{ACO}_{10}\}$ because $D_{4,10}(\theta) = 0.177$, as fourth group $\{\mathrm{ACO}_1, \mathrm{ACO}_8\}$ because $D_{18}(\theta) = 0.345$, and finally we could also cluster ACO$_5$ with ACO$_7$ because $D_{57}(\theta) = 0.391$. one sees that there is no strong correlation between results obtained from BF-ICrA based on fast PCR6 and those based on averaging rule, which is not surprising because the rules are totally different. Nevertheless the group $\{\mathrm{ACO}_4, \mathrm{ACO}_{10}\}$ is agreed by both methods here.

VI. CONCLUSIONS

The fast Belief Function based Inter-Criteria Analysis method, using the canonical decomposition of basic belief assignments defined on a dichotomous frame of discernment was applied, tested and analysed in this paper. This new method was applied for evaluating the Multiple-Objective Ant Colony Optimization (MO-ACO) algorithm for Wireless Sensor Networks (WSN) deployment. Based on the BF-ICrA outcomes it was shown a very high correlation with fast PCR6 rule for the ACO$_1$, ACO$_3$ and ACO$_9$ group, for the ACO$_4$, ACO$_5$ and ACO$_6$ group, and for the ACO$_7$, ACO$_8$ and ACO$_{10}$ group of algorithms in example 1 (case of size 350×350) as intuitively expected. This is because the considered ACO algorithms can solve the problem with good solution quality in example 1. These high correlations were not observed in the other two cases for example 2 (case of size 500×500) and 3 (case of size 700×700) because only few ACO algorithms perform actually very well for these examples. So, if we considered results in case of larger problem sizes, the BF-ICrA results show that the number of ants has the significant influence on the obtained results, as already pointed out in [20].

ACKNOWLEDGMENTS

This work is partially supported by the grant No BG05M20P001-1.001-0003, financed by the Science and Education for Smart Growth Operational Program (2014-2020) and co-financed by the European Union through the European structural and Investment funds and the Bulgarian scientific fund by the grant DN 12/5.

REFERENCES

[1] J. Dezert, A. Tchamova, D. Han, J.-M. Tacnet, Simplification of multi-criteria decision-making using inter-criteria analysis and belief functions, in Proc. of Fusion 2019 Int. Conf. on Information Fusion, Ottawa, Canada, July 2-5, 2019.

[2] K. Atanassov, D. Mavrov, V. Atanassova, Intercriteria decision making: a new approach for multicriteria decision making, based on index matrices and intuitionistic fuzzy sets. Issues IFSs GNs 11, pp. 1–8, 2014.

[3] K. Atanassov, V. Atanassova, G. Gluhchev, InterCriteria Analysis: Ideas and problems, Notes on IFS, Vol. 21, No. 1, pp. 81–88, 2015.

[4] K. Atanassov et al., An approach to a constructive simplification of multiagent multicriteria decision making problems via intercriteria analysis, C.R. de l'Acad. Bulgare des Sci., Vol. 70, No. 8, 2017.

[5] J. Dezert, F. Smarandache, A. Tchamova, D. Han, Fast Fusion of Basic Belief Assignments Defined on a Dichotomous Frame of Discernment, In Proc. of Fusion 2020 (Online) conference, Pretoria, South Africa, July 2020.

[6] J. Dezert, F. Smarandache, Canonical Decomposition of Dichotomous Basic Belief Assignment, International Journal of Intelligent Systems, pp. 1–21, 2020.

[7] G. Shafer, *A Mathematical Theory of Evidence*, Princeton Univ. Press, 1976.

[8] J. Dezert, P. Wang, A. Tchamova, *On the validity of Dempster-Shafer theory*, Proc. of Fusion 2012, Singapore, July 9–12, 2012.

[9] A. Tchamova, J. Dezert, On the Behavior of Dempster's Rule of Combination and the Foundations of Dempster-Shafer Theory, IEEE IS-2012, Sofia, Bulgaria, Sept. 6-8, 2012.

[10] J. Dezert, A. Tchamova, *On the validity of Dempster's fusion rule and its interpretation as a generalization of Bayesian fusion rule*, Int. J. of Intelligent Syst., Vol. 29, Issue 3, pages 223–252, March 2014.

[11] F. Smarandache, J. Dezert, *On the consistency of PCR6 with the averaging rule and its application to probability estimation*, Proc. of Fusion 2013, Istanbul, Turkey, July 2013.

[12] F. Smarandache, J. Dezert (Editors), *Advances and applications of DSmT for information fusion*, American Research Press, Vol. 1–4, 2004–2015, http://www.onera.fr/staff/jean-dezert?page=2

[13] F. Smarandache, J. Dezert, J.-M. Tacnet, *Fusion of sources of evidence with different importances and reliabilities*, in Proceedings of Fusion 2010 conference, Edinburgh, UK, July 2010.

[14] https://bfasociety.org/

[15] R. Yager, *On the Dempster-Shafer framework and new combination rules*, Information Sciences, Vol. 41, pp. 93–138, 1987.

[16] D. Dubois, H. Prade, *Representation and combination of uncertainty with belief functions and possibility measures*, Comput. Intell., 4, 1988.

[17] S. Fidanova, J. Dezert, A. Tchamova, Inter-criteria analysis based on belief functions for GPS surveying problems, in Proc. of IEEE Int. Symp. on INnovations in Intelligent SysTems and Applications (INISTA 2019), Sofia, Bulgaria, July 3-5, 2019.

[18] J. Dezert, D.Han, H. Yin, A New Belief Function Based Approach for Multi-Criteria Decision-Making Support, Proc. of Fusion 2016 Conf.

[19] D. Han, J. Dezert, Y. Yang, New Distance Measures of Evidence based on Belief Intervals, Proc. of Belief 2014, Oxford, UK, Sept. 2014.

[20] S. Fidanova, O. Roeva, Multi-objective ACO Algorithm for WSN Layout: InterCriteria Analysis, in Large-Scale Scientific Computing, Springer, 2020.

[21] M. Dorigo, T. Stutzle, Ant Colony Optimization, MIT Press, Cambridge, 2004.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 251–255 DOI: 10.15439/2020F91 ISSN 2300-5963 ACSIS, Vol. 21

A comparison of evolutionary and simulated annealing algorithms for bi-criteria location-scheduling problem

Mirosław Ławrynowicz
Wroclaw University of Science and Technology
27 Wyb. Wyspianskiego St, 50-370 Wroclaw, Poland
Email: miroslaw.lawrynowicz@pwr.edu.pl

Grzegorz Filcek
Wroclaw University of Science and Technology
27 Wyb. Wyspianskiego St, 50-370 Wroclaw, Poland
Email: grzegorz.filcek@pwr.edu.pl

***Abstract*—A comparison of two heuristic algorithms solving a bi-criteria joint location and scheduling (ScheLoc) problem is considered. In this strongly NP-hard problem the sum of job completion times and location investment costs are used to evaluate the solution. The first solution algorithm (EV) uses an evolutionary approach, and the second more time-efficient algorithm (SA) is based on Simulated Annealing.**

I. Introduction

IN RECENT years, the location-scheduling problem, referred to as ScheLoc, and its applications have attracted attention of many researchers (see e.g.[3], [5], [6], [10], [15], [20], [22]) The ScheLoc has been considered for the first time in [5]. Then it has been discussed and developed in many works. They differ depending on a kind of location area, type and number of machines, criterion evaluating a schedule of jobs, as well as used solution algorithms. A majority of works deals with a discrete area for the deployment of machines where a finite set of available positions for machines is known and given a priori, and a non-empty subset of this set is to be selected, (e.g., [6], [17], [15]). The evaluation of job schedule, noticed in the literature, is based on the makespan C_{max} and the sum of completion times $\sum C_j$ (see [21]). The criteria serve in the analyzed works as the assessment of ScheLoc as a whole. Some works include also other criteria for evaluation of the deployment of machines, (e.g., [11], [15]). Let us remind that the job scheduling sub-problem with different release dates is strongly NP-hard for a single machine and criterion $\sum C_j$ [16]. The other sub-problem is also NP-hard when the deployment of machines is treated as a particular case of the uncapacitated facility location problem (UFLP) or p-median problem (e.g., [13]). In consequence, ScheLoc which joins and extends those problems is at least as hard and complex as each of them, thus Scheloc is strongly NP-hard. This fact justifies the development of efficient heuristic algorithms used to solve the problem. The ScheLoc problem has been also extended to multiple criteria versions, e.g, in [17] the expected value of $\sum C_j$ together with the total location costs of machines is considered, and in [25], four criteria are taken into consideration and solution algorithm, based on the NSGA II approach, has been developed.

This paper deals with one of the bi-criteria version of the ScheLoc problem. In this problem, it is assumed that a finite number of jobs are deployed at given original locations in a planar area. Every job has to be moved from its original location to the position of the corresponding machine site which is not known in advance. Then all the jobs moved to the same machine are scheduled. In this problem the number of machines is not given, but results from its solution. The machines can be deployed only in the locations from a given finite set, and not more then one machine can be launched at each location. The solution is evaluated by two criteria: the sum of jobs completion times and investment costs of deployment of machines in particular locations. To solve this strongly NP-hard problem (authors provide the proof in other work), two heuristic solution algorithms have been adopted and compared. The first algorithm (EV) is based on the evolutionary ([2]) approach, and the second (SA) on Simulated Annealing ([12]) approach. The algorithms are compared with the use of hyper-volume indicator (see [14]) to evaluate the quality of Pareto fronts obtained by the algorithms.

The remainder of the paper is organized as follows. The mathematical model is provided in Section II, which is followed by the presentation in Section III of the solution algorithms. Section IV is devoted to the computational experiments, which allowed us to evaluate the algorithms. Final remarks complete the paper.

II. Problem formulation

We consider a set $J = \{1, 2, ..., j, ..., n\}$ of n jobs, a set $A = \{a_1, a_2, ..., a_j, ..., a_n\}$ of their origin locations where $a_j = [a_j^{(1)}, a_j^{(2)}]^{\mathrm{T}}$ represents the location of job j. The job j is characterized by execution time p_j, ready time ρ_j, and transportation speed v_j. The job needs to be performed by a single machine selected from a set of identical machines. The machines' prospective locations should be selected from a set $B = \{b_1, b_2, ..., b_i, ..., b_\mu\}$ of μ possible locations where $b_i = [b_i^{(1)}, b_i^{(2)}]^{\mathrm{T}}$. It is assumed that the number $m, 1 \leq m \leq \mu$ of employed machines is not a priori known.

Let us introduce a binary vector $y = [y_i]^{\mathrm{T}}_{i=\overline{1,\mu}}$ where $y_i = 1(0)$ if the location b_i is selected for the deployment of a machine taken from the set of identical machines (otherwise). Consequently, the index i denotes a machine deployed at the location b_i. Let the schedule of jobs be represented by a three-dimensional binary matrix $x = [x_{jik}]_{\substack{j,k=\overline{1,n}\\ i=\overline{1,\mu}}}$ in which current entry x_{jik} is equal 1(0) if the jth job is scheduled on the ith machine as the kth (otherwise). A performance of job j by machine i follows its transportation at a distance $d(a_j, b_i)$ between locations a_j and b_i if $y_i = 1$. The transport needs time $r_j(x, y) = \rho_j + \frac{1}{v_j}\sum_{k=1}^{n}\sum_{i=1}^{\mu} d(a_j, b_i) y_i x_{jik}$, which is interpreted as a release date for the job j. The p_j is the execution time of the job on the machine.

The evaluation of decisions x and y is done with the use of two criteria. The first one is the sum of completion times of all the jobs

$$q^{(1)}(x, y) \triangleq \sum\nolimits_{i=1}^{\mu}\sum\nolimits_{k=1}^{n} y_i C_{ik}(x, y) \tag{1}$$

where the auxiliary variable $C_{ik}(x, y)$ stands for the completion time of a job performed by the ith machine as the kth and depends on decision variables via constraints (5) and (6). The second criteria used is the total cost of all locations used by machines

$$q^{(2)}(y) \triangleq \sum\nolimits_{i=1}^{\mu} c_i y_i \tag{2}$$

where c_i is the location cost of b_i.

The decision variables x and y must satisfy the following constraints:

$$\sum\nolimits_{i=1}^{\mu}\sum\nolimits_{k=1}^{n} x_{jik} = 1, j = 1, 2, ..., n, \tag{3}$$

$$\sum\nolimits_{j=1}^{n} x_{jik} \le y_i, i = 1, 2, ..., \mu, k = 1, 2, ..., n, \tag{4}$$

$$\begin{aligned} C_{ik}(x, y) \ge \sum\nolimits_{j=1}^{n} (r_j(x, y) + p_j) x_{jik}, \\ i = 1, 2, ..., \mu, k = 1, 2, ..., n, \end{aligned} \tag{5}$$

$$\begin{aligned} C_{ik}(x, y) \ge C_{i,k-1}(x, y) + \sum\nolimits_{j=1}^{n} p_j x_{jik}, \\ i = 1, 2, ..., \mu, k = 2, 3, ..., n, \end{aligned} \tag{6}$$

$$1 \le \sum\nolimits_{i=1}^{\mu} y_i \le \mu, \tag{7}$$

$$\begin{aligned} \sum\nolimits_{j=1}^{n} (x_{j,i,k+1} - x_{jik}) \le 0, \\ i = 1, 2, ..., \mu, k = 1, 2, ..., n-1, \end{aligned} \tag{8}$$

$$y_i, x_{jik} \in \{0, 1\}, j, k = 1, 2, ..., n, i = 1, 2, ..., \mu. \tag{9}$$

Constraints (3) and (4) ensure that each job is performed on one position of a single machine in a launched location. The job j assigned to a single machine cannot start before its release date or completion time of the job scheduled to the same machine before j, what ensure constraints (5) and (6). The constraint (6) guarantees that the number of launched locations is between 1 and μ. To limit the number of equivalent solutions represented by matrix x, (8) is present. The last constraint defines the decision variables domains.

In consequence, the following bi-criteria optimization problem, referred to as BC_ScheLoc (Bi-Criteria ScheLoc), is solved. Given n, A, B, μ, p_j, v_j, ρ_j, c_i, $j = 1, 2, ..., n$, $i = 1, 2, ..., \mu$ find the schedule of jobs x and the locations of machines y minimizing a vector of criteria $q(x, y) = [q^{(1)}(x, y), q^{(2)}(y)]^{\mathrm{T}}$ subject to constraints (3)-(9). Let us point out that the resulting from the optimisation number of used machines m can be calculated using the formula $m = \sum_{i=1}^{\mu} y_i$.

III. Solution algorithms

Considered bi-criteria optimization problem, as it was stated in the Introduction, is strongly NP-hard. Hence, the general schemes of an evolutionary approach [2] and the Simulated Annealing metaheuristic [12] with some improvements are used.

A. Algorithm EV

The proposed algorithm uses a general evolutionary approach. The next subsection provide information about encoding of the chromosome, and used selection, mutation, and crossover operators, as well as stop condition.

1) Encoding: A chromosome E = (e_1^m, ..., e_μ^m, $e_{\mu+1}^s$, ..., $e_{\mu+n}^s$, $e_{\mu+n+1}^{job}$, ..., $e_{\mu+2n}^{job}$) $\triangleq$ ((e^m), (e^s), (e^{job})) encoded as a three-part sequence represents a candidate solution (y, x). It contains binary values e^m and e^s along with integer values from the set J for e^{job}. The binary values e_i^m directly represent y_i and allow calculating m. The value of $\sum_{l=1}^{t} e_l^m = i$ calculated for every $e_t^m = 1$ indicates the index i of the machine deployed at location t. The following mappings decode the optimization variable y and the number of launched locations (machines) m:

$$y = f_y(E) = [e_i^m]_{i=\overline{1,\mu}}^{\mathrm{T}}, \tag{10}$$

$$m = f_m(E) = \sum\nolimits_{t=1}^{\mu} e_t^m. \tag{11}$$

The $m-1$ ‘1’s in e^s located at positions w_i, i = 1, 2,..., $m-1$ and referred to as indices of separation specify the division of the set of all jobs into m subsequences of jobs assigned to individual machines. Namely, the ith index of separation indicates the first position of the $(i+1)$th subsequence of e^{job} representing jobs assigned to the machine deployed at the opened location pointed out by the $(i+1)$th in order ‘1’ in e^s. The first subsequence of jobs in e^{job}, assigned to the machine pointed out by the first in order ‘1’ in e^m, starts at $e_{\mu+n+1}^{job}$. Consequently, the three-dimensional binary matrix x can be retrieved from E by the mapping:

$$x = f_x(E) = \left[x_{j,i+1,k} = \begin{cases} 1, \quad if & \begin{aligned} & j = e_{\mu+n+w_i+k-1}^{job}, \\ & k = 1, 2, ..., w_{i+1} - w_i, \\ & w_0 = 1, \\ & i = 0, 1, ..., m-1, \end{aligned} \\ 0, \quad \text{otherwise.} \end{cases} \right] \tag{12}$$

The algorithm searches solutions iteratively and independently for each subproblem that takes into account the fixed number of machines. It starts with the generation of an initial population $\tilde{S}_i = \left\{\tilde{E}_{il}\right\}_{l=\overline{1,\alpha_i}}$ of size α_i, $i = 1, 2, ..., \mu$, which cardinality $\left|\tilde{S}_i\right| = \alpha_i$ is equal for each subproblem. The creation of solutions is made by uniformly setting the values within the genome with the assumption about the Hamming distance between each pair of chromosomes higher than ϑ. This diversification strategy helps to avoid premature convergence during the initial iterations.

2) Evolutionary operators: The Stochastic Universal Sampling method for equally weighted criteria is employed for the selection process [1]. Next, the chromosomes undergo a parallel crossover with the use of two different operators with the probability $\varphi_c = 0.95$ [4]. For the binary parts, the Count-Preserving Crossover (CPC-2) operator is applied [8]. For the integer part, the Order-based Crossover Operator (OX2) recombines solutions [23]. The Simple Inversion Mutation operator (SIM) changes the integer part of E with probability $\varphi_m = 0.01$ [7]. The binary parts of E are randomly modified by the Swap Mutation (SM) operator [19].

Finally, the stop condition is fulfilled if there is no improvement of solutions representing the Pareto front through Γ consecutive iterations. The parameter Γ_{max} restricts the maximal number of iterations for each subproblem.

B. Algorithm SA

The generation of the initial solutions set I involves a randomly assignment of all the jobs to each possible number of machines $1, 2, ..., \mu$. For $s \in \{1, 2, ..., \mu\}$, the different initial solutions are created K times, $|I| = K\mu$. The time of performing computations is constrained by $r^{\max}$ and the cooling schedule $T(i) = T(0) - \gamma i$ where i stands for the current iteration index, γ is the adjustable parameter and, the initial temperature $T(0)$ chosen experimentally so that an acceptance probability of worse solutions is close to one in the first iteration [9], [12]. The set $S(i)$ stores each best found (non-dominated in Pareto's sense) pair (x, y) of the decision variables. The new solution $(\tilde{x}, \tilde{y})$ referred to as a neighboring solution of (x, y) is determined by reassignment of all the jobs being processed on the β (adjustable parameter) machines providing the largest sum of completion times:

$$\sum_{k=1}^{n} y_{s_w} C_{s_w k}(x,y) \geq \sum_{k=1}^{n} y_{s_{w-1}} C_{s_{w-1} k}(x,y) \geq ... \\ \geq \sum_{k=1}^{n} y_{s_1} C_{s_1 k}(x,y), \beta < w, \forall_{l \in \{s_w, s_{w-1}, ..., s_1\}}(y_l = 1), \tag{13}$$

to the nearest β unoccupied feasible locations. The machines at new locations $\tilde{y}_{s_k}, \tilde{y}_{s_{k-1}}, ..., \tilde{y}_{s_\beta}$ receive the jobs from the set $\tilde{J} = \{j \in J : x_{jlk} = 1, k = 1, 2, ..., n, l = s_w, s_{w-1}, ..., s_\beta\}$. The reassignment preserves a greedy approach and requires $|\tilde{J}|$ steps. At each step, the job $j \in \tilde{J}$ is assigned in such a way that (2) is minimized with the exclusion of non-chosen jobs from $\tilde{J}$. The neighboring solution $(\tilde{x}, \tilde{y}) \in N_{(x,y)}$ replaces $(x, y) \in S(i)$ only if the dominance condition is met:

$$E(x, y, \tilde{x}, \tilde{y}) = \begin{cases} true, & q^{(1)}(\tilde{x}, \tilde{y}) < q^{(1)}(x, y) \\ & \quad \vee \quad q^{(2)}(\tilde{y}) < q^{(2)}(y) \\ false, & \text{otherwise} \end{cases} . \tag{14}$$

Consequently, the decisions $(\tilde{x}, \tilde{y})$ may undergo further changes only if the acceptance probability $\big(1 + \exp(\frac{Dist(\tilde{x},\tilde{y}) - Dist(x,y)}{T(i)})\big)^{-1}$ exceeds a value of the acceptance threshold Λ and $Dist(x, y)$ means the Euclidean distance from the ideal point $(0, 0)$. The introduced probability function is the modified version of the function described in [18] that enables an evaluation of a bi-criteria solution.

Finally, the iteration count depends directly on the initial temperature $T(0)$. Moreover, the calculation time is constrained by the parameter $r^{\max}$.

C. Tuning procedure

The tuning procedure has been conducted for each dataset, which is based on the offline approach described in detail in [24]. Three parameters of the EV with restricted domains of their possible values have been selected for tuning: $\alpha_i \in \{100, 150, .., \mathbf{350}, .., 500\}$, $\tilde{\alpha} = \{0.6\alpha_i, 0.7\alpha_i, \mathbf{0.8}\alpha_i, 0.9\alpha_i\}$, $\Gamma \in \{20, 30, .., \mathbf{50}, .., 80\}$, where the default values are marked by bold types. The default values of other parameters have been assumed as follows: $\varphi_c = 0.95$, $\varphi_m = 0.01$, $\Gamma_{\max} = 3500$, $\vartheta = 0.1(n + \mu)$. Analogously, three parameters of the SA have been selected for tuning: $\gamma \in \{0.85, \mathbf{0.9}, 0.95, 0.99\}$, $T(0) \in \{10^2, 10^3, \mathbf{10^4}, 10^5\}$, $\beta \in \{30, \mathbf{50}, .., 100\}$. The default values of other parameters have been assumed as follows: $r^{\max} = 15$ min, $K = 20$, $\Lambda = 0.5$.

IV. Computational Experiments

The purpose of the conducted computational experiments is the comparison of the EV and SA. Research includes the detailed analysis of Pareto fronts generated by the developed algorithms. The definition of quality indices and the instance generation assumptions in Subsection IV-A is followed by the evaluation in Subsection IV-B All the computations have been made using a PC with AMD Ryzen Threadripper 2970WX equipped with 32GB of RAM. The EV and SA have been implemented in the Haskell language.

A. Foundations of computational experiments

The parameters values have been randomly generated according to the uniform distribution. The locations of jobs a_j, and available locations for machines b_i, have been drown from set $[0, 1000]$, job processing times p_j from $\{20, 21, ..., 50\}$, costs of machine locations c_i from $\{70, 71, ..., 139\}$. The speed of job j movement can be calculated with the formula $v_j = 350/p_j$. We have also assumed the domains for the number of jobs n and the number of available locations for machines μ as $\{25, 50,...,150\}$ and $\{5, 10, ..., 25\}$, respectively. Finally, $\rho_j = 0$ for all the jobs.

Let us introduce a Pareto front PF $\triangleq (q_l = (q_l^{(1)}, q_l^{(2)}) : q_{l-1}^{(1)} < q_l^{(2)})_{l=\overline{1,L}}$, which is an L-element sequence of points, resulted from the run of solution algorithm. We distinguish within PF the point q_λ, which is the closest to $(0, 0)$ according to the Euclidean distance $Dist = \sqrt{(q_\lambda^{(1)})^2 + (q_\lambda^{(2)})^2}$. The values of q_λ express the trade-off between (1) and (2). For evaluation we propose a well-known hypervolume indicator

TABLE I
DEPENDENCE OF $I_{\mathrm{H.EV}}$ AND $I_{\mathrm{H.SA}}$ ON μ AND n

$(\mu.n)$	$I_{\mathrm{H.EV}}$			$I_{\mathrm{H.SA}}$			$1-(I_{\mathrm{H.SA}}/I_{\mathrm{H.EV}})$
	Max	Min	Avg	Max	Min	Avg	Avg
(5,25)	0.794	0.703	0.763	0.760	0.686	0.733	0.039
(5,50)	0.779	0.711	0.756	0.757	0.702	0.740	0.021
(5,75)	0.721	0.654	0.702	0.706	0.621	0.656	0.066
(5,100)	0.736	0.661	0.711	0.740	0.678	0.714	**-0.004**
(5,125)	0.701	0.655	0.682	0.718	0.681	0.701	**-0.028**
(5,150)	0.727	0.659	0.703	0.712	0.643	0.687	0.023
(15,25)	0.807	0.745	0.780	0.778	0.702	0.734	0.059
(15,50)	0.769	0.699	0.739	0.761	0.691	0.736	0.004
(15,75)	0.708	0.645	0.679	0.712	0.649	0.688	**-0.013**
(15,100)	0.687	0.622	0.667	0.697	0.622	0.681	**-0.021**
(15,125)	0.717	0.636	0.683	0.699	0.629	0.659	0.035
(15,150)	0.735	0.659	0.704	0.707	0.621	0.673	0.044
(25,25)	0.691	0.636	0.655	0.667	0.612	0.632	0.035
(25,50)	0.713	0.649	0.673	0.689	0.617	0.635	0.056
(25,75)	0.699	0.624	0.651	0.669	0.587	0.641	0.015
(25,100)	0.689	0.603	0.639	0.699	0.609	0.646	**-0.011**
(25,125)	0.667	0.599	0.634	0.670	0.603	0.639	**-0.008**
(25,150)	0.706	0.631	0.671	0.688	0.617	0.649	0.033

TABLE II
DEPENDENCE OF $Dist_{EV}$ AND $Dist_{SA}$ ON μ AND n

(μ,n)	$Dist_{EV}$			$Dist_{SA}$			$(1-Dist_{SA}/Dist_{EV})$*100%
	Max	Min	Avg	Max	Min	Avg	Avg
(25,25)	951	934	945	957	927	945	0.01
(15,25)	952	934	940	944	918	933	0.71
(5,25)	985	944	970	981	952	965	0.48
(25,50)	1959	1915	1934	1971	1896	1936	**-0.11**
(15,50)	1931	1896	1916	1929	1875	1905	0.56
(5,50)	2886	2824	2858	2942	2791	2867	**-0.34**
(25,75)	2986	2909	2942	3018	2920	2971	**-0.98**
(15,75)	3089	2991	3043	3037	2980	3017	0.84
(5,75)	6191	5995	6074	6314	5927	6131	**-0.94**
(25,100)	3978	3929	3954	4035	3882	3961	**-0.18**
(15,100)	4866	4753	4821	4730	4530	4662	3.29
(5,100)	10810	10312	10500	10915	10241	10508	**-0.08**
(25,125)	5388	5149	5251	5365	5146	5278	**-0.53**
(15,125)	7076	6987	7046	7077	6686	6927	1.68
(5,125)	16352	16021	16135	16465	15686	16102	0.20
(25,150)	7042	6977	7011	7090	6631	6903	1.53
(15,150)	9976	9859	9902	9984	9396	9635	**-0.53**
(5,150)	23477	22844	23059	23781	22205	22897	0.70

I_{H} [14] and $Dist$ as quality indices. Indicator I_{H} uses normalized values of criteria and measure the part of the criteria area weakly dominated by an evaluated Pareto front PF. Its analytic formula is $I_{\mathrm{H}} = \sum\limits_{l\in\overline{1,L}} (1-\bar{q}_l^{(1)})(\bar{q}_{l-1}^{(2)} - \bar{q}_l^{(2)})$ where $\bar{q}_0^{(2)} = 1$, $\bar{q}_l^{(i)} = q_l^{(i)}/q_{\max}^{(i)}$, $i = 1, 2$ are normalized values of both criteria for PF. Normalization is done with $q_{\max}^{(2)} = \sum_{i=1}^{\mu} c_i$, and $q_{\max}^{(1)} = \max\limits_{i\in\{1,2,\ldots,\mu\}} \big\{ \max\limits_{j\in\{1,2,\ldots,n\}} n * (\rho_j + \frac{1}{v_j}\sum_{k=1}^{n}\sum_{i=1}^{\mu} d(a_j, b_i))\big\} + \sum_{j=1}^{n}(n+1-j) * p_{s_j}$, where the processing times $p_{s_1} \geq p_{s_2} \geq \ldots \geq p_{s_n}$ are sorted in descending order. Increasing value of the I_{H} arranges a greater area of dominated solutions and an evenly Pareto front. In effect, the I_{H} is desired to be close to 1. Smaller $Dist$ values are preferred as the trade-off point is then closer to the reference point $(0, 0)$.

B. Results of experiments

The evaluation of the EV and SA comprises the comparison of quality indices calculated for Pareto front $\mathrm{PF}_{\mathrm{EV}}$ generated by the EV with the corresponding indices calculated for Pareto front $\mathrm{PF}_{\mathrm{SA}}$ generated by the SA. Additionally, we apply processing time $Time$ as the performance indicator. The indices calculated for the EV and SA are marked by the subscripts EV and SA. The values of both criteria and quality indices are presented in Tables I-II, where the Avg, Min, and Max mean average, minimum, and maximum, respectively. Each value is obtained through 30 independent runs of the EV and SA. Table I presents the Avg, Min, and Max values of the indicator I_{H} for the EV and SA, and comparison between them in average as $1 - (I_{\mathrm{H.SA}}/I_{\mathrm{H.EV}})$. With reference to this indicator, both algorithms have achieved similar results as values in the last column are very small (the difference

TABLE III
DEPENDENCE OF COMPUTATIONAL TIMES (IN MILLISECONDS) T_{EV} AND T_{SA} ON μ AND n

(μ, n)	T_{EV}			T_{SA}			$T_{\mathrm{EV}}/T_{\mathrm{SA}}$
	Min	Max	Avg	Min	Max	Avg	Avg
(5,50)	8246	10281	9124	6016	6804	6287	1.45
(5,150)	27526	32046	30822	20397	22219	21016	1.47
(15,50)	36251	40403	38900	17005	17664	17211	2.26
(15,150)	96155	102302	98669	44753	45996	45256	2.18
(25,50)	75371	104258	85577	28089	30676	29232	2.93
(25,150)	159050	181497	172228	57998	59936	59268	2.91

between algorithms does not exceed $6,6\%$ in favour of the EV). In majority of analyzed instances, the EV algorithm outperforms SA, but it cannot be concluded that EV is better. For some instances, the SA gets better results for all the indicators. The analysis of Table II shows that both algorithms have reached similar results and there is no possibility to indicate the better approach. However, the SA has ensured better results in most cases. The slight differences between the both algorithms do not exceed 3.3% on average. In addition, the comparison of computation times illustrated in Table III shows that the SA has surpassed the EV. In some cases, the EV has been about 3 times slower than SA, and the difference increases along with the size of the problem instance.

V. FINAL REMARKS

The primary contribution of this study deals with the analysis of evolutionary and mataheuristic-based approaches for the bi-criteria Scheloc problem. Two algorithms have been developed and evaluated with the use of the hypervolume I_{H}, $Dist$ and $Time$ indicators. As the SA has ensured faster execution times, it is strongly recommended for the critical-time applications. In addition, the minimal distance from the highly preferable point $(0, 0)$ has been mainly delivered by the SA. On the contrary, the evenly arranged Pareto fronts have been provided by the EV. Although the unambiguous interpretation of execution times, the better algorithm cannot be clearly chosen on the basis of the I_{H} and $Dist$ indicators. Finally, further research may include other criteria or seeking for more efficient approximation schemes.

REFERENCES

[1] J. E. Baker. Reducing bias and inefficiency in the selection algorithm. In *Proceedings of the Second International Conference on Genetic Algorithms on Genetic Algorithms and Their Application*, pages 14–21, Hillsdale, NJ, USA, 1987. L. Erlbaum Associates Inc.

[2] K. Deb. *Multi-objective optimization using evolutionary algorithms*, volume 16. John Wiley & Sons, 2001.

[3] D. Elvikis, H. W. Hamacher, and M. T. Kalsch. Simultaneous scheduling and location (ScheLoc): the planar ScheLoc makespan problem. *Journal of Scheduling*, 12(4):361–374, 2009.

[4] J. J. Grefenstette. Optimization of control parameters for genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, 16(1):122–128, Jan 1986.

[5] H. Hennes and H. Hamacher. *Integrated Scheduling and Location Models: Single Machine Makespan Problems*. Report in Wirtschaftsmathematik. Univ., Fachbereich Mathematik, 2002.

[6] C. Heßler and K. Deghdak. Discrete parallel machine makespan ScheLoc problem. *Journal of Combinatorial Optimization*, 34(4):1159–1186, 2017.

[7] J. H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. University of Michigan Press, 1975.

[8] Y.-C. Hou and Y.-H. Chang. A new efficient encoding mode of genetic algorithms for the generalized plant allocation problem. *Journal of Information Science and Engineering*, 20:1019–1034, 09 2004.

[9] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon. Optimization by simulated annealing: An experimental evaluation; part i, graph partitioning. *Operations research*, 37(6):865–892, 1989.

[10] M. T. Kalsch. *Scheduling-Location (ScheLoc): Models, Theory and Algorithms*. Verlag Dr. Hut, 2009.

[11] M. T. Kalsch and Z. Drezner. Solving scheduling and location problems in the plane simultaneously. *Computers & Operations Research*, 37(2):256–264, 2010.

[12] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.

[13] J. Krarup and P. M. Pruzan. The simple plant location problem: Survey and synthesis. *European Journal of Operational Research*, 12(1):36 – 81, 1983.

[14] M. Laszczyk and P. B. Myszkowski. Survey of quality measures for multi-objective optimization. Construction of complementary set of multi-objective quality measures. *Swarm and Evolutionary Computation*, 48:109–133, 2019.

[15] M. Ławrynowicz and J. Józefczyk. A memetic algorithm for the discrete scheduling-location problem with unrelated executors. In *Proc. of 24th Int. Conf. on Models and Methods in Automation and Robotics MMAR*, pages 158–163, 2019.

[16] J. K. Lenstra, A. R. Kan, and P. Brucker. Complexity of machine scheduling problems. In *Annals of discrete mathematics*, volume 1, pages 343–362. Elsevier, 1977.

[17] M. Liu, X. Liu, E. Zhang, F. Chu, and C. Chu. Scenario-based heuristic to two-stage stochastic program for the parallel machine ScheLoc problem. *International Journal of Production Research*, 57(6):1706–1723, 2019.

[18] Z. Michalewicz and D. B. Fogel. *How to solve it: modern heuristics*. Springer Science & Business Media, 2013.

[19] I. M. Oliver, D. Smith, and J. R. Holland. Study of permutation crossover operators on the traveling salesman problem. In *Genetic algorithms and their applications: proceedings of the second International Conference on Genetic Algorithms at the Massachusetts Institute of Technology, Cambridge, MA*. Hillsdale, NJ: L. Erlhaum Associates, 1987.

[20] B. Piasecki and J. Józefczyk. Evolutionary algorithm for joint task scheduling and deployment of executors. *In: Automation of Discrete Processes. Theory and Applications, Silesian University of Technology*, 1:169–178, 2018.

[21] M. Pinedo. *Scheduling: theory, algorithms and systems development*, volume 29. Springer-Verlag NY, 2012.

[22] M. Rajabzadeh, M. Ziaee, and A. Bozorgi-Amiri. Integrated approach in solving parallel machine scheduling and location (ScheLoc) problem. *International Journal of Industrial Engineering Computations*, 7(4):573–584, 2016.

[23] G. Syswerda. Scheduling optimization using genetic algorithms. *Handbook of genetic algorithms*, pages 332 – 349, 1991.

[24] E.-G. Talbi. *Metaheuristics: from design to implementation*, volume 74. John Wiley & Sons, 2009.

[25] S. Wesolkowski, N. Francetić, and S. C. Grant. TraDE: Training device selection via multi-objective optimization. In *2014 IEEE Congress on Evolutionary Computation (CEC)*, pages 2617–2624. IEEE, 2014.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 257–261 DOI: 10.15439/2020F134 ISSN 2300-5963 ACSIS, Vol. 21

A Reactive Search-Based Algorithm for Scheduling Multiprocessor Tasks on Two Dedicated Processors

Méziane Aïder
LaROMaD, USTHB
BP 32 El Alia, 16111 Alger, Algérie
Email: m-aider@usthb.dz

Fatma Zohra Baatout
LaROMaD, USTHB
BP 32 El Alia, 16111 Alger, Algérie
Email: fbaatout@usthb.dz

Mhand Hifi*
EPROAD, UPJV
7, rue du Moulin Neuf, 80000 Amiens, France
Email: hifi@u-picardie.fr

***Abstract*—In this paper, we propose a reactive search-based algorithm for solving the problem of scheduling multiprocessor tasks on two dedicated processors. An instance of the problem is characterized by a set of tasks divided into three subsets and two processors, where some tasks can be executed either on one processor or two processors. The goal of the problem is to determine the scheduling of all tasks minimizing the execution of the last assigned task. The proposed reactive search starts with a starting greedy solution. Next, a series of local operators combined with a tabu list are introduced in order to intensify the search process. The method is also reinforced with a drop and rebuild operator that is applied for diversifying the search process. Finally, the performance of the proposed method is evaluated on a set of benchmark instances, where its provided results are compared to those achieved by a recent method available in the literature. Encouraging results have been reached.**

I. Introduction

The problem of Scheduling multiprocessor Tasks on Two dedicated Processors (noted ST2P) is an NP-hard combinatorial optimization problem (cf. Hoogeveen *et al.* [8]), where its aims is to assign available tasks to two different processors. Generally, for the scheduling problems, the measures of performance are often categorized into three main groups: criteria based on completion time, criteria based on due dates, and those based on inventory cost and use. The studied problem is a special case of the scheduling problems family, where the set of tasks is divided into three groups, where the first group contains the tasks that need to be performed on the first processor, the second group contains those executed on the second processor while the third group contains the tasks that must be performed simultaneously on both processors. For such problem, on the one hand, several objective functions can be considered, like (i) minimizing the makespan, (ii) to minimize the summation of the delays of all tasks, (iii) to minimize both delays and makespan, etc. On the other hand, several versions of the scheduling problem can be accessed (i) on the number of available processors, (ii) how tasks are assigned on certain processors, etc.

Herein, we study the multiprocessor tasks scheduling on two dedicated processors problem. Its goal is to minimize the completion time of the last assigned/executed task (makespan). Such a version of the problem can be encountered in several real-world applications, like production and data transfer (cf. Manaa and Chu [10]). An instance of ST2P problem may be defined as follows: let N denote the set containing n tasks to scheduling on two dedicated processors (namely P_1 and P_2) such that a task j is released at time r_j and has to be processed without preemption during its processing time p_j and C_j is the completion time of the j-th task while C_{max} denotes the makespan of the schedule to minimize. As described in Graham *et al.* [5], such a problem is defined as $P2|f_ix_j, r_j|C_{max}$, where:

- $P2$: represents two processors on which all tasks must be executed.
- f_ix_j: means that task j is affected to both processors.
- r_j: denotes the release date of the j-th task.
- p_j: is the processing time of the j-th task when executed on the processors.
- C_{max}: denotes the makespan (completion time) of the last assigned / executed task.

The remainder of the paper is organized as follows. Section II reviews some related works tackling scheduling problems. A nice decomposition of ST2P, proposed by Manaa and Chu [10], providing a tight lower bound is given in Section III. Section IV describes the proposed reactive search-based algorithm for approximately solving ST2P. A starting solution, using a knapsack greedy rule, is described in Section IV-A. The intensification operators, combined with a tabu list, are discussed in Section IV-B. The diversification strategy, using the drop and rebuild operator, is discussed in Section IV-C. Section V exposes the experimental part, where the performance of the proposed method is evaluated on a set of benchmark instances. The provided results are compared to those achieved by a recent algorithm of the literature and to the results achieved by Manaa and Chu's lower bound. Finally, Section VI summarizes the content of the paper.

II. Related Works

The scheduling problems family contains a huge number of problem types as underlined in Brucker [3]. Generally, the performance measures for scheduling problems are often categorized into three main groups of criteria: those based on completion time, those based on due dates, and those based on inventory cost and utilization. Due to the complexity of the studied problem, there are few available papers tackling it in the literature.

Bianco *et al.* [1] tackled the problem of scheduling tasks on two dedicated processors with preemptive constraints (noted

*Corresponding Author (M. Hifi)

$P2|f_i x_j, r_j, pmtn|C_{max}$), where the task can be interrupted and completed later. An exact algorithm has been designed that is based on two steps polynomial time complexity.

Manaa and Chu [10] proposed an exact algorithm for solving the problem studied in this paper. The method is based upon a branch and bound where the internal nodes are bounded with special lower and upper bounds. The experimental part showed the performance of such a method, where it was able to solve instances up to thirty tasks within fifty minutes.

Kacem and Dammak [9] tailored an effective genetic algorithm for approximately solving the same problem. The principle of the algorithm is based upon the classical genetic principle reinforced with a constructive procedure able to provide feasible solutions for the problem. The resulting algorithm was evaluated on random instances generated following Manaa and Chu's [10] generator and the experimental evidence showed that the method was able to achieve bounds closest to those provided by Manaa and Chu's [10] tight lower bounds.

Thesen [11] designed a tabu search for tackling general multiprocessor scheduling problems. The method combines tabu strategy and local search operator. Several strategies have been considered, like random blocking related to the size of the tabu list, frequency-based penalties for diversifying the search, and the hashing operator for stocking high solutions. The experimental part showed that some combinations have better behavior than others.

Blazewicz *et al.* [2] tackled the problem of scheduling multiprocessor tasks on three dedicated processors. The authors studied the complexity analysis, where different cases were considered for which they proposed optimal solutions in polynomial time complexity.

Buffet *et al.* [4] developed two tabu search for solving the scheduling problem with m processors. A standard tabu search was followed, where a starting solution is built by respecting a legal schedule, the intensification strategy that checks possible permutations between tasks, the diversification strategy using a local search for exploring unvisited subspaces.

III. A Lower Bound for ST2P

Manaa and Chu [10] proposed a nice lower bound for ST2P that is based on relaxing the original problem into two subproblems to solve. They also proved that bound provides an optimal solution for the preemptive case of the problem, i.e., $P2|f_i x_j, r_j, pmtn|C_{max}$. The calculation of such a bound is explained in what follows.

Let $N = \{1, \ldots, n\}$ be the set of tasks and P_1 and P_2 two processors such that a task j is released at time r_j and has to be processed without preemption during its processing time p_j and C_j is the completion time of the j-th task while C_{max} denotes the makespan of the schedule to minimize. A task $j \in N$ is called a $P_1 - task$ (resp. $P_2 - task$) if it is affected to the processor P_1 (resp. P_2) while it is called $P_{12} - task$ whenever the j-th task requires simultaneously both processors P_1 and P_2; that is a bi-processor task. Then, the lower bound can be computed by splitting ST2P into two subproblems, where all bi-processor tasks are divided into two sets of mono-processor tasks each. In this case, the first (resp. second) set, noted $P_{12}^1 - Tasks$ (resp. $P_{12}^2 - Tasks$) are separately scheduled on each processor. Thus,

- $P_1 - Tasks$ and $P_{12}^1 - Tasks$ should be scheduled on processor P_1.
- $P_2 - Tasks$ and $P_{12}^2 - Tasks$ should be scheduled on processor P_2.

Finally, the optimal solution for each subproblem can be provided by processing tasks in nondecreasing order of their release dates r_j on each processor. Positioning step by step the tasks affected to each processor induces an optimal solution for each subproblem, an optimal solution C_1^{opt} for the first subproblem with processor P_1 and C_2^{opt} for the second one with processor P_2. Hence, ST2P's lower bound corresponds to

$$\max(C_1^{opt}, C_2^{opt}).$$

Note that the solution procedure used for computing the aforementioned bound is a polynomial-time algorithm with an order time complexity of $O(n \log n)$.

IV. A Reactive Search for ST2P

In this section, we expose the cooperative method for scheduling tasks on two dedicated processors problem. The main principle of the reactive search can be summarized as follows:

1) Starting the search process by an initial solution using a basic knapsack's greedy procedure (cf. Section IV-A).
2) Building an improved solution using a series of permutations (cf. Section IV-B).
3) Perturbing the search process and re-constructing a new current solution with a basic greedy procedure according to the new order (cf. Section IV-C).
4) Steps (1)-(3) are repeated until a satisfactory solution is reached.

A. A Constructive Procedure

Generating a solution is equivalent to generate a sequence of positions of the tasks on the processors. Herein, the starting solution can be provided by using a standard scheduling's greedy procedure that can be adapted for ST2P. The procedure can be viewed as a Constructive Procedure (noted CP) that applies two main steps: (i) reordering the objects (tasks) according to given criteria and (ii) selecting step by step a non-affected item (task) and assigning it to a knapsack (processor). The second step is repeated until positioning all the items (tasks) on their corresponding knapsack (processor).

Indeed, let r_j be the release date of the j-th task and p_j its processing time. Then,

1) Compute all ratios representing the processing time per release date, i.e., $\frac{p_j}{r_j}$, $j \in N$.
2) Reorder all ratios in non-increasing order; that is $\frac{p_1}{r_1} \geq \cdots \geq \frac{p_j}{r_j} \geq \cdots \geq \frac{p_n}{r_n}$.

Finally, by applying the principle of the greedy knapsack procedure to each task, according to the aforementioned order, a starting solution is provided for ST2P; that forms a sequence of tasks assigned to either the first processor, or the second processor, or both processors.

B. Intensification Search

Determining an improved solution (with a new sequence) is equivalent to solve a reduced problem by fixing some tasks. Making some moves between tasks is equivalent to fix some of them and to reassign the rest of the tasks on their corresponding processors(s).

1) A 2-opt Operator: A 2-opt operator is a simple local search/improvement procedure, which is even based upon simple local modifications of the current solution. Given a (current) feasible solution, the operator repeatedly makes some moves/swaps/shakes as long as the quality of the induced solution is improved. Whenever the improvement stagnates around the same objective value, we say that the 2-opt operator reaches its limits; that is a situation where the method is trapped into a local optimum. Herein, the 2-opt operator consists of swapping two randomly chosen positions of the sequence. The series related to these swaps induces the current neighborhood around the solution at hand.

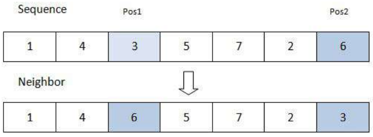


Fig. 1. The 2-opt operator

Figure 1 illustrates the swapping operator used at each step of the intensification search. One can observe that making a simple swapping between two tasks may provide either a feasible solution or (i) an unfeasible one. In the case of the unfeasible solution, we propose a repairing operator, which can be viewed as a two-step procedure. Let i and j denote the two positioned tasks (after a swap), such that i is positioned before j. Then the following two-steps procedure is applied to the provided configuration.

The first-step. The first step of the repairing operator can be applied as follows: (**i**) According to the position of the i-th task, move all tasks from the left to the right till removing all overlapping; (**ii**) According to the position of the j-th task (with its new position), move all tasks from the left to the right till removing all overlapping.

The second-step. Observe that swapping two tasks induces a new sequence and so, a simple knapsack greedy procedure CP can be applied to that order.

Hence, by applying both steps for the current solution, a series of solutions are built; that are the solutions forming the current 2-opt neighborhood.

2) A 3-opt Operator: In this section, we propose a local search based upon the 3-opt operator. As observed above (Section IV-B1), a current solution may be locally improved by using a simple 2-opt operator that is based on small moves. Herein, we propose to introduce a neighbor operator with higher freedom, which can mix two consecutive solutions around the current solution.

The idea is to repeat a series of small moves around the current solution. After some iterations, apply another search operator with higher moves and continue searching with small moves. Such a search is repeated until satisfying a predefined stopping criteria. One step of the higher move-based operator can be described as follows (let $\underline{S}$ be the current solution): (**i**) Select two random tasks from $\underline{S}$, permute both tasks for forming a new configuration S'; (**ii**) Select two random tasks from S' (different from the already swapped tasks), permute both tasks for forming a new configuration S''; (**iii**) Call the 2-opt operator on S'' for providing the best solution (noted $\underline{S}'$) around the solution at hand $\underline{S}$.

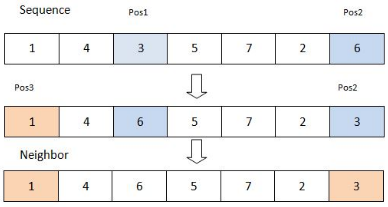


Fig. 2. The 3-opt operator

Figure 2 illustrates the steps used when applying the 3-opt operator that is applied to the current feasible solution.

3) Using a Tabu List: Generally, both 2-opt and 3-opt operators try to build a series of solutions belonging to a series of subspaces. Because a new solution built can be provided by exchanging the positions of two tasks, one can observe that repeating the same process may lead toward the same local optimum and so, the method can be trapped into that optimum. Among the techniques that can be introduced to avoid cycling towards the same solutions, the tabu search remains one of the simplest strategies that can be introduced whenever the studied problem belongs to the combinatorial optimization problems family. Because the method uses swaps between tasks, it is interesting to reinforce the search process by adding a tabu list. It contains a list of temporarily inverse-moves that avoids returning to the solutions already visited.

C. Diversification Search

The intensification strategy tries to find a series of feasible solutions to the problem, which are often considered as local optima. The objective of the building procedure is to provide a series of neighborhoods, issuing from the solution at hand, which might contain better solutions. Despite some improvements that can be realized, and because of the number of achievable solutions with the same objective value, it is interesting to provide a manner capable to drive the search process through other unvisited subspaces.

Herein, we propose a diversification search that consists of removing a subset of tasks from the current sequence (i.e. a feasible solution of the problem). The removing strategy tries to diversify the search process by degrading the quality of the solution at hand with the aim of avoiding stagnating in a local optimum. Then, a partial solution is obtained and it is completed using the constructive procedure as a tool for refining the quality of the partial solution, according to the new order associated with the remaining tasks. Such a strategy was already used with success for solving variants of the knapsack type problems (cf., Hifi [6] and Hifi and Michrafy [7]).

Herein, the diversification strategy can be applied by using the Drop and Rebuild Operator (DRO) that is described as follows. According to the current solution $\underline{S}$, DRO tries to reduce the problem, by randomly fixing a subset of tasks of $\underline{S}$, as follows. **Step 1:** From the solution $\underline{S}$, drop $\beta\%$ of the tasks belonging to that sequence; **Step 2:** Solve the reduced instance by calling the constructive procedure CP (cf., Section IV-A) and **Step 3:** Complete the current solution by calling CP, with the already removed tasks.

Algorithm 1 A Reactive Search-Based Algorithm (RSBA)

```
Input. An instance of SP2P.
Output. A feasible solution $S^\star$ with its objective value $C^\star_{max}$.
1: Set $S^\star = \emptyset$ and $C^\star_{max} = +\infty$.
2: Call CP for solving the original problem providing the solution $\underline{S}$ with objective
   value $C_{max}$.
3: repeat
4:   while (the stopping criterion is not performed) do
5:     if ($C_{max} < C^\star_{max}$) then
6:       set $S^\star = \underline{S}$ and $C^\star_{max} = C_{max}$.
7:     end if
8:     while (2-opt local iterations is not matched) do
9:       Call 2-opt using $\underline{S}$'s neighborhood and let $\underline{S}'$ be the neighbor solution
         with the best objective value $C'_{max}$.
10:      if ($C'_{max} < C^\star_{max}$) then
11:        set $S^\star = \underline{S}$ and $C^\star_{max} = C'_{max}$.
12:      end if
13:      Update the local iterations and set $\underline{S} = \underline{S}'$.
14:    end while
15:    (i) Call 3-opt using $\underline{S}$'s neighborhood and let $\underline{S}'$ be the neighbor solution with
           the best objective value $C'_{max}$.
       (ii) Set $\underline{S} = \underline{S}$ and $C_{max} = C'_{max}$.
16:  end while
17:  (i) Apply DRO to the best current solution $S^\star$ and let $\underline{S}$ be the solution reached.
     (ii) Reinitialize the 2-opt local iterations.
18: until (the global criterion is performed).
19: return $S^\star$ with its objective value $C^\star_{max}$.
```

D. An Overview of the Reactive Search

Algorithm 1 describes the main steps of the Reactive Search-Based Algorithm (denoted RSBA). The input of RSBA is an instance of SP2P and its output is an (near)optimal solution $S^\star$ with its objective value $C^\star_{max}$. The algorithm begins by generating a starting solution (line 2) provided by calling the constructive procedure CP. RSBA is composed of three loops, a global loop, and two internal loops. The global loop `repeat` from line 3 to line 18 that is applied for generating a series of current solutions, which are enhanced by using both intensification and diversification phases. Its stopping condition is defined according to the number of iterations based on the size of the instance. The first internal loop `repeat` from line 8 to line 14 serves to intensify the search by using the 2-opt operator while the second internal loop (from line 4 to line 16) is used for calling the 3-opt operator. The diversification procedure is considered whenever both internal loops stagnate on a local optimum (points (i) and (ii) of line 17). Both internal loops are embedded into the global loop `repeat` which serves to repeat the enhancement and the scattering on a new solution generated by the drop and rebuild operator DBO. The global loop is iterated until either the runtime limit or the number of global iterations is performed. Finally (line 19), RSBA returns $S^\star$, the best solution found so far with its objective value $C^\star_{max}$.

V. Computational Results

The solution method proposed in this study, the Reactive Search-Based Algorithm (noted RSBA), is evaluated on two sets instances, where each set is composed of five groups such that each group is related to the type of instances considered (as suggested in Manaa and Chu [10]). The proposed method was coded in C++ language and run on an Intel Pentium Core i7-8550U 1.99 GHz and 16 Gb of RAM. In order to evaluate the behavior of the proposed RSBA, we also compared its provided results to those achieved by both the Genetic Algorithm (noted GA) proposed in Kacem and Dammak [9]([1]) and the tight Lower Bound (noted LB) proposed in Manaa and Chu [10] (as used in Kacem and Dammak [9]).

TABLE I
Characteristics of the instances

Type of task	T1	T2	T3	T4	T5
$n1$	n	n	n	n	$[n/2]$
$n2$	$[n/2]$	n	$[n/2]$	n	$[n/2]$
$n12$	$[n/2]$	$[n/2]$	n	n	n

TABLE II
Performance of both RSBA and GA on instances of Set 1: small and medium instances

	Tasks		GA			RSBA		
	$n = 10$	LB	UB	Av. UB	T_{GA}	UB	Av. UB	T_{R}
T1	$\alpha = 0.5$	400.90	441.30	467.80	0.068	**407.20**	407.20	0.2064
	$\alpha = 1$	478.70	540.90	575.50	0.0654	**496.40**	496.40	0.1961
	$\alpha = 1.5$	789.30	845.50	907.20	0.0734	**797.90**	797.90	0.225
T2	$\alpha = 0.5$	402.40	519.10	556.00	0.0935	**476.60**	476.60	0.2606
	$\alpha = 1$	651.00	738.10	810.30	0.1025	**648.80**	648.80	0.1975
	$\alpha = 1.5$	914.80	1002.70	1067.20	0.0929	**925.90**	925.90	0.1824
T3	$\alpha = 0.5$	494.90	594.70	640.20	0.1042	**528.50**	528.50	0.1924
	$\alpha = 1$	664.00	841.30	895.20	0.0943	**696.60**	696.60	0.2354
	$\alpha = 1.5$	924.80	1062.80	1125.40	0.079	**936.10**	936.10	0.1512
T4	$\alpha = 0.5$	547.60	690.90	751.30	0.1175	**582.10**	582.10	0.1808
	$\alpha = 1$	732.20	959.10	1025.80	0.1149	**781.70**	781.70	0.2287
	$\alpha = 1.5$	674.50	767.10	811.60	0.0931	**681.20**	681.20	0.232295
T5	$\alpha = 0.5$	373.00	444.80	475.30	0.0727	**397.00**	397.00	0.2149
	$\alpha = 1$	545.00	655.40	709.40	0.0627	**560.80**	560.80	0.1938
	$\alpha = 1.5$	595.50	667.70	712.30	0.061	**601.60**	601.60	0.1736
	Average	612.57	718.09	768.70	**0.086**	**634.56**	**634.56**	0.205
	$n = 20$	LB	UB	Av. UB	T_{GA}	UB	Av. UB	T_{RSBA}
T1	$\alpha = 0.5$	354.50	405.80	427.70	0.16234	**353.20**	353.20	0.280971
	$\alpha = 1$	411.60	523.40	567.10	0.164691	**418.40**	418.50	0.191233
	$\alpha = 1.5$	536.30	656.10	691.30	0.163243	**554.10**	554.30	0.196438
T2	$\alpha = 0.5$	318.60	466.00	491.50	0.258109	**387.10**	387.10	0.232317
	$\alpha = 1$	471.10	630.60	666.40	0.256941	**491.00**	491.30	0.226608
	$\alpha = 1.5$	703.50	838.40	882.30	0.259082	**708.90**	708.90	0.234278
T3	$\alpha = 0.5$	392.80	519.90	541.50	0.245839	**396.00**	396.00	0.218839
	$\alpha = 1$	491.80	690.50	722.40	0.246592	**507.10**	507.80	0.219218
	$\alpha = 1.5$	670.70	842.50	884.50	0.245212	**680.20**	680.60	0.224639
T4	$\alpha = 0.5$	443.40	611.40	640.00	0.356819	**504.90**	504.90	0.263979
	$\alpha = 1$	568.50	810.20	853.50	0.357753	**596.00**	597.00	0.252089
	$\alpha = 1.5$	843.30	1059.20	1107.30	0.357673	**854.20**	854.20	0.265056
T5	$\alpha = 0.5$	287.10	391.30	413.10	0.156887	**319.20**	319.20	0.187833
	$\alpha = 1$	395.10	549.30	582.50	0.163567	**422.60**	422.60	0.182792
	$\alpha = 1.5$	643.00	755.80	793.60	0.158044	**651.00**	651.00	0.1956
	Average	502.09	650.03	684.31	0.237	**522.93**	**523.11**	**0.225**

The generator suggested by Manaa and Chu [10] considered five types of instances, related to the number of tasks n to use and those affected to both P_1 and P_2 and the bi-processor tasks affected to both P_1 and P_2 simultaneously: (i) the number of tasks $n = 10$ for small instances, $n = 20$ for medium-sized ones and $n = 100$ for large-scale ones, where thirty instances are considered for each value, (ii) the number n_1 (resp. n_2 and n_{12}) denotes the number of tasks assigned to the processor

[1] The code was provided by the first author for generating and testing the behavior of all methods on the same instances.

P_1 (resp. P_2 and P_{12}) and generated according to the values illustrated in Table I, where $[x]$ denotes the integral value of x, (iii) the processing time p_j related to the duration of the j-th task is randomly generated in $\{1, \dots, 50\}$ and (iv) the release date r_j of task j, is randomly generated in the interval $\{1, \dots, k\}$, where k is setting equal to $\alpha \times \frac{(s_{12}+(s_1+s_2)}{2}$ such that $\alpha \in \{0.5; 1; 1.5\}$ (the density of the instance) and s_1 (resp. s_2 and s_{12}) denotes the total duration related of the tasks belonging to P_1 (resp. P_2 and P_{12}).

TABLE III
PERFORMANCE OF BOTH RSBA AND GA ON INSTANCES OF SET 2: $n = 100$ (LARGE-SCALE INSTANCES)

	Tasks		GA			RSBA		
	n=100	LB	UB	Av. UB	T_{GA}	UB	Av. UB	T_R
T1	$\alpha = 0.5$	3708.6	5 649.8	5 839,7	4.035	**3742.4**	3 748.4	1.161
	$\alpha = 1$	4 885.8	7 711.7	7 935.4	4.014	**5 167.3**	5 261.3	1.136
	$\alpha = 1.5$	7 465.6	10 260.1	10 537.2	4.053	**7 508.4**	7 595.9	1.172
T2	$\alpha = 0.5$	3 793.1	6 753.6	6 940.8	6.502	**4 875.8**	4 877.1	1.532
	$\alpha = 1$	6 113.9	9 581.9	9 770.8	6.488	**6 463.1**	6 609.1	1.452
	$\alpha = 1.5$	9 374.6	12 641.5	12 958.6	**6.485**	9 518.9	9 621.8	1.453
T3	$\alpha = 0.5$	4 931.6	7 617.0	7 805.9	6.440	**5 012.1**	5 026.5	1.421
	$\alpha = 1$	6 181.4	10 312.3	10 570.1	6.410	**6 790.5**	6 896.2	1.399
	$\alpha = 1.5$	9 019.7	12 908.8	13 181.3	6.422	**9 134.5**	9 285.4	1.410
T4	$\alpha = 0.5$	4 981.6	8 821.4	9 032.2	10.245	**6 073.3**	6 079.3	1.877
	$\alpha = 1$	7 270.7	12 010.3	12 265.5	9.580	**8 098.7**	8 214.4	1.727
	$\alpha = 1.5$	10 918.8	15 372.1	15 769.1	9.658	**11 120.4**	11 259.6	1.739
T5	$\alpha = 0.5$	3 855.9	6 204.3	6 394.1	3.974	**4 383.8**	4 386.2	1.141
	$\alpha = 1$	4 951.8	8 257.7	8 476.0	3.960	**5 383.8**	5 468.5	1.084
	$\alpha = 1.5$	7 378.2	10 436.4	10 748.6	3.973	**7 408.7**	7 473.1	1.122
	Average	6 322.1	9 635.9	9 881.7	6.15	**6 712.11**	**6 786.85**	**1.388**

A. Behavior of RSBA vs GA on small and medium instances

First, in order to evaluate the performance of the proposed method RSBA, we compare its provided results to those of GA and to the lower bound LB of Manaa and Chu [10]. Table II shows LB, Kacem and Dammak's algorithm (GA) and those provided by RSBA. Columns 1 and 2 display the data information, column 3 reports LB of each instance, column 4 (resp. column 5 and column 6) tallies the GA's bound (resp. the average value and the average runtime over the ten trials) while column 7 (resp. column 8 and column 9) reports the best RSBA's bound (resp. the average values and the average runtime needed for the same trials). Finally, the last line of the table displays the average values of all values represented in each column (we note that the value in "boldface" (last line of the table) means that the best (average) solution values have been obtained by the considered algorithm). According to Table II, for the small instances with $n = 10$, RSBA outperforms GA although when considering the average value (the solution values over the ten trials). Indeed, RSBA realizes an average global value of 634.56 while GA provides an average global value equals to 768.70. The Gap between both values is closest to 134 units even GA's average runtime remains smaller than that of RSBA. For the medium-sized instances with $n = 20$, the same phenomenon can be observed. Indeed, the global RSBA's best value (522.93) is better than that achieved by GA (650.03). For the achieving results, GA's global average runtime is slightly greater (0.237 sec) than that needed by RSBA (0.225 sec), for the medium instances.

B. Behavior of RSBA vs GA on large-scale instances: Set 2

Herein, RSBA's behavior is analyzed on the instances of Set 2 which contains thirty instances representing more largest benchmark instances. Its achieved results are also compared to those achieved by GA and Manaa and Chu's lower bound. Table III reports the bounds achieved by RSBA, GA and LB on the instances of `Set 2`. From the table, one can observe that RSBA remains competitive when comparing its results to those achieved by GA. RSBA's average best solution value is equal to 6712.11 while that of GA is equal to $9635, 93$, which achieves a significant Gap closest to 2924. The global RSBA's average solution values are also better than those matched by GA and the average RSBA's runtime, in this case, is smaller than that needed by GA, i.e., 1.388 sec versus 6.150 sec. The average RSBA's best solution value provides an experimental approximation ratio of 1.062 when compared to Manaa and Chu's lower bound LB while GA's reaches an approximation ratio equal to 1.524. The larger the instance, more the behavior of RSBA is interesting, which also consumes a smaller runtime for this type of instance.

VI. CONCLUSION

The problem of scheduling tasks on two dedicated processors is solved with a reactive search-based algorithm. The method combines three main features: a starting solution built by tailoring a constructive greedy procedure, an intensification search introduced in order to visit a series of local solutions and a diversification strategy using the drop and rebuild operator. Finally, the experimental part showed the effectiveness of the proposed method when compared to the best available method in the literature.

REFERENCES

[1] Bianco, L., Blazewicz, J., Dell'Olmo, P. and Drozdowski, M. (1997). 'Preemptive multiprocessor task scheduling with release times and time windows', Annals of Operations Research,Vol. 70, No. 1, pp.43-55, https://doi.org/10.1023/A:1018994726051.

[2] Blazewicz, J., Dell'Olmo, P., Drozdowski, M. and Speranza, M.G (1992). 'Scheduling multiprocessor tasks on three dedicated processors'. Information Processing Letters 41 (1992) 275-280, https://doi.org/10.1016/0020-0190(92)90172-R.

[3] P. Brucker. Scheduling algorithms. Springer, ISBN 978-3-540-20524-1 4th ed. Springer Berlin Heidelberg New York, 2007.

[4] Buffet. O, Cucu. L, Idoumghar. L and Schott. R. (2010). 'Tabu Search Type Algorithms for the Multiprocessor Scheduling Problem'. Conference: Artificial Intelligence and Applications, https://hal.archives-ouvertes.fr/hal-00435241.

[5] Graham, R.L., Lower, E.L, Lenstra, J.K., Rinnoy, A.H.G. (1979) 'Optimization and Approximation in Deterministic Sequencing and Scheduling Theory': A Survey. Annals of Discrete Mathematics.V5, p287-326, https://doi.org/10.1016/S0167-5060(08)70356-X.

[6] Hifi, M. (2014). An iterative rounding search-based algorithm for the disjunctively constrained knapsack problem. Engineering Optimization. 46(8), 1109–1122, https://doi.org/10.1080/0305215X.2013.819096.

[7] Hifi M. and Michrafy M. (2006). A reactive local search-based algorithm for the disjunctively constrained knapsack problem. Journal of the Operational Research Society. 57(6), 718-726, https://doi.org/10.1057/palgrave.jors.2602046.

[8] Hoogeveen, J.A., van de Velde, S.L. and Veltman, B. (1994) 'Complexity of scheduling multiprocessor tasks with prespecified processor allocations', Discrete Applied Mathematics, Vol. 55, pp.259-272, https://doi.org/10.1016/0166-218X(94)90012-4.

[9] Kacem, A., Dammak, A. (2014)'A genetic algorithm to minimize the makespan on two dedicated processors', In IEEE, International Conference on Control, Decision and Information Technologies (CoDIT), pp.400-404, 3-5 Nov. 2014 Metz, France, doi: 10.1109/CoDIT.2014.6996927.

[10] Manaa, A., Chu, C. (2010) 'Scheduling multiprocessor tasks to minimise the makespan on two dedicated processors'. European Journal of Industrial Engineering, 4(3), https://dx.doi.org/10.1504/EJIE.2010.033331.

[11] Thesen. A. (1998) 'Design and evaluation of tabu search algorithms for multiprocessor scheduling'. Journal of Heuristics, 4: 141-160, https://doi.org/10.1023/A:1009625629722

DOI: 10.15439/2020F33
ISSN 2300-5963 ACSIS, Vol. 21

Multiprocessor Scheduling Problem with Release and Delivery Times

Natalia Grigoreva
St.Petersburg State University
Universitetskay nab. 7/9, St.Petersburg, Russia
Email: n.s.grig@gmail.com

***Abstract*—The multiprocessor scheduling problem is defined as follows: set of jobs have to be executed on parallel identical processors. For each job we know release time, processing time and delivery time. At most one job can be performed on every processor at a time, but all jobs may be simultaneously delivered. Preemption on processors is not allowed. The goal is to minimize the time, by which all tasks are delivered. Scheduling tasks among parallel processors is a NP-hard problem in the strong sense. The best known approximation algorithm is Jackson's algorithm, which generates the list schedule by selecting the ready job with the largest delivery time. This algorithm generates no delay schedules. We define an IIT (inserted idle time) schedule as a feasible schedule in which a processor can be idle at a time when it could begin performing a ready job. The paper proposes the approximation inserted idle time algorithm for the multiprocessor scheduling. We proved that deviation of this algorithm from the optimum is smaller then twice the largest processing time. To illustrate the efficiency of our approach we compared two algorithms on randomly generated sets of jobs.**

I. Introduction

We consider the problem of scheduling jobs with release and delivery times on parallel identical processors.

We consider a set of jobs $U = \{u_1, u_2, \dots, u_n\}$. For each job we know its processing time $t(u_i)$, its release time $r(u_i)$ the time at which the job is ready for performing and its delivery time $q(u_i)$. All data are integer. Set of jobs is performed on m parallel identical processors. Any processor can run any job and it can perform no more than one job at a time. Preemption is not allowed. The schedule defines the start time $\tau(u_i)$ of each job $u_i \in U$. The makespan of the schedule S is the quantity

$$C_{\max} = \max\{\tau(u_i) + t(u_i) + q(u_i) | u_i \in U\}.$$

The goal is to minimize $C_{\max}$, the time by which all jobs are delivered. Following the classification scheme proposed by Graham *et al.* [12], this problem is denoted by $P|r_i, q_i|C_{max}$.

The problem is equivalent to model $P|r_i|L_{\max}$ with due dates $d(u_i)$, rather than delivery times $q(u_i)$. The equivalence is shown by replacing each delivery time $q(u_i)$ by due date $d(u_i) = q_{\max} - q(u_i)$, where $q_{\max} = \max\{q(u_i) \mid u_i \in U\}$. In this problem the objective is to minimize the maximum lateness of jobs $L_{\max} = \max\{\tau(u_i) + t(u_i) - d(u_i) | u_i \in U\}$.

This problem relates to the scheduling problem [3], very similar problems can arise in different application fields [23]. The problem plays the main role in some important applications, for example, in the Resource Constrained Project Scheduling Problem [3], and it is NP-hard [27].

The single machine problem with release and delivery times is denoted by $1|r_j, q_j|C_{max}$ and it is NP-hard too [27]. The $1|r_j|q_j|C_{\max}$ is also a main component of several more complex scheduling problems, such that flowshop and jobshop scheduling [1], [8] and uses in real industrial application [8]. The problem $1|r_j, q_j|C_{\max}$ has been studied by many researches [5], [16], [22], [26].

The problem $P|r_j, q_j|C_{\max}$ is a generalization of the single-machine scheduling problem with release and delivery times $1|r_j, q_j|C_{\max}$. The problem arises as a strong relaxation of the multiprocessor flow shop problem [4]. The problem has been the subject of numerous papers, some of these works focus on problems with a precedence constrains [29].

Most of these studies have focused to obtain lower bounds [6], [18], the development of exact solution of the problem [7], [8] or a polynomial time approximation scheme (PTAS) [17], [21].

However, despite its practical importance, only Jackson's algorithm is used as a simple list heuristic algorithm for the $P|r_j, q_j|C_{\max}$.

The worst-case performance of Jackson's algorithm has been investigated by Gusfield [15] and Carlier [7]. Gusfield [15] examined Jackson's heuristic for the problem to minimize the maximum lateness of jobs with release times and due dates and proved that difference between the lateness given by Jackson's algorithm and the optimal lateness is bounded by $(2m-1)t_{max}/m$ and this bound is tight.

Carlier [7] proved that $C_{\max} - C_{opt} \leq 2t_{\max} - 2$, where $C_{\max}$ is the objective function of Jackson's rule schedule, and C_{opt} is the optimal makespan.

Gharbi and Haouari [11] proposed improved Jackson's algorithm which uses an $O(n \log n)$-time preprocessing procedure in order to reduce the number of jobs to be scheduled and investigated its worst-case performance.

The preprocessing procedure can be briefly described as follows. Let $j(k)$ is the job with the kth smallest release time. A condition which allows to define the start time of a job $j_0 \in \{j_1, j_2, ..., j_m\}$ at $r(j_0)$ in an optimal schedule is $r(j_0) + t(j_0) = \min\{r(j_k) + t(j_k) | k \in 1..m\} \leq r(j_{m+1})$. Then a job j_0 can be deleted from the set of jobs. This deleting rule is recursively applied to the new jobset $U \setminus \{j_0\}$. Let U_r be the set of jobs deleted according to this rule. Then the above deleting rule can be applied to the reversing problem (where by reversing the roles of the release and delivery times). Let U_q be the set of jobs deleted according to this second rule.

Therefore, the problem can be solved on a reduced jobset, denoted by UJ. Let S_{UJ} is a feasible schedule with makespan equal to $C_{max}(S_{UJ})$. Then the improved Jackson's algorithm constructs a complete schedule with makespan equal to $C_{\max} = \max\{C_{\max}(S_{UJ}), \max(r_j+t_j+q_j|j \in U_r \cup U_q)\}$.

Most of research in scheduling is devoted to the development of nondelay schedule. A nondelay schedule has been defined by Baker[2] as a feasible schedule in which a processor cannot be idle at a time when it could start performing a ready job. Kanet and Sridharam [19] defined an inserted idle time schedule (IIT)as a feasible schedule in which a processor can idle, if there is the ready job and reviewed the literature with problem setting where IIT scheduling may be required. Most of papers considered problem with single processor. It is known that an optimal schedule can be IIT schedule. Therefore,it is important to develop algorithms that can build IIT schedule.

In [13] we considered multiprocessor scheduling problem with precedence constrained and proposed the branch and bound algorithm, which use an inserted idle time algorithm for m parallel identical processors.

In [14] we investigated the inserted idle time algorithm for single machine scheduling with release times and due dates.

The goal of this paper is to propose an approximation IIT algorithm for $P|r_j, q_j|C_{max}$ problem and investigate its worst-case performance.

In order to confirm the effectiveness of our approach we tested our algorithms on randomly generated examples.

First in section 2, we propose an approximation IIT algorithm named MDT/IIT (maximum delivery time/ inserted idle time). In section 3 we investigate the worst-case performance of MDT/IIT algorithm. In section 4 we present the results of testing the algorithm. Summary of this paper is in section 5.

II. Approximation algorithm MDT/IIT

Algorithm MDT/IIT generates the schedule, in which a processor can be idle at the time when it could begin performing a job.

Let $r_{\min} = \min\{r(i) \mid i \in U\}$ and $q_{\min} = \min\{q(i) \mid i \in U\}$.

First we calculate the lower bound LB of the optimal makespan [7]:
$LB = \max\{r_{\min} + \sum_{i=1}^{n} t(i)/m + q_{\min}, \max\{r(i) + t(i) + q(i) \mid i \in U\}\}$.

Let $t_{\max} = \max\{t(i) \mid i \in U\}$.

Let a partial schedule S_k have been constructed, where k is the number of scheduling jobs. Let $C_{\max}(S_k))$ be the makespan of S_k.

Let $time_k[i]$ be the time of the termination of the processor i after completion all its jobs.

Procedure $SET(i, j, k, C_{\max}(S_k))$ sets a job j on processor i at step k and include the job j in S_k.

$SET(i, j, k, C_{\max}(S_k))$.

1) $\tau(j) := \max\{time_k[i], r(j)\}$.
2) $k := k + 1$.
3) $time_k[i] := \tau(j) + t(j)$.
4) $C_{\max}(S_k) := \max\{C_{\max}(S_{k-1}), \tau(j) + t(j) = q(j)$.

The approximation schedule S is constructed by MDT/IIT algorithm as follows:

1) Determine the processor l_0 such that
$$t_{\min}(l_0) = \min\{time_k[i] | i \in 1..m\}.$$
2) If there is no job u_i, such that $r(u_i) \le t_{\min}(l_0)$ then $t_{\min}(l_0) := \min\{r(u_i) \mid u_i \notin S_k\}$.
3) Select a job u with the largest delivery time $q(u) = \max\{q(u_i) \mid r(u_i) \le t_{\min}(l_0)\}$.
4) If $t_{\min}(l_0) > t_{\max}$ then $SET(l_0, u, k, C_{\max}(S_k))$; go to 11.
5) Select a job u^* such that $q(u^*) = \max\{q(u_i) \mid t_{\min}(l_0) < r(u_i) < t_{\min}(l_0) + t(u)\}$.
6) If there is no such job u^* or one of inequality is hold $q(u) \ge q(u^*)$ or $q(u^*) \le LB/3$, or $r(u^*) \ge t_{\max}$ then $SET(l_0, u, k, C_{\max}(S_k))$. Go to 11.
7) Calculate the idle time of the processor l_0 before the start of job u^*
$idproc(l_0) = r(u^*) - t_{\min}(l_0)$.
If $q(u^*) - q(u) < idproc(l_0)$, then $SET(l_0, u, k, C_{\max}(S_k))$. Go to 11.
8) Select a job u_1 which can be executed during the time interval $[t_{\min}(l_0), r(u^*)]$, namely such that $q(u_1) = \max\{q(u_i) \mid t_{\min}(l_0) \ge r(u_i) \ \& \ t(u_i) \le idle(u^*)\}$.
If job u_1 exists, then $SET(l_0, u1, k, C_{\max}(S_k))$. Go to 11.
9) Select the ready job u_2 such that $q(u_2) = \max\{q(u_i) \mid t_{\min}(l_0) < r(u_i) \ \& \ r(u_i) + t(u_i) \le r(u^*)\}$.
If we find u_2, then $SET(l_0, u2, k, C_{\max}(S_k))$. Go to 11.
10) $SET(l_0, u^*, k, C_{\max}(S_k))$.
11) If $k < n$, then go to 1.
12) If $k = n$, we construct the approximation schedule $S = S_n$ and we have the objective function $C_{\max}(S) = C_{\max}(S_n)$.

The algorithm sets on the processor l_0 the job u^* with the largest delivery time $q(u^*)$. If job u^* is not ready, then the processor l_0 does not work in the interval $[t_1, t_2]$, where $t_1 = t_{\min}(l_0)$, $t_2 = r(u^*)$.

In order to avoid too much idle of the processor the inequality $q(u^*) - q(u) \ge idproc(l_0)$ is verified on step 7 and if it is hold, we select job u^*. In order to use the idle time of the processor l_0 we look for job u_1 or u_2 to perform in this interval (see steps 8 and 9). Job u^* starts at $\tau(u^*) = r(u^*)$.

The MDT/IIT algorithm generates the schedule in $O(mn^2)$ times. It generates the schedule by n iterations, the processor selection requires $O(m)$ times and the job selection requires $O(n)$ time on each iteration.

III. Property of MDT/IIT algorithm

Let algorithm generate a schedule S, and for each job j we have the start time $\tau(j)$. The makespan is $C_{\max}(S) = \max\{\tau(j) + t(j) + q(j) \mid j \in U\}$.

Definition 3.1:

Critical job j_c is the first processed job such that $C_{\max}(S) = \tau(j_c) + t(j_c) + q(j_c)$.

Let C_{opt} be the length of an optimal schedule.

Theorem 3.2: $C_{\max}(S) - C_{opt} < t_{\max}(2m-1)/m$, and this bound is tight.

Proof:

Let c be the critical job then $C_{\max}(S) = \tau(c)+t(c)+q(c)$. If the processors do not idle in the time interval $[0, \tau(c)]$, then we set $\tau^* = 0$, else let

$$\tau^* = \max\{t \mid 0 < t < \tau(c)\},$$

where t is the time, when the number of processors working from time $t-1$ to t is smaller then m.

Let $J = \{v_i \in U | \tau^* \leq \tau(v_i) < \tau(c)\}$ be the set of jobs, which begin in interval $[\tau^*, \tau(c))$.

Let $\tau(j_0) = \max\{\tau(v_i) | \tau(j_0) < \tau(c) \ \& \ q(v_i) < q(c)\}$. The job j_0 is the last scheduling job with $q(j_0) < q(c)$ and $\tau(j_0) < \tau(c)$.

If there is no such work j_0, then we set $\tau(j_0) = 0$.

We consider four cases.

Case 1. There is not any idle time of processors before $\tau(c)$ and then $\tau^* = 0$.

Let $\tau(j_0) = 0$, then all jobs, which start time $\tau(v_i) < \tau(c)$, have delivery time $q(v_i) \geq q(c)$. The jobs from J must start in interval $[0, \tau_c)$, then

$$\sum_{v_i \in J} t(v_i) \geq m\tau(c)$$

and

$$C_{opt} \geq \sum_{v_i \in J} t(v_i)/m + t(c)/m + q(c) \geq \tau(c) + t(c)/m + q(c).$$

Then

$$C_{\max}(S) - C_{opt} \leq t(c) - t(c)/m < t_{\max}$$

.

Case 2. Let $0 \leq \tau(j_0) < \tau^* < t_{\max}$.

Then $q(v_i) \geq q(c), \forall v_i \in J$.

We can consider three sets of jobs:

$A_1 = \{v_i \in J | r(v_i) \geq \tau^*\}$, the jobs can start in interval $[\tau^*, \tau_c)$,

$A_2 = \{v_i \in J | r(v_i) < \tau^*\}$, the jobs can start before τ^*,

$A_3 = \{v_i \in U | \tau(v_i) \leq \tau^* - 1 \ \& \ \tau(v_i) + t(v_i) \geq \tau^*\}$. A_3 contains not more $m-1$ jobs and this jobs process in the interval $[\tau^* - 1, \tau^*]$. There are no any idle time of processors in the interval $[\tau^*, \tau(c)]$, then

$$T_A = \sum_{v_i \in A_3} (t(v_i) - 1) + \sum_{v_i \in A_1} t(v_i) + + \sum_{v_i \in A_2} t(v_i) \geq m(\tau(c) - \tau^*).$$

The jobs from set A_1 can process only after the time τ^*, but the jobs from sets A_2 and A_3 can process before τ^*. The job c can process before τ^*, if $r(c) < \tau^*$.

$$C_{opt} \geq (T_A + t(c))/m + q(c) \geq \tau(c) - \tau^* + t(c)/m + q(c).$$

Hence

$$C_{\max}(S) - C_{opt} \leq \tau^* + t(c) - t(c)/m < t_{\max}(2 - 1/m),$$

because $\tau^* < t_{max}$ (see step 3 of MDT/IIT algorithm).

Case 3. Let $t_{\max} \leq \tau^*$ and $\tau(j_0) < \tau^*$.

If $t_{\max} \leq \tau^*$ then $A_2 = \emptyset$ and the job c can process only after τ^*. Then

$$\sum_{v_i \in A_3} (t(v_i) - 1) + \sum_{v_i \in A_1} t(v_i) \geq m(\tau(c) - \tau^*).$$

$$C_{opt} \geq \tau^* + \sum_{v_i \in A_1} t(v_i)/m + t(c)/m + q(c) \geq$$

$$\geq \tau(c) - \sum_{v_i \in A_3} (t(v_i) - 1)/m + t(c)/m + q(c)$$

A_3 contains not more $m-1$ jobs, hence

$$C_{\max}(S) - C_{opt} \leq t(c) - t(c)/m + 1/m \sum_{v_i \in A_3} (t(v_i) - 1) \leq$$

$$\leq t(c) - t(c)/m + (m-1)/m(t_{\max} - 1) \leq$$

$$\leq (2t_{\max} - 1)(m-1)/m < t_{\max}(2 - 1/m).$$

Case 4. Consider the case $0 \leq \tau^* \leq \tau(j_0)$.

Let $J = \{v_i \in U | \tau(j_0) < \tau(v_i) < \tau(c)\}$.

For all $v_i \in J$ it is true, that $r(v_i) > \tau(j_0)$, otherwise the processor must process job v_i instead of j_0. $q(v_i) \geq q(c)$.

Then

$C_{opt} \geq \tau(j_0) + 1 + \sum_{v_i \in J} t(v_i)/m + t(c)/m + q(c)$.

We can see the set of jobs:

$A_3 = \{v_i \in U | \tau(v_i) \leq \tau(j_0) \ \& \ \tau(v_i) + t(v_i) \geq \tau(j_0) + 1\}$, the jobs must process in interval $[\tau(j_0), \tau(j_0)+1]$. A_3 contains m jobs. Then

$$\sum_{v_i \in A_3} (t(v_i) - 1) + \sum_{v_i \in J} t(v_i) \geq m(\tau(c) - \tau(j_0) - 1).$$

$$C_{opt} \geq \tau(j_0) + 1 + \tau(c) - \tau(j_0) - 1 - 1/m \sum_{v_i \in A_3} (t(v_i) - 1) +$$

$$+ t(c)/m + q(c) =$$

$$= \tau(c) + t(c)/m + q(c) - 1/m \sum_{v_i \in A_3} (t(v_i) - 1).$$

A_3 contains m jobs, hence

$$C_{\max}(S) - C_{opt} \leq 1/m \sum_{v_i \in A_3} (t(v_i) - 1) + t(c)(m-1)/m \leq$$

$$\leq t_{max} - 1 + t_{\max}(m-1)/m$$

TABLE I
MDT SCHEDULE $C_{\max}(MDT) = 5m - 2$

t	$m-1$	$m-1$		m	m	m
$P1$	idle	u_1	u_4	a	v_3	$v6$
$P2$	idle	u_2	u_5	v_1	v_4	idle
$P3$	idle	u_3	u_6	v_2	v_5	idle

TABLE II
OPTIMAL SCHEDULE $C_{\max} = 3m$

t	m	m			m
$P1$	v_3	a			v_6
$P2$	v_1	u_1	u_4	u_3	v_4
$P3$	v_2	u_2	u_5	u_6	v_5

TABLE IV
THE JACKSON'S SCHEDULE $C_{\max}(JR) = 4m - 1$.

t	m	$m-1$		m
$P1$	v_1	u_1	u_4	a
$P2$	v_2	u_2	u_5	idle
$P3$	v_3	u_3	u_6	idle

TABLE V
OPTIMAL SCHEDULE $C_{\max} = 2m + 1$

t	1	m			m
$P1$	idle	a			v_6
$P2$	idle	u_1	u_4	u_3	v_4
$P3$	idle	u_2	u_5	u_6	v_5

$$C_{\max}(S) - C_{opt} \leq t_{\max}(2m-1)/m - 1.$$

Now, we show that this bound is tight.

Example 3.3: Consider the m^2+m+1 jobs and m machine instance. There are $2m$ jobs $v_i : r(v_i) = 0$; $t(v_i) = m$; $q(v_i) = 0$. There are $m(m-1)$ jobs $u_i : r(u_i) = m-1$; $t(u_i) = 1$; $q(u_i) = m$ and job $a : r(a) = m-1$; $t(a) = m$; $q(a) = m$.

The makespan of MDT/IIT schedule is $C_{\max}(MDT) = 5m - 2$. The makespan of the Jackson's schedule is $C_{\max}(JR) = 4m - 1$. The optimal makespan is equal $3m$.

Table 1 shows the schedule posted by algorithm MDT/IIT, and Table 2 shows the optimal schedule, for the case $m = 3$. The first row of the table shows the time of the assignments. The next three lines indicate the tasks performed on the processors $P1, P2, P3$, respectively.

We can see that $C_{\max}(MDT) - C_{opt}$ is equal $2m-2$, that is $2t_{max} - 2$. ■

We compare schedules constructed by MDT/IIT algorithm with schedules constructed by nondelay Jackson's algorithm. Consider next example.

Example 3.4: Consider the $m^2 + 1$ jobs and m machine instance.

There are m jobs $v_i : r(v_i) = 0; t(v_i) = m; q(v_i) = 0$. There are $m(m-1)$ jobs $u_i : r(u_i) = 1$; $t(u_i) = 1$; $q(u_i) = m$ and there is job $a : r(a) = 1$; $t(a) = m$; $q(a) = m$.

The makespan of the Jackson's schedule is $C_{\max}(JR) = 4m-1$. The makespan of MDT/IIT schedule is $C_{\max}(MDT) = 3m$. The makespan of an optimal schedule is $C_{opt} = 2m + 1$.

Table 3 shows the schedule posted by algorithms MDT/IIT, Table 4 shows the Jackson's schedule schedule and Table 5 shows the optimal schedule for the case $m = 3$.

The algorithms JR and MDT are in a certain sense opposites: if the algorithm JR generates a schedule with a large error, the algorithm MDT/IIT works well and vice versa. Examples 3.3 and 3.4 illustrate this property of the algorithms. We propose the combined algorithm that builds two schedules: one by the algorithm JR, the other by the algorithm MDT and selects the best.

TABLE III
MDT SCHEDULE $C_{\max}(MDT) = 3m$

t	1	$m-1$		m	m
$P1$	idle	u_1	u_4	a	v_3
$P2$	idle	u_2	u_5	v_1	idle
$P3$	idle	u_3	u_6	v_2	idle

IV. COMPUTATION RESULT

In this section we present the results of testing the proposed algorithm on several types of tests. The quality of the schedules we estimated the average relative gap produced by each algorithm, where the gap is equal to $RT = (C_{\max} - LB)/LB$. We compared algorithms JR, MDT/IIT and the combined algorithm CA, that builds two schedules (one schedule by the algorithm JR, the other by the algorithm MDT) and selects the best solution.

The experiment considered several types of examples. The number of jobs n changed from 100 to 500.

In examples type A job processing time, release and delivery times are generated with discrete uniform distributions between 1 and n. Groups for $m = 20$ and $n = 100, 200, 300, 400, 500$ were tested. For each n we generate 30 instances. 150 instances of type A are tested. The results are given in Table 6. The first column of this table contains the number of jobs n.The columns $N_{opt}(MDT)$, $N_{opt}(JR)$ and $N_{opt}(CA)$ shows the cases (in percents) where optimal schedules were obtained by MDT method, JR method and combined method.

We can see that the problem becomes easier as n increases, because the average number of jobs per processor tends to increase. The average relative gap ranges from 4 % to 21 % for CA algorithm. The combined algorithm allows to improve RT in all cases.

TABLE VI
TYPE A. VARIATION OF n.

n	$RT(MDT)$	$RT(JR)$	$N_{opt}(CA)$	$RT(CA)$
100	0.219	0.228	0	0.216
200	0.147	0.159	0	0.141
300	0.061	0.066	0	0.058
400	0.053	0.051	0	0.052
500	0.047	0.042	0	0.039

In the next experiment we fix the number of jobs $n = 500$ and change the number of processors m from 3 to 170. For each m we generate 30 instances and a total of 240 instances are tested. The results of the experiments are shown in Table 7. The first column of this table contains the number of processors m. Table 7 shows the performance of JR, MDT and CA algorithms.

Table 7 shows that average relative gap increases when m changes from 3 to 100 and reaches a maximum at $m = 100$. Then it decreases and when $m = 170$ algorithm MDT generates 98 % optimal solutions, algorithm JR 96 % and algorithm CA generates optimal solutions for all instances. Algorithms JR and MDT give very close solutions and only with $m = 3, 20, 30, 130, 170$ the algorithm MDT has an advantage. The combined algorithm allows to improve RT in all cases.

We can see from tables 6 and 7 that the most difficult examples occur when the average number of jobs per processor is equal 5.

In the next series of tests, we restricted our instances to those types that found hard. The number of jobs n is equal to 100 and the number of processors m is equal to 20 (5 jobs on average per processor). In instances of type C we change $t_{\max}$. Type C, that were randomly generated as follows: the job processing time is generated with discrete uniform distributions between 1 and $t_{\max}$, where $t_{\ max}$ changes from 20 to 500. For each $t_{\max}$ we generate 30 instances.240 instances of type C are tested. Release and delivery times are generated with discrete uniform distributions on [1,100]. The results of the work are given in Table 8.

We can see that the problem becomes more difficult with increasing $t_{\max}$, the average relative gap increases and remains large at a $t_{\max}$ from 100 to 500. The maximum deviation is reached at $t_{\max} = 200$. The combined algorithm allows to increase (at $t_{\max} = 50$) the number of optimal solutions by 9% and to improve RT in all cases.

In the series of tests considered, the average deviation was slightly different for the algorithms JR and MDT. The combined algorithm allowed us to slightly improve the value of the objective function.

In order to get a better picture of the actual effectiveness of MDT/IIT we consider other types of instances.

In next series we consider instances in which jobs have the same processing time.

Type EJ (Equal job): The heads are drawn from the discrete uniform distribution on [1, 10] and tails from [1, 60], $n = 100$. All processing times $t_i = 60$. We can see the computational results in Table 9, where the last column F contains the difference $RT(JR) - RT(MDT)$.

We can see that for examples of Type EJ the average relative gap is less for algorithm MDT/IIT for all values of m. For $m = 50$, the average relative gap for the JR algorithm is equal 0.40, but for the MDT/IIT algorithm it is only 0.17.

Type SG (small-great) : The heads are generated from the discrete uniform distribution on [1, 10] and tails on [1, 80], $n = 100$. The processing times are drawn from the discrete uniform distribution on $[40, 60]$.

Table 10 shows the results of examples of type SG. For cases $m = 40$ and $m = 50$, there is a significant difference between the results obtained by different algorithms. The average relative gap for MDT algorithm and JR algorithm is equal 0.14 and 0.24, respectively, for $m = 40$. Algorithm MDT/IIT generated 100% of optimal solutions, whereas algorithm JR only 25% for $m = 50$. We observe from Tables 9 and 10 that MDT/IIT exhibits a good performance with instances of types EJ and SL.

Type GS(great-small): The $r(u)$ are drawn from the discrete uniform distribution on [1, 100] and $q(u)$ on [1, 20], $n = 100$. Table 11 shows the results of examples of type LS. The processing times are drawn from the discrete uniform distribution on $[1, n]$.

For examples of the type LS, the greatest deviation is observed at $m = 20$ and $m = 30$. The optimal solutions were obtained only at $m = 50$. The combined algorithm works better than each of the algorithms separately in all types of examples.

V. Conclusion

We propose an approximation IIT algorithm named MDT/IIT (maximum delivery time/ inserted idle time) for $P|r_j, q_j|C_{max}$ problem. We proved that $C_{\max}(S) - C_{opt} < t_{\max}(2m - 1)/m$, and this bound is tight, where $C_{\max}$ is the objective function of MDT/IIT schedule, and C_{opt} is the makespan of an optimal schedule. We observe that MDT/IIT algorithm exhibits a good performance with instances in which delivery times are large compared with processing times and release times.

We propose the combined algorithm that builds two schedules (one by the algorithm JR, the other by the algorithm MDT) and selects the best solution.The algorithms JR and MDT are in a certain sense opposites: if the algorithm JR generates a schedule with a large error, the algorithm MDT works well and vice versa. Computational experiments have shown that the combined algorithm works better than each of the algorithms separately.

References

[1] C. Artigues, D. Feillet, "A branch and bound method for the job-shop problem with sequence-dependent setup times",*Annals of Operations Research*, vol. 159,2008, pp.135—159.

[2] K.R. Baker,*Introduction to Sequencing and Scheduling.* John Wiley & Son, New York, 1974.

[3] P. Brucker, *Scheduling Algorithms. fifth ed.* Springer,Berlin, 2007.

[4] J. Carlier, E. Néron, "An exact algorithm for solving the multiprocessor flowshop," *RAIRO Operations Research*, vol. 34, 2000, pp. 1—25.

[5] J. Carlier, "The one machine sequencing problem." *European Journal of Operational Research*,vol.11,1982, pp. 42–47.

[6] J. Carlier, E. Pinson, " Jackson's pseudo preemptive schedule for the $Pm|rj, qj|Cmax$ scheduling problem," *Annals of Operations Research*, vol. 83, 1998, pp.41–58.

[7] J. Carlier, "Scheduling jobs with release dates and tails on identical machines to minimize the makespan."*European Journal of Operational Research*, vol. 29, 1987,pp.298—306.

TABLE VII
TYPE A. VARIATION OF m.

m	$N_{opt}(MDT)$	$RT(MDT)$	$N_{opt}(JR)$	RT(JR)	$N_{opt}(CA)$	$RT(CA)$
3	0	0.003	0	0.004	0	0.002
10	0	0.019	0	0.016	0	0.014
20	0	0.042	0	0.046	0	0.041
30	0	0.068	0	0.075	0	0.066
50	0	0.146	0	0.135	0	0.132
100	0	0.201	0	0.195	0	0.191
130	0	0.021	0	0.025	0	0.019
170	98	0.001	97	0.001	**100**	**0.000**

TABLE VIII
TYPE C. VARIATION OF $t_{\max}$.

$t_{\max}$	$N_{opt}(MDT)$	$RT(MDT)$	$N_{opt}(JR)$	RT(JR)	$N_{opt}(CA)$	$RT(CA)$
20	100	0.000	99	0.000	100	0.000
50	51	0.004	52	0.005	61	0.003
70	0	0.016	0	0.014	0	0.013
100	0	0.212	0	0.219	0	0.210
200	0	0.223	0	0.221	0	0.220
300	0	0.209	0	0.218	0	0.207
400	0	0.207	0	0.213	0	0.206
500	0	0.203	0	0.206	0	0.202

TABLE IX
TYPE EJ. VARIATION OF m.

m	$RT(MDT)$	$RT(JR)$	$RT(CA)$	F
20	0.03	0.04	0.03	0.01
30	0. 22	0.24	0.22	0.02
40	**0.26**	0.32	0.26	0.06
50	**0.17**	0.40	**0.17**	0.23

TABLE X
TYPE SG. VARIATION OF m

m	$RT(MDT)$	RT(JR)	$RT(CA)$	F
20	0.10	0.11	0.10	0.01
30	0.19	0.23	0.19	0.04
40	**0.14**	0.24	**0.14**	0.10
50	**0.000**	0.23	**0.000**	0.23

TABLE XI
TYPE GS. VARIATION OF m.

m	$RT(MDT)$	$RT(JR)$	$RT(CA)$
3	0.015	0.014	0.014
10	0.100	0.110	0.092
20	0. 239	0.236	0.227
30	0.205	0.198	0.192
50	**0.006**	0.008	**0.000**

[8] C. Chandra, Z.Liu, J. He, J,T. Ruohonen, "A binary branch and bound algorithm to minimize maximum scheduling cost," *Omega*, vol. 42, 2014,,pp.9–15.

[9] A.,Gharbi, M.,Haouari, "Minimizing makespan on parallel machines subject to release dates and delivery times," *Journal of Scheduling*,vol. 5, 2002, pp.329—355.

[10] A. Gharbi, M. Haouari, "Optimal parallel machines scheduling with availability constraints,"*Discrete Applied Mathematics* vol. 148, 2005, pp.63—87.

[11] A. Gharbi,M. Haouari, "An approximate decomposition algorithm for scheduling on parallel machines with heads and tails," *Computers & Operations Research*, vol. 34, 2007, pp.868 —883.

[12] R.L. Graham, E.L. Lawner, A.H.G. Rinnoy Kan, "Optimization and approximation in deterministic sequencing and scheduling," A survey.*Ann. of Disc. Math.*, vol. 5 (10),1979, pp. 287–326.

[13] N.S.Grigoreva, "Branch and bound method for scheduling precedence constrained tasks on parallel identical processors", Lecture Notes in Engineering and Computer Science. *In proc. of The World Congress on Engineering 2014, WCE 2014 London, U.K.* 2014, pp.832–836.

[14] N.Grigoreva, "Single Machine Inserted Idle Time Scheduling with Release times and Due Dates," *Proc.DOOR2016. Vladivostoc,Russia. Sep.19-23.2016. CEUR-WS.*2016, vol.1623, pp. 336—343.

[15] D.Gusfield, "Bounds for naive multiple machine scheduling with release times and deadlines,"*Journal of Algorithms* vol.5,1984, pp.1—6.

[16] L.A.Hall, D.B. Shmoys, "Jackson's rule for single-machine scheduling: making a good heuristic better," *Mathematics of Operations Research.* vol.17 (1),1992, pp.22–35.

[17] L.A.Hall,D.B. Shmoys, "Approximation schemes for constrained scheduling problems", *Proceedings of the 30th IEEE Symposium on Foundations of Computer Science,* 1989, pp. 134 —139.

[18] M. Haouari, A. Gharbi, "Lower bounds for scheduling on identical parallel machines with heads and tails,"*Annals of Operations Research* vol. 129, 2004, pp.187—204.

[19] J. Kanet, V. Sridharan, "Scheduling with inserted idle time:problem taxonomy and literature review," *Oper.Res.* vol.48 (1),2000, pp. 99–110.

[20] J.A Lenstra, A.H.G. Rinnooy Kan, P. Brucker, "Complexity of machine scheduling problems,"*Ann. of Disc. Math.*, vol. 1,1977, pp.343–362.

[21] M. Mastrolilli, "Efficient approximation schemes for scheduling prob-

lems with release dates and delivery times,"*Journal of Scheduling*, vol. 6, 2003, pp.521—531.

[22] E. Nowicki, C. Smutnicki, "An approximation algorithm for a single-machine scheduling problem with release times and delivery times," *Discrete Applied Mathematics* ,vol. 48, 1994, pp.69–79.

[23] J. Omer, A. Mucherino, "Referenced Vertex Ordering Problem",*Theory, Applications and Solution Methods HAL open archives, hal-02509522, version 1,* March, 2020.

[24] K. Sourirajan, R. Uzsoy, "Hybrid decomposition heuristics for solving large-scale scheduling problems in semiconductor wafer fabrication," *Journal of Scheduling,* vol. 10, 2007, pp.41–65.

[25] Y. Pan, L. Shi, "Branch and bound algorithm for solving hard instances of the one-machine sequencing problem,"*European Journal of Operational Research,* 168, 2006, pp. 1030—1039.

[26] C.N. Potts, "Analysis of a heuristic for one machine sequencing with release dates and delivery times," *Operational Research.* vol. 28 (6), 1980, pp. 445–462.

[27] J. Ullman, "NP-complete scheduling problems," *J. Comp. Sys. Sci.* vol. 171, 1975,pp. 394—394.

[28] Y. Zinder, D. Roper, "An iterative algorithm for scheduling unit-time operations with precedence constraints to minimize the maximum lateness,"*Annals of Operations Research,* 81, 1998, pp.321–340.

[29] Y. Zinder, " An iterative algorithm for scheduling UET tasks with due dates and release times," *European Journal of Operational Research,* vol.149, 2003, pp.404–416.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 271–275 DOI: 10.15439/2020F4 ISSN 2300-5963 ACSIS, Vol. 21

On Finding the Optimal Tree of a Complete Weighted Graph

Seyed Soheil Hosseini, Nick Wormald, Tianhai Tian
School of Mathematics,
Monash University,
Victoria 3800, Australia
Email: {soheil.hosseini, nick.wormald, tianhai.tian}@monash.edu

***Abstract*—We want to find a tree where the path length between any two vertices on this tree is as close as possible to their corresponding distance in the complete weighted graph of vertices upon which the tree is built. We use the residual sum of squares as the optimality criterion to formulate this problem, and use the Cholesky decomposition to solve the system of linear equations to optimize weights of a given tree. We also use two metaheuristics, namely Simulated Annealing (SA) and Iterated Local Search (ILS) to optimize the tree structure. Our results suggest that SA and ILS both perform well at finding the optimal tree structure when the dispersion of distances in the complete graph is large. However, when the dispersion of distances is small, only ILS has a solid performance.**

I. Introduction

We want to find an edge-weighted tree that best estimates the complete weighted graph of distances between vertices such that the discrepancy between the path length between any two vertices in the tree, and their distance in the complete graph, is minimized. To this end, we use the residual sum of squares (RSS) as it is a typical optimality criterion for these types of problems. We call the resulting tree, the residual sum of squares optimal tree (RSSOT). The underlying idea for this problem originates from three areas: stock-correlation networks, phylogenetic trees [1] and t-spanners [2] in graph theory. In the first two areas, several algorithms have been proposed to build a network based on the complete weighted graph of distances between stocks [3]–[10] or species information [11]. In the third area, the problem is similar to estimating the t-spanner tree of K_n.

We take an approach similar to some investigations in phylogenetic trees [11], but we have a different treatment of a basic improvement step used in local search heuristics. Also, in contrast to phylogenetic trees, we consider distances between all vertices of the tree, not just leaves. We investigate two metaheuristics—Simulated Annealing (SA) and Iterated Local Search (ILS)—for this problem.

In Section II, we discuss how to optimize edge weights of a given tree. In Section III, we use the aforementioned metaheuristics to optimize the tree structure—find RSSOT—and ultimately, Sections IV and V include our results and conclusion respectively.

II. Sub-problem: Tree weight optimization

For the complete weighted graph $K_n = (V, \mathbf{E}, d)$, we want to come up with a weighted spanning tree $T = (V, E, w)$ where $E \subset \mathbf{E}$ such that the path length between any two vertices on the tree best estimates the distance between them in K_n. To be precise, we want to minimize the RSS between path lengths in T and their corresponding edge distances in K_n such that

$$RSS\left(T, K_n\right) = \sum_{\substack{m,k\\ m<k}} \left(S\left(P_{m,k}\right) - d_{mk}\right)^2 . \tag{1}$$

In the equation above, $P_{m,k}$ denotes the path connecting vertices v_m and v_k, and $S(P_{m,k})$ denotes the sum of edge weights on this path. For example, for the path $P_{m,k} = (e_{ma}, e_{ab}, e_{bc}, \ldots, e_{dk})$, $S(P_{m,k}) = w_{ma} + w_{ab} + w_{bc} + \ldots + w_{dk}$. Thus, equation (1) can be reformulated as

$$RSS(T, K_n) = \sum_{\substack{m,k\\ m<k}} \left(\sum_{\substack{i,j\\ e_{ij}\in P_{m,k}}} w_{ij} - d_{mk} \right)^2 . \tag{2}$$

In order to find the edge weights for a given spanning tree, we take the derivative of RSS with respect to the w_{ij}'s, so that $\frac{\partial RSS}{\partial w_{ij}} = 0$. It gives us

$$\begin{aligned}\frac{\partial RSS}{\partial w_{ij}} = 2 \Bigg(&\sum_{\substack{m,k:\, e_{ij}\in P_{m,k}\\ m<k}} \Bigg(w_{ij} + \sum_{\substack{e_{rs}\in P_{m,k}\\ e_{rs}\neq e_{ij}}} w_{rs} \Bigg) \\ &- \sum_{\substack{m,k:\, e_{ij}\in P_{m,k}\\ m<k}} d_{mk} \Bigg) = 0 \qquad \forall e_{ij} \in E.\end{aligned} \tag{3}$$

The equation above can be written as

$$\begin{aligned}\frac{\partial RSS}{\partial w_{ij}} = 2 \Bigg(&\alpha_{ij} w_{ij} + \sum_{e_{rs}\neq e_{ij}} \beta_{rsij} w_{rs} \\ &- \sum_{\substack{m,k:\, e_{ij}\in P_{m,k}\\ m<k}} d_{mk} \Bigg) = 0 \qquad \forall e_{ij} \in E\end{aligned} \tag{4}$$

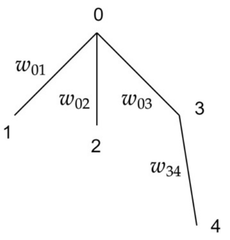


Fig. 1: An example of what matrix A and vector $\mathbf{d}$ look on this tree

where α_{ij} denotes the number of paths that edge e_{ij} is on, and β_{rsij} denotes the number of paths on which both edges e_{ij} and e_{rs} are. The reason being each term $(.)^2$ in RSS denotes the square error between a path length in T and its corresponding edge distance in K_n. We have $\binom{n}{2}$—equal to the number of paths between each two vertices in T—of these terms. Taking the derivative with respect to a w_{ij}, we are considering only the terms $(.)^2$ that include the edge e_{ij} which correspond to the paths that include edge e_{ij}. From equations (3) and (4), we have the following $n-1$ equations

$$\frac{\partial RSS}{\partial w_{ij}} = \alpha_{ij} w_{ij} + \sum_{e_{rs}\neq e_{ij}} \beta_{rsij} w_{rs} = \sum_{\substack{m,k:\, e_{ij}\in P_{m,k}\\ m<k}} d_{mk} \qquad \forall e_{ij} \in E. \tag{5}$$

The above linear system can be expressed in matrix form as $A\mathbf{w} = \mathbf{d}$, where the entries of matrix A are as follows. a_{ij} denotes the number of paths including the edge corresponding to the i*-th* entry of the vector $\mathbf{w}$ where $i = j$, and where $i \neq j$, it denotes the number of paths including the edges corresponding to the i*-th* and the j*-th* entries of the vector $\mathbf{w}$. Let us go through the following example to make it more clear.

For the tree in Fig. 1, the system of equations is as below.

$$\underbrace{\begin{bmatrix} 4 & 1 & 2 & 1 \\ 1 & 4 & 2 & 1 \\ 2 & 2 & 6 & 3 \\ 1 & 1 & 3 & 4 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} w_{01} \\ w_{02} \\ w_{03} \\ w_{34} \end{bmatrix}}_{\mathbf{w}} = \underbrace{\begin{bmatrix} d_{01}+d_{12}+d_{13}+d_{14} \\ d_{02}+d_{12}+d_{23}+d_{24} \\ d_{03}+d_{04}+d_{23}+d_{13}+d_{14}+d_{24} \\ d_{34}+d_{04}+d_{24}+d_{14} \end{bmatrix}}_{\mathbf{d}} \tag{6}$$

In the linear system above, the diagonal entries of A—a_{11}, a_{22}, a_{33} and a_{44}—are the number of paths passing respectively through the edges e_{01}, e_{02}, e_{03} and e_{34}. Also, for example, a_{12} is the number of paths passing through both edges e_{01} and e_{02}, and a_{34} is the number of paths passing through both edges e_{03} and e_{34}. In vector $\mathbf{d}$, in the first entry—$d_{01}+d_{12}+d_{13}+d_{14}$—the indices correspond to the beginning vertex and end vertex of the paths that the edge e_{01} is on, and d_{ij} is the distance between the vertices v_i and v_j in the complete graph.

The question is how do we count the number of paths passing through one specific edge or two specific edges in a tree effectively? Let us take one vertex of the tree as the root vertex and consider the tree directed based on that vertex where D_i denotes the descendants of vertex v_i. Also, α_{ij} and β_{ijrs} are as defined in equation (4). To answer the first part of the question—the number of paths passing through e_{ij} where $v_j \in D_i$—$\alpha_{ij} = (|D_j|+1)\,(n-(|D_j|+1))$. To answer the second part of the question—to count the number of edges passing through two edges—say, e_{ij} and e_{rs} where $v_j \in D_i$ and $v_s \in D_r$,

$$\beta_{ijrs} = \begin{cases} (|D_j|+1)(|D_s|+1) & D_j \cap D_s = \emptyset \\ (|D_j|+1)\,(n-(|D_s|+1)) & D_j \subset D_s \\ (|D_s|+1)\,(n-(|D_j|+1)) & D_s \subset D_j. \end{cases} \tag{7}$$

It can be seen that only the number of descendants of the bottom vertices of the edges e_{ij} and e_{ijrs} is factored in α_{ij} and $\beta ijrs$.

After finding all entries of A, we can find the edge weights by solving $A\mathbf{w} = \mathbf{d}$. Yet, is the matrix A necessarily invertible? In the following, we prove that not only is A invertible, but positive-definite.

Lemma 1. *A is a positive-definite matrix.*

Proof. We define the function Z on a spanning tree T as follows. For each edge e_{ij}, we assign a variable v_{ij}. Then we define $Z = \sum\limits_{\substack{m,k\\ m<k}} \left(\sum\limits_{\substack{i,j\\ e_{ij}\in P_{m,k}}} v_{ij} \right)^2$. We can see that the terms $(.)^2$ in Z are the same as those in RSS (equation (2)). The only difference being the variables w_{ij} are replaced with v_{ij} and the constants d_{ij} are replaced with 0. Z can be written as $Z = \mathbf{v}^\top B\mathbf{v} > 0$ where $\mathbf{v}$ is the vector of variables v_{ij}, and B is a matrix whose entries are as follows. b_{pq} is the number of terms $(.)^2$ in Z including the variable v_{ij} assigned to the p*-th* entry of vector $\mathbf{v}$ for $p = q$, and for $p \neq q$, b_{pq} is the number of terms $(.)^2$ including both variables v_{ij} and v_{rs} assigned to the p*-th* and q*-th* entries of vector $\mathbf{v}$. Since each term $(.)^2$ denotes a path in T, we can say that b_{pq} is the number of paths including the edge e_{ij} assigned the p*-th* entry of vector $\mathbf{v}$ for $p = q$, and for $p \neq q$, b_{pq} is the number of paths including both edges e_{ij} and e_{rs} assigned the p*-th* and q*-th* entries of vector $\mathbf{v}$. Thus, $B = A$, and since B is positive-definite, A is also a positive definite matrix. □

Since A is positive-definite, we can use the Cholesky decomposition of A in the form $A = LL^\top$ where L is a unique lower triangular matrix whose entries are computed by equations (8) and (9). From there, we can solve $Ly = \mathbf{d}$, and then $L^\top\mathbf{w} = y$ to find the weights. In the following, we discuss how to optimize the tree structure—find RSSOT.

$$L_{ii} = \sqrt{A_{ii} - \sum_{k=1}^{i-1} L_{ik}^2} \tag{8}$$

$$L_{ij} = \frac{1}{L_{jj}} \left(A_{ij} - \sum_{k=1}^{j-1} L_{ik} L_{jk} \right) \tag{9}$$

III. Problem: Tree structure optimization

So far, we discussed how we can find the edge weights for a given tree based on the distances in the complete graph. The question is, how can we find the tree with minimum RSS? In other words, how can we optimize the tree structure to find RSSOT? We can build n^{n-2} spanning trees on any n number of labeled vertices. That means for as few as 50 labeled vertices, we can have roughly as many spanning trees as the number of atoms in the known universe. Due to the large scale of the problem, we make use of two metaheuristics—in this case, Simulated Annealing (SA) and Iterated Local Search (ILS)—to approximate the optimal tree. These are two of the typical metaheuristics applied to such difficult optimization problems. Below, we explain how to make a structure change in a tree, and how to use SA and ILS to optimize the tree structure based on the structure change.

A. Tree structure change for optimization

Before discussing SA and ILS on a tree, let us explain how we make a change in the structure of a given tree in order ot accept or reject the transition between two states. Let T_t be the tree at time t and let us denote its corresponding structure by $T(V, E)$. Let us also denote the structure after change by $T'(V, E')$—the structure that we want to accept or reject. For $v_i \in V$, we denote the neighbours of v_i by $N(v_i)$. We pick one edge $e_{ij} \in E$. Then we define set C as $C = N(v_i) \cup N(v_j) \setminus \{v_i, v_j\}$. We pick $v_k \in C$ uniformly at random. If $v_k \in N(i)$, then $E' = E \cup \{e_{jk}\} \setminus \{e_{ik}\}$; otherwise, if $v_k \in N(j)$, then $E' = E \cup \{e_{ik}\} \setminus \{e_{jk}\}$. We denote the former structure change by $SC(T, e_{ij}, e_{jk}, e_{ik})$ and the latter by $SC(T, e_{ij}, e_{ik}, e_{jk})$. In $SC(T, ., ., .)$, the second, third, and forth terms are respectively the picked edge, the edge that is added to, and the edge that is removed from the tree.

The other thing we investigate before discussing SA and ILS algorithms on a tree is the change in matrix A and vector $\mathbf{d}$ following the structure change in $T(V, E)$. Should we recompute every entry of A and $\mathbf{d}$ after every structure change? Let us define A' and $\mathbf{d}'$ as the matrix and vector corresponding to $T'(V, E')$.

Lemma 2. *Suppose we have the structure change $SC(T, e_{ij}, ., .)$ resulting in tree $T'(V, E')$. All the entries of A and A' are the same except the rows and columns corresponding to e_{ij}. So are all the entries in $\mathbf{d}$ and $\mathbf{d}'$ except the entry corresponding to e_{ij}. Thus, we only need to recompute the entry in $\mathbf{d}$, and the rows and columns in A corresponding to e_{ij}, to obtain A' and $\mathbf{d}'$.*

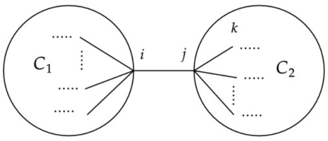
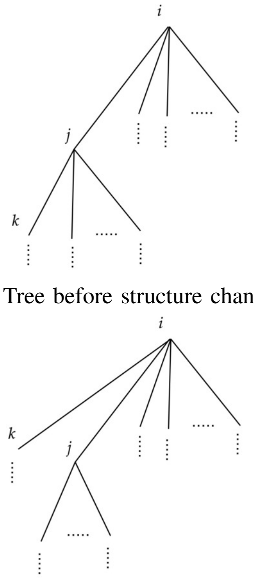


Fig. 2: Tree $T(V, E)$ before the structure change with picked edge e_{ij} connecting components C_1 and C_2, and randomly picked vertex $v_k \in C$



(a) Tree before structure change

(b) Tree after structure change

Fig. 3: Demonstration of the structure change $SC(T, e_{ij}, e_{ik}, e_{jk})$. Only v_j has a different number of descendants in T' than it has in T.

Proof. Consider the tree $T(V, E)$ in Fig. 2 on which we want to make the structure change based on the picked edge e_{ij} and $v_k \in C$—C as defined above. α and β are as defined in equation (4) for $T(V, E)$, and the equivalents of them are α' and β' for $T'(V, E')$. If $v_k \in N(v_j) \setminus \{v_i\}$, the structure change is $SC\,(T, e_{ij}, e_{ik}, e_{jk})$.

Let us look at $T(V, E)$ as a directed tree with the root vertex v_i—Fig. 3a. This tree before and after the structure change is illustrated in Fig. 3. Consider the subgraph $S = G(V, E'')$ in $T'(V, E')$—Fig. 3b—where $E'' = E' \setminus \{e_{ij}, e_{ik}\}$. It can be seen that every vertex but v_j in this subgraph has the exact same descendants in T' as they have in T. Thus, since $E'' \subset E'$ and $E'' \subset E$ and based on the calculation of α and β in Setion II, we can say that the number of paths that pass through any edge or any two edges in E'' is the same in T and T'. Similarly, regarding $e_{ik} \in E'$ and $e_{jk} \in E$, $\alpha'_{ik} = \alpha_{jk}$ and $\beta'_{ikrs} = \beta_{jkrs}$ for all $e_{rs} \in E''$. Hence, we see that e_{ij} is the only edge for which $\alpha'_{ij} \neq \alpha_{ij}$ and $\beta'_{ijrs} \neq \beta_{ijrs}$ where $e_{rs} \in E \cap E' \setminus e_{ij}$.

□

B. Simulated Annealing (SA)

As mentioned above, let us say the structure of the tree at time t is $T(V,E)$—$T_t \leftarrow T(V,E)$. Let us also denote $RSS(T', K_n)$ and $RSS(T, K_n)$ by RSS' and RSS respectively. Starting from a random initial tree structure T_0, we make the transition from $T_t \leftarrow T(V,E)$ to $T_{t+1} \leftarrow T'(V,E')$ in either of the following two cases:

1) $RSS' < RSS$
2) $P\left(\frac{RSS'-RSS}{RSS}, t\right) < \text{random}(0,1) \quad \text{if } RSS' > RSS$.

Otherwise, $T_{t+1} \leftarrow T$. In the above, random$(0,1)$ denotes a number picked uniformly at random in the interval $(0,1)$. The second case accepts the new tree structure with a worse RSS value with a certain probability. $P(RSS', RSS, t) = a_1 e^{-a_2 (\ln t)^{a_3} \frac{RSS'-RSS}{RSS'}}$, and it can be seen that the probability of accepting $RSS' > RSS$ decreases with time t. The parameters a_1, a_2 and a_3 are tuned according to how often we are willing to accept a transition with a larger RSS' than RSS, and such that accepted RSS' values roughly converge for a large t.

C. Iterated Local Search (ILS)

In ILS, we make the transition from $T_t \leftarrow T$ to $T_{t+1} \leftarrow T'$ only if $RSS' < RSS$—so far, it is a descent-only algorithm. However, in contrast to a descent-only algorithm, when we get stuck in a local minimum, we restart the algorithm—by modification of the current local minimum—to a new tree structure. Basically, ILS consists of the following two steps:

1) Modification of the current local minimum by kicking it far enough from its current basin
2) Descent to get to a new local minimum.

We want to try every possible structure change to make sure the function RSS is stuck at a local minimum. To this end, for any picked edge e_{ij}, the number of structure changes that we can make depends on $|C_i| = |N(v_i) \setminus \{v_j\}|$ and $|C_j| = |N(v_j) \setminus \{v_i\}|$. If we remove the edge e_{ij} from T, the resulting graph $G(V, E \setminus \{e_{ij}\})$ consists of two trees T_i and T_j where $v_i \in T_i$ and $v_j \in T_j$. We assume the average degree of a tree to be two; thus, we assume the degree of both v_i and v_j to be 2. With this assumption, the number of possible structure changes based on the picked edge e_{ij} is four, so for the whole tree, we estimate the number of possible structure changes at $4n$. If we try structure changes on a tree uniformly at random, the average number of times that we need to try all possible structure changes is $4nH_{4n}$—based on the well-known Coupon collector's problem—where H_{4n} is the $4n$-th harmonic number defined as $H_k = \sum_{i=1}^{k} \frac{1}{i}$. That is why we set $4nH_{4n}$ as the threshold to determine the algorithm is stuck at a local minimum.

IV. Results

We applied SA and ILS as described in Section III to evaluate the performance of these two metaheuristics in different scenarios. We evaluated whether bias towards smaller edges—picking an edge e_{ij} with a smaller weight for the tree structure change $SC(T, e_{ij}, ., .)$ with higher probability—has any advantage in SA over no bias—picking e_{ij} uniformly at random—in SA. After extensive experiments, we found that biased SA in general has a slight advantage over unbiased SA, so in the following, SA refers to biased SA.

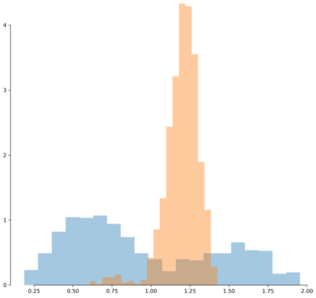


Fig. 4: Dispersion of sample of size 50 in Tables I and II

We compared the performance of SA and ILS based on running each of them ten times over the complete graph—where the distances in K_n are derived from stock-correlation data. See Tables I and II for a performance comparison of SA and ILS. In these tables, in each of the 10 runs, we ran each algorithm—SA and ILS—on trees with sizes of 20, 30, and 50 respectively for 10 minutes, two hours, and 18 hours. The values in the tables are for the minimum RSS value found in its corresponding run—according to which we evaluate the performance of the algorithm. In Table I, it can be seen that the performance of ILS is much better than that of SA. However, in Table II, we can see that there is no apparent difference between SA and ILS performance.

The reason for performance inconsistency of SA in Tables I and II seems to be the dispersion in distances of the complete weighted graph used in each of them. For example, for tree of size 50 in each table, dispersion of distances in the complete weighted graph is illustrated in Fig. 4 with a histogram. It can be seen that for distances with high dispersion, SA and ILS have a similar performance while for distances with low dispersion, ILS maintains a solid performance, but SA performance sharply decreases. We got the same result by running SA and ILS on the trees of many other complete weighted graphs of distances. It is noteworthy that for distance values with low dispersion, both the biased and unbiased SA, where the biased SA picks lightweight edges with a higher probability, have a poor performance. The reason possibly being, when distance values are close to each other, smaller distance values are not considerably smaller than the large distance values—giving no edge to biased over unbiased.

V. Conclusion

We have presented a scheme to optimize the edges weights and structure of a tree to approximate a complete weighted graph using a measure involving the path distances in the tree.

TABLE I: SA vs ILS on a complete weighted graph with low dispersion of distances. For each tree, the metaheuristic with a better performance has been highlighted.

	Tree size					
	20		30		50	
Run	SA	ILS	SA	ILS	SA	ILS
1	5.291951597	**4.84414153**	9.501749529	**8.4082242**	24.8579367	**16.8823573**
2	8.21566953	**4.84414153**	11.66835384	**8.4082242**	26.7081655	**16.8823573**
3	7.797470793	**4.84414153**	11.16650355	**7.97443788**	19.6536142	**16.8823573**
4	6.875995126	**4.84414153**	11.66835384	**7.97443788**	25.5941918	**16.8823573**
5	6.875995126	**4.84414153**	8.408224203	**7.97443788**	26.284257	**16.8823573**
6	6.019906558	**4.84414153**	12.98953771	**7.97443788**	30.2829294	**16.8823573**
7	7.016393465	**4.84414153**	10.3933439	**7.97443788**	34.8261135	**16.8823573**
8	7.016393465	**4.84414153**	11.80185307	**7.97443788**	31.0322914	**16.8823573**
9	**4.844141529**	5.2919516	13.02265027	**7.97443788**	31.1982311	**16.8823573**
10	7.016393465	**4.84414153**	10.58531443	**7.97443788**	26.265805	**16.8823573**
Average	6.697031065	**4.88892254**	11.12058844	**8.06119515**	27.6703536	**16.8823573**

TABLE II: SA vs ILS on a complete weighted graph with high dispersion of distances. For each tree, the metaheuristic with a better performance has been highlighted.

	Tree size					
	20		30		50	
Run	SA	ILS	SA	ILS	SA	ILS
1	5.75665216	5.75665216	**12.7055242**	12.8162316	32.18863674	**31.8890846**
2	5.75665216	5.75665216	**12.7055242**	12.7495148	**31.78033885**	31.8140415
3	5.75665216	5.75665216	**12.7464911**	12.8550396	**31.68615883**	31.8986455
4	5.75665216	5.75665216	**12.7055242**	12.8345686	32.0511834	**31.9636283**
5	5.75665216	5.75665216	**12.7055242**	13.0379292	**31.65783212**	31.6641679
6	5.75665216	5.75665216	**12.7055242**	12.832506	32.07947769	**31.7900896**
7	5.75665216	5.75665216	**12.7615612**	12.8750193	**31.91069847**	32.0325964
8	5.75665216	5.75665216	**12.7055242**	12.8785592	**32.00314739**	32.2380063
9	5.75665216	5.75665216	**12.7055242**	12.7207733	**31.96688445**	31.9796543
10	5.93707406	**5.75665216**	**12.7207733**	12.8472376	31.93720134	**31.792651**
Average	**5.77469435**	5.75665216	**12.7167495**	12.8447379	31.92615593	**31.9062565**

We have proposed a very efficient way of computing modifications to the tree that assist with local search metaheuristics, and evaluate the performance of two of these: SA and ILS.

References

[1] J. Felsenstein and J. Felenstein, *Inferring phylogenies*. Sinauer associates Sunderland, MA, 2004, vol. 2.

[2] G. Narasimhan and M. Smid, *Geometric spanner networks*. Cambridge University Press, 2007.

[3] R. N. Mantegna, "Hierarchical structure in financial markets," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 11, no. 1, pp. 193–197, 1999. [Online]. Available: https://doi.org/10.1007/s100510050929

[4] M. Tumminello, T. Aste, T. Di Matteo, and R. N. Mantegna, "A tool for filtering information in complex systems," *Proceedings of the National Academy of Sciences*, vol. 102, no. 30, pp. 10 421–10 426, 2005. [Online]. Available: https://doi.org/10.1073/pnas.0500298102

[5] V. Boginski, S. Butenko, and P. M. Pardalos, "Statistical analysis of financial networks," *Computational statistics & data analysis*, vol. 48, no. 2, pp. 431–443, 2005. [Online]. Available: https://doi.org/10.1016/j.csda.2004.02.004

[6] M. Tumminello, C. Coronnello, F. Lillo, S. Micciche, and R. N. Mantegna, "Spanning trees and bootstrap reliability estimation in correlation-based networks," *International Journal of Bifurcation and Chaos*, vol. 17, no. 07, pp. 2319–2329, 2007. [Online]. Available: https://doi.org/10.1142/S0218127407018415

[7] A. Kocheturov, M. Batsyn, and P. M. Pardalos, "Dynamics of cluster structures in a financial market network," *Physica A: Statistical Mechanics and its Applications*, vol. 413, pp. 523–533, 2014. [Online]. Available: https://doi.org/10.1016/j.physa.2014.06.077

[8] J.-P. Onnela, A. Chakraborti, K. Kaski, J. Kertesz, and A. Kanto, "Asset trees and asset graphs in financial markets," *Physica Scripta*, vol. 2003, no. T106, p. 48, 2003.

[9] J. Birch, A. A. Pantelous, and K. Soramäki, "Analysis of correlation based networks representing dax 30 stock price returns," *Computational Economics*, vol. 47, no. 4, pp. 501–525, 2016. [Online]. Available: https://doi.org/10.1007/s10614-015-9481-z

[10] D. Han *et al.*, "Network analysis of the chinese stock market during the turbulence of 2015–2016 using log-returns, volumes and mutual information," *Physica A: Statistical Mechanics and its Applications*, vol. 523, pp. 1091–1109, 2019. [Online]. Available: https://doi.org/10.1016/j.physa.2019.04.128

[11] R. Desper and O. Gascuel, "Theoretical foundation of the balanced minimum evolution method of phylogenetic inference and its relationship to weighted least-squares tree fitting," *Molecular Biology and Evolution*, vol. 21, no. 3, pp. 587–598, 2004. [Online]. Available: https://doi.org/10.1093/molbev/msh049

DOI: 10.15439/2020F100
ISSN 2300-5963 ACSIS, Vol. 21

Reinforcement Learning Algorithms for Online Single-Machine Scheduling

Yuanyuan Li*, Edoardo Fadda†, Daniele Manerba‡,
Roberto Tadei† and Olivier Terzo*

*LINKS Foundation - Advanced Computing and Applications, 10138 Torino, Italy
Email: {yuanyuan.li, olivier.terzo}@linksfoundation.com
†Department of Control and Computer Engineering, Politecnico di Torino, 10129 Torino, Italy
Email: {edoardo.fadda, roberto.tadei}@polito.it
‡Department of Information Engineering, University of Brescia, 25123 Brescia, Italy
Email: daniele.manerba@unibs.it

***Abstract*—Online scheduling has been an attractive field of research for over three decades. Some recent developments suggest that Reinforcement Learning (RL) techniques have the potential to deal with online scheduling issues effectively. Driven by an industrial application, in this paper we apply four of the most important RL techniques, namely *Q-learning*, *Sarsa*, *Watkins's Q(λ)*, and *Sarsa(λ)*, to the online single-machine scheduling problem. Our main goal is to provide insights on how such techniques perform. The numerical results show that Watkins's Q(λ) performs best in minimizing the total tardiness of the scheduling process.**

I. Introduction

PRODUCTION scheduling is one of the most important aspects to address in many manufacturing companies (see [1]). The optimization problems arising within production scheduling can be of *static* or *dynamic* type (see [2]). In contrast with the static case, in which specifications and requirements are fully and deterministically known in advance, in the dynamic one, additional information (e.g., new orders, changes of available resources) may arrive during the production process itself. In this paper, we will consider the latter case, commonly called *online scheduling*, mainly fostered by our experience on an industrial project (Plastic and Rubber 4.0[1]) in which frequent occurrences of unexpected events call for more dynamic and flexible scheduling (see [3]).

In particular, we will focus on online single-machine scheduling problems with release dates and preemption allowed, in which the objective is to minimize the total tardiness. Let us consider a set $\mathcal{J}$ of jobs that are released over time. As soon as a job arrives, it is added to the end of a waiting queue. For each job $j \in \mathcal{J}$, let dt_j be its due time and ct_j its completion time. The goal of the problem is to arrange the jobs of the queue, so to minimize the total tardiness calculated as $\Gamma = \sum_{j\in\mathcal{J}} ta_j$, where $ta_j := \max\{0, ct_j - dt_j\}$. The motivation of studying a single-machine problem relies on the fact that, in the plastic and rubber manufacturing, the process of transforming raw material into a final product just goes through one or two machines. On the other hand, even for those manufacturing requiring multiple-machine scheduling problems, each machine represents a basic block of a chain. Thus improper usage of a machine can slow down the whole production process.

The easiest way to deal with scheduling in a dynamic context is the use of the so-called *dispatching rules*. These rules first prioritize jobs waiting for being processed and then select the job with a greedy evaluation whenever a machine gets free (see Section II for more details). While most dispatching rules simply schedule on a local view basis, other smarter approaches can be used to provide better results in the long run. For instance, Reinforcement Learning (RL) is a continuing and goal-directed learning paradigm, and it represents a promising approach to deal with online scheduling. The potential of RL on online scheduling has been revealed in several works (see, e.g., [4], [5], [6]). However, while most works compare a single RL algorithm with commonly-used dispatching rules, they lack in comparing different RL algorithms. A research question naturally arises: how do different RL algorithms perform on online scheduling?

Motivated by investigating the applicability of RL algorithms on online single-machine scheduling in detail, in this work, we will compare the following approaches' performance:

- a random assignment (*Random*) which simply selects a job randomly;
- one of the most popular dispatching rules, namely the *earliest due date* (*EDD*) rule;
- four RL approaches, namely *Q-learning*, *Sarsa*, *Watkins's Q(λ)*, and *Sarsa(λ)*.

Furthermore, we will test the algorithms under different operating conditions (e.g., the frequency of job arrivals). *Watkins's Q(λ)* seems the most promising method in most of the cases. Therefore, we contribute the literature on two different aspects: getting insights on the compared methods, and giving

[1] Plastic&Rubber 4.0. Piattaforma Tecnologica per la Fabbrica Intelligente (Technological Platform for Smart Factory), URL: https://www.regione.piemonte.it/web/temi/fondi-progetti-europei/fondo-euro\ \peo-sviluppo-regionale-fesr/ricerca-sviluppo-tecnologico-innovazione/piatta\ \forma-tecnologica-fabbrica-intelligente

practitioners suggestions on selecting the best method against the specific situation. Notice that comparing and evaluating different algorithms against various aspects and performance indicators is a commonly adopted research methodology (see, e.g., [7], [8], [9], [10], [11], and [12]). The specific comparison of RL algorithms can be found, for instance, in the game field. In [13], the authors compared two RL algorithms (*Q-learning* and *Sarsa*) through the simulation of bargaining games. Even though the two algorithms present slight differences, they might have essentially different simulation results, as reflected in our experiment (see Section IV).

Finally, we also propose some preliminary results obtained by the use of *Deep Q Network* (*DQN*), which utilizes the power of neural networks to approximate the value function (see [14] for a review about DQN). However, our experiments will show that *DQN* is better suited for high-dimensional inputs. In contrast, with smaller input settings, *DQN* has a longer training time and obtains results that are far from the performance of *Watkins's Q(λ)*.

The rest of the paper is organized as follows. Section II is dedicated to a general overview of RL techniques, while Section III introduces and reviews some previous works using RL approaches on scheduling problems. Section IV describes the algorithmic framework for the online single-machine problem. Section V defines the simulation procedure, and the simulation results from three different types of experiments (Section VI). Finally, in Section VII, the paper concludes with a summary of the findings and some future lines.

II. Reinforcement Learning

RL is a branch of Machine Learning that improves automatically through experience. It comes from three main research branches: the first relates to learning by trial-and-error, the second relates to optimal control problems, and the last relates to temporal-difference methods (see [15]). The three approaches converged together in the late eighties to produce the modern RL.

RL approaches can be applied to scenarios in which a decision-maker called *agent* interacts with a set of *states* called *environment* by means of a set of possible *actions*. A *reward* is given to the agent in each specific state. In this paper, we consider a discrete time system, i.e. defined over a finite set $\mathcal{T}$ of time steps with its cardinality being called *time horizon*. As shown in Figure 1, at each time step $t \in \mathcal{T}$, an agent in state S_t takes action A_t, then, the environment reacts by changing into state S_{t+1} and by rewarding the agent of R_{t+1}. The interaction starts from an initial state, and it continues until the end of the time horizon. Such a sequence of actions is named an *episode*. In the following, $\mathcal{E}$ will represent the set of episodes.

Each *state* of the system is associated with a *value function* that estimates the expected future reward achievable from that state. Each state-action pair (S_t, A_t) is associated with a so-called *Q-function* $Q(S_t, A_t)$ that measures the future reward achievable by implementing action A_t in state S_t. The agent's goal is to find the best *policy*, which is a function mapping the set of states to the set of actions, maximizing the cumulative *reward*. If exact knowledge of the Q-function is available, the best policy for each state is defined by $\max_a Q(S_t, a)$.

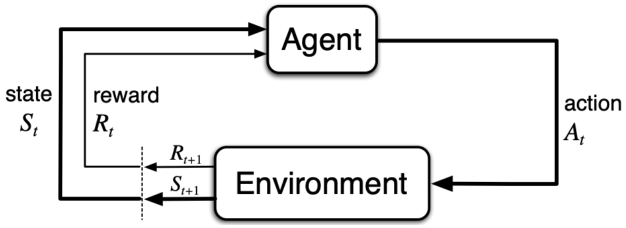


Fig. 1. The agent-environment interaction in RL [15].

To estimate the value functions $Q(s, a)$ and discover the optimal policies, three main classes of RL techniques exist Monte Carlo (MC)-based, Dynamic Programming (DP)-based methods, and temporal-difference (TD)-based methods. Unlike DP-based methods, which require complete knowledge of all the possible transitions, MC-based methods only require some experience and the possibility to sample from the environment randomly. TD-based methods are a sort of combination of MC-based and DP-based ones: they sample from the environment like in MC-based methods and perform updates based on current estimates like DP-based ones. Moreover, TD-based methods are also appreciated for being flexible, easy to implement, and computationally fast. For these reasons, in this paper, we will consider only RL algorithms belonging to the TD-based methods. Even if several TD-based RL algorithms have been introduced in the literature, the most used are *Sarsa* (an acronym for State-Action-Reward-State-Action), *Q-learning* and their variations, e.g. the *Watkins's Q(λ)* method and the *Sarsa(λ)* (see [16]).

III. Literature Review

Since online scheduling has been an active field for several decades, an in-depth analysis of the literature review is out of scope for the present paper. Thus, in this section, we recall some of the most traditional approaches to online scheduling, and we review the main applications of RL to this problem.

Differently from tailored algorithms (heuristic and exact methods), which might require effort in implementation and calibration over a broad set of parameters, dispatching rules are widely adopted for online scheduling for their simplicity (see, e.g., [17]). For instance, the *earliest due date* (*EDD*) dispatching rule is one of the most commonly used ones in practical applications [18]. *EDD* simply schedules first the job with the earliest due date. Again, in [19], the authors propose a deterministic greedy algorithm known as *list scheduling* (*LS*), which simply assigns each job to the machine with the smallest load. For more details, we refer the reader to the work [20] that classified over one hundred dispatching rules. In [21], the authors designed a deterministic algorithm and a randomized one for online machine sequencing problems using Linear Programming techniques. At the same time, in [22], the authors proposed an algorithm to make jobs artificially available to the online scheduler by delaying the release time of jobs.

In online scheduling, a decision-maker is regularly scheduling jobs over time, attempting to reach the overall best performance. Therefore, it is reasonable that RL represents one of the possible techniques able to exploit such a setting.

In [4], the authors interpreted job-shop scheduling problems as sequential decision processes. They try to improve the job dispatching decisions of the agent by employing an RL algorithm. Experimental results on numerous benchmark instances showed the competitiveness of the RL algorithm. More recently, in [6], the authors modeled the scheduling problem as a Markov Decision Process and solved it through a simulation-based value iteration and a simulation-based *Q-learning*. Their results clearly showed that such RL algorithms could achieve better performance concerning several dispatching heuristics, disclosing the potential of RL application in the field. In the context of an online single-machine environment, in [23], the authors compared the performance of *neural fitted Q-learning* techniques using combinations of different states, actions, and rewards. They proved that taking only the necessary inputs of states and actions is more efficient.

While all the discussed works revealed the competitiveness of RL on scheduling problems, a further comparison of the performance among various RL algorithms is still missing in the scheduling literature. With the knowledge of the available studies showing the potential of RL and the demand from the industrial application, we are motivated to compare different RL approaches' performance on online scheduling for getting more insights. In particular, we carry out experimental studies on four of the most commonly used model-free RL algorithms, namely *Q-learning*, *Sarsa*, *Watkins's Q(*λ*)*, and *Sarsa(*λ*)*. Our comparison methodology is inspired by [23], in which the best configuration for minimizing maximal lateness is pursued. In our work, instead, we aim at minimizing the total tardiness of the scheduling process. Moreover, another major difference with their work lies in the way we evaluate the results. While they used the result from one run, our results come from 50 runs with different random seeds, and two different time step sizes are tested (the interaction between agent and environment is checked in each step). Also, we further test a neural network-based RL technique showing that it is not necessary to use such a combination when the state space is limited.

IV. Reinforcement Learning Algorithms for Online Scheduling

In this section, we describe the algorithmic framework used to deal with our online single-machine scheduling problem. In particular, we provide several variants based on different RL techniques.

A. States, actions, and rewards

To be approached by RL techniques, we define our problem setting along the lines used in [23]. In particular:

- *state*: a state is associated with each possible length of the jobs in the waiting queue;
- *action*: if not all the jobs are finished, the action is either to select one new job from a specific position of the waiting queue and start processing it (we recall that preemption is allowed), or to continue processing the job which has been already assigned to the machine in the previous step;
- *reward*: since RL techniques aim at maximizing rewards while our problem aims at minimizing the total tardiness, we set the reward of a state as the opposite value of its total tardiness.

When the action implies the selection of a job from a certain position in the waiting queue, it is important to decide the order in which jobs are stored inside the queue. Therefore, we implemented three possible ordering of jobs which provide very different scheduling effects:

- jobs are unsorted (*UNSORT*), i.e., they have the same order as the arrivals;
- jobs are sorted by increasing value of due time (*DT*);
- all unfinished jobs are sorted by increasing the value of the sum of due time and processing time (*DT+PT*).

For instance, by using *DT*, if the action is to select a job in the second position of the queue, the job with the second earliest due time will be processed.

B. RL algorithms adopted

We have decided to implement four different RL algorithms, namely *Q-learning*, *Sarsa*, *Watkins's Q(*λ*)*, and *Sarsa(*λ*)*. They are described in the following. Here are some notations used:

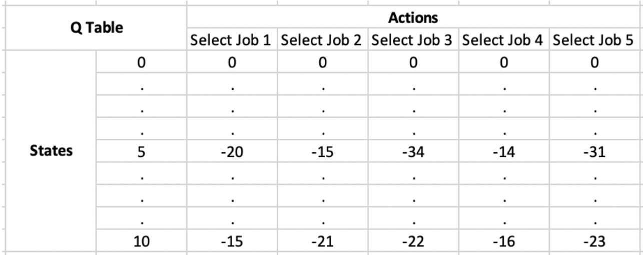
- s state;
- a action;
- $\mathcal{S}$ set of nonterminal states;
- $\mathcal{A}(s)$ set of actions possible in state s;
- S_t state at t;
- A_t action at t;
- R_t reward at t.

1) Q-learning: Q-learning is a technique that learns the value of an optimal policy independently of the agent's action. It is largely adopted for its simplicity in the analysis of the algorithm and for the possibility of early convergence proofs by directly approximating the optimal action-value function (see [16] and [15]). The updating rule for the estimation of the Q-function is:

$$Q(S_t,A_t) \leftarrow Q(S_t,A_t) + \alpha[R_{t+1} + \gamma \max_a Q(S_{t+1},a) - Q(S_t,A_t)]. \quad (1)$$

The $Q(S_t, A_t)$ function estimates the quality of state-action pair. At each time step t, the reward R_{t+1} from state S_t to S_{t+1} is calculated and $Q(S_t, A_t)$ is updated accordingly. The coefficient α is the learning rate ($0 \leq \alpha \leq 1$); it determines the extent that new information overrides the old information. Furthermore, γ is the discount factor determining the importance of future reward and finally, $\max_a Q(S_{t+1}, a)$ is the estimation of best future value.

The values of the Q-function are stored in a look-up table called *Q table*. Figure 2 displays an example of Q table storing Q-function values for states from 0 to 10 (in row) and actions from selecting *Job 1* to *Job 5* (in column). By overlooking the actual policy being followed in deciding the next action, *Q-learning* simplifies the analysis of the algorithm and enabled early convergence proofs.



Q Table		Actions				
		Select Job 1	Select Job 2	Select Job 3	Select Job 4	Select Job 5
States	0	0	0	0	0	0
	.	.	.	.	.	.
	.	.	.	.	.	.
	.	.	.	.	.	.
	5	-20	-15	-34	-14	-31
	.	.	.	.	.	.
	.	.	.	.	.	.
	.	.	.	.	.	.
	10	-15	-21	-22	-16	-23

Fig. 2. An example of Q table.

2) Sarsa: Sarsa is a technique that updates the estimated Q-function by following the experience gained from executing some policies (see [24] and [15]). The updating rule for the estimation of the Q-function is:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)]. \quad (2)$$

The structure of formula (2) is similar to (1). The only difference is that (2) considers the actual action implemented in the next step A_{t+1}, instead of the generic best action $\max_a Q(S_{t+1}, a)$.

As for *Q-learning*, also in *Sarsa* the values of the Q-function are stored in a *Q table*. Despite the more expensive behaviour with respect to *Q-learning*, *Sarsa* may provide better online performances in some scenarios (as shown by the *Cliff Walking* example in [15]).

3) Watkins's Q(λ): Watkins's Q(λ) is a well-known variant of *Q-learning*. The main difference with respect to classical *Q-learning* is the presence of a so-called *eligibility trace*, i.e. a temporary record of the occurrence of an event, such as the visiting of a state or the taking of an action. The trace marks the memory parameters associated with the event as eligible for undergoing learning changes. A trace is initialized when a state is visited or an action is taken, and then the trace gets decayed over time according to a decaying parameter λ (with $0 \leq \lambda \leq 1$). Let us call $e_t(s, a)$ the trace for a state-action pair (s, a). Let us also define an indicator parameter $\mathbb{1}_{xy}$ that takes value 1 if and only if x and y are the same, and 0 otherwise. Then, for any (s, a) pair (for all $s \in \mathcal{S}$, $a \in \mathcal{A}$), the updating rule for the estimation of the Q-function is:

$$Q_{t+1}(s, a) = Q_t(s, a) + \alpha \delta_t e_t(s, a) \quad (3)$$

where

$$\delta_t = R_{t+1} + \gamma \max_{a'} Q_t(S_{t+1}, a') - Q_t(S_t, A_t) \quad (4)$$

and

$$e_t(s, a) = \gamma \lambda e_{t-1}(s, a) + \mathbb{1}_{sS_t} \mathbb{1}_{aA_t} \quad (5)$$

if $Q_{t-1}(S_t, A_t) = \max_a Q_{t-1}(S_t, a)$, and $\mathbb{1}_{sS_t} \mathbb{1}_{aA_t}$ otherwise.

As the reader can notice, by plugging Eq. (4) into Eq. (3), we obtain an equation similar to (1) but with the additional eligibility term that increments the value of δ_t if the state and action selected by the algorithm are one of the eligibility states. In the rest of the paper we use *Q(λ)* referring to *Watkins's Q(λ)*.

4) Sarsa(λ): Similarly to *Q(λ)*, the *Sarsa(λ)* algorithm represents a combination between *Sarsa* and eligibility traces to obtain a more general method that may learn more efficiently. Here, for any (s, a) pair (for all $s \in \mathcal{S}$, $a \in \mathcal{A}$), the updating rule for the estimation of the Q-function is:

$$Q_{t+1}(s, a) = Q_t(s, a) + \alpha \delta_t e_t(s, a) \quad (6)$$

where

$$\delta_t = R_{t+1} + \gamma Q_t(S_{t+1}, A_{t+1}) - Q_t(S_t, A_t) \quad (7)$$

and

$$e_t(s, a) = \gamma \lambda e_{t-1}(s, a) + \mathbb{1}_{sS_t} \mathbb{1}_{aA_t} \quad (8)$$

Unlike Eq. (5), there is no other condition (set the eligibility traces to 0 whenever a non-greedy action is taken) added. A deeper discussion about the interpretation of the formulas is given in [15].

V. Simulation Procedure

In order to perform the comparison under interest, we create an online scheduling simulation procedure as described in Algorithm 1.

Algorithm 1 Online scheduling simulation through RL algorithms

```
Require: $|\mathcal{E}|$ number of episodes; $|\mathcal{T}|$ number of time-steps;
Initialize $Q(s,a) = 0, \forall\ s \in \mathcal{S}, a \in \mathcal{A}$;
for $\eta \leftarrow 1$ to $|\mathcal{E}|$ do
    Initialize $S$
    for $t \leftarrow 1$ to $|\mathcal{T}|$ do
        if new jobs arrive then
            Update waiting list $L$
        end if
        if $L$ is not empty then
            Take $A_t$ in $S_t$, observe $R_t$, $S_{t+1}$
            Calculate $A_{t+1}$ and update $Q_t$
            $S_t \leftarrow S_{t+1}$, $A_t \leftarrow A_{t+1}$
        end if
    end for
end for
```

We first update Q tables through a training phase then use the Q tables to select actions in the test phase.

The arrival time of job j are distributed according to an exponential distribution, i.e., $X_j \sim exp(r)$ with the rate parameter valued $r = 0.1$. It is simulated in this way: at the first time step, a random number of jobs (from 1 to 6 jobs) and an interval time (following the exponential distribution) are generated. Once a job is generated (simulating the arrival of the job), it will be put into the waiting queue immediately. Then at the next time step, if the interval time is passed, new jobs will be generated and put into the waiting queue; meanwhile, a new interval time will be created. Otherwise, nothing is created. Then the same procedure repeats till reaching a final state.

For the settings regarding RL algorithms:

- In the policy, $\epsilon = 0.1$ enabling highly greedy actions while keeping some randomness in job selections;
- In the value function, $\alpha = 0.6$, i.e., there is a bit higher tendency to explore more possibilities while a bit lower in keeping exploiting old information, whereas $\gamma = 1.0$, which means it strives for a long-term high reward;
- In the eligibility traces, λ is 0.95, a high decaying value is leading to a longer-lasting trace.

It is worth noting that all the algorithms considered are heuristics. Thus they focus on finding a good solution in a short amount of time by finding a balance between intensified and diversified explorations of the solution space. Nevertheless, the plain implantation of the algorithms above does not ensure enough diversification. For this reason, it is common to use a ϵ-greedy method. Thus, with probability ϵ, exploration is chosen, which means the action is chosen uniformly at random between the available ones. Instead, with probability $1 - \epsilon$, exploitation is chosen by taking the actions with the highest values greedily. After knowing the way to balance exploration and exploitation, we need to define a learning method for finding out policies leading to higher cumulative rewards.

In an episode, we start a new schedule by initializing state S and terminates when either reaching the maximum steps or no jobs to process. To simulate real-time scheduling, for each episode, we check the arrivals of new jobs and update the waiting queue if there are, then we choose the action A, and calculate the reward R and the next state S' accordingly. The Q functions are updated according to the exact RL algorithms used. The same procedure is carried out in both training and test phases except that in the test, the Q table is not initialized with empty values but obtained from the training phase.

Let us see a training example with *Q-learning* to see for the same schedule how the reward is accumulated, and objective value evolves with more episodes passing by. In Fig. 3, the graph on the bottom shows after around 80 episodes, the reward reaches the maximum and holds steady. Accordingly, the objective value - total tardiness drops more slowly after around 80 episodes. While the reward keeps stable, total tardiness continues dropping to around 4,0000. To summarize, using total tardiness as a reward is useful, but it is still challenging to represent the trend of the objective value adequately.

VI. Numerical Experiments

In this section, we propose three different experimental results. Section VI-A compares the performance among random assignment (*Random*), *EDD*, and the four RL approaches implemented. Section VI-B investigates the possible impact of different operating conditions (i.e., frequency of jobs arrivals) on the RL approaches. Finally, Section VI-C compares *Q(λ)* and *DQN*.

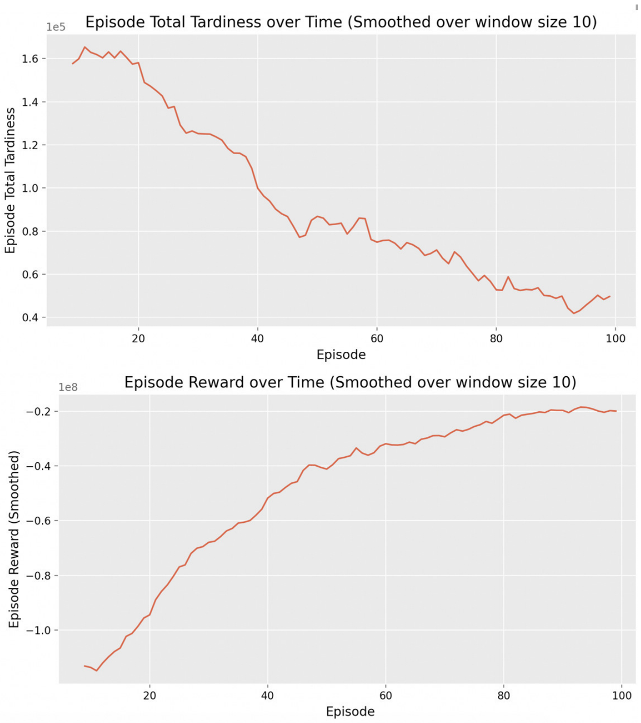


Fig. 3. The changes to reward and the objective value (total tardiness) of 100 episodes.

The algorithms have been implemented in Python 3.6. To avoid possible ambiguities, we locate the related code in a public repository[2]. All the experiments are carried out on an *Intel Core i5* CPU@2.3GHz machine equipped with 8GB RAM and running *MacOS* v10.15.4 operating system.

A. RL algorithms vs Random and EDD

To check if considering different time horizons leads to different results, we consider two experiments in which the time horizon $\mathcal{T}$ is set to 2500 and 5000, respectively. For each of the settings, we ran 50 tests with different random seeds. For each algorithm Θ, we call $\Gamma_{\zeta\Theta}$ the total tardiness achieved in simulation ζ . Furthermore, we define $\rho_{\zeta\Theta}$ to be the percentage gap between the total tardiness achieved by the best algorithm and by algorithm Θ during run ζ, i.e.,

$$\rho_{\zeta\Theta} = \frac{\Gamma_{\zeta\Theta}}{\min_{\zeta\Theta} \Gamma_{\zeta\Theta}}. \tag{9}$$

To compare the different algorithms, we consider the average value of $\rho_{\zeta\Theta}$ concerning all the runs.

The simulation results with the algorithms (under different job orders, time horizons) are displayed in Table I, where

[2] URL: https://github.com/Yuanyuan517/RL_OnlineScheduling.git

avg($\rho_{\zeta\Theta}$), std($\rho_{\zeta\Theta}$) are respectively the mean and standard deviations of $\rho_{\zeta\Theta}$. The best value among all the combinations of algorithms and jobs order policies for each time horizon is highlighted in bold font.

TABLE I: Experiment cases of the algorithms with different settings

		$\|\mathcal{T}\|$=2500		$\|\mathcal{T}\|$=5000	
Algorithm	**Jobs order**	avg($\rho_{\zeta\Theta}$)	std($\rho_{\zeta\Theta}$)	avg($\rho_{\zeta\Theta}$)	std($\rho_{\zeta\Theta}$)
Random	-	2.59	0.50	3.06	0.69
EDD	-	7.67	1.76	9.19	1.47
Q-learning	*UNSORT*	2.15	0.43	2.04	0.35
Q-learning	*DT*	1.45	0.28	1.29	0.20
Q-learning	*DT+PT*	1.44	0.30	1.25	0.18
Sarsa	*UNSORT*	2.55	0.53	2.47	0.39
Sarsa	*DT*	1.65	0.40	1.76	0.36
Sarsa	*DT+PT*	1.66	0.47	1.68	0.33
Sarsa(λ)	*UNSORT*	4.42	0.93	5.04	0.93
Sarsa(λ)	*DT*	7.04	1.35	7.73	1.34
Sarsa(λ)	*DT+PT*	3.08	1.03	7.70	1.33
Q(λ)	*UNSORT*	2.04	0.42	2.01	0.40
Q(λ)	*DT*	**1.11**	**0.18**	1.13	0.17
Q(λ)	*DT+PT*	1.19	0.26	**1.09**	**0.14**

While [23] shows *EDD* gets a better result than RL to minimize the maximum tardiness, with the new objective of minimizing total tardiness in our experiments, all RL algorithms get better results than *EDD*.

As shown in Table I, the size of running time steps influenced the result on job order but does not influence the algorithm. And for the case with 2500 steps, the configuration *Q(λ)* plus *DT* gets the best result, instead for 5000 steps, the configuration *Q(λ)* plus *DT+PT* gets the best result.

Besides, we find with the sorting choice *DT+PT* that all algorithms get smaller average values except for the configuration *Q(λ)* with 2500 steps. Comparatively, a randomly sorting job leads to a much worse result.

B. Q(λ) performance against different job arrival rates

Another test is on the operating condition - the frequency of job arrivals for the two best combinations *Q(λ)* plus *DT* and *Q(λ)* plus *DT+PT*, which is controlled by the rate parameter r. To understand whether the value of r affects performance, we experimented with 2 more values, i.e. $r = \{0.05, 0.2\}$ in addition to the previous one $r = 0.1$.

In Table II, the results are also normalized by following Eq. (9) with 50 tests and $|\mathcal{T}| = 2500$ for each test. As shown in the table, with small $r = 0.05$, $r = 0.1$ (indicating jobs arrive much less frequently than the last one), the version with jobs ordered by *DT* performs better. When jobs arrive much more frequently, the version sorted by *DT + PT* wins. Hence a careful selection of algorithms and settings according to the operating conditions matters.

TABLE II: Experiment cases of the rate parameter with best settings from *Q(λ)*.

Jobs order	r	avg($\rho_{\zeta\Theta}$)	std($\rho_{\zeta\Theta}$)
DT	0.05	**1.14**	**0.18**
DT+PT	0.05	1.17	0.55
DT	0.10	**1.10**	**0.17**
DT+PT	0.10	1.17	0.26
DT	0.20	1.17	0.28
DT+PT	0.20	**1.12**	**0.24**

C. Comparison between Q(λ) and DQN

In the third test we compare a four-layer DQN and *Q(λ)* plus *DT+PT*, i.e. the better performing RL algorithm according to Table I. Figure 4 shows such a comparison. The result is from running 50 tests and $|\mathcal{T}| = 5000$ in each test. The horizontal axis represents the total tardiness and the vertical axis shows the probability the objective value falls in. The dark yellow area indicates the overlapping between *Q(λ)* and DQN.

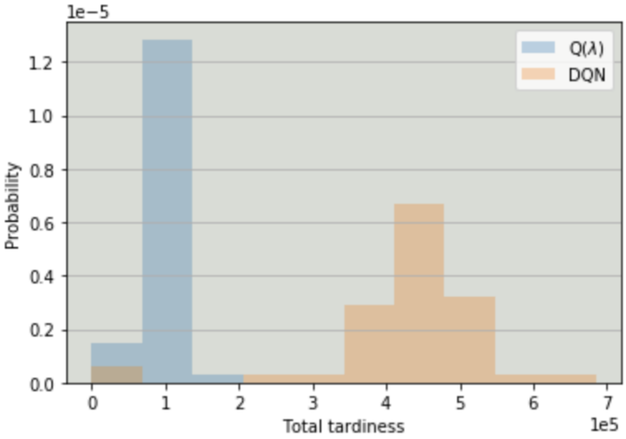


Fig. 4. The comparison between *Q(λ)* and DQN on the total tardiness of 50 runs with different seeds representing different schedules.

We can see *Q(λ)* has much higher probability with smaller objective value, which indicates *Q(λ)* outperforms DQN. Taking into account the time spent in training DQN is almost 10 times of *Q(λ)*, *Q(λ)* is a better option, especially for guaranteeing a flexible and adaptive scheduling in realtime.

VII. Conclusions and Future Work

In this paper, we compared four RL methods, namely *Q-learning*, *Sarsa*, *Watkins's Q(λ)*, and *Sarsa(λ)*, with *EDD* and random assignment on an online single-machine scheduling problem. The experiments show that:

- better scheduling performance is achieved by the RL method *Watkins's Q(λ)*, especially when the action concerns the selection of jobs sorted by due date for the smaller time horizon ($|\mathcal{T}| = 2500$) and the selection of jobs sorted by due date and processing time for bigger time horizon ($|\mathcal{T}| = 5000$).
- the tests on r disclose the combination of *Q(λ)* and job orders have different performances in various operating conditions.

- slight difference in algorithms can profoundly change the results.

Besides, with limited input, using DQN is too costly for extended running time and energy spent in adjusting parameters to guarantee a good result. The results above indicate careful analysis should be done from different angles (running time, operating conditions, average results from multiple experiments) for making a wiser selection of algorithms.

Furthermore, with multiple machines, more transitions must be considered, which need more representational state information. Thus it will be impossible to store values of all state-action pairs in a Q table. DQN may take a leading role then. As indicated by the work [25], unpredictable changes may happen at different places in the state-action space, and more care should be taken to avoid instabilities of DQN. One techniques that can acheive this goal is the usage of kernel function (see [26]), this builds a future research avenue.

Acknowledgement

This research was partially supported by the Plastic and Rubber 4.0 (P&R4.0) research project, POR FESR 2014-2020 - Action I.1b.2.2, funded by Piedmont Region (Italy), Contract No. 319-31. The authors acknowledge all the project partners for their contribution.

References

[1] P. Brucker, *Scheduling Algorithms*, 5th ed. Springer Publishing Company, Incorporated, 2010.

[2] S. C. Graves, "A review of production scheduling," *Operations Research*, vol. 29, no. 4, pp. 646–675, 1981.

[3] Y. Li, S. Carabelli, E. Fadda, D. Manerba, R. Tadei, and O. Terzo, "Machine learning and optimization for production rescheduling in industry 4.0," *The International Journal of Advanced Manufacturing Technology*, pp. 1–19, 2020.

[4] T. Gabel and M. Riedmiller, "Adaptive reactive job-shop scheduling with reinforcement learning agents," *International Journal of Information Technology and Intelligent Computing*, vol. 24, no. 4, pp. 14–18, 2008.

[5] H. Sharma and S. Jain, "Online learning algorithms for dynamic scheduling problems," in *2011 Second International Conference on Emerging Applications of Information Technology*, 2011, pp. 31–34.

[6] T. Zhang, S. Xie, and O. Rose, "Real-time job shop scheduling based on simulation and markov decision processes," in *2017 Winter Simulation Conference (WSC)*. IEEE, 2017, pp. 3899–3907.

[7] P. Castrogiovanni, E. Fadda, G. Perboli, and A. Rizzo, "Smartphone data classification technique for detecting the usage of public or private transportation modes," *IEEE Access*, vol. 8, pp. 58 377–58 391, 2020.

[8] E. Fadda, P. Plebani, and M. Vitali, "Optimizing monitorability of multi-cloud applications," 06 2016, pp. 411–426.

[9] E. Fadda, G. Perboli, and G. Squillero, "Adaptive batteries exploiting on-line steady-state evolution strategy," in *Applications of Evolutionary Computation*, G. Squillero and K. Sim, Eds. Cham: Springer International Publishing, 2017, pp. 329–341.

[10] E. Fadda, D. Manerba, R. Tadei, P. Camurati, and G. Cabodi, "KPIs for Optimal Location of charging stations for Electric Vehicles: the Biella case-study," in *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 18. IEEE, 2019, pp. 123–126. [Online]. Available: http://dx.doi.org/10.15439/2019F171

[11] E. Fadda, D. Manerba, G. Cabodi, P. Camurati, and R. Tadei, "Comparative analysis of models and performance indicators for optimal service facility location," *Transportation Research part E: Logistics and Transportation Reviews (submitted)*, 2020.

[12] R. Giusti, C. Iorfida, Y. Li, D. Manerba, S. Musso, G. Perboli, R. Tadei, and S. Yuan, "Sustainable and de-stressed international supply-chains through the synchro-net approach," *Sustainability*, vol. 11, p. 1083, 02 2019.

[13] K. Takadama and H. Fujita, "Toward guidelines for modeling learning agents in multiagent-based simulation: Implications from q-learning and sarsa agents," in *International Workshop on Multi-Agent Systems and Agent-Based Simulation*. Springer, 2004, pp. 159–172.

[14] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[15] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[16] C. J. C. H. Watkins, *Learning from delayed rewards*. Thesis Submitted for Ph.D., King's College, Cambridge, 1989.

[17] A. Kaban, Z. Othman, and D. Rohmah, "Comparison of dispatching rules in job-shop scheduling problem using simulation: a case study," *International Journal of Simulation Modelling*, vol. 11, no. 3, pp. 129–140, 2012.

[18] H. Suwa and H. Sandoh, *Online scheduling in manufacturing: A cumulative delay approach*. Springer Science & Business Media, 2012.

[19] R. L. Graham, "Bounds for certain multiprocessing anomalies," *Bell System Technical Journal*, vol. 45, no. 9, pp. 1563–1581, 1966.

[20] S. S. Panwalkar and W. Iskander, "A survey of scheduling rules," *Operations Research*, vol. 25, no. 1, pp. 45–61, 1977.

[21] J. R. Correa and M. R. Wagner, "Lp-based online scheduling: from single to parallel machines," *Mathematical Programming*, vol. 119, no. 1, pp. 109–136, 2009.

[22] X. Lu, R. Sitters, and L. Stougie, "A class of on-line scheduling algorithms to minimize total completion time," *Operations Research Letters*, vol. 31, no. 3, pp. 232–236, 2003.

[23] S. Xie, T. Zhang, and O. Rose, "Online single machine scheduling based on simulation and reinforcement learning," in *Simulation in Produktion und Logistik 2019*. Simulation in Produktion und Logistik 2019, 2019.

[24] S. Singh, T. Jaakkola, M. L. Littman, and C. Szepesvári, "Convergence results for single-step on-policy reinforcement-learning algorithms," *Machine learning*, vol. 38, no. 3, pp. 287–308, 2000.

[25] V. François-Lavet, R. Fonteneau, and D. Ernst, "How to discount deep reinforcement learning: Towards new dynamic strategies," *arXiv preprint arXiv:1512.02011*, 2015.

[26] V. Cerone, E. Fadda, and D. Regruto, "A robust optimization approach to kernel-based nonparametric error-in-variables identification in the presence of bounded noise," in *2017 American Control Conference (ACC)*. IEEE, may 2017. [Online]. Available: https://doi.org/10.23919

Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 285–288
DOI: 10.15439/2020F135
ISSN 2300-5963 ACSIS, Vol. 21

A Two-Stage Monte Carlo Approach for Optimization of Bimetallic Nanostructures

Rossen Mikhov[1], Vladimir Myasnichenko[2], Leoneed Kirilov[1],
Nickolay Sdobnyakov[2], Pavel Matrenin[3], Denis Sokolov[2], and Stefka Fidanova[1]

[1]Bulgarian Academy of Sciences
Acad. G. Bonchev Str., bl. 2
1113 Sofia, Bulgaria
Email: l_kirilov_8@abv.bg

[2]Tver State University
33, Zhelyabova Str.
170100 Tver, Russia
Email: virtson@gmail.com

[3]Novosibirsk State Technical University
20, Prospekt K. Marksa
630087 Novosibirsk, Russia

□

***Abstract*—In this paper we propose a two-stage lattice Monte Carlo approach for optimization of bimetallic nanoalloys: simulated annealing on a larger lattice, followed by simulated diffusion. Both algorithms are fairly similar in structure, but their combination was found to give significantly better solutions than simulated annealing alone. We also discuss how to tune the parameters of the algorithms so that they work together optimally.**

I. Introduction

THE fundamental and practical significance of studying the structural characteristics and transformations in nanoparticles and nanosized heterostructures is associated with the wide prospects for their use in various fields of nanotechnology. For example, they may serve as nanocontacts/nanowires, as sensors, or as catalysts. In this context, the search for stable configurations is a very important research problem [1], [2], [3]. A configuration is stable when its potential energy is minimal. This is a global optimization problem: traditional numerical methods are impractical because they need huge amounts of computational resources [4]. Therefore, the global minimum has to be approximated using time-efficient optimization strategies (metaheuristics).

A lot of methods are available for the prediction of nanoparticle structures [5]. For example, metal nanowires are studied in [6] by means of canonical Monte Carlo simulations and embedded atom potentials, demonstrating some advantages of Monte Carlo simulations over molecular dynamics simulations. In [7], grand and semigrand canonical global optimization approaches are presented, using basin-hopping with an acceptance criterion based on the local contribution of each potential energy minimum to the (semi)grand potential. Details regarding the implementation of the basin-hopping method are also given in relation to Monte Carlo moves that change the system size. The basin-hopping Monte Carlo algorithm was modified to determine a global minimum structure in Ag and AgPd nanoclusters [8]. For a pure metallic silver nanocluster, the newly developed quadratic basin-hopping Monte Carlo algorithm is more efficient than the standard basin-hopping Monte Carlo algorithm. For a bimetallic AgPd nanocluster, the new algorithm succeeds in finding the global minimum structure even though the standard algorithm fails. It is important that such approach as the formation energy machine learning model [9] can be used to predict the stable metal element distribution in the nanoparticles via Monte Carlo simulations. In [10], Monte Carlo sampling for pure random selection of sample points is used. It can be useful when implementing the so-called surrogate models, which can be a suitable replacement for complex simulation models in applications.

II. The Basic Algorithms

Our method performs the optimization on a lattice, combining two Monte Carlo algorithms. The energy of the system is given by the multi-particle tight-binding potential of Cleri–Rosato [11], having the following form:

$$E = \sum_i \sum_{j \neq i} E_{ij,ab} - \sum_i \sqrt{\sum_{j \neq i} B_{ij,ab}} \tag{1}$$

$$E_{ij,ab} = A_{ab} \exp\left(- p_{ab} \left(\frac{r_{ij}}{r_{0,ab}} - 1 \right) \right) \tag{2}$$

$$B_{ij,ab} = \xi_{ab}^2 \exp\left(- 2q_{ab} \left(\frac{r_{ij}}{r_{0,ab}} - 1 \right) \right) \tag{3}$$

where i ranges over all atoms; j ranges over all atoms other than i but within distance R_{cut} from i; a and b represent the species of the atoms i and j; $E_{ij,ab}$ and $B_{ij,ab}$ are the repulsive and binding components of the potential due to the atom pair (i, j); r_{ij} is the distance between the atoms; $r_{0,ab}$, A_{ab}, p_{ab}, ξ_{ab}, q_{ab} are constants particular to the chemical elements under considera-

□ This work was partially funded by Russian Federation of Basic Research, project number 20-37-70007, by the Ministry of Science and Higher Education of the Russian Federation in the framework of the State Program in the Field of the Research Activity, project number 0817-2020-0007, by the National Scientific Program "Information and Communication Technologies for a Single Digital Market in Science, Education and Security (ICTinSES)", Ministry of Education and Science – Bulgaria, and by the Bulgarian NSF under the grant DFNI-DN 02/10.

tion. We use a value for R_{cut} corresponding to five coordination spheres, beyond which the interaction is assumed to be zero.

A. The Wide-Lattice Monte Carlo Algorithm

The first algorithm, which we will call the "wide-lattice" Monte Carlo, is specified in [12]. It starts by placing the atoms at random on a lattice several times larger than the total number of atoms. At each iteration, one atom and one neighboring empty node are chosen at random. If the potential energy would decrease by the atom moving into the empty node, the jump is performed unconditionally. Otherwise, the jump may still be performed, with a probability given as:

$$P = \exp\left(-\Delta E/kT\right), \tag{4}$$

where ΔE is the energy difference of the configurations and T is the current temperature of the system. The iteration ends either with or without a jump.

The temperature is set high at the beginning, and then gradually decreases as the algorithm proceeds. We use a linear formula for the cooling, subtracting a small amount once every several thousand iterations. The algorithm ends when the temperature reaches 1 K.

The appropriate initial temperature strongly depends on the size and type of lattice used, as well as on the size and chemical composition of the nanostructure [13], therefore it is best determined experimentally.

Due to its simplicity and its particular form, this algorithm lends itself to a highly optimized computer implementation. It can run for billions of iterations within minutes on a standard personal computer.

B. The Diffusion Algorithm

The second algorithm, which we will call the "diffusion," is specified in [14]. It runs on a lattice filled with atoms of two different kinds, plus a small number of empty nodes (~4 for a 200-atom structure). At each iteration, one empty node is chosen at random, and the iteration always ends with a jump of a neighboring atom into that empty node. Which atom jumps is determined by calculating (4) separately for each candidate, and picking a random number in the interval from zero to the sum of all P s.

Note that the term *neighboring atom* is defined here as being within a radius of three coordination spheres. This is different from the wide-lattice Monte Carlo algorithm above, where nodes are neighbors only within one coordination sphere. This difference is due to the scarcity of the empty nodes during diffusion, and we have verified experimentally that three coordination spheres seem to be optimal for this purpose.

Temperature is managed similarly to the wide-lattice Monte Carlo algorithm.

The running time, while slower than the wide-lattice Monte Carlo, is still on the order of millions of iterations per minute.

III. THE COMBINED METHOD

The starting point of this research is the observation that combining the two algorithms above may produce better solutions than just a single-staged approach. The proposed combined method has the following steps:

Step 1: Parameter tuning. Repeatedly run the wide-lattice Monte Carlo algorithm from random initial configurations, for $N/10$ (~ 40 million) iterations at each trial, to determine the optimal initial temperature, cooling speed, and scaling factors (along each of the x, y, and z axes) for the lattice. Due to the lower number of iterations used, each trial completes quickly to save time for the more important following steps.

Step 2: Shape fixing. Using the best values from Step 1 for the initial temperature and scale factors, repeatedly run the wide-lattice Monte Carlo algorithm afresh from random initial configurations, with 1/10th the cooling speed, i.e. for N (~ 400 million) iterations at each trial. However, set up the nanoalloy to have α (~ 2) extra atoms of each type. The goal of this step is to obtain the advantageous geometric shapes appropriate for the nanoparticle (3D), film (2D) or wire (1D) under consideration on the given lattice. The resulting configurations also have a somewhat low-energy ordering of the atoms, to be further improved by diffusion.

Step 3: Diffusion. From each resulting configuration, delete all empty nodes, then convert α atoms of each type into empty nodes, to use as vacancies during diffusion. This is the same number of extra atoms added in Step 2, but the converted atoms are selected randomly. Run the diffusion algorithm once per configuration, for $N/2$ (~ 200 million) iterations at each trial.

After all Monte Carlo simulation is finished, following existing practice [12], relaxation with molecular dynamics (MD) may be used to further improve the energy of the system before selecting the best solution as the final result.

IV. VERIFICATION

We do a number of tests to verify that the proposed method is a significant improvement over simpler approaches.

All trials are performed for a 200-atom AuAg nanoparticle (gold and silver in 1:1 proportion), on a 309-node lattice with a twinned bi-pyramid shape. One example configuration of this particle is illustrated on Fig. 1. It is expected that the results also hold for other chemical compositions and other lattices (we have observed this in our preliminary testing). Instead of relaxation with MD, all comparisons of final solutions below are done after applying an additional round of scaling of the lattice (with separate factors for the x, y, and z axes).

In Fig. 2, the combined method (Steps 1+2+3) is compared against running only the wide-lattice Monte Carlo algorithm (Steps 1+2, but for a larger number of iterations).

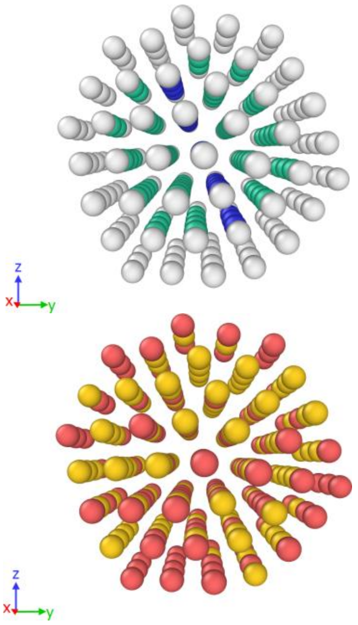
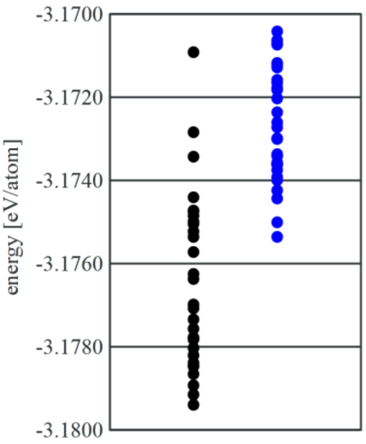


Fig. 1 One example configuration of Au100Ag100 (top: blue – fcc atoms, green – hcp atoms, grey – unknown atoms; bottom: yellow – Au atoms, red – Ag atoms)

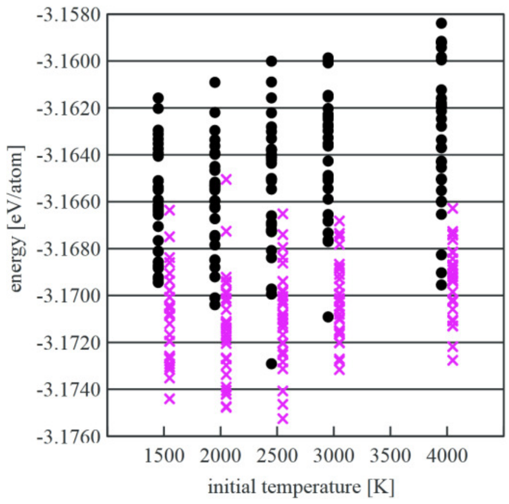
The number of iterations is chosen such that the total wall-clock running time is the same in both cases. The combined method gives clearly better solutions, which is the main result of this research.

In Fig. 3, tests verify that the parameter tuning approach of Step 1 is sound. In other words, that the optimal initial temperature determined at the higher cooling speed of Step 1 ($N/10$ iterations) is still optimal when the algorithm is run at the much lower cooling speed of Step 2 (N iterations). From the figure, it can be read out that the initial temperature giving best results is 2500 K, the same in both cases.



left side (black) dots: 30 trials	right side (blue) dots: 30 trials
combined method	wide-lattice algorithm only
Step 1: 40 million iterations	Step 1: 40 million iterations
Step 2: 400 million iterations	Step 2: 4 billion iterations
Step 3: 170 million iterations	(so that total running time is the same)

Fig. 2 Comparing the combined method (Steps 1+2+3) against using only the wide-lattice simulated annealing algorithm (Steps 1+2)



for each initial temperature:	
left side (black dots): 30 trials	right side (pink crosses): 30 trials
fast cooling (40 million iterations)	slow cooling (400 million iterations)

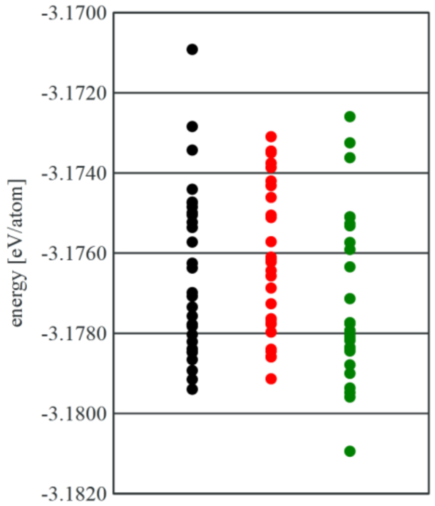
Fig. 3 Comparing the optimal initial temperature for fast cooling (such as in Step 1) versus slow cooling (such as in Step 2). The tried temperatures are the same for both cases but the crosses are shown slightly to the right of the dots for clarity.

We note that for this lattice, the differences between initial temperatures seem to be small, to the point that any temperature from the chosen range may be adequate. With larger lattices and more atoms, however, the influence of the initial temperature is more dramatic [13], and in that case the advantage of knowing the best initial temperature before running the main algorithm may also be more substantial.

The last series of tests evaluate whether the parameter tuning approach of Step 1 is advantageous. That is, whether the solutions by the full combined method are better than if we only ran Steps 2+3.

In Fig. 4, comparing the left (black) column with the middle (red) column, it can be seen that there is not much difference between the full combined method and a variant omitting Step 1 but running Step 2 for 10% more iterations (and since the best initial temperature is assumed unknown, from a random temperature in the range of 1000-4000 K). This corresponds to the fact that the influence of the initial temperature is small, as was already observed in Fig. 3. It remains open whether choosing the initial temperature at random is adequate for larger numbers of atoms, where the temperature effects are amplified.

In Fig. 4, the right (green) column shows the case of an alternative way of determining the lattice scaling factors to be applied before Step 3: instead of performing the scaling after Step 1 and using that for Steps 2+3, omit Step 1 and perform the scaling after Step 2. Comparing these solutions to the left (black) column, we see that the alternative ap-



left side (black) dots: 30 trials	middle (red) dots: 30 trials	right side (green) dots: 30 trials
full combined method	alternative 1	alternative 2
Step 1: 40 million iterations	random initial temperature	scaling of the lattice after Step 2
Step 2: 400 million iterations	Step 2: 440 million iterations	Step 2: 440 million iterations
Step 3: 170 million iterations	Step 3: 170 million iterations	Step 3: 170 million iterations

Fig. 4 Influence on the final results of using Step 1 to tune the parameters: the initial temperature (comparing left vs. middle columns), and the lattice scaling factor (comparing left vs. right columns)

proach is not substantially different. If anything, it may even look like the original (black) approach is slightly worse, but this cannot be determined from our data—the test design allows only confirming or failing to confirm a potential advantage, not a disadvantage. (The alternative (green) trials run Step 2 for 10% more iterations while still starting from the known best initial temperature, which gives them an unfair start.)

V. Conclusion

We have proposed a lattice Monte Carlo method for optimization of bimetallic nanoalloys, combining two previous algorithms: a wide-lattice simulated annealing algorithm and a simulated diffusion algorithm. We have verified that the combined method gives significantly better solutions than using only wide-lattice simulated annealing. We have discussed several ways to tune the algorithm parameters, proposing one particular approach (namely, with an additional tuning step), and verifying its soundness. As regards the alternative parameter tuning approaches investigated, they were found to be equally good to the proposed one on the relatively small lattice on which the tests were performed.

There are other important parameters to be tuned that we have not discussed here. In particular, better managing the temperature during diffusion is something that, in our experience, becomes much more important when a large number of atoms are involved. Further research will be needed to determine the optimal strategy for particle sizes where this becomes relevant. It is also worth mentioning that appropriate size effects at the nanoscale region should be taken into account, in particular, at the interface between components [15], [16].

References

[1] D. J. Wales and J. P. K. Doye, "Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms," *J. Phys. Chem. A.*, vol. 101, no. 28, pp. 5111–5116, July 1997.

[2] X. Wu and Y. Sun, "Stable structures and potential energy surface of the metallic clusters: Ni, Cu, Ag, Au, Pd, and Pt," *J. Nanopart Res.*, vol. 19, art. no. 201, July 2017.

[3] K. Michaelian, N. Rendón, and I. L. Garzón, "Structure and energetics of Ni, Ag, and Au nanoclusters," *Phys. Rev. B*, vol. 60, no. 3, pp. 2000–2010, July 1999.

[4] J. P. K. Doye, "Physical perspectives on the global optimization of atomic clusters," in *Global Optimization. Nonconvex Optimization and Its Applications*, vol. 85, J. D. Pintér, Ed. Boston, MA: Springer, 2006, pp. 103–139.

[5] S. B. Gelfand and S. K. Mitter, "Metropolis-type annealing algorithms for global optimization in {R}^d," *SIAM J. Control Optim.,* vol. 31, no. 1, pp. 111–131, 1993.

[6] M. C. Giménez and W. Schmicker, "Monte Carlo simulation of nanowires of different metals and two-metal alloys," *J. Chem. Phys.*, vol. 134, pp. 064707-1–064707-6, Febr. 2011.

[7] F. Calvo, D. Schebarchov, and D. J. Wales, "Grand and semigrand canonical basin-hopping," *J. Chem. Theory Comput.*, vol. 12, no. 2, pp. 902–909, Dec. 2015.

[8] H. G. Kim, S. K. Choi, and H.M. Lee, "New algorithm in the basin hopping Monte Carlo to find the global minimum structure of unary and binary metallic nanoclusters," *J. Chem. Phys.*, vol. 128, no. 14, pp.144702-1–144702-4, Apr. 2008.

[9] C. Chen, Y. Zuo, W. Ye, *et al.*, "Critical review of machine learning of energy materials," *Adv. Energy Mater.*, vol. 10, no. 8, 1903242-1–1903242-36, Jan. 2020.

[10] S. Balduin, F. Oest, M. Blank-Babazadeh, A. Nieße, and S. Lehnhoff, "Tool-assisted surrogate selection for simulation models in energy systems," in *Proc. 2019 FedCSIS*, pp. 185–192.

[11] F. Cleri and V. Rosato, "Tight-binding potentials for transition metals and alloys," *Phys. Rev. B*, vol. 48, no. 1, pp. 22–33, July 1993.

[12] V. Myasnichenko, N. Sdobnyakov, L. Kirilov, R. Mikhov, and S. Fidanova, "Structural instability of gold and bimetallic nanowires using Monte Carlo simulation," in *Recent Advances in Computational Optimization: Results of the Workshop on Computational Optimization and Numerical Search and Optimization 2018*, S. Fidanova, Ed. Springer, 2020, pp. 133–145.

[13] R. Mikhov, V. Myasnichenko, S. Fidanova, L. Kirilov, and N. Sdobnyakov, "Influence of the temperature on simulated annealing method for metal nanoparticle structures optimization," in *Advanced Computing in Industrial Mathematics: 13th Annu. Meet. Bulg. Sect. of SIAM, Dec. 2018, Sofia, Bulgaria,* Springer, to be published.

[14] V. Myasnichenko, R. Mikhov, L. Kirilov, N. Sdobnykov, D. Sokolov, and S. Fidanova, "Simulation of diffusion processes in bimetallic nanofilms," in *Advanced Computing in Industrial Mathematics: 14th Annu. Meet. Bulg. Sect. of SIAM, Dec. 2019, Sofia, Bulgaria,* Springer, submitted for publication.

[15] V. M. Samsonov, N. Yu. Sdobnyakov, A. G. Bembel, D. N. Sokolov, and N. V. Novozhilov, "Size dependence of the melting temperature of metallic films: two possible scenarios," *J. Nano-Electron. Phys.*, vol. 5, no. 4, pp. 04005-1–04005-3, Dec. 2013.

[16] V. M. Samsonov, N. Yu. Sdobnyakov, A. G. Bembel, D. N. Sokolov, and N. V. Novozhilov, "Thermodynamic approach to the size dependence of the melting temperatures of films," *Bull. Russ. Acad. Sci. Phys.*, vol. 78, no. 8, pp. 733–736, Sept. 2014.

Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 289–294
DOI: 10.15439/2020F35
ISSN 2300-5963 ACSIS, Vol. 21

MD-JEEP: a New Release for Discretizable Distance Geometry Problems with Interval Data

A. Mucherino,* D.S. Gonçalves,† L. Liberti,‡ J-H. Lin,§
C. Lavor,¶ N. Maculan‖

*IRISA, University of Rennes 1, Rennes, France.
antonio.mucherino@irisa.fr

†Centro de Ciências Físicas e Matemáticas, Universidade Federal de Santa Catarina, Florianópolis, Brazil.
douglas.goncalves@ufsc.br

‡LIX, École Polytechnique, Palaiseau, France.
liberti@lix.polytechnique.fr

§Research Center for Applied Sciences, Academia Sinica, Taipei, Taiwan.
jhlin@gate.sinica.edu.tw

¶Department of Applied Mathematics (IMECC-UNICAMP), University of Campinas, Campinas (SP), Brazil.
clavor@unicamp.br

‖COPPE, Federal University of Rio de Janeiro, Rio de Janeiro (RJ), Brazil.
maculan@cos.ufrj.br

***Abstract*—With the most recent releases of MD-JEEP, new relevant features have been included to our software tool. MD-JEEP solves instances of the class of Discretizable Distance Geometry Problems (DDGPs), which ask to find possible realizations, in a Euclidean space, of a simple weighted undirected graph for which distance constraints between vertices are given, and for which a discretization of the search space can be supplied. Since its version 0.3.0, MD-JEEP is able to deal with instances containing interval data. We focus in this short paper on the most recent release MD-JEEP 0.3.2: among the new implemented features, we will focus our attention on three features: (i) an improved procedure for the generation and update of the boxes used in the coarse-grained representation (necessary to deal with instances containing interval data); (ii) a new procedure for the selection of the so-called discretization vertices (necessary to perform the discretization of the search space); (iii) the implementation of a general parser which allows the user to easily load DDGP instances in a given specified format. The source code of MD-JEEP 0.3.2 is available on GitHub, where the reader can find all additional details about the implementation of such new features, as well as verify the effectiveness of such features by comparing MD-JEEP 0.3.2 with its previous releases.**

I. Introduction

Let $G = (V, E, d)$ be a simple weighted undirected graph, where vertices represent given objects (depending on the applications), and edges between two vertices indicate that the relative distance between the two vertices is known [6]. The weight function d associates the numerical value of the distance to every edge of E. This numerical value can be represented either by a singleton (in this case, we say that the distance is *exact*), or rather by an interval representing several possible distance values, delimited by a lower and an upper bound. In the general case, therefore, for two vertices u and $v \in V$ for which $\{u, v\} \in E$, the distance $d(u, v)$ is an interval $[\underline{d}(u, v), \bar{d}(u, v)]$.

Definition 1 *Given a simple weighted undirected graph $G = (V, E, d)$ and a positive integer K, the Distance Geometry Problem (DGP) asks whether a function*

$$x : v \in V \longrightarrow x_v \in \mathbb{R}^K$$

exists such that

$$\forall \{u, v\} \in E, \quad ||x_u - x_v|| \in d(u, v), \tag{1}$$

where $|| \cdot ||$ represents the Euclidean norm.

The function x is called a *realization* of the graph G. We say that a realization x that satisfies all constraints in equ. (1) is a *valid realization*. The DGP has several interesting applications, such as the one arising in structural biology for the determination of protein structures (see for example [1]), and the sensor network localization problem [4]; the reader is referred to [9] for more information about the applications.

In the last years, we have been focusing our research on a special class of DGP instances where the search space can be discretized and reduced to a tree, by transforming in this way the problem into a combinatorial problem [8]. Let E' be the subset of the edge set E such that the weight associated to the edges are "degenerate" intervals, i.e. intervals such that $\underline{d}(u,v) = \bar{d}(u,v)$.

Definition 2 *A simple weighted undirected graph G represents an instance of the Discretizable DGP (DDGP) in dimension K if and only if there exists a vertex ordering on V such that the following two assumptions are satisfied:*

(a) $G[\{1,2,\ldots,K\}]$ *is a clique whose edges are in E';*

(b) $\forall v \in \{K+1,\ldots,|V|\}$, *there exist $u_1,u_2,\ldots,u_K \in V$ such that*

- **(b.1)** $u_1 < v$, $u_2 < v$, $\ldots$, $u_K < v$;
- **(b.2)** $\{\{u_1,v\},\{u_2,v\},\ldots,\{u_{K-1},v\}\} \subset E'$, $\{u_K,v\} \in E$;
- **(b.3)** $\mathcal{V}_S(u_1,u_2,\ldots,u_K) > 0$ *(if $K > 1$),*

where $G[\cdot]$ is the subgraph induced by a subset of vertices of V, and $\mathcal{V}_S(\cdot)$ is the volume of the simplex generated by a valid realization of the vertices $u_1,u_2,\ldots,u_K$.

We will refer to assumptions **(a)** and **(b)** as the *discretization assumptions*. Such assumptions can be verified once a vertex ordering has been associated to the vertex set V, which we call a *discretization order* when the two assumptions are satisfied. For more details about discretization orders, the reader is referred to [3], [13].

In the following, for a given vertex v, we will refer to all vertices u such that $u < v$ and $\{u,v\} \in E$ as *reference vertices*. When constructing the realization x in the vertex ordering given by the discretization order, feasible positions for the reference vertices have already been computed when one is searching for positions for the current vertex v. The positions for the reference vertices, together with the corresponding distances, can therefore be exploited for defining the set of feasible positions for v.

Not all reference vertices are however necessary for the definition of a *preliminary set* of possible positions for v. As assumption **(b)** suggests, only K reference vertices are actually necessary. However, not all possible subsets of K reference vertices are able to satisfy assumption **(b)**, i.e. at least $K-1$ vertices need to be connected to a distance which is considered as exact (belonging to the subset E'), and they need to admit a realization for which the volume $\mathcal{V}_S(u_1,u_2,\ldots,u_K)$ is strictly positive (see definition above). We refer to the selected subset of reference vertices as *discretization vertices*; similarly, the distances related to discretization vertices are called *discretization distances*. Notice that the concept of discretization vertex and distance is local and related to the current vertex v.

Historically, the word "discretization" has been employed because the corresponding set of vertices and distances (when they are exact) allows us to reduce the set of feasible positions for v to a discrete set; when not enough exact distances are present, however, so that the K^{th} distance is actually represented by a nondegenerate interval (for which $\underline{d}(u,v) < \bar{d}(u,v)$), then the preliminary position set for the current vertex v is instead continuous. In the latter case, this set has the property of being the set with minimal dimensionality that can be obtained by exploiting the smallest subset of reference distances: the use of any other available distance (not marked as a "discretization" distance) would in fact not reduce the dimensionality of the set, which is bounded to remain equal to 1 (only the use of an exact distance can make the dimensionality drop to 0, which goes against our hypothesis).

In the general case [8], the preliminary position set which can be obtained by using the discretization vertices and distances for a given vertex v consists of two singletons (when all discretization distances are exact), or two arcs (otherwise). The Branch-and-Prune (BP) algorithm (see Section II) is based on the idea of constructing and exploring a search tree containing, at every layer v, the subsets of vertex positions extracted from the preliminary position sets. In case of a discrete set, every position can be assigned to a different tree node; otherwise, every node can rather contain the disjoint components of the set (corresponding to the two arcs). The additional distances (not used for the construction of the preliminary set) can be exploited to verify the feasibility of the points assigned to the tree nodes: when none of the node points are feasible w.r.t. these additional distances, then the corresponding branch of the tree can be pruned. For this reason, these additional distances are named *pruning distances*. Notice that, even if locally the nodes can contain continuous geometrical objects having dimension up to 1, the tree is a discrete structure and the general problem is therefore combinatorial, even if locally continuous. This particular structure of the search tree has in fact inspired the BP algorithm.

The first releases of MD-JEEP were able to deal with instances consisting of exact distances only [10]. Since its version 0.3.0, MD-JEEP is on GitHub[1] and implements a coarse-grained representation for the nodes of the search tree, which makes it possible to solve instances containing interval data (more details will be given in Section II). The basic idea behind the coarse-grained representation is to assign, to every node of the search tree, a pair consisting of one selected vertex position and of a K-dimensional box containing additional positions that are feasible w.r.t. all reference distances. In fact, as the search proceeds by realizing more and more vertices by following the discretization order, more and more pruning distances need to be verified and satisfied, so that the initial selections of vertex positions in the K-dimensional boxes may not always be feasible. In case the minimum and maximum distance between pairs of boxes indicates that the distances may be satisfied by other points in the boxes, then a *refinement step* can be performed, which basically consists in selecting new positions for previous vertices inside their own boxes.

In this short paper, we will present some new features introduced in the last release of MD-JEEP (version 0.3.2), which mainly aim at improving the performances of the implemented BP algorithm. In Section II, we will give a brief description of the BP algorithm and of the coarse-grained representation. In Section III, we will present three of the most relevant features introduced in last MD-JEEP release: a new

[1] https://github.com/mucherino/mdjeep

procedure to create and expand the K-dimensional boxes used in the coarse-grained representation, two new small procedures for the selection of the discretization vertices (from the set of reference vertices), and a general parser for loading instances having various formats (allowing in this way a larger target of applications without the need of format conversions). The source code of MD-JEEP 0.3.2, as well as the code of its previous releases, is available on GitHub, together with some sets of DDGP instances: the effectiveness for these newly introduced features can as well be verified by the reader. Section IV will conclude the paper with some future works.

II. A COARSE-GRAINED REPRESENTATION FOR THE BP ALGORITHM

The basic idea behind the Branch-and-Prune (BP) algorithm is to construct the search tree by performing a depth-first tree search where new potential candidate positions (to be assigned to tree nodes) for the current vertex v are computed by using its discretization vertices and distances, and by verifying the feasibility of such new positions by exploiting the pruning distances [5]. When a new position is feasible, then the branch having this position as a root is subsequently explored, until a leaf node of the tree is reached; otherwise, when the position does not respect the pruning distances, then this new branch is *virtually* pruned: it is removed from the tree which one may construct by using only discretization distances, but in the implementations it is actually not generated at all.

The coarse-grained representation initially proposed in [12] is based on the idea to assign, to every node of the search tree, a K-dimensional box B_v containing feasible positions for the current vertex v, together with a selected position $x_v \in B_v$. The function

$$z : v \in V \longrightarrow (x_v, B_v) \in \mathbb{R}^K \times \mathbb{R}^{2K}$$

is constructed as the search proceeds over the tree branches. When the vertex u is used as a reference for current vertex v, the selected position x_u is the one used as a reference to compute the preliminary position set for the vertex v. However, this position x_u can be subsequently changed to another one in B_u to allow satisfying the discretization, as well as the pruning distances, available at the tree layer v. The change of the selected position in the set B_u is performed in the part of the algorithm that we name the *refinement step*. Since the version 0.3.0 of MD-JEEP, the refinement step is performed by invoking a Spectral Projected Gradient (SPG) with non-monotone line-search [2].

Once a tree leaf node is reached, the solution to the DDGP instance can be simply obtained by extracting the positions x_v from the function z. The use of the boxes B_v can then be useful to verify how different this latest found solution is from other possible solutions that the algorithm will encounter in the further exploration of the tree (for more details, see [11] and the *resolution parameter* recently introduced in BP).

III. NEW IMPLEMENTED FEATURES

This section describes the three main features recently implemented in MD-JEEP; we will make reference to the release MD-JEEP 0.3.2. For lack of space, we will not present computational experiments in this short paper: the reader can easily verify, by using for example the instances available on MD-JEEP's GitHub repository, the usefulness of these new features by comparing the performances of MD-JEEP 0.3.2 with its previous versions. As for example, MD-JEEP 0.3.1 was not able to solve (the given answer was: 0 found solutions) some of the instances in `proteinSet2`. This set of instances (available on the repository) consists in the same instances of `proteinSet1`, but where the distances are given at low resolution (only 3 decimal digits).

A. Bound expanding procedure

Since the version 0.3.0 of MD-JEEP, it was empirically noticed that the size of the boxes B_v (over the various dimensions) was too small to have a successful refinement step in the BP algorithm (see Section II). This issue was initially solely attributed to the convergence properties of the SPG method, where it is known that too tight bounds (the size of the boxes B_v) may harm its capability to converge to a local optimum. For this reason, since the version 0.3.0 of MD-JEEP, even if initially in a very primitive form, a *bound expanding* procedure was included in the code.

A recent deeper analysis shows however that the necessity to expand the boxes does not come solely from the need of less stringent bounds in SPG. Fig. 1 (left-hand side) shows an illustration in 2D of the procedure to generate the position x_v (in green) and the box (in gray) from a computed arc (in magenta) for a three-dimensional instance (see Section II). At the center of the dashed circle, it is indicated that its center depends on both the reference vertices u_1 and u_2; the reference vertex u_3 is used for delimiting the arcs on the circle. Except for the very first vertices having a small rank in the discretization ordering (for which the preliminary position sets are discrete), the boxes related to u_1 and u_2 contain an infinite number of possible positions for the two reference vertices, from which two positions, say x_u^1 and x_u^2, are selected. Moreover, these two positions x_u^1 and x_u^2 are subject to change after every refinement step. When the box in Fig. 1 (left-hand side) is constructed, however, only the currently selected positions x_u^1 and x_u^2 are taken into consideration.

In the hypothesis where the distance between u_1 and u_2 is fixed to an exact value, the radius of all circles, obtained with different positions x_u^1 and x_u^2, is constant. Its placement, however, *floats* in the space, and so does its projection in 2D (see Fig. 1, right-hand side). There are therefore several arcs to take into consideration, which depend on the positions of the reference vertices: the actual region of space containing the feasible positions for the current vertex v is actually a larger box (in blue in the picture).

From a technical point of view, it is not advisable to compute *all* such arcs to construct the boxes B_v. Rather, in MD-JEEP 0.3.2, we have implemented a simple heuristic for the bound expanding procedure where, from the initial box for the current vertex v obtained as in the previous MD-JEEP

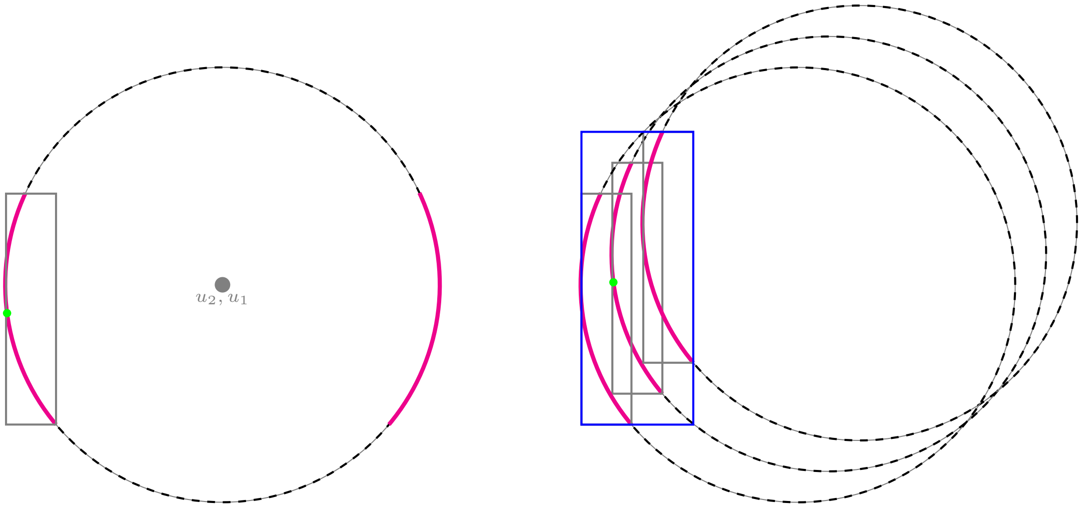


Fig. 1. On the left-hand side, the projection in 2D of two arcs representing the positions that are feasible w.r.t. the discretization distances for a given vertex v, from which the position x_v (in green) and the box B_v (in gray) can be computed. On the right-hand side, we have a similar illustration with several different positions for u_1 and u_2, which shows that the actual box (in blue) of feasible positions is larger. For simplicity, it is supposed that the distance between u_1 and u_2 is exact, and that the arc length does not depend on the positions of the reference vertices.

versions (see Fig. 1, left-hand side), we keep expanding this box as far as the new added box layer contains points that satisfy all available distances (both discretization and pruning distances). The selected position (in green in Fig. 1) is still extracted from the initial arc.

Naturally, what is obtained for B_v is only an approximation of the actual box depicted in Fig. 1 (right-hand side), but our initial experiments where BP is integrated with this heuristic are already providing promising results. In fact, MD-JEEP 0.3.2, equipped with this heuristic for the definition of the boxes, is able to perform better than its previous versions. The technical details about the implementation of the heuristic can be found on the GitHub repository.

B. *Selecting the discretization vertices*

The DDGP general case is the one where the set of discretization distances consists of $K-1$ exact distances, and 1 interval distance. There can however be some vertices that, in the given discretization order, refer only to exact discretization distances. In previous MD-JEEP versions, the selection of such discretization distances was performed by simply choosing the distances to the vertices that are closer in rank to the current vertex v. In MD-JEEP 0.3.2, we have implemented two new procedures for the selection of the discretization distances.

In both procedures, the main idea is to attempt the selection of the discretization distances that are likely to lead to the least error propagation. In case our set of reference distances contains K exact distances, then we have no possible choice: this set will be our set of discretization distances. But if more than K exact distances are actually available, it is possible to verify all combinations of selected distances for which the corresponding reference vertices form a clique: we therefore choose the set for which the angles formed by pairs of reference vertices is as far as possible from a multiple of $\pi/2$. Formulae for the computation of candidate positions for the vertices make in fact use of trigonometric functions such as *sine* and *cosine* [7], and it is well-known that an error in an angle near $\pi/2$ (or one of its multiples) can be consistently amplified in the sine and cosine values.

Finally, in case only $K-1$ reference distances are exact and the K^{th} distance needs to be selected among the available distances represented by an interval, then we simply make the choice of selecting the interval distance with the smallest range.

C. *Adding a general parser*

One of the most important technical features included in the current release of MD-JEEP is its new parser. Naturally, this is only a technical feature and does not provide any help in the solution of DDGP instances, but we decided to devote one short section to this feature because it opens the possibility for an easier exploration of the use of MD-JEEP in new applications, where the research community may have been using file formats different from the ones MD-JEEP was able to read until its previous release.

MD-JEEP 0.3.2 introduces the "MD-JEEP files" (MDfiles), with extension `mdf` (an example is given in Fig. 2). This is basically a text file containing the main specifications for the DDGP instances to be solved. In the file, every *field* is supposed to begin with its string identifier, followed by a colon and then by its name. For example, a colon needs to be positioned between the field `instance` and the instance name. Once specified in the file, every field can be followed by a certain number of lines starting with the key-word `with`, which allows the user to define the value of the attributes related to the current field. The syntax is similar: the key-word `with` needs to be followed by the attribute name, and then by its value preceded by a colon. For example, for the

```
# an example of MDfile #

# instance field
instance: protein
with file: instances/0.3/proteinSet2/1hj0.nmr
with format: Id1 Id2 groupId1 groupId2 lb ub Name1 Name2 groupName1 groupName2
with separator: ' '

# method field
method: bp
with resolution: 5.0
with tolerance: 0.001
with maxtime: 6000

# refinement field
# (all attribute values below are the default values)
refinement: spg
with eta: 0.99
with gamma:  1.e-4
with epsobj: 1.e-7
with epsg: 1.e-8
with epsalpha: 1.e-12
with mumin: 1.e-12
with mumax: 1.e+12
```

Fig. 2. An example of MDfile.

field `instance`, the attribute `file` needs to be followed by the a string with the name of the text file containing the distance list defining the DDGP instance.

The `format` attribute of the `instance` field is what gives the new parser a general-purpose type of flexibility in accepting different file formats. This attribute is a string of characters specifying the meaning of every element in the generic line of the distance list (contained in the text file specified after the key-word `file`). With this new format specification, MD-JEEP is able to point to the necessary information such as the identifiers of the two vertices (`Id1` and `Id2`) related to this distance specified at the current line, the value of the lower (`lb`) and upper (`ub`) bound for this distance, and other additional information. When some information contained in the file is not strictly necessary for MD-JEEP to solve the instance, then the key-word `ignore` may be employed as for a format element. Additional details about this new general parser can be found in the documentation available on the GitHub repository (see `README` file).

IV. Conclusions

We have presented three of the most important features introduced in the latest release of MD-JEEP. They represent another step ahead in the development of a general tool for solving DDGP instances. As for example, the implemented heuristic for expanding the boxes B_v used in the coarse-grained representation in the BP algorithm may be replaced, in the near future, with a more efficient deterministic procedure. We expect then to provide more formal descriptions, as well as pseudo-codes, for these MD-JEEP features. Moreover, a longer term project consists in extending MD-JEEP to instances consisting of *only* interval distances, which represents another important step for the generalization of the software tool.

As for the general parser that was introduced in the current MD-JEEP release, we intend to extend it in the next releases so that it will be able to load some more complex data formats, such as the ones that are generally employed by the structural biology community. One example is given by a certain number of data formats that are used for storing information obtained by Nuclear Magnetic Resonance (NMR) [1].

Acknowledgments

AM, LL and JHL are supported by the international project MultiBioStruct funded by ANR in France (ANR-19-CE45-0019) and by the Ministry of Science and Technology of Taiwan (MoST 109-2923-M-001 -004 -MY3). DG and CL are supported by CNPq. DG is also supported by CAPES/Print Process 88881.310538/2018-01, and CL is also thankful to FAPESP. NM thanks the COPPETEC Foundation.

References

[1] F.C.L. Almeida, A.H. Moraes, F. Gomes-Neto, *An Overview on Protein Structure Determination by NMR, Historical and Future Perspectives of the Use of Distance Geometry Methods*. In: [9], 377–412, 2013.

[2] E.G. Birgin, J.M. Martínez, M. Raydan, *Nonmonotone Spectral Projected Gradient Methods on Convex Sets*, SIAM Journal on Optimization **10**, 1196–1211, 2000.

[3] D.S. Gonçalves, A. Mucherino, *Optimal Partial Discretization Orders for Discretizable Distance Geometry*, International Transactions in Operational Research **23**(5), 947–967, 2016.

[4] N. Krislock, H. Wolkowicz, *Explicit Sensor Network Localization using Semidefinite Representations and Facial Reductions*, SIAM Journal on Optimization **20**, 2679–2708, 2010.

[5] L. Liberti, C. Lavor, N. Maculan, *A Branch-and-Prune Algorithm for the Molecular Distance Geometry Problem*, International Transactions in Operational Research **15**, 1–17, 2008.

[6] L. Liberti, C. Lavor, N. Maculan, A. Mucherino, *Euclidean Distance Geometry and Applications*, SIAM Review **56**(1), 3–69, 2014.

[7] D. Maioli, C. Lavor, D. Gonçalves, *A Note on Computing the Intersection of Spheres in* $\mathbb{R}^n$, ANZIAM Journal **59**, 271–279, 2017.

[8] A. Mucherino, C. Lavor, L. Liberti, *The Discretizable Distance Geometry Problem*, Optimization Letters **6**(8), 1671–1686, 2012.

[9] A. Mucherino, C. Lavor, L. Liberti, N. Maculan (Eds.), *Distance Geometry: Theory, Methods and Applications*, 410 pages, Springer, 2013.

[10] A. Mucherino, L. Liberti, C. Lavor, *MD-jeep: an Implementation of a Branch & Prune Algorithm for Distance Geometry Problems*, Lectures Notes in Computer Science **6327**, K. Fukuda et al. (Eds.), Proceedings of the 3^{rd} International Congress on Mathematical Software (ICMS10), Kobe, Japan, 186—197, 2010.

[11] A. Mucherino, J-H. Lin, *An Efficient Exhaustive Search for the Discretizable Distance Geometry Problem with Interval Data*, IEEE Conference Proceedings, Federated Conference on Computer Science and Information Systems (FedCSIS19), Workshop on Computational Optimization (WCO19), Leipzig, Germany, 135–141, 2019.

[12] A. Mucherino, J-H. Lin, D.S. Gonçalves, *A Coarse-Grained Representation for Discretizable Distance Geometry with Interval Data*, Lecture Notes in Computer Science **11465**, Lecture Notes in Bioinformatics series, I. Rojas et al (Eds.), Proceedings of the 7^{th} International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO19), Part I, Granada, Spain, 3–13, 2019.

[13] J. Omer, A. Mucherino, *Referenced Vertex Ordering Problem: Theory, Applications and Solution Methods*, HAL open archives (hal-02509522, version 1), March 2020.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 295–301 DOI: 10.15439/2020F106 ISSN 2300-5963 ACSIS, Vol. 21

Exact and approximation algorithms for sensor placement against DDoS attacks

Konstanty Junosza-Szaniawski*, Dariusz Nogalski†, Agnieszka Wójcik*
* *Warsaw University of Technology, Faculty of Mathematics and Information Science*
ul. Koszykowa 75, 00-662 Warszawa, Poland
email: {k.szaniawski, a.wojcik}@mini.pw.edu.pl
† *Military Communication Institute, C4I Systems Department*
ul. Warszawska 22A, 05-130 Zegrze, Poland
email: d.nogalski@wil.waw.pl

***Abstract*—In DDoS attack (Distributed Denial of Service), an attacker gains control of many network users by a virus. Then the controlled users send many requests to a victim, leading to lack of its resources. DDoS attacks are hard to defend because of distributed nature, large scale and various attack techniques.**

One of possible ways of defense is to place sensors in the network that can detect and stop an unwanted request. However, such sensors are expensive so there is a natural question about a minimum number of sensors and their optimal placement to get the required level of safety.

We present two mixed integer models for optimal sensor placement against DDoS attacks. Both models lead to a trade-off between the number of deployed sensors and the volume of uncontrolled flow. Since above placement problems are NP-hard, two efficient heuristics are designed, implemented and compared experimentally with exact linear programming solvers.

***Index Terms*—DDoS, sensor placement, network safety optimization, minimum multicut, heuristics.**

I. Introduction

DENIAL of Service (DoS) attacks are intended attempts to stop legitimate users from accessing a specific network resource (Zargar et al. [1]). DoS attack is an attack on availability, which is one of the three dimensions from the well known CIA security triad - Confidentiality, Integrity and Availability. Availability is a guarantee of reliable access to information by authorized people. In 1999 the Computer Incident Advisory Capability (CIAC) reported the first Distributed DoS (DDoS) attack incident (Criscuolo [2]). The attacker gets the control of a large number of users by a virus and then simultaneously performs a large number of requests to a victim server via infected machines. As a result of a large number of tasks, the victim server is out of resources and it cannot provide its service to legitimate users. DDoS attacks are also a problem in the context of Smart Grid environments (SG) (Wang et al. [3], Provos and Holz [4], Cameron et al. [5]). According to [5] availability is more critical than integrity and confidentiality for SG environments.

DDoS attacks are difficult to defend because a large number of machines may be controlled by botnets and participate in an attack, and in consequence, an attack may be launched from a large number of directions. A single bot (compromised machine) sends a small amount of traffic which looks legitimate, but the total traffic at target from the whole botnet is very high. This leads to exhaustion of resources and disruption of legal users (Mirkovic and Reiher [6], Ranjan et al. [7]). Another difficulty is that the attack pattern may be changed frequently. Typically only a subset of botnet nodes conduct the attack at the same time. After certain time, the botnet commander switches to another subset of nodes that conduct the attack.

One of the ways to defend against a DDoS attack is to place sensors in the network which recognize and stop unauthorized demands. However, placing such sensors in every node of the network would be very expensive and inefficient. Hence, a natural question which arises is what should be the number of sensors and where to place such sensors. The detection precision may be higher closer to attack sources since its easier to detect spoofed addresses and other anomalies. On the other hand the traffic closer to targets is big enough to accurately recognize actual flooding attack. In order to efficiently control the flooding, sensors should be placed in the core of the network, where most of the traffic can be observed. A taxonomy of defense mechanisms against DDoS flooding attacks including source-based, destination-based, network-based, and hybrid (a.k.a. distributed) defense mechanisms is discussed in [1].

Jeong et al. [8] and Islam et al. [9] minimize the number of sensors such that every path of a given length (r) contains a sensor. Any node less than r hops away is permitted to attack another node, since the impact of the attack is regarded low, especially for low r. In this paper we consider the problem of sensor placement under a different assumption.

In literature, one can find a well-known class of interdiction problems, which can be related to our DDoS problem. Altner et al. [10] study the Maximum Flow Network Interdiction Problem (MFNIP). In MFNIP a capacitated s-t (directed) network is given, where each arc has a cost of deletion, and a budget for deleting arcs. The objective is to choose a subset of arcs to delete, without exceeding the budget, that minimizes the maximum flow that can be routed through the network induced on the remaining arcs. The special case of MFNIP when an interdictor removes exactly k arcs from the network to minimize the maximum flow in the resulting network is known as the Cardinality Maximum Flow Network Interdiction Problem (CMFNIP) (Wood [11]). One of the recent works on interdiction problem addresses a two-stage defender-attacker game that takes place on a network whose nodes can be influenced by competing agents (Hemmati et al. [12]). A more general problem on graphs was proposed by Omer and Mucherino [13], which includes, among the others, the interdiction problem. In our DDoS problem we delete vertices, instead of arcs in CMFNIP. Additionally, we consider multiflow instead of a single flow.

From an attacker point of view, a DDoS attack can be modeled as a flow from multiple sources to single target (single commodity flow). We define a directed graph with a capacity function on edges, a set of sources (S) and a set of targets (T). We receive a set of possible attacks $P = \{(S, t_i)$, where $t_i \in T$ is a target of an attack$\}$. An attacker can conduct a single or multiple attack $p \in P$. The strength of an attack is given by a value of a maxflow for $p \in P$. The single attack $maxflow(p)$ can be computed efficiently by Ford-Fulkerson algorithm [14].

Within our DDoS defense approach we want to place sensors in network nodes, which recognize and stop unwanted traffic. If a sensor is placed in a vertex $v \in V$ then all the edges incident to v are assumed controlled. We call $D \subset V$ a set of sensors. The goal of our defense is to limit maximum uncontrolled flow towards each $t \in T$. To achieve that we minimize multi-cut. The case of a single target $|T| = 1$ can be reduced to single pair min-cut/max-flow problem and solved efficiently by a well-known Ford-Fulkerson algorithm [14]. Additionally, in such case the maximum flow is equal to the minimum cut. When the number of pairs is two the problem can be reduced to the single pair case in an undirected graph (Hu's two-commodity flow theorem [15]). When the number of pairs is more than two the problem of multi-cut becomes NP-hard. The reduction goes from the *multiterminal cuts* problem (Dahlhaus et al. [16]), also known as *multiway cuts* (Garg et al. [17]). In the multiterminal cuts problem we are given an edge-weighted graph and a subset of the vertices called terminals, and asked for a minimum weight set of edges that separates each terminal from all the others. When the number of terminals is more than two the multiterminal cuts is NP-hard (proved by Dahlhaus et al. [16]). Garg et al. [17] proved that the undirected 3-way edge cut problem can be reduced to the directed 2-way cut problem.

The main result of this paper are two mixed integer models for optimal sensor placement against DDoS attacks. One model generalises the edge multiterminal cut problem, and the other generalizes node multiterminal cut problem. Hence, both problems described by our models are NP-hard. Moreover we present two efficient heuristics (one for each problem). Finally, we present experimental comparison of solutions given by the heuristics and the mixed-integer programming solvers.

II. Problem definition

A. Problem of optimal sensor placement

Network Model: We assume that the network is modeled as a directed graph $G = (V, E)$ without multiple edges. Every directed edge has assigned a capacity by a function $c : E \rightarrow [0, \infty)$. Each node in the network can be interpreted as a router or an autonomous system.

Protected nodes: We use $T \subset V$ to denote a set of $protected$ nodes (a.k.a target nodes) in a network. Each node $v \in T$ contains a protected resource and is a target of a possible malicious flow.

Attack sources: We assume that network flooding targeted at protected nodes T can start from any network node ($source$). In practical scenario however we want to limit our attention to a set of sources $S \subset V$.

Attacks: We define a set of possible attacks $P = \{(S, t_i)$, where $t_i \in T$ is a target of an attack$\}$. We don't assume which traffic from a source $s_j \in S$ is legitimate and which one is hostile. Every potential attack $p \in P$ starts from S, is targeted at $t(p)$ and is modeled as a single-commodity flow. Routing policies allow multi-path transmissions from $s_j \in S$ to $t(p)$.

Sensors: A detection sensor can be placed in each network node. When a sensor is placed in a node $v \in V$, then all the incoming and outgoing nodes' edges are assumed controlled. A set of nodes where sensors are placed is called D.

Definition 1: **Attack flow** A function $f : P \times E \rightarrow [0, \infty)$ is called $attack\ \ flow$ if both conditions are satisfied: conservation of flow (1) and capacity constraints (2).

$$\forall_{p \in P} \forall_{u \in V \setminus \{S, t(p)\}}$$
$$\sum_{v:(v,u) \in E} f_p(v, u) = \sum_{w:(u,w) \in E} f_p(u, w), \tag{1}$$

where $f_p(u, v) = f(p, (u, v))$.

$$\forall_{e \in E} f_p(e) \leq c(e). \tag{2}$$

The attack flow value is given by

$$f_p = \sum_{v:(v,t(p)) \in E} f_p(v, t(p)) - \sum_{w:(t(p),w) \in E} f_p(t(p), w). \tag{3}$$

Definition 2: **Maxflow** By $maxflow_G(p)$, where $t(p) \in T$, we denote the maximum value of f_p (3).

Definition 3: $\mathbf{G} \setminus \mathbf{D}$ Having a graph $G = (V, E)$ and a set of sensors D. A graph $G \setminus D$ is a graph $(V \setminus D, E \setminus E_{incident(D)})$, where $E_{incident(D)}$ is a set of edges incident to $d \in D$.

Definition 4: **Uncontrolled flow** An uncontrolled flow for $t \in T$ is a flow for which $f_t > 0$ in a graph $G \setminus D$.

For example, in Fig. 2 all the incoming and outgoing edges of node 5 and 7 are controlled. An uncontrolled flow exists in a graph $G \setminus \{5, 7\}$.

In order to defend against DDoS attack we want to place sensors in a network in such a way that they can observe all or most of the traffic coming from sources S to targets T. Placing sensors in every node of the network would be very expensive and inefficient. Having a limited number of sensors available, we search for a placement such that uncontrolled flows are "distributed" among all $t_j \in T$. We want to avoid the situation in which some targets are fully protected (all traffic from S is controlled) and the other targets receive a high portion of an uncontrolled traffic, and in consequence are vulnerable to DDoS attack.

We consider two models *PC* (Placement with required Cardinality) and *PQ* (Placement with required Quality).

In the ***PC*** model we assume that the number of sensors k is given and the task is to find a k-element set $D \subset V$ such that $\max_{t \in T} maxflow_{G-D}(t)$ is minimal. Such model is important from a practical perspective. In many cases the number of available sensors is limited and one needs to find an optimal placement.

In the second model, denoted by ***PQ***, we want to minimize the number k of sensors under the assumption that the amount of uncontrolled flow does not exceed a given value. Formally, for a given number $q \in [0, 1]$, we ask what a minimal number $k \in \mathbb{N}$ is such that there exists a k-element set $D \subset V$ such that

$$\max_{t \in T} maxflow(t)_{G-D} \leq (1-q) \cdot \max_{t \in T} maxflow_G(t).$$

For $q = 1$ we get the question: what is the minimal number of sensors that guarantee the total control in the network.

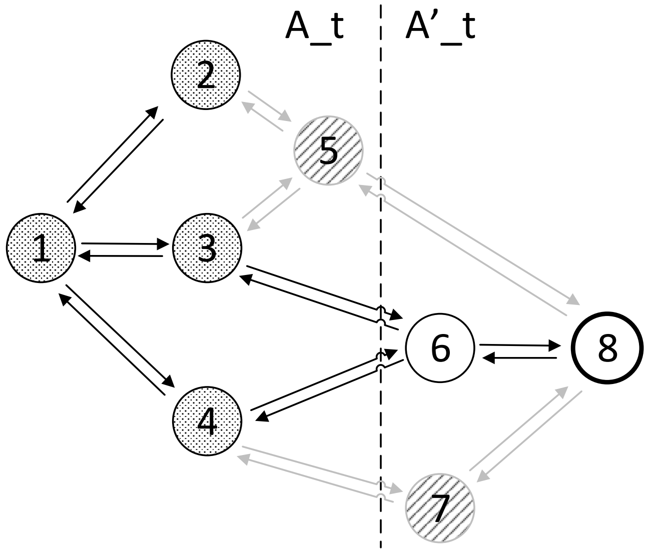


Fig. 1. Cut for $t = 8 \in T$ in G, source nodes $S = \{1, 2, 3, 4\}$, protected nodes $T = \{8\}$ and sensors $D = \{5, 7\}$.

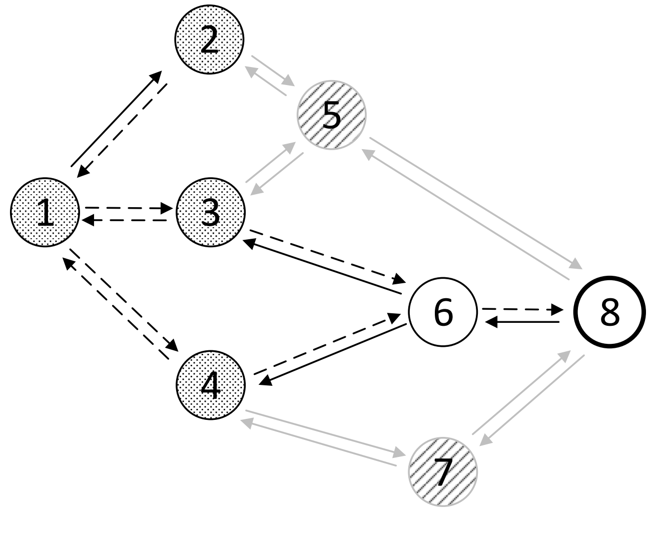


Fig. 2. Uncontrolled flow in G (dashed lines), source nodes $S = \{1, 2, 3, 4\}$, protected nodes $T = \{8\}$ and sensors $D = \{5, 7\}$.

B. Complexity of the optimal sensor placement

In the multiway cuts problem we are given a directed graph $G = (V, E)$ with edge capacities $c : E \to \mathbb{R}^+$, and a set of t terminals $S = \{s_1, s_2 ... s_t\}$. A (edge) multiway cut in G is a set of nodes (edges) whose deletion separates every terminal from every other terminal (i.e. the remaining graph does not contain any path from s_i to s_j for $i \neq j$). The problem of computing the minimum (edge) multiway cut in directed graphs is NP-hard in case $t \geq 2$ (Garg et al. [17], Theorem 3.1).

Notice that the minimum node multiway cut can be reduced to the problem described by PQ model for $q = 1$. Moreover the minimum edge multiway cut problem can be reduced to the problem described by PC model for $k = 0$. Hence both our problems are NP-hard.

III. Models description

Basic formulation of PC and PQ models: To solve the problem of optimal sensor placement in the sense of models *PC* and *PQ* we use mix-integer programming. Our solution is based on a well-known Ford-Fulkerson Theorem [14] stating that the maximum flow cannot exceed the minimum cut and actually, in our solution we minimize the min-cuts. To compute minimum cuts for every target $t \in T$ we introduce a set A_t such that any edge u, v is in a cut for t if and only if $u \in A_t$ and $v \notin A_t$ (Fig. 1). The set $D \subseteq V$ denotes the set of vertices in which we place sensors. We start with the *PC* model.

Formally, we define the following variables:

- For every $v \in V$ a binary variable $d[v]$ with the meaning $d[v] = 1$ if and only if $v \in D$ (there is a sensor in the vertex v).
- For every $t \in T$ and $v \in V$ a binary variable $a[t, v]$ with the meaning $a[t, v] = 1$ if and only if $v \in A_t$. The sets A_t allow us to compute a cut for the target $t \in T$.
- For every $t \in T, e \in E$ a binary variable $cutT[t, e]$ with the meaning $cutT[t, e] = 1$ if and only if $e \in$ belongs to a cut in $G - D$ for t.
- A real variable $M \in \mathbb{R}$, that denotes the value of the minimum cut in $G - D$.

A function to minimize is just M with respect to the below restrictions (4)-(8). For every target $t \in T$ each vertex $s \in S$ belongs to A_t (4). For every target $t \in T$ the vertex t does not belong to A_t (5). The restriction (6) guarantees that an edge belongs to a cut if none of its ends is in a set D, the first vertex is in A_t and the second vertex is not. The restriction (7) makes sure that the number of sensors is fixed. Finally, the equation (8) bounds the value of the cut with M.

$$\forall_{t \in T} \ \forall_{s \in S} \ \ a[t,s] == 1 \tag{4}$$

$$\forall_{t \in T} \ \ a[t,t] == 0 \tag{5}$$

$$\begin{aligned} &\forall_{t \in T} \ \forall_{(u,v) \in E} \\ &cutT[t,u,v] \geq a[t,u] - a[t,v] - d[u] - d[v] \end{aligned} \tag{6}$$

$$\sum_{v \in V} d[v] = k \tag{7}$$

$$\forall t \in T \quad \sum_{(u,v) \in E} cutT[t,u,v] \cdot c[u,v] \leq M \tag{8}$$

To obtain the *PQ* model it is enough to replace the target function to minimize by $\sum_{v \in V} d[v]$, omitting the restrictions (7) and (8), and adding the restriction

$$\begin{aligned} \forall t \in T \sum_{(u,v) \in E} & cutT[t,u,v] \cdot c[u,v] \leq \\ & (1-q) \cdot \max_{t \in T} maxflow_G(t) \end{aligned} \tag{9}$$

where $\max_{t \in T} maxflow_G(t)$ is equal to the value of max minimum cut M_t in G (result of *PC* model for $k=0$).

IV. Algorithms description

Relaxed formulation of PC and PQ models: In this formulation we relax two types of variables to allow the fractional sensor placement (first) and fractional traffic control (second):

- For every $v \in V$ a real variable $d[v] \in [0,1]$
- For every $t \in T, e \in E$ a real variable $cutT[t,e] \in [0,1]$.

In the basic model formulation (section III) when an edge u,v is in a cut for some t ($u \in A_t$ and $v \notin A_t$), placing a sensor in either u or v classifies such edge as fully controlled. When no sensor is placed in neither u nor v such edge is uncontrolled. Whereas in the relaxed formulation we allow fractional sensor placement (d variables) and fractional control of edges in a cut ($cutT$ variables).

To solve the *PC* and *PQ* problems, additionally to our two models (section III), we have designed and implemented two algorithms:

1) *PCIterativeBestSensor* (see alg 1)
2) *PQIterativeBestSensor* (see alg 2).

Both algorithms assume the following common *input* parameters: G graph representing a network with c capacity function, T set of targets and S set of sources. Additionally, *PCIterativeBestSensor* heuristics takes k (number of sensors) as input and *PQIterativeBestSensor* heuristics q (quality factor).

A. PC Iterative Best Sensor Placement

The algorithm *PCIterativeBestSensor* constitutes $k+1$ iterations. In each $\{1,..,k\}$ iteration, linear program relaxation is solved (line 5). From the relaxed LP solution a subset of vertices L is selected from the set $V \setminus D$ such that $d[v] \neq 0$ and $d[v] == max\{d[j]\}_{j \in V \setminus D}$ (line 6). Among the $|L|$ best sensor locations, the single best (max) one v_{max} is selected and added to the model as a constraint (line 8). The constraint fixes a sensor in the location v_{max} in the next iterations.
In the last iteration, the LP relaxation is solved assuming fixed sensor placements for all $v \in D$ (line 11).

Algorithm 1 PCIterativeBestSensor

```
1:  Require $G, c, T, S, k$
2:  Create the relaxed PC problem (section IV) with goal
    $minimize\ M$. Add constraints {(4),(5),(6),(7),(8)} to the
    $problem$.
3:  Let's initiate a set of vertices in which we place sensors
    $D = \emptyset$
4:  for $i = 1,..,k$ do
5:      Solve the $problem$
6:      Let $L = \{v$, s.t. $v \in V \setminus D$ and $d[v] \neq 0$ and $d[v] ==$
        $max\{d[j]\}_{j \in V \setminus D}\}$
7:      Choose randomly $v_{max} \in L$, where probability of
        selecting an element $v_{max}$ equals $\frac{1}{|L|}$
8:      Add constraint $d[v_{max}] == 1$ to the $problem$
9:      $D = D \cup \{v_{max}\}$
10: end for
11: Solve the $problem$
12: Retrieve $M$ from the $problem$ solution
13: Return $(D, M)$
```

B. PQ Iterative Best Sensor Placement

The preparatory step of the algorithm *PQIterativeBestSensor* is a computation of the value of $\max_{t \in T} maxflow_G(t)$ (line 2). In each while loop, linear program relaxation is solved (line 6). From the relaxed LP solution a subset of vertices L is selected from the set $V \setminus D$ such that $d[v] \neq 0$ and $d[v] == max\{d[j]\}_{j \in V \setminus D}$ (line 7). Among the $|L|$ best sensor locations, the single best (max) one v_{max} is selected and added to the model as a constraint (line 9). The constraint fixes a sensor in the location v_{max} in the next iterations.

V. Computational results

A. Experiment Setup

The two models *PC* and *PQ* and two algorithms *PCIterativeBestSensor* and *PQIterativeBestSensor* were run with the use of CPLEX 12.10 for Python. Python 3.7 was utilized to implement heuristics and automate simulations. The simulations were run on a personal computer with 1.9GHz CPU, 16GB RAM and 64-bit Windows platform.

The experiments were conducted on 9 types of grid networks: $Net|V|$, where $|V| = \{64, 81, 100, 121, 144, 169, 196, 225, 256\}$ indicates the

Algorithm 2 PQIterativeBestSensor

```
Require $G, c, T, S, q$
Compute a value of $\max_{t \in T} maxflow_G(t)$
Create the relaxed $PQ\ problem$ (section IV) with goal $minimize \sum_{v \in V} d[v]$. Add constraints {(4),(5),(6),(9)} to the $problem$
Let's initiate a set of vertices in which we place sensors $D = \emptyset$
while ($\exists t \in T \sum_{(u,v) \in E} cutT[t,u,v] \cdot c[u,v] > (1-q) \cdot \max_{t \in T} maxflow_G(t)$) do
    Solve the $problem$
    Let $L = \{v$, s.t. $v \in V \setminus D$ and $d[v] \neq 0$ and $d[v] == max\{d[j]\}_{j \in V \setminus D}\}$
    Choose randomly $v_{max} \in L$, where probability of selecting an element $v_{max}$ equals $\frac{1}{|L|}$
    Add constraint $d[v_{max}] == 1$ to the $problem$
    $D = D \cup \{v_{max}\}$
end while
Return $D$
```

number of vertices in a network. All of these networks are directed graphs, with a single edge in each direction u, v and v, u. An example of a small grid network is demonstrated in Fig. 3. Each vertex in a graph may correspond to a router or an autonomous system in telecommunication network.

For the purpose of simulation scenarios, for each network type, four random instances of each network type were generated, each with randomly selected capacities (c). Each edge capacity was randomly selected from the range $c(e)_{e \in E} \in < 100, 200 >$ (random selection with uniform distribution). Additionally, for each simulation scenario, four random instances of target locations $T_{i=1..4} \subset V$) were generated (all vertices V have equal probabilities). For each target instance T_i, four random instances of source locations were generated ($S_{j=1..4} \subset V \backslash T_i$)(all vertices $V \backslash T_i$ have equal probabilities). As a result, each value (volume of uncontrolled flow; execution time) presented on each diagram is an average computed from 64 measurements. Finally, for all scenarios we assumed $|T| = 10$ and $|S| = 40$.

B. Scenario1: PC problem, Net100, increasing number of sensors

The experiments were conducted for the grid network *Net100*. The number of sensors was increasing from $k = 0$ to $k = 10$.

The diagram Fig. 4 demonstrates the average volume of uncontrolled traffic (y axis) depending on the number of sensors. As the number of sensors increases, the average volume of uncontrolled traffic decreases to zero (for $k = |T|$), for both *PC* model and *PCIterativeBestSensor* heuristics. The observed average objective values of *PCIterativeBestSensor* are higher than those of *PC* by up to 8%.

The diagram Fig. 5 demonstrates the average time of execution (y axis). The observed average values of execution time of *PC* are up to 10 times higher than those of *PCIterativeBestSensor*.

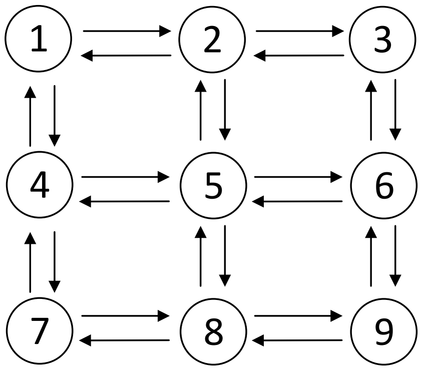


Fig. 3. An example of a small grid network $|V| = 9$

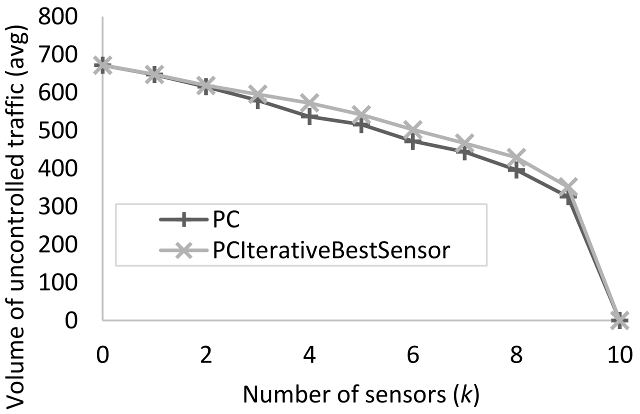


Fig. 4. Scenario1, volume of uncontrolled traffic (avg), *PC* vs. *PCIterativeBestSensor*

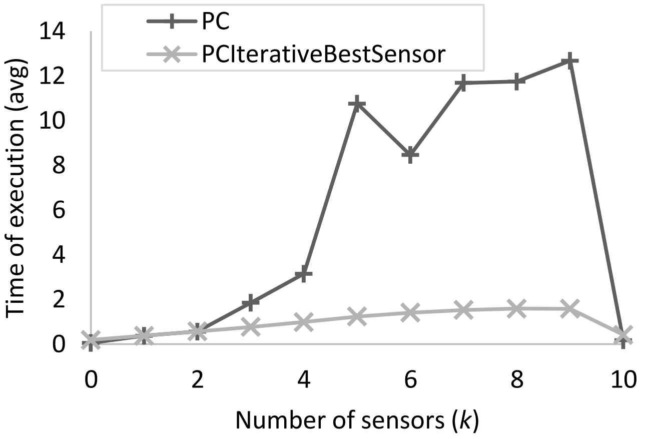


Fig. 5. Scenario1, time of execution (avg) [s], *PC* vs. *PCIterativeBestSensor*

C. Scenario2: PC problem, k=5, increasing size of the grid Net64, Net81, ... , Net169

The experiments were conducted for the grid networks: *Net64, Net81, Net100, Net121, Net144, Net169*. The number of sensors was fixed $k = 5$.

The diagram Fig. 6 demonstrates the average time of execution (y axis) as the size of the network increases ($|V|$). As $|V|$ grows, the gap between *PCIterativeBestSensor* and *PC* increases significantly in favour of the heuristics.

D. Scenario3: PQ problem, Net196, increasing value of quality factor

The experiments were conducted for the grid network *Net196*. The value of quality factor was increasing $q \in \{0.1, 0.2, ..., 1.0\}$.

The diagram Fig. 7 demonstrates the average number of sensors (y axis) required to control the q-factor of the network traffic (x axis). As the value of q-factor increases, the number of required sensors increases on average, for both *PQ* model and *PQIterativeBestSensor* heuristics. However, at a certain point sensor usage becomes saturated. In the worst observed cases *PQIterativeBestSensor* required approximately one sensor more than *PQ* to achieve the same quality.

The diagram Fig. 8 demonstrates the average time of execution (y axis). The observed average values of execution time of *PQ* are up to 5 times higher than those of *PQIterativeBestSensor*.

E. Scenario4: PQ problem, q=0.5, increasing size of the grid Net121, Net144, ... , Net256

The experiments were conducted for the grid networks: *Net121, Net144, Net169, Net196, Net225, Net256*. The quality factor was fixed $q = 0.5$.

The diagram Fig. 9 demonstrates the average time of execution (y axis) as the size of the network increases ($|V|$). As $|V|$ grows, the gap between *PQIterativeBestSensor* and *PQ* increases significantly in favour of the heuristics.

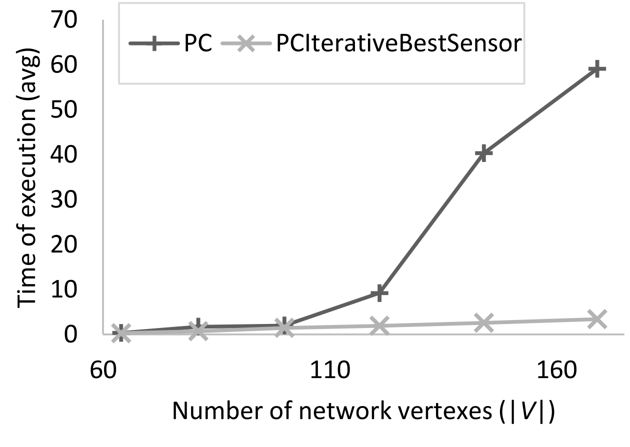


Fig. 6. Scenario2, time of execution (avg) [s], *PC* vs. *PCIterativeBestSensor*

F. Summary of simulation results

The *PC* algorithms simulations lead to a number of observations. Firstly, for all test networks, as the number of sensors

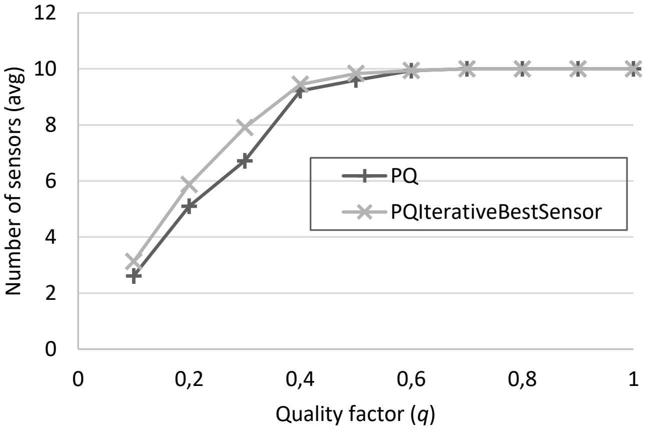


Fig. 7. Scenario3, number of sensors (avg), *PQ* vs. *PQIterativeBestSensor*

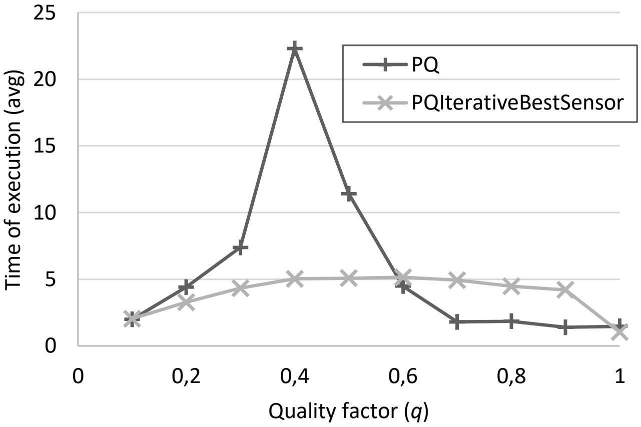


Fig. 8. Scenario3, time of execution (avg) [s], *PQ* vs. *PQIterativeBestSensor*

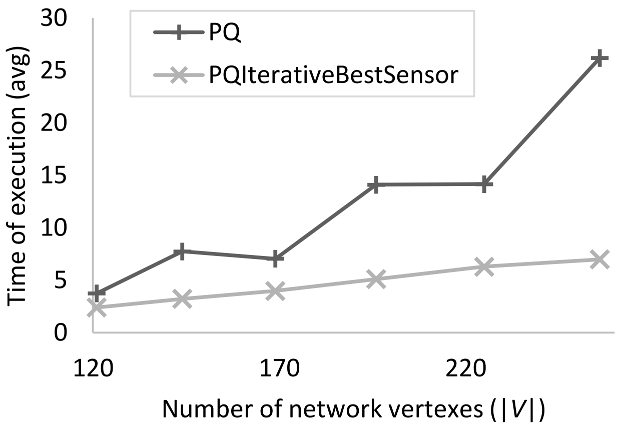


Fig. 9. Scenario4, time of execution (avg) [s], *PQ* vs. *PQIterativeBestSensor*

increases, the volume of uncontrolled traffic decreases to zero, for both *PC* model and *PCIterativeBestSensor* heuristics. Secondly, the observed average objective values of *PCIterativeBestSensor* are higher than those of *PC* by up to 8% for tested networks. Finally, as the size of the grid network increases, for fixed k, the execution time gap between *PCIterativeBestSensor* and *PC* increases significantly in favour of the heuristics.

The *PQ* algorithms simulations lead to the following observations. Firstly, as the quality factor increases, the number of sensors increases on average, however, at a certain point sensor usage becomes saturated, for both *PQ* model and *PQIterativeBestSensor* heuristics. Secondly, in the worst observed cases *PQIterativeBestSensor* required approximately one sensor more than *PQ* to achieve the same quality. Finally, as the size of the grid network increases, for fixed q, the execution time gap between *PQIterativeBestSensor* and *PQ* increases significantly in favour of the heuristics.

VI. Conclusions

As demonstrated for some medium-sized grid networks, computation time is not high and qualifies both *PC* and *PQ* models for practical applications. The models respond to the challenges of the real DDoS problem. One challenge is that an attack can be conducted from any network node. The other is that sensors are expensive and placing them in all network nodes is not possible in many cases. Sensors can be placed dynamically based on perceived network indicators. The models expose a highly desirable feature, such that dislocation of relatively small number of sensors (proportional to the number of protected nodes) can obtain a significant quality. Both models lead to a trade-off between the number of deployed sensors and the volume of uncontrolled flow.

Finally, for large networks, the execution time gap between the two models and their corresponding heuristics increases significantly in favour of the heuristics.

References

[1] S. T. Zargar, J. Joshi, and D. Tipper, “A survey of defense mechanisms against distributed denial of service (DDOS) flooding attacks,” *IEEE Communications Surveys and Tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013. doi: 10.1109/SURV.2013.031413.00127

[2] P. J. Criscuolo, “Distributed Denial of Service Trin00, Tribe Flood Network, Tribe Flood Network 2000, And Stacheldraht, CIAC-2319,” *Department of Energy Computer Incident Advisory Capability (CIAC), Lawrence Livermore National Laboratory*, 2000.

[3] K. Wang, M. Du, S. Maharjan, and Y. Sun, “Strategic honeypot game model for distributed denial of service attacks in the smart grid,” *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2474–2482, Sep. 2017. doi: 10.1109/TSG.2017.2670144

[4] N. Provos and T. Holz, *Virtual Honeypots: From Botnet Tracking to Intrusion Detection*. Addison-Wesley, 2007. ISBN 978-0321336323

[5] C. Cameron, C. Patsios, P. C. Taylor, and Z. Pourmirza, “Using Self-Organizing Architectures to Mitigate the Impacts of Denial-of-Service Attacks on Voltage Control Schemes,” *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 3010–3019, 2019. doi: 10.1109/TSG.2018.2817046

[6] J. Mirkovic and P. Reiher, “A taxonomy of DDoS attack and DDoS defense mechanisms,” *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, p. 39, 2004. doi: 10.1145/997150.997156. [Online]. Available: http://portal.acm.org/citation.cfm?doid=997150.997156

[7] S. Ranjan, R. Swaminathan, M. Uysal, A. Nucci, and E. Knightly, “DDoS-shield: DDoS-resilient scheduling to counter application layer attacks,” *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, pp. 26–39, 2009. doi: 10.1109/TNET.2008.926503

[8] S. B. Jeong, Y. Choi, and S. Kim, “An effective placement of detection systems for distributed attack detection in large scale networks,” in *Information Security Applications, 5th International Workshop, WISA 2004, Jeju Island, Korea, August 23-25, 2004, Revised Selected Papers*, ser. Lecture Notes in Computer Science, C. H. Lim and M. Yung, Eds., vol. 3325. Springer, 2004. doi: 10.1007/978-3-540-31815-6_17 pp. 204–210. [Online]. Available: https://doi.org/10.1007/978-3-540-31815-6_17

[9] M. H. Islam, K. Nadeem, and S. A. Khan, “Efficient placement of sensors for detection against distributed denial of service attack,” *2008 International Conference on Innovations in Information Technology, IIT 2008*, pp. 653–657, 2008. doi: 10.1109/INNOVATIONS.2008.4781681

[10] D. S. Altner, Ö. Ergun, and N. A. Uhan, “The maximum flow network interdiction problem: Valid inequalities, integrality gaps, and approximability,” *Oper. Res. Lett.*, vol. 38, no. 1, pp. 33–38, 2010. doi: 10.1016/j.orl.2009.09.013. [Online]. Available: https://doi.org/10.1016/j.orl.2009.09.013

[11] R. Wood, “Deterministic network interdiction,” *Mathematical and Computer Modelling*, vol. 17, no. 2, pp. 1 – 18, 1993. doi: 10.1016/0895-7177(93)90236-R. [Online]. Available: http://www.sciencedirect.com/science/article/pii/089571779390236R

[12] M. Hemmati, J. Cole Smith, and M. T. Thai, “A cutting-plane algorithm for solving a weighted influence interdiction problem,” *Computational Optimization and Applications*, vol. 57, no. 1, pp. 71–104, Jan. 2014. doi: 10.1007/s10589-013-9589-9. [Online]. Available: https://doi.org/10.1007/s10589-013-9589-9

[13] J. Omer and A. Mucherino, “Referenced vertex ordering problem: Theory, applications and solution methods,” Mar. 2020, working paper or preprint. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02509522

[14] L. R. Ford and D. R. Fulkerson, “Maximal flow through a network,” *Canadian Journal of Mathematics*, vol. 8, p. 399–404, 1956. doi: 10.4153/CJM-1956-045-5

[15] T. Hu, “Multi-commodity network flows,” *Operations Research*, vol. 11, no. 3, p. 344–360, 1963. doi: 10.1287/opre.11.3.344

[16] E. Dahlhaus, D. Johnson, C. Papadimitriou, P. Seymour, and M. Yannakakis, “The complexity of multiterminal cuts,” *SIAM Journal on Computing*, vol. 23, no. 4, pp. 864–894, 1994. doi: 10.1137/S0097539792225297

[17] N. Garg, V. V. Vazirani, and M. Yannakakis, “Multiway cuts in directed and node weighted graphs,” in *Automata, Languages and Programming, 21st International Colloquium, ICALP94, Jerusalem, Israel, July 11-14, 1994, Proceedings*, ser. Lecture Notes in Computer Science, S. Abiteboul and E. Shamir, Eds., vol. 820. Springer, 1994. doi: 10.1007/3-540-58201-0_92 pp. 487–498.

Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 303–306 DOI: 10.15439/2020F13 ISSN 2300-5963 ACSIS, Vol. 21

Pipe-lining Dynamic Programming Processes in Order to Synchronize Energy Production and Consumption

Fariha Bendali
LIMOS CNRS/UCA
Clermont-Fd, France
Email: bendali@isima.fr

Jean Mailfert
LIMOS CNRS/UCA
Clermont-Fd, France
Email: mailfert@isima.fr

Alain Quilliot
LIMOS CNRS UMR 6158
LABEX IMOBS3, Université
Clermont-Auvergne
Bat. ISIMA, BP 10125
Campus des Cézaux,
63173 Aubière, France
Email: quilliot@isima.fr

Eloise Mole Kamga
LIMOS CNRS UMR 6158
63173 Aubière, France
Email: eloise@isima.fr

Hélène Toussaint
LIMOS CNRS UMR 6158
LABEX IMOB3
Clermont-Fd, France

***Abstract*—Synchronizing heterogeneous processes remains a difficult issue in Scheduling area. Related ILP models are in trouble. So we propose here a pipe-line collaboration of a dynamic programming process for energy production and consumption scheduling.**

I. Introduction

EFFICIENTLY synchronizing heterogeneous process remains a difficult issue when it comes to scheduling. ILP models are flawed by large gaps induced by the relaxation of the integrality constraint (the *Big M* problem). This difficulty also arises when one wants to plan industrial, domestic or local logistics activities, while relying on local renewable energy production: Due to both market deregulation and emergent technologies, the rise local producers (factories, farms,…) while simultaneously remaining consumers (see [1, 6]) tends to make this issue a trend in Energy Economics. In the context of Labex IMOBS3 project in Clermont-Fd, France, devoted to *Innovative Mobility*, we are involved into the control of a local micro-plant for hydrogen (H^2) production, which provides autonomous vehicles in H^2 fuel. Researchers rely here on solar power and photolysis ([4, 6, 7]): so the productivity of the process deeply depend on the intensity of solar illumination. Few works address resulting synchronization issue (see general contribution in [1], studies about electric vehicles (*Green VRP*, *Pollution-Routing Problem*,…in [3, 8]), and industrial processes (see [2, 5]) under time-dependent energy costs and access restrictions.

Because of the IMOBS3 project, present contribution is about the synchronous management of, on one side, a fleet of small electric vehicles provided with H^2 power cells, and, on the other side, a micro-plant in charge of local H^2 fuel production. Taken as a whole, resulting model of Section II involves forecasting, safety management and scheduling. We only address here the last issue, while considering only one vehicle, required to perform tasks according to a pre-fixed order, which periodically goes back to the micro-plant in order to refuel. The micro-plant has its own production/storage restrictions. Relying on ILP is inefficient, and so we first propose in Section III an exact *Dynamic Programming Scheme* (DPS). But, though this DPS allows us to state a PTAS (*Polynomial Time Approximation Scheme*) result, it remains time-costly in practice. So we decompose it (Section IV) into 2 DPS sub-processes, one related to the vehicle, and the other one to the micro-plant which collaborate through a *pipe-line*.

II. The Energy Production/consumption problem (EPC)

Some vehicle has to perform internal logistics tasks, while following a route Γ which starts from some *Depot* node and ends in the same way after going through stations $j = 1, \ldots, M$, according to this order. Start-node *Depot* has label 0 and End-node *Depot* has label $M + 1$. The time required by the vehicle in order to go from j to $j + 1$ is equal to t_j, (including *service* time). The vehicle may leave *Depot* at time 0 and should finish its route no later than some threshold time *TMax*. It is powered by hydrogen (H^2) fuel. The capacity of its tank is denoted by C^{Veh} and we know, for any $j = 0, \ldots, M$, the H^2 amount e_j required in order to move from station j to station $j + 1$. The initial H^2 load of the vehicle is denoted by E_0, and the vehicle is required to end its trip with at least the same energy load. It comes that the vehicle must periodically refuel. Refueling transactions take place at a *micro-plant*, close to *Depot*: The time required by the vehicle in order to move from station j to the micro-plant (from the micro-plant to j) is denoted by d_j (d^*_j); by the same way, the energy required in order to move from j to the micro-plant (from the micro-plant to j) is denoted by ε_j (ε^*_j). Figure 1 displays an example of a trip performed by the vehicle along station *Depot* = 0, 1, 2, 3, 4, 5, 6 = *Depot*.

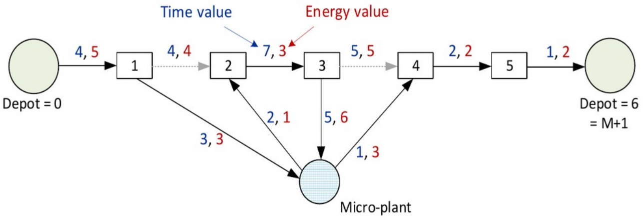


Figure 1. *A vehicle trip, with its refueling transactions*

On another side, the micro-plant produces H^2 *in situ* through photolysis&electrolysis. Resulting H^2 is stored inside the micro-plant's tank, with capacity equal to C^{MP}. We suppose that the time space {0, ..., *TMax*} is divided into periods $P_i = [p.i, p.(i + 1)[$, $i = 0, \ldots, N - 1$, with $TMax = N.p$ (see Figure 3). We identify index i and period P_i. If the micro-plant is *active* at some time during period i, then it is active during the whole period i, and produces R_i hydrogen fuel units. At time 0, the load of the micro-plant tank is $H_0 \leq C^{MP}$ and the micro-plant is idle. This should also hold at time *TMax*. Because of safety concerns, the vehicle cannot refuel while the micro-plant is producing and any vehicle refueling transaction takes a whole period i. Besides, producing H^2 fuel has a cost, which may be decomposed into:

- A constant *activation cost* $Cost^F$, which is charged every time the micro-plant is activated.

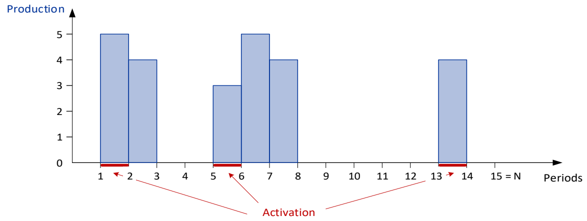


Figure 2. ***An example of micro-plant activity, with N = 15***

- A *time-dependent* production cost $Cost^V_i$ which reflects the time-indexed prices charged by the electricity provider.

Then the *Energy Production/Consumption* (**EPC**) Problem consists in scheduling both the vehicle and the micro-plant in such a way that:

- The vehicle starts from *Depot* = 0, visits all stations $j = 1, \ldots, M$ and comes back to *Depot* at some time $T \in [0, TMax]$, while refueling every time it is necessary;
- The micro-plant produces and stores in time the H^2 fuel needed by the vehicle;
- Both induced H^2 production cost *Cost* and time T are the smallest possible: $Min = Cost + \alpha.T$, where α is some scaling coefficient.

Figure 3 below shows the synchronization between the vehicle and the micro-plant of fig. 1, 2, in case $p = 2$, $E_0 = 8$, $H_0 = 4$, $TMax = 30$, $Cost^F = 7$, $C^{MP} = 15$, $C^{Veh} = 15$, $\alpha = 1$.

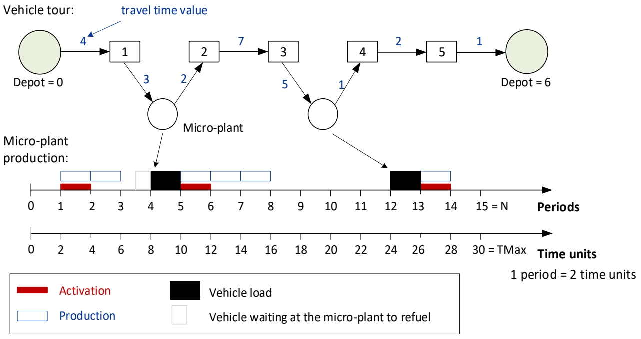


Figure 3. *A feasible solution related to Fig. 1 and 2.*

III. A *DPS-EPC* Algorithm.

EPC is *NP-Hard*: It can be reduced to Knapsack. We first handle it through DPS (*Dynamic Program Scheme*):

DPS *Time Space* and *States*: The *time space* is the set Δ of *time pairs* (i, j), $i = 0, \ldots, N$, $j = 0, \ldots, M + 1$.We link periods i and stations j through relations (<<, >>, ==) which locate period i with respect to time value $T \in \{0, \ldots, TMax\}$:

- $T << i$ if $T < p.i$; $T >> i$ if $T \geq p.(i + 1)$;
- $T == i$ if $p.i \leq T < p.(i + 1)$.

For any such a *time pair* (i, j), a related *state* is a 4-uple $s = (Z, T, V^{Tank}, V^{Veh})$, with:

- $Z = 1$: micro-plant active at the end of period $i - 1$.
- V^{Tank} and V^{Veh} are respectively the loads of the micro-plant at the beginning of i and the vehicle when it arrives at j;
- T is a value in 0, …, *TMax* with the meaning:
  - ○ $T >> i$: the vehicle will reach j at time T;
  - ○ $T << i$: the vehicle is between j and the micro-plant, possibly waiting for being refueled;
  - ○ $T == i$: the vehicle is in j, and decides between riding to $j + 1$ or to the micro-plant.

Initial *state* corresponds to *time pair* (0, 0) and 4-uple $s_0 = (0, 0, H_0, E_0)$. Final *state* corresponds to any *time pair* $(i \leq N, M + 1)$, and any 4-uple $(Z, T \leq TMax, V^{Tank} \geq H_0, V^{Veh} \geq E_0)$.

Decisions/Transitions/Costs. Then a decision D is a 3-uple $D = (z, x, \delta)$ in $\{0, 1\}^3$, with the meaning:

- $z = 1$ ~ the micro-plant produces during period i;
- x refers the case $T == i$: $x = 0$ means that the vehicle rides from j to $j + 1$ without refueling; $x = 1$ means that it refuels at the micro-plant while ridingfrom j to $j + 1$.
- $\delta = 1$ ~ the vehicle is located at the micro-plant and decides to refuel during period i, forbidding the micro-plant to be active during this period. It requires $T << i$ and $p.i - T \geq d_j$.

Decision is taken at the end of period $i - 1$. For any *time pair* (i, j) and *state* $s = (Z, T, V^{Tank}, V^{Veh})$, no more than 4 decisions D are feasible:

- **1 th case**: $T >> i$. Then the only choice is about z.
- **2 th case:** $T << i$ and $p.i - T < d_j$. The vehicle is moving from j to the micro-plant and cannot refuel yet. Once again, the only choice is about z.
- **3 th case:** $T << i$ and $p.i - T \geq d_j$. Then, we have 3 choices: 1). **Producing**: $z = 1$; $\delta = 0$; 2). **Refueling**: $z = 0$; $\delta = 1$; 3). **Doing nothing**: $z = 0$; $\delta = 0$. l.
- **4 th case:** $T == i$. Then we have 4 choices:
  - ○ **Producing and riding towards j+1:** $z = 1$, $x = 0$.
  - ○ **Not Producing, riding towards j+1:** $z = 1$, $x = 0$.
  - ○ **Not Producing, riding to micro-plant:** $z = 0$, $x = 1$.
  - ○ **Producing, riding to micro-plant:** $z = 1$, $x = 1$.

We implement Bellman Equations through a *Forward Driven* Strategy and denote by ***DPS-EPC*** the algorithm designed this way. In order to control th number of states, we need to enhance it with filtering devices.

A. Filtering through Rounding: A PTAS Result.

DPS-EPC is in trouble when M and N are large. Still, by considering that 2 states are equivalent when they are equal modulo the K largest bits and extending the notion of state in a well-fitted way, we turn ***DPS-EPC*** into an algorithm ***DPS-EPC***(K) which allows to state:

Theorem 2 (Polynomial Time Approximation Scheme): *For any value $\varepsilon > 0$, we may choose $K = K(\varepsilon)$ large enough in such a way that in case EPC admits an optimal solution with value W^{Opt}, then **DPS-EPC(K(ε))** yields in polynomial time a solution which is feasible with regards to initial values $(1 + \varepsilon/2).H_0$ and $(1 + \varepsilon/2).E_0$, threshold values $(1 + \varepsilon).C^{MP}$, $(1 + \varepsilon).C^{Veh}$ and $(1 + \varepsilon).TMax$ and whose cost value is no larger than W^{Opt}.*

B. Logical Filtering Devices.

First, we apply the standard *Dominance* Rule: If, for a given time pair (i, j), state s_1 dominates state s_2, $(W_1 \leq W_2\ ; T_1 \leq T_2\ ; Z_1 \geq Z_2; V^{Tank}{}_1 \geq V^{Tank}{}_2; V^{Veh}{}_1 \geq \mathrm{V}^{\mathrm{Veh}}{}_2)$, then we *kill* s_2. But this has little filtering power. So, for any time pair (i, j), and related state $s = (Z, T, V^{Tank}, V^{Veh})$, we get rough estimations *Fuel* and *Time* of respectively energy and time required in order to allow the vehicle to return from j to *Depot*, and derive the following *logical filtering rules*:

1).***Makespan Based*** **filtering rule**: If ($Time \geq TMax - T + 1$) then *kill state* $s = (Z, T, V^{Tank}, V^{Veh})$ related to *time pair* (i, j), since there is not enough time left for the vehicle to achieve its trip.

2) ***Energy Based*** **filtering rule**: If $Fuel > V^{Veh}\ \Sigma_{k \geq i} R_k + V^{Tank}$ then *kill state* $s = (Z, T, V^{Tank}, V^{Veh})$ related to *time pair* (i, j), since there won't be enough energy for the vehicle to achieve its trip.

We go further and pre-compute, for any energy amount V, any period number i, and any micro-plant Z value, the minimal cost $Cost\text{-}Min(i, V, Z)$ required from the micro-plant to produce V energy units from time $p.i$ on, Z denoting the state of the micro-plant at the end of period $i - 1$.Then, for any *time pair* (i, j) and any state $s = (Z, T, V^{Tank}, V^{Veh})$ with value W, we derive a lower bound LB of a best **EPC** trajectory involving (i, j) and s, by setting: $LB((i, j), s) = \alpha.Time + Cost\text{-}Min(i, (Fuel - V^{Tank})^+, Z) + W$. This lower bound allows us to turn ***DPS-EPC*** into a greedy procedure ***GREEDY-EPC***, by keeping, for any time pair (i, j), only the state $s(i,j)$ which minimizes $LB((i, j), s)$. ***GREEDY-EPC*** provides us with some feasible value *Current-Value* and we may apply the following ***Upper/Lower Bound Based*** **filtering rule** 3): If $LB((i, j), s) \geq Current\text{-}Value$, then *kill state* $s = (Z, T, V^{Tank}, V^{Veh})$, related to time pair (i, j).

IV. Pipe-line Decomposition of ***DPS-EPC.***

*A. The **DPS-Vehicle** Scheme.*

We do here as if micro-plant were able to provide, at any time, the vehicle with as much as energy it needs. we optimize the *Refueling Strategy* of the vehicle, that is the {0, 1} valued vector $x = (x_j, j = 0..M)$ and the load vector $L = (L_j, j = 0..M)$ which tell us at which stations j vehicle will refuel between j and $j + 1$, and how much, while minimizing some quantity: $\alpha.T^{End} + \beta.(\Sigma_j L_j.x_j)$, where T^{End} means the ending date of the vehicle trip, and β is an auxiliary *cost* coefficients. We notice that every time the vehicle arrives to the micro-plant, it is sufficient for him to refuel exactly the H^2 it needs in order to reach the next refueling transaction. This leads us to the following ***DPS-Vehicle*** scheme, whose components *time*, *state* and *decision* come as follows:

- **Time Space**: the set $J = \{0, 1, \ldots., M, M+1\}$.
- **State Space**: A state s is a 2-uple $s = (T, V^{Veh})$: T is the time necessary in order to come back to *Depot* and V^{Veh} the load at j of the vehicle tank. Its *value* $W = \alpha.T + \beta.U$, involves the energy amount U which will be wasted by the vehicle before the end of its trip. **Initial state** (in the sense of a ***backward driven*** DPS) is the state $(0, E_0)$ related to $j = (M+1)$. **Final states**, related to $j = 0$, should be any state ($\mathrm{T} \leq TMax$, $V^{Veh} \leq E_0$).
- **Decision Space**: A decision $x \in \{0, 1\}$: $x = 0$ means a no *refueling move* to j+1, and $x = 1$ means a *refueling move* to the micro-plant before reaching j+1.
- **Backward Driven Strategy**: In order to store, for any pair (j, V^{Veh}), the time and energy amount required in order to achieve tour, we implement Bellman Principle according to a backward driven strategy.

We denote by ***DPS-Vehicle*** the resulting DPS algorithm. In order to synchronize it with the H^2 Production control, we retrieve from any run a *Reduced Refueling Strategy*, that is:

- S = number of refueling transactions; Loads μ_s = quantities of H^2 which is loaded for every value $s = 1\ldots S$;
- Lower bounds $m_1,..m_Q$ and upper bounds $M_1,..,M_S$ for the related period numbers $i_1,.., i_S \in \{0,.., N\text{-}1\}$, as well as *Time Lag* coefficients $B_1, .., B_S$ which means: For any $s = 1..S\text{-}1$, $i_{s+1} \geq i_s + B_s$.

*B. The **DPS-Prod** Scheme*

Let S, m, M, μ be a *Reduced Refueling Strategy*, as above. Then we want to schedule the activity of the micro-plant, that is compute {0,1}-valued vectors z and δ with indexation on $i = 0..N\text{-}1$ as in ***DPS-ECP***, in such a way that:

- The vehicle may refuel at some periods $i_1,.., i_S$ in a way consistent with time lags and time window constraints induced by the *Reduced Refueling Strategy*
- The micro-plant ends with the H^2 load as it started;
- We minimize $\alpha.i_S + \Sigma_{i = 0..N\text{-}1} (Cost^F.y_i + Cost^V{}_{i}.z_i)$.

We apply a **forward driven** DPS algorithm ***DPS-Prod*** with the following *Time, State,* and *Decision* components:

- **Time Space**: the set $I = \{0..N\}$.
- **State Space**: For any $i = 0..N$, a state is a 4-uple $E = (Z, V^{Tank}, Rank, Gap)$, with *Rank* in $0..S$:
  - o $Z = 1$ ~ the micro-plant is active at the end of i-1.
  - o V^{Tank} is the load of the micro-plant when i starts.
  - o $Rank \in 1..S$ ~ the $Rank^{th}$ refueling transaction has been performed and we are waiting for the $(Rank + 1)^{th}$ refueling transaction. *Gap* means the difference between i and the period when the $Rank^{th}$ refueling transaction was performed.

  For every $i = 0..N$, a state E is provided with its current Bellman value W^{Prod}.
  - o **Initial state** is $E^{Start} = (0, H_0, 0, 0)$, with related value W^{Prod} =0, and time value i= 0;
  - o **Final states** are states $E^{End} = (Z, V^{Tank} \geq H_0, S, 0)$, associated with a time value $i \leq N$;
- **Decision/Transitions**: For any $i = 0..N$, $E = (Z, V^{Tank}, Rank, Gap)$, a decision is defined as a 2-uple (z, δ) in $\{0,1\}^2$, with the following meaning:
  - o $z = 1$ ~ the micro-plant will produce during period i;
  - o $\delta = 1$ ~ the vehicle will perform its $(Rank+1)^{th}$ refueling transaction during period i.

  Since production and refueling cannot be performed simultaneously, there are only 3 possible decisions:
  1). $\underline{z = 1, \delta = 0}$; 2). $\underline{z = 0, \delta = 0}$; 3). $\underline{z = 0, \delta = 1}$.

As for ***DPS-EPC***, we may enhance ***DPS-Prod*** through *logical* and *upper/lower bound based* filtering devices.

C. The Pipe-Line Scheme

Clearly, the simplest way to make above ***DPS-Vehicle*** and ***DPS-Prod*** interact, is to design the following heuristic ***Pipe-Vehicle->Production***:

Main Steps of *Pipe-Vehicle->Production*:
1). Fix β and Apply ***DPS-Vehicle***, to the *Vehicle* instance related to α, β: get related *Reduced Refueling Strategy*;
2). Apply ***DPS-Prod*** to resulting *Production* instance;
3). **Reconstruct the whole EPC solution**.

Choosing β: β should reflect the energy production cost. Since we do not know when the refueling transactions take place, we do as if were to be uniformly distributed.

V.NUMERICAL EXPERIMENTS.

Purpose and Technical Context: We evaluate: 1). the pipe-line decomposition ***DPS-Vehicle*** and ***DPS-Prod***; 2). the filtering devices and the greedy procedure described in III.2, while using C++, on Windows 10 with IntelCore i5-6500@3.20 GHz CPU, 16 Go RAM.

Instances: We fix N and M, and randomly generate stations j and *Depot* and the *Micro-Plant* as point of the R^2 space. Then d_j, d^*_j and t_j, e_j, ε_j, ε^*_j respectively corresponds to Euclidean and Manhattan distances. Then we fix C^{MP}, C^{Veh}, *TMAX*, $Cost^F \geq \text{Inf}_i\, Cost^V_i$, $i = 0,\ldots, N-1$.

Outputs: We first run ***Greedy-EPC*** with 50 replications, => gap *G-Gap* to optimality. Next we run ***DPS-EPC***:
1) Only with the *Strong Dominance Rule* => $ST(1)$
   = Maximal number of states for a given pair (i,j),
2) With the *2 Logical Filtering* rules => $ST(2)$;
3) With all filtering rules => $ST(3)$ (i,j).

Next we run ***Pipe-Vehicle->Production*** and get max state/time *ST-Veh*, *ST-Prod*, and gap *P-Gap* to optimality.

Instance (*M*, *N*)	*G-GAP*	*ST*(3)	*ST*(2)	*ST*(1)
1, (6, 27)	18.4	11870	11369	394754
2, (6, 26)	1.5	447	2619	299933
3, (10, 25)	0.0	10642	25636	107228
4, (10, 31)	11.4	17526	26254	310543
5, (10, 46)	0.0	21404	45014	425009

TABLE 1: VALUES *N, M, G-Gap, ST(1), ST(2), ST(3)*

Inst (*M, N*)	*ST-Veh*	*ST-Prod*	*P-Gap*
1, (6, 27)	22	895	6.4
2, (6, 26)	18	105	0
3, (10, 25)	43	902	0
4, (10, 31)	50	1088	2.9
5, (10, 46)	52	1385	0

TABLE 2: VALUES *N, M, ST-Veh, ST-Prod, P-Gap*

Comment: *Dominance* rule has little impact, logical anticipation and optimistic estimation rules are significantly more efficient, while the pipe-line scheme ***DPS-Vehicle -> DPS-Production*** offers a good tradeoff time/accuracy.

VI. CONCLUSION

In the future, we shall deal with uncertainties, address the *vehicle route* issue, manage the *on line* context.

REFERENCES

[1]. S.Albers: *Energy-efficient algorithms*; **Communications of ACM** 53, 4, p 86-96, (2010). https://dl.acm.org/doi/10.1145/1735223.1735245

[2]. ARTIGUES, E.HEBRARD, A.QUILLIOT, H.TOUSSAINT: *Models and algorithms for evacuation problems*; **IEEE Proc. FEDCIS WCO12**, Leipzig, 4 pages, (2019). https://doi.org/10.15439/2019F90

[3]. L.Benini, , A.Bogliolo, G.De Micheli: *A survey of design techniques for system level dynamic power management*; **IEEE Transactions of Very Large Scale Integratio Systems**, 8, 3, p 299-316, (2000). https://dl.acm.org/doi/10.1145/1403375.1403402

[4]. C.C.Chan. *The state of the art of fuel cell vehicles*. **Proc. of the IEEE**, 95, p 704-718, (2007). DOI: 10.1109/JPROC.2007.892489

[5]. P.Chretienne, A.Quilliot: *A polynomial algorithm for the non idling scheduling problem*; **DAM**, 20 pages, (2018). https://doi.org/10.1016/j.dam.2013.01.019

[6]. C.Grimes, O.Varghese, S.Ranjan. *Light, water, hydrogen: photoelectrolysis*. **Springer US**, (2008). ISBN 978-0-387-33198-0

[7]. S.Licht. *Thermochemical and Thermal/Photo Hybrid Solar Water Splitting*, **Springer New York, NY**, (2008). https://link.springer.com/chapter/10.1007/978-0-387-72810-0_5

[8]. C. Lin, K.L.Choy, G.T.Ho, S.H. Chung, H.Lam. *Survey of green vehicle routing problem*. **Expert Systems Applications**, 41, p 1118–1138, (2014). https://doi.org/10.1016/j.eswa.2013.07.107.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 307–315 DOI: 10.15439/2020F127 ISSN 2300-5963 ACSIS, Vol. 21

Stochastic multi-depot vehicle routing problem with pickup and delivery: an ILS approach

Brenner H. O. Rios, Eduardo C. Xavier, Flávio K. Miyazawa
Institute of Computing
University of Campinas
São Paulo, Brazil
Email: brenner@students.ic.unicamp.br, {ecx,fkm}@ic.unicamp.br

Pedro Amorim
INESC TEC
Faculty of Engineering
University of Porto
Porto, Portugal
Email: amorim.pedro@fe.up.pt

***Abstract*—We present a natural probabilistic variation of the multi-depot vehicle routing problem with pickup and delivery (MDVRPPD). In this paper, we present a variation of this deterministic problem, where each pair of pickup and delivery points are present with some probability, and their realization are only known after the routes are computed. We denote this stochastic version by S-MDVRPPD. One route for each depot must be computed satisfying precedence constraints, where each pickup point must appear before its delivery pair in the route. The objective is to find a solution with minimum expected traveling distance. We present a closed-form expression to compute the expected length of an *a priori* route under general probabilistic assumptions. To solve the S-MDVRPPD we propose an Iterated Local Search (ILS) that uses the Variable Neighborhood Descent (VND) as local search procedure. The proposed heuristic was compared with a Tabu Search (TS) algorithm based on a previous work. We evaluate the performance of these heuristics on a data set adapted from TSPLIB instances. The results show that the ILS proposed is efficient and effective to solve S-MDVRPPD.**

I. Introduction

VEHICLE routing problems (VRPs) have been extensively studied over the last three decades, mainly due to their economic importance and their theoretical challenges. The diversity of applications has motivated the study of several variants of VRPs. One of its more challenging variants is the multi-depot vehicle routing problem (MDVRP), where the well know TSP is a particular case of this problem. On the other hand, uncertainty is a characteristic of many real VRPs. Some common stochastic elements are customer requests, travel time and service time. The stochastic VRP (SVRP) is basically any VRP where one or more parameters are stochastic.

The VRP variant we consider is the stochastic version of MDVRP with pickup and delivery (MDVRPPD). The MDVRPPD is closely related to the problem proposed in [1]. In this work, the authors introduced the multi-depots pick-up and delivery problem with time windows and multi-vehicles. The principle of MDVRPPD is to design an optimal set of routes for a fleet of vehicles, each one located in a different depot. The set of routes allows serving a set of pickup and delivery points geographically dispersed. The number of vehicles is equal to the number of depots. Each vehicle must start and end the route in its assigned depot. Vehicles must visit once and only once each node. In Figure 1 we show the tour of three vehicles belonging each one to a different depot.

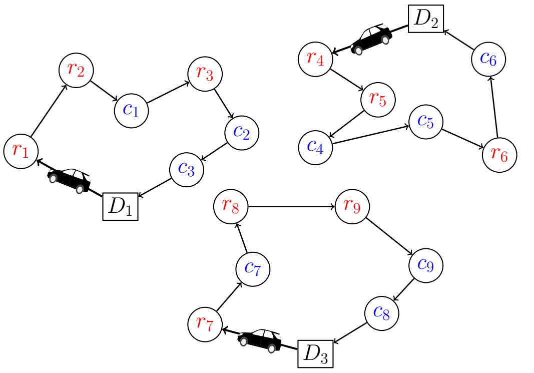


Fig. 1. Example of a solution for the MDVRPPD with three vehicles. Each vertex r_i represents a pickup point while c_i represents its corresponding delivery point.

In the stochastic version of the MDVRPPD the pickup and delivery points are uncertain. Consider for example an online marketplace provider, which is an e-commerce platform owned and operated by the provider, where third-parties can sell their products. It is common for the provider to be responsible for the gather and delivery of sold products (specially in the case of food delivery for example). If, based on past data, the provider has access to a probability distribution of the chance of a request from costumer A from seller B to occur, better routes can be constructed.

We propose an Iterated Local Search (ILS) heuristic to solve this problem that uses the Variable Neighborhood Descent (VND) heuristic as a local search. We denote the proposed algorithm by ILS-VND. The ILS-VND is based on the ILS heuristics presented by Subramanian et. al. [2] for the VRP with simultaneous pickup and delivery. To evaluate its performance we compare it with an adaptation of the TABUSTOCH algorithm proposed by Gendreau et. al. [3]. The tabu search algorithm is one of the main methods to deal with SVRPs [4].

The remainder of this paper is organized as follows. Section II-A presents the description and mathematical formulation of the MDVRPPD. Section III presents the closed-form expression to compute the expected length of an *a priori* route. Section IV introduces the proposed heuristic ILS-VND. Section V presents the adapted tabu search heuristic. Section VI describes the computational experiments, and Section VII presents the conclusions of this work.

A. Related work

The proposed problem has a close connection with the well known multi-depot traveling salesman problem (mTSP) [5]. Specifically, with the special case of multi-depot multiple travelling salesman problem (MmTSP). In the MmTSP each salesman starts from a unique city, travels to a set of cities and completes the route by returning to his original city with each city visited once [6]. Kara and Bektas [7] presents a mTSP review and explores connections with VRPs. The MDVRPPD is also closely related to the steiner multi cycle problem (SMCP), recently introduced in [8]. The SMCP arises in the scenario where a company has to periodically exchange goods between two different locations, and different companies can collaborate to create a route that visits all its pairs of locations sharing the total cost of the route [8]. The MDVRPPD can be seen as a version of the SMCP with depots. There are several heuristic approaches to solving VRPs and its stochastic variant. State of the art solutions include: particle swarm optimization approach [1], VNS [9], adaptive large neighbourhood search [10], ant colony optimization [11], genetic approach [12], tabu search [13], simulated annealing [14] and hybrid heuristic with exact methods [15] [16]. A review of the solution methods used in the past 20 years for the SVRP is presented in [17].

II. Problem description and model

In this section, we present the MDVRPPD and model it as an integer linear program first, then we define the S-MDVRPPD.

A. MDVRPPD

In this work, MDVRPPD is defined as follows. Let $G = (V, E)$ be a complete undirected graph, where $V = \{v_1, \ldots, v_n\}$ is the vertex set and $E = \{(v_i, v_j) : v_i, v_j \in V, i < j\}$ is the edge set. With each edge (v_i, v_j), it is associated a non-negative cost or distance d_{ij}. A subset of vertices $D = \{v_1, \ldots, v_m\}$ represents the depots, and the remaining vertices $V' = \{v_{m+1}, \ldots, v_n\}$ corresponds to pickup and delivery points. Let $w = |V'|/2$, then w vertices are pickup points and w vertices are delivery points. Each pickup point v_i is associated with a unique delivery point v_{i+w}, and vice versa, for $m + 1 \leq i \leq m + w$. There are m identical vehicles of unlimited capacity such that each one is located in a single depot. Each vehicle leaves its depot, serves a subset of pickup and delivery vertices and returns to its depot, forming a cycle (or route). The problem consists in determining a set of m vehicle cycles of minimal total cost considering the following constraints: a) each cycle starts and ends at the corresponding vehicles depot; b) each $v \in V'$ is visited exactly once by one vehicle c) each pair of pickup and delivery points, e.g $\{v_i, v_{i+w}\}$ for $m + 1 \leq i \leq m + w$, must belong to the same cycle and d) each cycle has an orientation where each pickup vertex in this cycle appears before its delivery pair.

The MDVRPPD is NP-hard since it includes the Traveling Salesman Problem (TSP) as a special case (e.g. if each pair of pickup and delivery are in the same location and there is only one depot).

We adapt the mathematical formulation proposed in [5] for the deterministic static version of MDVRPPD. In this formulation we assume G is a complete symmetric directed graph. The parameters and variables of the formulation are defined in Table I.

TABLE I
Parameters and decision variables for the MDVRPPD

D	Set of depots, $\{v_1, ..., v_m\}$.
V'	Set of nodes (pickup and delivery), $\{v_{m+1}, ..., v_n\}$
H^+	Set of pickup nodes, $\lvert H^+\rvert = w$.
u_i^k	A positive integer variable that indicates the order vertex i is visited by vehicle k, and $u_i^k = 0$ if i is not visited by k, $i \in V'$, $k \in D$.
d_{ij}	Distance between vertices i and j.
x_{ij}^k	If the vehicle from depot k travel along arc (i, j), then $x_{ij}^k = 1$, otherwise $x_{ij}^k = 0$.

$$\text{minimize} \sum_{k\in D} \sum_{j\in V'} (d_{kj}x_{kj}^k + d_{jk}x_{jk}^k) + \sum_{k\in D} \sum_{i\in V'} \sum_{j\in V'} d_{ij}x_{ij}^k \tag{1}$$

$$s.t. \quad \sum_{j\in V'} x_{kj}^k = 1, \quad k \in D, \tag{2}$$

$$\sum_{j\in V'} x_{jk}^k = 1, \quad k \in D, \tag{3}$$

$$\sum_{k\in D} x_{kj}^k + \sum_{k\in D} \sum_{i\in V'} x_{ij}^k = 1, \quad \forall j \in V', \tag{4}$$

$$x_{kj}^k + \sum_{i\in V'} x_{ij}^k = x_{jk}^k + \sum_{i\in V'} x_{ji}^k, \quad \forall k \in D, j \in V', \tag{5}$$

$$u_i^k \leq n(\sum_{j\in V'} x_{ij}^k + x_{ik}^k), \quad i \in V', \quad k \in D, \tag{6}$$

$$x_{ki}^k \leq u_i^k, \quad i \in V', \quad k \in D, \tag{7}$$

$$u_i^k + 1 \leq u_j^k + (1 - x_{ij}^k)n, \quad i, j \in V', \quad k \in D, \tag{8}$$

$$u_i^k + 1 \leq u_{i+w}^k + (1 - \sum_{j\in V'} x_{ij}^k)n, \quad i \in H^+, \quad k \in D, \tag{9}$$

$$x_{ij}^k \in \{0, 1\}, \quad i, j \in V, \tag{10}$$

$$u_i^k \in \mathcal{Z}^+, \quad i \in V, \quad k \in D, \tag{11}$$

In this formulation, constraint (2) ensures that exactly one vehicle depart from each depot $k \in D$, while (3) assures the vehicle returns to the depot. Constraint (4) ensures that each node is visited exactly once. Route continuity is ensured by the flow conservation constraints (5). Constraints (6) assures that the order of client i is 0 if it is not in the route of vehicle k. Constraints (7) impose that if i is the first vertex visited in route k, than its order in this route is at least 1. Constraint (8) is a subtour elimination constraint, since if j is visited after i in route k, then the visit order of j must be larger than the one of i in this route. Constraint (9), ensures that each pick-up node (i) must be visited before the corresponding delivery node ($i+w$). Finally we have the integrality constraints (10) and (11) of the variables in the model.

B. S-MDVRPPD

Now, we define the particular S-MDVRPPD considered in this work. This problem has one type of uncertainty: stochastic pickup and delivery points. Each pair $\{v_i, v_{i+w}\} \in V'$, for $m+1 \leq i \leq m+w$, has a probability p_i of being present when traveling along the route. When pickup point v_i is absent, delivery point v_{i+w} is also absent. We consider the S-MDVRPPD as a two stage stochastic problem. In the *first stage*, a set of cycles satisfying constraints (a) - (d) of the MDVRPPD are computed. The presence or absence of $\{v_i, v_{i+w}\}$ is revealed at the latest time upon leaving the preceding vertex of v_i. We suppose that the demand of every delivery point v_{i+w} is the same e.g. one unit. In the *second stage*, the first stage routes are followed as planned, with the following exception: any absent node is skipped. The S-MDVRPPD consists of designing a first stage solution that *minimizes the expected cost of the second stage solution*.

The S-MDVRPPD can be formulated as a stochastic integer program. We will use the parameters and variables defined in Table I. Let $T(x,\xi)$ be the cost of second stage solution if $x = (x_{ij}^k)$ is the first stage solution, and $\xi = (\xi_i)$ is the vector of non-negative random variables associated with the vertices of V'. The S-MDVRPPD is then formulated as

$$\min_x E_\xi[T(x,\xi)] \tag{12}$$

subject to equations (2)-(11).

III. The expected cost of an *a priori* route

Given a priori computed route $s = (v_0, v_1, \ldots, v_{2q}, v_0)$, where v_0 is a depot, let l_s be the cost/length of s. Our goal is to compute efficiently the expected length $E[l_s]$ of route s, given that during its execution, each pair $\{v_i, v_{i+w}\}$ of pickup and delivery points in this route have a probability of occurring during s's execution. We may also refer to node v_i as r_i, and v_{i+w} as c_i. Let $P(v_i)$ be the probability that node v_i appears in s. Note that we have the following relationship for a pair of pickup and delivery points r_i and c_i: $P(r_i) = P(c_i)$, $P(c_i|r_i\,\text{appears}) = P(r_i|c_i\,\text{appears}) = 1$, and $P(c_i|r_i\,\text{not appear}) = P(r_i|c_i\,\text{not appear}) = 0$.

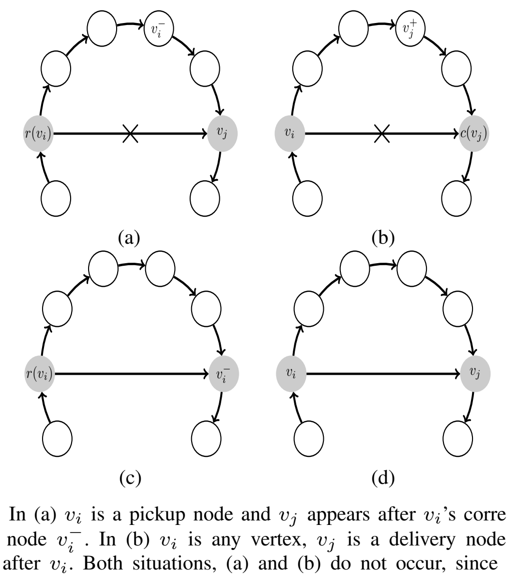
In this theorem we assume that R is the set of pickup points that appear in s ($|R| = q$). The pickup points are numbered in the superscript, in the order they appear in s, from r^1 until r^q. Likewise, C is the set of the corresponding delivery points and are also numbered in the superscript, from c^1 until c^q in the order they appear in s. We also use the following notation. If v_i is a pickup point we denote this by writing $r(v_i)$, and its corresponding delivery point as v_i^-, and if v_i is a delivery node, we denote this by writing $c(v_i)$ and denote its corresponding pickup point as v_i^+. Finally, given a subsequence $s_i^j = (v_i, v_{i+1}, \ldots, v_j)$ of s, let $R(s_i^j)$ denote the set containing the pickup points that appear in s_i^j, and also containing the pickup points of the delivery vertices that appear in s_i^j. Notice that $|R(s_i^j)| \leq j - i + 1$, and it is strictly small only when a pickup point appears in s_i^j and its delivery point also appears. Then we can compute $E[l_s]$ as follows.

Theorem 1. *Given a priori route $s = (v_0, v_1, \ldots, v_{2q}, v_0)$, then:*

$$\begin{aligned} E[l_s] = {} & \sum_{i=1}^{q} d_{v_0,r^i} P(r^i) \prod_{k=1}^{i-1} (1 - P(r^k)) \\ & + \sum_{i=1}^{q} d_{c^i,v_0} P(c^i) \prod_{k=i+1}^{q} (1 - P(r^k)) \\ & + \sum_{i=1}^{2q} \sum_{j=i+1}^{2q} f(v_i, v_j) \end{aligned} \tag{13}$$

where

$$f(v_i,v_j) = \begin{cases} 0 & \text{, (a) or (b)} \\ d_{v_i,v_j} P(v_i) \prod_{v \in R(s_{i+1}^{j-1})} (1 - P(v)) & \text{, (c)} \\ d_{v_i,v_j} P(v_i) P(v_j) \prod_{v \in R(s_{i+1}^{j-1})} (1 - P(v)) & \text{, (d)} \end{cases} \tag{14}$$



(a) (b) (c) (d)

Fig. 2. In (a) v_i is a pickup node and v_j appears after v_i's corresponding delivery node v_i^-. In (b) v_i is any vertex, v_j is a delivery node and v_j^+ appears after v_i. Both situations, (a) and (b) do not occur, since in (a), if $r(v_i)$ is present in the route then v_i^- must appear as well, and in (b) if $c(v_j)$ is present then v_j^+ must be present as well, so going from v_i directly to v_j skipping vertices in between is not a valid route. In (c) we have the case where v_i is a pickup point and v_j is its corresponding delivery point. In (d) we have all other cases that do not belong to one of the previous cases.

Proof: In equation (13) we are basically computing the probability of each edge between vertices in s to appear, in the execution of s, and multiplying this probability by the edge's cost. We have three terms in this equation.

In the first term, we are computing the expected cost of each possible initial edge of the route, such that if the route starts with (v_0, r^i) then all previous pickup points r^k, $k = 1, \ldots, i-1$ must not be present in the route. In the second term, we are computing the expected cost of each possible final edge in the route, similar to the first term.

In the last term we compute the expected cost of each edge from any pair of vertices in the route. Figure 2 represents all the possible cases between vertices v_i and v_j. Since cases (a) or (b) do not occur in practice the expected cost of an edge (v_i, v_j) in any one of these cases is zero. In case (c), the probability of going directly from $r(v_i)$ to $v_j = v_i^-$ is the probability of request of pickup point v_i to occur times the probability of requests of points in between v_i and v_j to not occur. Likewise, in case (d), if v_i and v_j are not related in any of the previous cases, then the probability of edge (v_i, v_j) to occur, is equal to the probability of v_i and v_j to occur times the probability of none of the requests of vertices in between them to occur. ■

We can compute the expected cost of an *a priori* route, $E(l_s)$, with time complexity $O(q^2)$, where $2q+2$ is the size of the route.

IV. ILS-VND

The proposed heuristic (ILS-VND) for the S-MDVRPPD works as follows. The method is executed $MaxIter$ times. In each iteration an initial solution is generated by a greedy algorithm, then this solution is improved using ILS. Internally, the ILS procedure uses the VND heuristic for performing the local search and a refinement heuristic for the initial solution called Random Mix-Shift. The ILS-VND is presented in Algorithm 1, where s^* corresponds to the best solution found during any iteration.

A. Initial solution generation

The method employed for building a feasible initial solution is based in the work of [18], so it is generated by following two steps. The first step is called *nodes assignment*. Each pair of pickup and delivery is assignment to one of the depots. After all vertices have been assigned to depots, the second step, called *nodes sequencing*, decides the service sequence of the pickup and delivery nodes. The details of these two steps are introduced in the following paragraphs.

1) Nodes assignment: This step assigns nodes to the depot which is closer to them. In the same way as in [18], in order to make the initial solution more flexible, the assignments of pickup an delivery nodes to depots are based on a probability. Suppose that $d(D_a, r_i, c_i)$ indicates the sum of the distances between pickup point r_i and depot D_a, and the distance between the delivery point c_i and depot D_a. Let $\overline{d(D, r_i, c_i)}$ the average distance between the pair of pickup and delivery $\{r_i, c_i\}$ and all depots. The probability of the set $\{r_i, c_i\}$ being assigned to depot D_a, is calculated by Eq. (15).

Algorithm 1: ILS-VND

```
for $k := 1, \ldots, MaxIter$ do
    $s :=$ GenerateInitialSolution(seed);
    $s' :=$ RandomMixShift($s$) ;
    $iterILS := 0$;
    while $iterILS < MaxIterILS$ do
        $r =$ number of neighborhoods ;
        $s := VND(N(.), r, s)$ ;
        if $f(s) < f(s')$ then
            $s' := s$ ;
            $s := Perturb(s')$ ;
            $iterILS := 0$ ;
        else
            $iterILS := iterILS + 1$ ;
    if $f(s') < f(s^*)$ then
        $s^* := s'$ ;
```

$$P(D_a, r_i, c_i) = \frac{\max\left\{\overline{d(D, r_i, c_i)} - d(D_a, r_i, c_i), 0\right\}}{\sum_{a=1}^{|D|} \max\left\{\overline{d(D, r_i, c_i)} - d(D_a, r_i, c_i), 0\right\}} \tag{15}$$

2) Nodes sequencing: Once we have assigned all pickup and delivery nodes to depots, we sequence the nodes to create cycles. Let D_i be an arbitrary depot, v_i a pickup or delivery point associated with D_i and t_i a tour containing D_i. The procedure begins by looking for a node v_i nearest to the last node of t_i, which initially contains only D_i, such that all constraints of the S-MDVRPPD are satisfied when v_i is appended to t_i. Then v_i is appended to t_i and the procedure is repeated until all the pickup and delivery nodes are inserted into t_i. The Figure 3 shows an example of sequencing one depot and four pairs of pickup and delivery points. The complexity time of this procedure is $O(n^2)$.

B. Local Search

The local search is based on the VND heuristic introduced by Mladenović and Hansen [19]. In the variable neighborhood descent method a change of neighborhoods is performed in a deterministic way. The proposed VND is presented in Algorithm 2.

A set $\{N^1, \ldots, N^6\}$ of six neighborhood operators were used by the proposed VND. All operators are exhaustively executed. These operators are adapted in such a way that they preserve feasibility. We divide these operators into three groups: inter-tour, intra-tour and inter&intra-tour operators. The inter-tour operators are: Shift(1,0) and Swap(1,1). The intra-tour neighborhood operators are: 2-opt, 3-opt and Reverse. Finally, Mix-shift(1,0) is the only operator that is inter&intra-tour operator. In the case of inter-route operators, to reduce the computational cost, each vertex removed of a

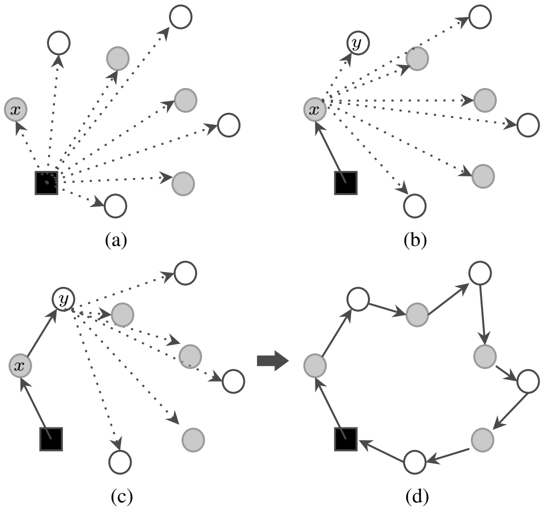


Fig. 3. Example of nodes sequencing in a route. Gray nodes are pickups and white nodes deliveries. In (a) distances from the depot to all vertices are calculated. In (b) node x is added to the route, since it is the closest to the depot and its addition does not break the constraints of the problem. Then, the distances from x to all the nodes that are not part of the route are computed. The node y is the closest to x, and its addition does not break the constraints of the problem. In (c) node y is added to the route. The process repeats until all nodes are added to the route. The obtained route is shown in (d).

route can only be inserted before or after one of its p closest neighbors in the other routes.

The list of neighborhoods considered are:

- **Shift(1,0)** – N^1 – A pickup and delivery pair r, c is removed from a route t_1 and each one is moved to the best position in route t_2 keeping the feasibility of the solution.
- **Swap(1,1)** – N^2 – An exchange between a pair r_1, v_1 from a route t_1 and another pair r_2, v_2 from route t_2. Each vertex of the pairs are inserted in the best possible position while maintaining the feasibility of the solution.
- **Mix-Shift(1,0)** – N^3 – This operator is similar to the Shift(1,0) operator with the difference that now it is allowed a movement within its own route.
- **2-opt** – N^4 – Two nonadjacent arcs are removed and other two are added to form a new route. We only consider movements that do not break the constraints of the problem.
- **3-opt** – N^5 – Three nonadjacent arcs are removed and other two are added to form a new route. We only consider movements that do not break the constraints of the problem.
- **Reverse** – N^6 – This operator reverses the direction of the route. Then swaps are performed between each pair of pickup and delivery.

In case of improvement of the current solution, the algorithm performs an intensification process on each route. The objective is to decrease the cost of each route. Therefore, the neighborhoods N^4, N^5 and N^6 are applied in this order in the current solution.

Given the versatility of the *Mix-Shift(1,0)* operator, and based on the neighborhood structure proposed by Gendreau et. al. [3], we present the Random Mix-Shift heuristic (Algorithm 3). In this heuristic a randomly selected pickup and delivery pair $\{r, c\}$ is removed, r is inserted immediately before or after one of its p closest neighbors. The vertex c is randomly inserted into the same route, without breaking the constraints of the problem. To avoid necessary iterations of Random Mix-Shift heuristic, we chose to use s' if it is *promising*. A solution s' is *promising* if it has the potential to become the new best solution, specifically, if its cost is at most $\alpha\%$ higher than the cost of the best solution so far, s. We use this heuristic in the ILS-VND as a refinement mechanism to improve the initial solution.

C. Perturbation Mechanism

A set P of two perturbation mechanisms were adopted in the ILS-VND heuristic. Every time the *perturb()* function is called one of the following operators is randomly selected and applied.

Algorithm 2: VND

```
Let $r$ be the number of neighborhoods
 structures and $s$ a current solution ;
$k := 1$; current neighborhood ;
while $k \leq r$ do
    Find the best neighbor $s'$ of $s \in N^k$ ;
    if $f(s') < f(s)$ then
        $s := s'$ ;
        $k := 1$ ;
        intensification in the modified
         routes ;
        $s' := 2 - opt(s)$ ;
        $s'' := 3 - opt(s')$ ;
        $s''' := Reverse(s''')$ ;
        if $f(s''' \leq f(s)$ then
            $s := s'''$
    else $k := k + 1$ ;
```

Algorithm 3: Random Mix-Shift

```
for $k := 1 \ldots, MaxIterShift$ do
    $r, c :=$ SelectRandomPair($r, c$) ;
    $s' :=$ Mix-Shift($s, r, c$) ;
    if $f(s') < f(s^*)$ then
        $s := s'$; $s^* := s'$; $k := 1$ ;
    else
        if $f(s') < \alpha f(s^*)$ then
            $s := s'$ ;
        else
            $s := s^*$;
```

Double-Swap – P^1 – Two Swap(1,1) operators are performed in sequence.

Depot Exchange – P^2 – The depot exchange operator select two depots at random, and exchange their routes.

V. Tabu Search

We present an adaptation of the TABUSTOCH heuristic proposed by Gendreau et. at. [3] that was originally designed to the Vehicle Routing Problem (VRP) with stochastic demands. The algorithm solves a two stage stochastic VRP, where in the first stage a feasible solution is constructed including all vertices (clients). In the second stage recourse actions maybe taken, since the real demands of costumers are realized, capacity constraints may become violated. In traversing a route, once a vehicle becomes full it returns to the depot and resumes the route in the next client to be visited. All the parameters in the adapted algorithm, are the same used in the original TABUSTOCH. We will only present the modifications made to TABUSTOCH in order to deal with the S-MDVRPPD. Let x^k be a solution in the first stage in iteration k of the algorithm. Let $T(x^k)$ be the expected value in the second stage. Let $T(x^k) = \sum_{i=1}^{m^k} T^i(x^k)$, where m^k is the number of routes at iteration k.

The initial solution is built by assigning to each depot the closest *candidate* pair of pickup and delivery. A pair of pickup and delivery is a *candidate* if it has not been assigned to some depot. The selected pickup and delivery pair is appended to the solution (first pickup and then delivery). The initial solution is always feasible. The neighbourhood structure used by TABUSTOCH is the Mix-shift(1,0) operator presented in section IV-B. Thus, there is the possibility of inserting pairs of pickup and delivery in different cycles.

We consider the movement of nodes in a solution as elements of the tabu list. There are two ways to move a vertex: 1) change the position of the vertex in the same tour and 2) move the vertex (and its corresponding pair) to another tour. Either of these two movements is tabu for θ iterations, where θ is randomly selected from the interval $[|V|-5, |V|]$. The search of solutions in each iteration considers the current solution x^k and the best non-tabu solution x^{k+1} in the neighborhood structure Mix-Shift(1,0). However, a tabu solution can be selected if it improves the best solution T^* (aspiration criteria).

Note that computing the expected value of a solution is expensive. Moving a pickup and delivery pair not only affects the cost related to their immediate neighbors, but also affects the cost of each node in the tour. Notice also that the movement affects the costs of both previous and new tours where they where inserted.

Suppose we wish to insert a pair of pickup and delivery $\{x^+, x^-\}$ into a route. Figures (4a) and (4b) represents the possible positions of $\{x^+, x^-\}$ in a route before removing them while Figures (4c) and (4d) represents all possible positions of x^+ and x^- after their insertion into the new route. Dotted arrows represent arcs before insertion. Red lines represent subtours and black arrows arcs. We denote the approximations of the effect of inserting a pickup and delivery in a route with A_i and $\bar{A}_i$. A_i refers to the insertion of $\{x^+, x^-\}$ as shown in the Figure (4c). $\bar{A}_i$ refers to the insertion of $\{x^+, x^-\}$ as shown in Figure (4d). Note that the approximations of the effect of removing a pickup and delivery in a route can be represented with $-A_i$ and $-\bar{A}_i$.

We use three easy-computational approximations of insertion cost to speed up the search process. The first approximation, given by equations (16) and (17), completely detach the stochastic nature of the problem. The second approximation, given by equations (18) and (19), partially remediate the first approximation, but these equations give all the weight for e, f, g, h and x^-. Taking into account P_e, P_f, P_g, P_h and P_{x^-}, the third approximation, given by equations (20) and (21), seeks to remedy the second approximation. The problem with this last approximation happens when P_e, P_f, P_g, P_h are small and P_{x^+} (and so P_{x^-}) is large.

Tests conducted on 600 randomly generated instances involving between 10 and 100 vertices indicates that the third approximation yields the best correlation with the true cost increase ($r = 0.89$).

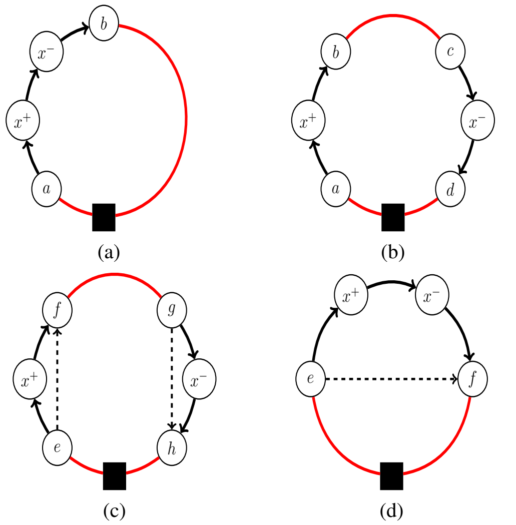
$$A_1(e,f,g,h,x^+,x^-) = d_{ex^+} + d_{x^+f} + d_{gx^-} + d_{x^-h} - d_{ef} - d_{gh} \tag{16}$$

$$\bar{A}_1(e,f,x^+,x^-) = d_{ex^+} + d_{x^+x^-} + d_{x^-f} - d_{ef} \tag{17}$$

$$A_2(e,f,g,h,x^+,x-) = (d_{ex^+} + d_{x^+f} + d_{gx^-} + d_{x^-h} - d_{ef} - d_{gh})P_{x^+} \tag{18}$$

$$\bar{A}_2(e,f,x^+,x^-) = (d_{ex^+} + d_{x^+x^-} + d_{x^-f} - d_{ef})P_{x^+} \tag{19}$$

$$\begin{aligned} A_3(e,f,g,h,x^+,x^-) = {} & d_{ex^+}P_eP_f + d_{x^+f}P_{x^+}P_f \\ & + d_{gx^-}P_gP_{x^-} + d_{x^-h}P_{x^-}P_h \\ & - d_{ef}P_eP_f - d_{gh}P_gP_h \end{aligned} \tag{20}$$



(a) (b)

(c) (d)

Fig. 4. Example of movements of nodes x^+ and x^-. Cases (a) and (b) represent different situations of nodes x^+ and x^- in a route, before the movement. Cases (c) and (d) represent possible insertion of x^+ and x^- in a route, after the movement.

$$\bar{A}_3(e,f,x^+,x^-) = d_{ex^+}P_eP_{x^+} + d_{x^+x^-}P_{x^+}P_{x^-} + d_{x^-f}P_{x^-}P_f - d_{ef}P_eP_f \tag{21}$$

We have the necessary terms to *approximate the cost of a movement*. The expressions (22)-(25) are used to evaluate the movement cost of x^+ and x^-. We will use the cases shown in Figure 3. If the movement of x^+ and x^- happens in the order: from case (4a) to case (4c) we use the equation (22); from case (4a) to case (4d) we use equation (23); from case (4b) to case (4c) we use (24); and, from case (4b) to case (4d) we use (25).

$$\Delta_1 = A_3(e,f,g,h,x^+,x^-) - \bar{A}_3(a,b,x^+,x^-) \tag{22}$$

$$\bar{\Delta}_1 = \bar{A}_3(e,f,x^+,x^-) - \bar{A}_3(a,b,x^+,x^-) \tag{23}$$

$$\Delta_2 = A_3(e,f,g,h,x^+,x^-) - A_3(a,b,c,d,x^+,x^-) \tag{24}$$

$$\bar{\Delta}_2 = \bar{A}_3(e,f,x^+,x^-) - A_3(a,b,c,d,x^+,x^-) \tag{25}$$

VI. Computational Experiments

We conducted experiments using a data set derived from six TSPLIB instances (*ulysses16, bayg29, dantzig42, eil51, st70* and *st76*). For each of these instances, n vertices in the interval of [2, 10] were randomly selected to be depots. A random matching was performed among the other vertices to create pickup and delivery pairs. The probability of presence of each pickup and delivery pair was chosen uniformly in the interval $[0,1]$. We generate 30 test instances.

The algorithms described above were coded in C++ and all experiments were run on a Linux operating system with 3 GB memory and Intel Core i5 2.54x4 Ghz processor. Computational times reported here are in CPU seconds on this machine. To evaluate our ILS-VND heuristic we compare it with an adaptation of TABUSTOCH algorithm. Ten independent runs of the algorithms were performed for each test case. The number of iterations (*MaxIter*) and perturbation allowed (*MaxIterILS*), was 10 and 15 respectively. The parameters α, $MaxIterShift$ and p were fixed to 1.05, 100 and 5, respectively. They were calibrated empirically after preliminary tests with different values.

Table II shows the results obtained by the ILS-VNS and the TABUSTOCH heuristics. The best solutions are in boldface. The columns related with the instances show: the instance name, *Name*, the number of vertices in the graph, $|V|$, and the number of depots $|D|$. Columns *Avg.* and *Best* show the average and the best solution costs found by the algorithms in their ten independent executions, respectively. Column *Time* presents the average processing time, in seconds, spend by each algorithm. The ILS-VND presented the best results for all instances.

Table III shows the percentage improvement of the best and average solutions obtained by the ILS-VND, against the TABUSTOCH. Negative values mean that ILS-VND was better than the TABUSTOCH. The formula $\frac{\text{ILS-VND}-\text{TABUSTOCH}}{\text{ILS-VND}}$ was used to generate the values presented in Table III. The results show that the ILS-VND was superior to the TABUSTOCH for all instances tested.

TABLE II
Results of the ILS-VND and TABUSTOCH to solve 30 instances of the S-MDVRPPD. Both heuristics were executed 10 times for each instance, and Avg. Cost and Time are the average solution cost and average time over these 10 executions.

Instance			ILS-VND			TS		
Name	$\|V\|$	$\|D\|$	**Best Cost**	**Avg. Cost**	**Time (s)**	**Best Cost**	**Avg. Cost**	**Time (s)**
ulysses16a	16	2	**66,44**	67,41	0,91	69,96	71,14	0,46
ulysses16b	16	2	**30,68**	30,69	0,72	31,94	31,94	0,41
ulysses16c	16	2	**44,92**	44,92	0,70	45,26	45,93	0,66
ulysses16d	16	4	**57,59**	57,59	0,26	65,05	65,55	0,53
ulysses16e	16	4	**53,21**	53,21	0,27	81,09	81,09	0.60
bayg29a	29	3	**7082,80**	7247,16	6,30	8010,40	9348,96	11,37
bayg29b	29	9	**11890,90**	11990,20	0,46	15745,10	15745,10	8,50
bayg29c	29	9	**10631,60**	10660,90	0,61	13856,80	13856,80	7,43
bayg29d	29	3	**7121,43**	7219,10	8,34	7705,80	8449,30	10,00
bayg29e	29	5	**8078,92**	8116,32	4,37	10800,10	11700,60	11,27
dantzig42a	42	8	**676,93**	696,76	5,11	985,70	1009,46	28,37
dantzig42b	42	4	**542,06**	557,25	28,43	676,76	723,88	69,13
dantzig42c	42	10	**714,80**	730,77	2,00	1130,56	1130,56	35,98
dantzig42d	42	2	**599,89**	628,23	55,22	617,35	696,75	41,13
dantzig42e	42	10	**671,76**	680,96	3,41	1244,39	1334,65	45,51
eil51a	51	3	**351,98**	369,15	90,39	418,03	447,26	122,13
eil51b	51	7	**368,78**	384,45	24,70	537,55	586,47	66,46
eil51c	51	9	**340,74**	354,02	11,94	579,64	596,45	45,63
eil51d	51	7	**323,60**	329,68	39,62	493,12	535,91	107,50
eil51e	51	5	**339,41**	344,22	22,97	409,80	484,96	93,16
st70a	70	4	**621,25**	690,76	224,80	810,95	875,85	408,45
st70b	70	6	**728,87**	760,81	88,89	871,67	945,15	765,90
st70c	70	8	**680,43**	710,06	115,43	1078,96	1249,14	459,02
st70d	70	8	**620,63**	669,17	87,50	1012,48	1071,10	398,99
st70e	70	6	**575,27**	616,96	200,63	960,09	1048,68	483,40
eil76a	76	6	**556,89**	599,37	171,47	703,79	790,67	480,91
eil76b	76	2	**464,73**	486,68	606,41	491,96	530,28	352,76
eil76c	76	4	**513,01**	564,59	309,40	559,31	636,99	381,38
eil76d	76	6	**522,01**	557,11	236,04	694,46	806,43	582,45
eil76e	76	6	**490,29**	534,55	176,54	642,77	723,63	404,50

TABLE III
PERCENTAGE DECREASE IN SOLUTIONS COST OBTAINED BY ILS-VND COMPARED TO TABUSTOCH.

	ILS-VND	
Instance	**Best (%)**	**Avg. (%)**
ulysses16a	-5,30	-5,25
ulysses16b	-4,12	-3,92
ulysses16c	-0,76	-2,20
ulysses16d	-12,96	-12,14
ulysses16e	-52,39	-34,38
bayg29a	-13,10	-22,48
bayg29b	-32,41	-23,85
bayg29c	-30,34	-23,06
bayg29d	-8,21	-14,56
bayg29e	-33,68	-30,63
dantzig42a	-45,61	-30,98
dantzig42b	-24,85	-23,02
dantzig42c	-58,17	-35,36
dantzig42d	-2,91	-9,83
dantzig42e	-85,24	-48,98
eil51a	-18,76	-21,16
eil51b	-45,77	-52,55
eil51c	-70,11	-68,48
eil51d	-52,39	-62,56
eil51e	-20,74	-40,88
st70a	-30,53	-26,79
st70b	-19,59	-24,23
st70c	-58,57	-75,92
st70d	-63,14	-60,06
st70e	-66,89	-69,98
eil76a	-26,38	-31,92
eil76b	-5,86	-8,96
eil76c	-9,02	-12,82
eil76d	-33,04	-44,75
eil76e	-31,10	-35,37

VII. CONCLUSION

This article described a new and practical SVRP involving multiple depots, and pickup and delivery (S-MDVRPPD). Contrary to the deterministic case, it is not easy to compute the objective function associated with a solution [20]. We presented a closed-form expression to compute the expected length of an *a priori* sequence under general probabilistic assumptions. In order to dealt with S-MDVRPPD, an algorithm based on the Iterated Local Search metaheuristic, which uses a VND heuristic as local search procedure was proposed. We use six local search operators, *Shift(1,0), Swap(1,1), 2-opt, 3-opt, Reverse, and Mix-Shift*. Also, we use two perturbation mechanisms, *Double-Swap and Depot-exchange*. We propose a heuristic based on *Mix-shift* operator to refine the initial solution of the ILS-VND. The ILS-VND was compare with a tabu search algorithm (TABUSTOCH). We report the results for 30 instances. The results show that the ILS-VND was superior to the TABUSTOCH for all instances tested. Our approach can be used as benchmark for future research in this area. The S-MDVRPPD can be further generalized to handle more practical constraints, e.g., limited capacity vehicles, time windows and stochastic demands.

ACKNOWLEDGMENT

This work is financed by FAPESP (Proc. 2015/11937-9, 2016/01860-1, 2018/08879-5), CNPq (Proc. 314366/2018-0, 425340/2016-3, 304856/2017-7), ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia within project POCI-01-0145-FEDER-031908.

REFERENCES

[1] I. H. Dridi, E. B. Alaïa, P. Borne, and H. Bouchriha, "Optimisation of the multi-depots pick-up and delivery problems with time windows and multi-vehicles using PSO algorithm," *International Journal of Production Research*, pp. 1–14, sep 2019. doi: 10.1080/00207543.2019.1650975

[2] A. Subramanian, L. Drummond, C. Bentes, L. Ochi, and R. Farias, "A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery," *Computers & Operations Research*, vol. 37, no. 11, pp. 1899–1911, nov 2010. doi: 10.1016/j.cor.2009.10.011

[3] M. Gendreau, G. Laporte, and R. Séguin, "A tabu search heuristic for the vehicle routing problem with stochastic demands and customers," *Operations Research*, vol. 44, no. 3, pp. 469–477, jun 1996. doi: 10.1287/opre.44.3.469

[4] H. N. Psaraftis, M. Wen, and C. A. Kontovas, "Dynamic vehicle routing problems: Three decades and counting," *Networks*, vol. 67, no. 1, pp. 3–31, aug 2015. doi: 10.1002/net.21628

[5] I. Kara and T. Bektas, "Integer linear programming formulations of multiple salesman problems and its variations," *European Journal of Operational Research*, vol. 174, no. 3, pp. 1449–1458, nov 2006. doi: 10.1016/j.ejor.2005.03.008

[6] M. Assaf and M. Ndiaye, "Multi travelling salesman problem formulation," in *2017 4th International Conference on Industrial Engineering and Applications (ICIEA)*. IEEE, apr 2017. doi: 10.1109/iea.2017.7939224

[7] T. Bektas, "The multiple traveling salesman problem: an overview of formulations and solution procedures," *Omega*, vol. 34, no. 3, pp. 209–219, jun 2006. doi: 10.1016/j.omega.2004.10.004

[8] V. N. Pereira, M. C. San Felice, P. H. D. Hokama, and E. C. Xavier, "The steiner multi cycle problem with applications to a collaborative truckload problem," in *17th International Symposium on Experimental Algorithms (SEA 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018. doi: 10.4230/LIPICS.SEA.2018.26

[9] M. Polacek, R. F. Hartl, K. Doerner, and M. Reimann, "A variable neighborhood search for the multi depot vehicle routing problem with time windows," *Journal of Heuristics*, vol. 10, no. 6, pp. 613–627, dec 2004. doi: 10.1007/s10732-005-5432-5

[10] G. Laporte, R. Musmanno, and F. Vocaturo, "An adaptive large neighbourhood search heuristic for the capacitated arc-routing problem with stochastic demands," *Transportation Science*, vol. 44, no. 1, pp. 125–135, feb 2010. doi: 10.1287/trsc.1090.0290

[11] P. Stodola, "Hybrid ant colony optimization algorithm applied to the multi-depot vehicle routing problem," *Natural Computing*, vol. 19, no. 2, pp. 463–475, jan 2020. doi: 10.1007/s11047-020-09783-6

[12] J. E. Mendoza and J. G. Villegas, "A multi-space sampling heuristic for the vehicle routing problem with stochastic demands," *Optimization Letters*, vol. 7, no. 7, pp. 1503–1516, sep 2012. doi: 10.1007/s11590-012-0555-8

[13] A. L. Erera, M. Savelsbergh, and E. Uyar, "Fixed routes with backup vehicles for stochastic vehicle routing problems with time constraints," *Networks*, vol. 54, no. 4, pp. 270–283, dec 2009. doi: 10.1002/net.20338

[14] J. C. Goodson, "A priori policy evaluation and cyclic-order-based simulated annealing for the multi-compartment vehicle routing problem with stochastic demands," *European Journal of Operational Research*, vol. 241, no. 2, pp. 361–369, mar 2015. doi: 10.1016/j.ejor.2014.09.031

[15] B. H. O. Rios, E. F. G. Goldbarg, and G. Y. O. Quesquen, "A hybrid metaheuristic using a corrected formulation for the traveling

car renter salesman problem," in *2017 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, jun 2017. doi: 10.1109/cec.2017.7969584

[16] B. H. O. Rios, E. F. G. Goldbarg, and M. C. Goldbarg, "A hybrid metaheuristic for the traveling car renter salesman problem," in *2017 Brazilian Conference on Intelligent Systems (BRACIS)*. IEEE, oct 2017. doi: 10.1109/bracis.2017.20

[17] J. Oyola, H. Arntzen, and D. L. Woodruff, "The stochastic vehicle routing problem, a literature review, part II: solution methods," *EURO Journal on Transportation and Logistics*, vol. 6, no. 4, pp. 349–388, nov 2016. doi: 10.1007/s13676-016-0099-7

[18] Y. Kuo and C.-C. Wang, "A variable neighborhood search for the multi-depot vehicle routing problem with loading cost," *Expert Systems with Applications*, vol. 39, no. 8, pp. 6949–6954, jun 2012. doi: 10.1016/j.eswa.2012.01.024

[19] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers & Operations Research*, vol. 24, no. 11, pp. 1097–1100, nov 1997. doi: 10.1016/s0305-0548(97)00031-2

[20] D. J. Bertsimas, "A vehicle routing problem with stochastic demand," *Operations Research*, vol. 40, no. 3, pp. 574–585, jun 1992. doi: 10.1287/opre.40.3.574

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 317–326 DOI: 10.15439/2020F19 ISSN 2300-5963 ACSIS, Vol. 21

An Experimental Study on Symmetry Breaking Constraints Impact for the One Dimensional Bin-Packing Problem

Khadija Hadj Salem
Université de Tours, LIFAT EA 6300, CNRS, ROOT ERL CNRS 7002,
64 avenue Jean Portalis, 37200 Tours
Email: khadija.hadj-salem@univ-tours.fr

Yann Kieffer
Univ. Grenoble Alpes, Grenoble INP, LCIS,
26000 Valence, France
Email: yann.kieffer@lcis.grenoble-inp.fr

***Abstract*—We consider the classical *One-Dimensional Bin Packing Problem* (1D-BPP), an $\mathcal{NP}$-hard optimization problem, where, a set of weighted items has to be packed into one or more identical capacitated bins. We give an experimental study on using symmetry breaking constraints for strengthening the classical integer linear programming proposed to optimally solve this problem. Our computational experiments are conducted on the data-sets found in BPPLib and the results have confirmed the theoretical results.**

I. Introduction

THE *one-dimensional Bin Packing Problem*, noted 1D-BPP from here on, has been widely studied in the literature both for its theoretical interest and its many practical applications. Several variants were considered as well as different approaches for its solution were proposed. The 1D-BPP can be informally defined as follows: n items have to be packed each into one of n available bins. Each item i has a non-negative weight w_i ($i = 1, \dots, n$) and all bins have the same positive integer capacity C. The objective is to find a packing with a minimum number of bins such that the total weights of the items in each bin does not exceed the capacity C.

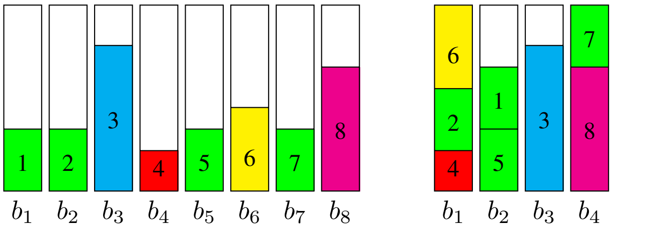
To illustrate these concepts, we consider the following example: give one instance of the 1D-BPP with a set of bins with capacity C equal to 6 and a set of items, indexed by i, with the weights w_i given in Table I.

TABLE I: An example of data, with 8 items

Items i	1	2	3	4	5	6	7	8
Weights w_i	2	2	5	1	2	3	2	4

An example of a feasible solution as well as an optimal solution, respectively, with 8 bins and 4 bins, are given in Figure 1.

A central theme for this study is the computational effect of the removal of symmetric solutions. To the best of our knowledge, no numerical studies have been published to ascertain the performance gain of symmetry breaking constraints for 1D-BPP. So we conducted such a study, including a new inequalities to strengthen its basic mathematical model.



(a) A feasible solution: 8 bins **(b)** An optimal solution: 4 bins

Fig. 1: Solutions for the 1D-BPP

The remainder of this paper is organized as follows. Section II formally introduces the basic mathematical model and briefly mentions the exact solution methods considered in the literature of 1D-BPP. Section III gives a brief review on symmetries in ILP formulations. Sections IV — VII describe some classes of symmetry breaking constraints. Computational results are reported and analyzed in Section VIII. Finally, the main conclusions of this work as well as some future research directions are drawn.

II. Basic mathematical models for the 1D-BPP

A. Assignment-based models

The compact ILP formulation for 1D-BPP, which Martello and Toth attribute to Kantorovich (see [19]), is the following, by introducing two types of binary decision variables for all $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, n\}$.

- $y_j \begin{cases} 1 & \text{if bin } j \text{ is used in the packing} \\ 0 & \text{otherwise} \end{cases}$
- $x_{ij} \begin{cases} 1 & \text{if item } i \text{ is packed into bin } j \\ 0 & \text{otherwise} \end{cases}$

The full model, hereafter denoted as `ILP-0`, is:

$$\mathtt{ILP-0}: \quad \min \sum_{j=1}^{n} y_j$$

$$s.t. \quad \begin{cases} \displaystyle\sum_{j=1}^{n} x_{ij} \quad = 1 \quad \forall i \in \{1,\ldots,n\} & (1)\\ \displaystyle\sum_{i=1}^{n} w_i * x_{ij} \le C * y_j \forall j \in \{1,\ldots,n\} & (2)\\ \qquad x_{ij} \in \{0,1\} \forall i \in \{1,\ldots,n\}, j \in \{1,\ldots,n\} & (3)\\ \qquad y_j \in \{0,1\} \forall j \in \{1,\ldots,n\} & (4) \end{cases}$$

In this model, constraints (1) ensure that each item is packed into exactly one bin, constraints (2) impose that the capacity of any used bin is not exceeded and both constraints (3) and (4) define the variable domains.

An obvious lower bound for the 1D-BPP, computable in $\mathcal{O}(n)$ time, is the optimal value of the *continuous relaxation* of `ILP-0`. Denoted by L_1 in the literature, this lower bound is given by the following equality:

$$\mathrm{L}_1 = \left\lceil \sum_{j=1}^{n} w_i/C \right\rceil \qquad (5)$$

It is easily seen that the *worst-case performance ratio* of L_1 is equal to $\frac{1}{2}$ (see, e.g., [20]).

The reader is referred to [11], which is a first survey on linear programming models for the 1D-BPP and its generalization, the Cutting Stock Problem (CSP).

B. Other methods for optimally solving the 1D-BPP

Among other methods for solving 1D-BPP exactly, we can find the branching algorithms and the pseudo-polynomial ILP formulations coming from a graph representation of the solution space.

The reader is referred to [15] for a recent survey on mathematical models and exact algorithms for both 1D-BPP and CSP and to [15] for a library, named BPPLIB and available at http://or.dei.unibo.it/library/bpplib. The BPPLIB provides a collection of computer codes of different types for the exact solution of the 1D-BPP and the CSP as well as a benchmark instance. It also includes a BibTeX file of more than 150 references on this topic and an interactive visual tool to manually solve both 1D-BPP and CSP.

An overview of these methods can be summarized as follows:

1) *Enumeration algorithms*, basically:
   - the branch-and-bound, in which three approaches are proposed: **MTP** (see [19]), **BISON** (see [1]) and **CVRPSEP** (see [9]).
   - the branch-and-price, in which one approach, called **SCIP-BP** (see [3]), is proposed.
2) *Pseudo-polynomial formulations solved through an ILP solver (like CPLEX, SCIP, GUROBI)*: we can find here both **ONECUT** (see [7]) and **DPFLOW** (see [6]).

According to the results discussed in [14], the **SCIP-BP** is effective on small-size instances ($n \leq 100$). In the same way, the **DPFLOW** has mainly theoretical interest, but has the advantage of being easily understandable.

III. A BRIEF REVIEW ON SYMMETRIES IN ILP FORMULATIONS

In a combinatorial optmization, symmetries increase the size of the search space and therefore, time to visiting symmetric solutions we will wasted. The most usual way to deal with symmetries is to add constraints that eliminate symmetric solutions. We give here a brief review on recent results in this area, focusing especially on the use of symmetry breaking constraints in mathematical programming models: LP [1], ILP [2] and MILP [3]. Please note that we refer here and after by the word ILP all variants of the mathematical programming models mentioned above.

Typical ILP formulations contain binary variables. An ILP is then symmetric if its variables can be permuted without changing the structure of the problem. For example, scheduling jobs on parallel identical machines or packing items into identical bins involve large symmetry groups.

For example, given a binary variable x_{ij}, where x_{ij} equal to one signifies that item i is assigned to bin j or that job i is assigned to machine j. The x variables can be interpreted as an $0-1$ matrix. Symmetry is often present in these kind of problems since there can be many identical bins/machines of a certain type. As result, given any feasible solution x, equivalent solutions can be generated by permuting the columns of x.

The presence of symmetry can have a significant negative effect on the performance of branch-and-bound algorithms. In the same way that it allows multiple equivalent solutions, symmetry also allows different sub-problems in the branch-and-bound tree to be equivalent.

First, Margot (2010) gives a survey of some of the approaches that have been developed for solving symmetric ILPs. These approaches are classified into four major groups: *perturbation*, *fixing variables*, *symmetry breaking inequalities*, and *pruning of the enumeration tree*. We refer to [5] for further details.

Similarly, Liberti (2012) gives a review of the most widespread approaches for breaking symmetries in ILPs together with a theorical and computational study of symmetries in the Kissing Number Problem (see [13]). In this paper, he used a generalization of the definition of formulation group given by Margot (2010), based on transforming an ILP into a DAG [4]. This allows automatic symmetry detection using graph isomorphism tools. Symmetries are then broken by means of static symmetry breaking inequalities.

In the same way, Sherali and Smith (2001) focus on the description of a natural method to remove symmetries in the context of the following problems: a *synchronous optical network (SONET) design problem*, a *minimax noise pollution problem*, and a *machine scheduling problem* (see [8]). Their method consists of augmenting the ILP model of that problem

[1] LP: Linear Programming

[2] ILP: Integer Linear Programming

[3] MILP: Mixed-Integer Linear Programming

[4] DAG: Directed Acyclic Graphs

with suitable symmetry breaking hierarchical constraints. The structure of the ILP can then be considerably improved by reducing the extent of the feasible region that must be explored by any algorithmic procedure.

Finally, Jans R. (2006) considers the issue of symmetry in the *lot-sizing problems on parallel identical machines* literature (see [18]). To break this symmetry, he simply enhances the existing ILPs by adding lexicographic ordering constraints. Other ways can be achieved by ordering the machines according to some natural logic (decreasing total setup cost per machine, decreasing total cost per machine or decreasing capacity utilization).

In summary, we can say that if symmetry is present in the ILP problem, it must be dealt with in an effective manner. There are many strategies that one can use to handle symmetries in the solution space. A most usual way is to add symmetry breaking constraints, as we can see in the next section in the case of 1D-BPP. In addition, an empirical evaluation of the impact of including separately or in combination the different symmetry breaking constraints to the ILP-0 formulation is presented in Section VIII.

IV. A BASIC SYMMETRY BREAKING CONSTRAINT

Due to the fact that all bins $j \in \{1, \ldots, n\}$ are identical (the same integer capacity C), there is complete symmetry with respect to bins. Thus, for any solution, an equivalent solution can be obtained by swapping the sets of items assigned to any pair of bins. To break this symmetry and limit the number of mathematical solutions to the actual number of different allocations of bins, we first add the following constraints:

$$y_j \geq y_{j+1} \qquad \forall j \in \{1, \ldots, n-1\} \tag{S0}$$

This constraint reduces the size of the enumeration tree by imposing that the bins are used in increasing order of index.

As we can see in Figures 2a and 2b, the optimal solution is defined by four bins. This means that the use of bins b_1, b_6, b_5, b_4 or bins b_1, b_2, b_3, b_4 is equivalent.

V. SORTING IN DECREASING ORDER OF BIN LOAD

The symmetry can be partially broken by stating that bins must be sorted by decreasing load. Consider the following constraint:

$$\sum_{i=1}^{n} w_i * x_{ij} \geq \sum_{i=1}^{n} w_i * x_{ij+1} \quad \forall j \in \{1, \ldots, n-1\} \tag{S1}$$

This constraint forces that the load of bin j must be greater than or equal to the load of bin $j+1$. An example of the effect of (S1) is given in Figure 3. The solution given in Figure 3a violates (S1), but the equivalent solution from Figure 3b respects it.

VI. SYMMETRY-LESS REFORMULATION: ILP-0-S2+S0

In this alternate formulation, the symmetry can be eliminated by stating that bins must be sorted by decreasing order of the maximum item index. To present this constraint, we define a new integer variable z_j ($j \in \{1, \ldots, n\}$) as the maximum index over all the items allocated to bin j.

In this case, the objective function is updated to the following weighted sum:

$$\min \sum_{j=1}^{n} y_j + \frac{2}{2+n(n+1)} * \sum_{j=1}^{n} z_j \tag{6}$$

in a way that minimizes the number of used bins $\sum_{j=1}^{n} y_j$ first (primary objective) and then the sum of its maximum item index $\sum_{j=1}^{n} z_j$ (secondary objective). The latter is also weighted by the following coefficient $\frac{2}{2+n(n+1)}$ to impose the lexicographic optimization ordering as mentioned before. This means that $\sum_{j=1}^{n} y_j$ is the integer part of the objective function and that $\frac{2}{2+n(n+1)} * \sum_{j=1}^{n} z_j < 1$. Indeed:

$$\sum_{j=1}^{n} z_j = \frac{n(n+1)}{2} < \frac{n(n+1)}{2} + 1 = \frac{2+n(n+1)}{2}$$

$$\frac{n(n+1)}{2} < \frac{2+n(n+1)}{2}$$

$$\text{hence} \quad \frac{2}{2+n(n+1)} \sum_{j=1}^{n} z_j < 1$$

The constraints of ILP-0-S2+S0 formulation are constraints (1)—(4) and the following set of inequalities which is denoted as (S2):

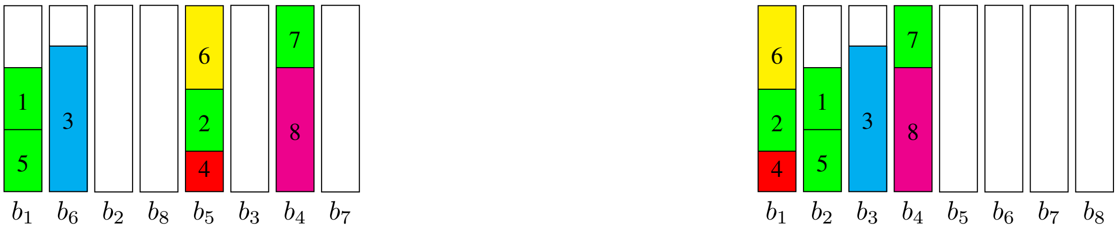
$$\begin{cases} i * x_{ij} \leq z_j & \forall\ i \in \{1, \ldots, n\}, \forall j \in \{1, \ldots, n\} \quad (7) \\ z_j \geq z_{j+1} & \forall\ j \in \{1, \ldots, n-1\} \quad (8) \\ z_j \geq 0 & \forall\ j \in \{1, \ldots, n\} \quad (9) \end{cases}$$

Constraints (7) ensure that the bin index z_j must be greater than or equal to the the maximum item index in bin j. Constraints (8) mean that the bin index bin z_j must be greater than or equal to the maximum item index in bin $j+1$. Constraints (8) guarantee the positivity of the bin index z_j. In addition, we consider the inequality constraint (S0) to reduce the size of the enumeration tree by imposing that the bins are used in increasing order of index.

As illustrated in Figure 4b, the corresponding solution provided by this new formulation ILP-0-S2+S0 is equivalent to the initial one given in Figure 4a.

VII. MATRIX-BASED SYMMETRY BREAKING CONSTRAINTS

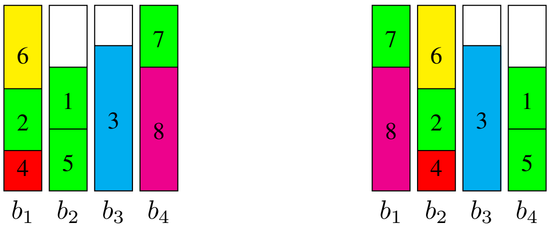
The matrix-based symmetry breaking constraints, proposed here, were inspired from those proposed by [2] and reused by [10] for the classical *Job Scheduling Problem* (specifically operating room scheduling problems), which is a well known


1
5
3
6
2
4
7
8
b_1 b_6 b_2 b_8 b_5 b_3 b_4 b_7
6
2
4
1
5
3
7
8
b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8

(a) A solution without using (S0)

(b) A solution using (S0)

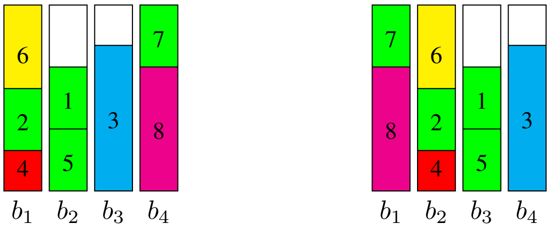
Fig. 2: Illustration of (S0)



(a) A solution without using (S1)

(b) A solution using (S1)

Fig. 3: Illustration of (S1)



(a) A solution without using S_2

(b) A solution using S_2

Fig. 4: Illustration of `ILP-0-S2+S0`

$\mathcal{NP}$-hard combinatorial optimization problem. These constraints restrict the feasible region to a minimal fundamental domain that has lexicographically decreasing columns.

For example, as illustrated in Figure 5, the 0/1 matrix x does not have lexicographically decreasing columns, so it is not in the fundamental domain. Permuting columns 1 and 2 gives the 0/1 matrix x', which has lexicographically-decreasing columns, so it is in the fundamental domain.

In the context of 1D-BPP, we introduce the following inequality (S3), which guarantees that for any x_{ij} equal to one with i and j greater than one, there is at least one lower-indexed item i assigned to bin $j-1$.

$$x_{ij} \leq \sum_{p=1}^{i-1} x_{p,j-1} \quad \forall i \in \{2,\dots,n\}, \forall j \in \{2,\dots,n\} \qquad \text{(S3)}$$

Now, as an alternative to asking for a lower-indexed item in the previous bin, one could as well ask for a bigger-indexed item in the previous bin. This condition is expressed by the following constraint:

$$x_{ij} \leq \sum_{p=i}^{n-1} x_{p,j-1} \qquad \forall i \in \{2,\dots,n\}, \forall j \in \{i,\dots,n\} \qquad \text{(S3Bis)}$$

Of course, only one of (S3) or (S3Bis) can be enforced, since the two constraints are incompatible.

While the only $0-1$ matrices that satisfy constraints (S3) have lexicographically decreasing columns, the constraints can be strengthened to create a tighter LP relaxation.
Using the property that each row of x must contain a single one, the general form of constraints in (S3) becomes:

$$\sum_{s=j}^{n} x_{is} \leq \sum_{p=1}^{i-1} x_{p,j-1} \quad \forall\ i \in \{2,\dots,n\}, \forall j \in \{2,\dots,n\} \qquad \text{(S4)}$$

In the same way, the general form of constraints in (S3Bis) becomes:

$$\sum_{s=j}^{n} x_{is} \leq \sum_{p=i}^{n-1} x_{p,j-1} \quad \forall\ i \in \{2,\dots,n\}, \forall j \in \{2,\dots,n\} \qquad \text{(S4Bis)}$$

VIII. Computational experiments and discussion

In this section, we test the effectiveness of the formulations described in the previous sections. Our experiments were motivated by this main goal: to evaluate, with respect to the standard formulation `ILP-0` (see Section II-A), the benefits obtained by including the valid inequalities of breaking symmetries previously described in Sections IV — VII.

A. Setup

We implemented formulations `ILP-0` and all its variants in Gurobi Optimizer 7.5.2 using Python 3.6 (https://www.gurobi.com/), running on a PC running Linux Debian 8.0 (“Jessy”). It has a Core 2 Duo CPU running at 3 GHz, and with 4 gigabytes of RAM. All executions where run within a single thread; only one core of the CPU was used.

We considered seven variants of the formulation `ILP-0`:

- one is the classical ILP model of 1D-BPP: `ILP-0` (see Section II-A);
- one including the symmetry breaking constraint, given by Eq. (S0), hereafter denoted as `ILP-0+S0` (see Section IV);

$$x = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} 0 & \colorbox{lightgray}{1} & 0 & 0 \\ \colorbox{lightgray}{1} & 0 & 0 & 0 \\ 0 & 0 & \colorbox{lightgray}{1} & 0 \\ \colorbox{lightgray}{1} & 0 & 0 & 0 \\ 0 & \colorbox{lightgray}{1} & 0 & 0 \\ \colorbox{lightgray}{1} & 0 & 0 & 0 \\ 0 & 0 & 0 & \colorbox{lightgray}{1} \\ 0 & 0 & 0 & \colorbox{lightgray}{1} \end{pmatrix} \end{array}$$

(a) 0/1 matrix x

$$x' = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} \colorbox{lightgray}{1} & 0 & 0 & 0 \\ 0 & \colorbox{lightgray}{1} & 0 & 0 \\ 0 & 0 & \colorbox{lightgray}{1} & 0 \\ 0 & \colorbox{lightgray}{1} & 0 & 0 \\ \colorbox{lightgray}{1} & 0 & 0 & 0 \\ 0 & \colorbox{lightgray}{1} & 0 & 0 \\ 0 & 0 & 0 & \colorbox{lightgray}{1} \\ 0 & 0 & 0 & \colorbox{lightgray}{1} \end{pmatrix} \end{array}$$

(b) 0/1 matrix x'

Fig. 5: Illustration of S_3

- one is the new alternate symmetry ILP model of 1D-BPP: `ILP-0-S2+S0` (see Section VI);
- one including the symmetry breaking constraints, given by Eq. (S0) and Eq. (S3), hereafter denoted as `ILP-0+S0+S3` (see Section VII);
- one including the symmetry breaking constraints, given by Eq. (S0) and Eq. (S3Bis), hereafter denoted as `ILP-0+S0+S3Bis` (see Section VII);
- one including the symmetry breaking constraints, given by Eq. (S0) and Eq. (S4), hereafter denoted as `ILP-0+S0+S4` (see Section VII);
- one including the symmetry breaking constraints, given by Eq. (S0) and Eq. (S4Bis), hereafter denoted as `ILP-0+S0+S4Bis` (see Section VII).

Moreover, for each implementation we considered two kinds of run-time settings. Specifically, we run each formulation by activating and deactivating Gurobi proprietary cuts, aiming at empirically validating the theoretical results presented in the previous sections.

As a Gurobi settings and in each implementation, we activated both Gurobi heuristics (with its default value of 0.05) and the presolving strategies and deactivated the Gurobi proprietary symmetry. In addition, in implementations `ILP-0+S0+S3`, `ILP-0+S0+S3Bis`, `ILP-0+S0+S4` and `ILP-0+S0+S4Bis`, we set the different added valid inequalities (symmetry breaking or cutting plane) as lazy inequalities, with a value of 2, i.e., all lazy constraints that are violated by a feasible solution will be pulled into the model. In contrast, we used a particular branching strategy for binary variables consisting of giving priority to x variables with respect to y variables during the branching process. We then set the ordering branch variable value to 1.

B. Data-sets

To test the performances of formulations `ILP-0`, `ILP-0+S0`, `ILP-0-S2+S0` and `ILP-0+S0+Si` ($i \in \{3, 3Bis, 4, 4Bis\}$), we considered the data-sets from the literature of the 1D-BPP, referred to in the following as the `BPPLIB` and described in [14]. All instances are downloaded from the web page http://or.dei.unibo.it/library/bpplib. The main characteristics of the used data-sets are summarized in Table II. Each data-set contains a number of tested instances (column **Tested inst.**) of the 1D-BPP, ad-hoc or uniformly distributed (column **Distribution**), characterized by having the same number of items (column **n**) and the same bin capacity (column **C**). Detailed information about the structure of each of these benchmarks can be found in [15] or in the the BPPLIB web page.

C. Comparison of the ILP Models

To evaluate the different proposed ILP formulations, described in a previous sections, we first compare its size complexity, which indicates how large a problem is in terms of binary variables and constraints as a function of n (the number of bins as well as of items). We note that in these formulations, no integer variables, except for the formulation `ILP-0-S2+S0` (n integer variables) as well as no Big-M constraints are considered. As we can see in Table III, the `ILP-0` and its variants are generally equivalent in terms of binary variables. On the other hand, we can see that the three formulations: `ILP-0`, `ILP-0+S0`, `ILP-0+S0+S1` have the same order of number of constraints: $\mathcal{O}(n)$. In the same way, formulations `ILP-0+S0+Si` for i in $\{3, 3Bis, 4, 4Bis\}$ have approximately the same order of $\mathcal{O}(n^2)$ of constraints number. Hence, the strengthening of the `ILP-0` by symmetry breaking inequalities seems to be more favorable for effectively reducing the search tree.

D. Numerical results

In this section, we analyze our results under one main axis, in which `ILP-0+S0`, `ILP-0-S2+S0` and `ILP-0+S0+Si` for i in $\{3, 3Bis, 4, S4Bis\}$ vs. `ILP-0` are compared. Our goal is to evaluate, with respect to `ILP-0`, the benefits obtained by including symmetry breaking constraints.

1) Analysis of the solution times: Table IV provides the results for the literature instances (described in table II) obtained by running the ILP formulations with a time limit of 60 seconds. Columns 1 and 2 identify the benchmarks (characterized by a specific number of items n and a bin capacity C) and give the number of instances for which the ILP formulations were executed. The column associated with each ILP formulation provides the number of such instances that were solved to proven optimality and, in parentheses, the average CPU time in the following two cases: `With GC` and

TABLE II: Main characteristics of the 9 used data-sets from the literature of the 1D-BPP (provided by the BPPLIB) considered in the experiments

Data-set	Reference	Parameters of the instances			
		Tested inst.	n	C	Distribution
Falkenauer T	[4]	40	$\{60,120\}$	1000	ad-hoc
Falkenauer U	[4]	40	$\{120,250\}$	150	uniform
Scholl 1	[1]	360	$\{50,100\}$	$\{100,120,150\}$	uniform
Scholl 2	[1]	240	$\{50,100\}$	1000	uniform
Scholl 3	[1]	10	200	100 000	uniform
Schwerin 1	[17]	100	100	1000	uniform
Schwerin 2	[17]	100	120	1000	uniform
Wascher	[21]	17	$[57-239]$	10 000	ad-hoc
Randomly Generated	[14]	240	$\{50,100\}$	$\{50,75,100,120,125,150,200,300,400,500,750,1000\}$	ad-hoc

TABLE III: Comparison of ILP formulations

Models	No. variables		No. Constraints
	binary	integer	
ILP-0	n^2+n	0	$2n$
ILP-0+S0	n^2+n	0	$3n-1$
ILP-0+S0+S1	n^2+n	0	$4n-1$
ILP-0-S2+S0	n^2+n	n	n^2+5n-2
ILP-0+S0+S3	n^2+n	0	$(n^2)/2+3n/2$
ILP-0+S0+S3Bis	n^2+n	0	$(n^2)/2+7n/2-1$
ILP-0+S0+S4	n^2+n	0	$(n^2)/2+7n/2-1$
ILP-0+S0+S4Bis	n^2+n	0	$(n^2)/2+7n/2-1$

No GC which refer to the activation and the deactivation of Gurobi proprietary cuts, respectively. For instances not solved, the time limit is considered as the solution time. A cell with a value of – means that no feasible solution found when solving an instance using Gurobi optimizer within the time limit. For each instance set, **boldface** highlights the highest number of instances optimally solved. In the same way, for each instance set, colored cell (Blue) highlights the cases where all instances were solved to proven optimality. Finally, row **Total (average)** reports the total number of instances optimally solved within the time limit for each formulation as well as the average CPU time in seconds for its resolution.

The results in Table IV provide the number of instances solved in less than one minute (average CPU time in seconds), by, respectively, ILP-0, ILP-0+S0, ILP-0-S2+S0 and ILP-0+S0+Si for i in $\{3, 3Bis, 4, S4Bis\}$.
The results showed that all formulations were unable to solve any instance in the data-sets **FalkT**$_{(60,1000)}$, **FalkT**$_{(120,1000)}$, **FalkU**$_{(250,150)}$ and **Scho3**$_{(200,100000)}$ within the time limit, either when activating or deactivating the Gurobi Optimizer proprietary cuts, expect in the case of formulation ILP-0+S0+S4Bis, i.e., it was able to optimally solve 19 instances in less than 20 seconds on average either both cases respect to the proprietary cuts.
This trend is also marked in the case of the data-set **Wae**$_{([57-239],10000)}$ when activating the Gurobi Optimizer proprietary cuts, expect in the case of both ILP-0+S0+S4 and ILP-0+S0+S4Bis formulations, i.e., they were able to optimally solve only one instance in less than 50 seconds.

Formulation ILP-0 was able to solve within the time limit only 8 instances in the data-set **FalkU**$_{(120,150)}$, both when activating and deactivating the Gurobi Optimizer proprietary cuts. Unfortunately, the combination of formulation ILP-0 with the symmetry breaking constraints (ILP-0+S0, ILP-0-S2+S0 and ILP-0+S0+S3Bis) did not prove successful, i.e., only one or at most two instances can be optimally solved. In contrast, formulation ILP-0+S4Bis was able to optimally solve the biggest number of instances (11 instances in 26.8 seconds on average) when activating the Gurobi Optimizer proprietary cuts.

Formulations ILP-0, ILP-0+S0 ,ILP-0-S2+S0 and ILP-0+S0+S3Bis generally have a similar performance, i.e., they give rise to too similar results in the data-sets **Scho1**$_{(50,C)}$, for C in $\{100,120,150\}$ in both cases regarding the cuts proprietary. However, formulation ILP-0+S4Bis performs clearly better than the other formulations by giving rise to the best results in the data-sets **Scho1**$_{(50,C)}$ for C in $\{100,120,150\}$, i.e., it was able to solve within the time limit all the instances when activating the Gurobi Optimizer proprietary cuts, expect in the case of the data-sets **Scho1**$_{(50,C)}$ for C in $\{120,150\}$ when deactivating the Gurobi Optimizer proprietary cuts.

Formulation ILP-0+S0+S4Bis provides the highest number of optimally solved instances in the data-sets **Scho1**$_{(100,C)}$, for C in $\{100,120,150\}$ when deactivating and activating the Gurobi Optimizer proprietary cuts, i.e., it was able to solve more than 40 instances in each case. In other hand, the behavior of both formulations ILP-0+S3 and ILP-0+S4 was similar. These formulations were unable to solve no instances, expect 3 instances in the data-set **Scho1**$_{(100,120)}$ were solved by ILP-0+S4 when activating the Gurobi Optimizer proprietary cuts.

Formulations ILP-0, ILP-0+S0 and ILP-0-S2+S0 give rise to the same results in the data-set **Scho2**$_{(50,1000)}$ when deactivating the Gurobi Optimizer proprietary cuts, i.e., they were able to optimally solve the highest number of instances (111 instances in less than 1 second on average). However, when activating the Gurobi Optimizer proprietary cuts, formulation ILP-0 provides the highest number of

TABLE IV: Number of instances solved in less than one minute (average CPU time in seconds), for formulations `ILP-0`, `ILP-0+S0`, `ILP-0-S2+S0` & `ILP-0+S0+Si` for i in $\{3, 3Bis, 4, 4Bis\}$

Set	Tested inst.	ILP-0		ILP-0+S0		ILP-0-S2+S0		ILP-0+S3		ILP-0+S3Bis		ILP-0+S4		ILP-0+S4Bis	
		No GC	With GC	No GC	With GC	No GC	With GC	No GC	With GC	No GC	With GC	No GC	With GC	No GC	With GC
FalkT$_{(60,1000)}$	20	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	- (60.0)	- (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	**19** (19.8)	**19** (15.1)
FalkT$_{(120,1000)}$	20	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	- (60.0)	- (60.0)	0 (60.0)	0 (60.0)	- (60.0)	- (60.0)	0 (60.0)	0 (60.0)
FalkU$_{(120,150)}$	20	**8** (30.0)	8 (16.8)	1 (1.1)	2 (1.7)	2 (28.6)	1 (38.5)	- (60.0)	- (60.0)	2 (24.9)	2 (37.3)	- (60.0)	- (60.0)	6 (26.9)	**11** (26.8)
FalkU$_{(250,150)}$	20	0 (60.0)	- (60.0)	0 (60.0)	- (60.0)	0 (60.0)	- (60.0)	- (60.0)	- (60.0)	0 (60.0)	- (60.0)	- (60.0)	- (60.0)	- (60.0)	- (60.0)
Scho1$_{(50,100)}$	60	18 (1.3)	19 (0.8)	16 (6.8)	20 (3.9)	16 (2.8)	20 (2.1)	- (60.0)	- (60.0)	18 (2.8)	16 (0.4)	3 (30.0)	37 (14.5)	**60** (0.3)	**60** (0.2)
Scho1$_{(50,120)}$	60	17 (2.2)	19 (2.9)	13 (9.1)	26 (2.7)	11 (3.1)	26 (3.6)	2 (46.1)	2 (27.3)	17 (4.2)	22 (1.7)	6 (7.3)	38 (10.0)	**57** (0.6)	**60** (0.4)
Scho1$_{(50,150)}$	60	47 (1.7)	47 (1.0)	40 (4.6)	43 (2.4)	40 (3.1)	44 (1.7)	6 (21.8)	6 (33.5)	47 (2.7)	48 (2.8)	16 (14.7)	18 (15.0)	**56** (0.9)	**60** (1.3)
Scho1$_{(100,100)}$	60	8 (12.9)	7 (4.3)	4 (12.2)	11 (13.0)	4 (12.7)	10 (7.5)	- (60.0)	- (60.0)	6 (9.9)	8 (9.9)	- (60.0)	- (60.0)	**54** (2.8)	**60** (2.3)
Scho1$_{(100,120)}$	60	3 (15.0)	6 (23.4)	1 (1.0)	8 (9.5)	0 (60.0)	8 (10.9)	- (60.0)	- (60.0)	6 (3.6)	6 (14.8)	- (60.0)	3 (36.9)	**51** (4.4)	**59** (5.0)
Scho1$_{(100,150)}$	60	29 (11.5)	29 (11.3)	17 (16.5)	17 (9.4)	17 (9.7)	17 (3.5)	- (60.0)	- (60.0)	21 (11.5)	26 (10.4)	0 (60.0)	0 (60.0)	**42** (13.6)	**43** (11.6)
Scho2$_{(50,1000)}$	120	**111** (0.7)	**113** (0.9)	**111** (0.3)	111 (0.3)	**111** (0.3)	111 (0.6)	97 (5.6)	98 (7.5)	110 (0.5)	111 (0.7)	108 (2.2)	110 (2.9)	107 (2.2)	111 (2.6)
Scho2$_{(100,1000)}$	120	**103** (1.5)	101 (1.7)	102 (2.5)	**102** (1.8)	101 (3.3)	101 (1.7)	46 (27.6)	44 (23.7)	92 (4.2)	98 (3.9)	70 (15.7)	72 (15.5)	57 (11.6)	57 (13.4)
Scho3$_{(200,100000)}$	10	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	0 (60.0)	- (60.0)	- (60.0)	0 (60.0)	0 (60.0)	- (60.0)	- (60.0)	- (60.0)	- (60.0)
Schw1$_{(100,1000)}$	100	52 (8.4)	48 (10.3)	40 (6.3)	52 (4.7)	40 (6.5)	51 (5.0)	- (60.0)	- (60.0)	**56** (13.4)	**62** (12.1)	12 (24.7)	11 (22.2)	15 (18.4)	10 (26.0)
Schw2$_{(120,1000)}$	100	**49** (8.8)	38 (9.8)	26 (8.4)	43 (10.8)	28 (10.0)	43 (11.8)	- (60.0)	- (60.0)	44 (13.2)	**46** (10.8)	2 (37.6)	1 (23.0)	9 (20.6)	6 (24.2)
Wae$_{([57-239],10000)}$	10	1 (40.6)	- (60.0)	8 (15.5)	- (60.0)	**9** (17.8)	- (60.0)	- (60.0)	- (60.0)	1 (37.2)	- (60.0)	2 (35.9)	**1** (42.3)	- (60.0)	**1** (49.8)
RG	240	151 (8.1)	161 (8.1)	103 (7.9)	117 (6.3)	97 (5.4)	108 (6.6)	3 (35.0)	44 (20.3)	140 (8.2)	153 (10.6)	16 (21.4)	22 (22.8)	**189** (4.5)	**212** (5.2)
Total (average)	1140	597 (11.0)	596 (7.6)	482 (7.1)	552 (5.6)	476 (8.6)	540 (7.8)	154 (27.2)	194 (22.4)	560 (10.5)	598 (9.6)	235 (21.0)	314 (20.51)	**722** (9.7)	**769** (13.1)

optimally solved instances, with a value of 113 (in less than 1 second on average).

In the case of the data-set **Scho2**$_{(100,1000)}$, formulation `ILP-0` performs better than the other formulations when deactivating the Gurobi Optimizer proprietary cuts, i.e., it was able to optimally solve 103 instances. In contrast, when activating the Gurobi Optimizer proprietary cuts, formulation `ILP-0-S2+S0` is better by optimally solving 102 instances in less than 2 seconds on average.

In both **Schw1**$_{(100,1000)}$ and **Schw2**$_{(120,1000)}$ data-sets, the table shows the clear superiority of formulation `ILP-0+S0+S3Bis` over the other formulations, either when activating or deactivating the Gurobi Optimizer proprietary cuts, expect in the case of the data-set **Schw2**$_{(120,1000)}$, for which formulation `ILP-0` performs better when activating Gurobi Optimizer proprietary cuts.

Among the proposed symmetry breaking constraints, formulation `ILP-0+S0+S4Bis` provides the highest number of optimally solved instances in the data-set **RG** (Randomly Generated) compared to the formulation `ILP-0`. It was able to solve 189 and 212 instances (in 5 seconds on average), when activating and deactivating the Gurobi Optimizer proprietary cuts, respectively.

The table confirms the clear superiority of `ILP-0+S0+S4Bis` over the other formulations. It was able to optimally solve within the time limit (60 seconds) in total 722 and 769 instances (in 9.7 and 13.1 seconds on average), respectively. In the same way, we can see that the behavior of `ILP-0+S0`, `ILP-0-S2+S0`, `ILP-0+S0+S3Bis` and `ILP-0+S0+S3Bis` formulations was generally similar to that of the standard formulation `ILP-0`. In addition, we can see that the formulation `ILP-0+S0+S4`, where constraints (S4) is the general form of constraints in (S3), performs better than formulation `ILP-0+S0+S3`, especially when activating the Gurobi Optimizer proprietary cuts. For example, formulation `ILP-0+S0+S3` was unable to solve within the time limit any instance in the data-sets **FalkU**$_{(120,150)}$, **Scho1**$_{(50,100)}$, **Scho2**$_{(100,C)}$ for C in $\{100, 120, 150\}$, **Schw1**$_{(100,1000)}$, **Schw2**$_{(120,1000)}$ and **Wae**$_{([57-239],10000)}$. This means that the constraint (S3) is the least effective among all the other symmetry breaking constraints.
However, the classical formulation `ILP-0` remains an effective model to solve some of the used BPPLIB data-sets, specifically the data-sets **Scho2**$_{(n,100)}$ for n in $\{50, 100\}$ and **Schw2**$_{(120,1000)}$.
As a general trend, the results showed that the use of Gurobi Optimizer proprietary cuts may increase the number of instances optimally solved and reduce the solution time in most formulations with the exception of the formulation `ILP-0` for certain data-sets: **Scho1**$_{(100,100)}$, **Scho1**$_{(100,150)}$, **Scho2**$_{(n,100)}$ for n in $\{50, 100\}$, **Schw2**$_{(100,1000)}$ and **Schw2**$_{(120,1000)}$.

2) Analysis of the gap : We analyze the performances of each proposed formulation with respect to both Gurobi gap `GGap (%)`, i.e., the difference between the best feasible solution and the best lower bound found by Gurobi at the end of CPU time limit (60 s) and the `Gap`, i.e., the difference between the best lower bound to a given instance of the 1D-BPP in a specific data-set and the objective function value of the linear programming relaxation at the root node of the respective search tree, divided by the best lower bound.

Figs. 6, 8, 7 and 9 show the results of our computational experiments in term of box-and-whisker plots. Specifically, the bottom and the top of each box represent the first and third quartiles; the band inside the box represents the second quartile (the median), and the ends of the whiskers represent the 9th percentile and the 91st percentile. Outliers are plotted as individual points.

In particular, Figs. 6 and 7 show the performances of the 7 formulations when disabling Gurobi Optimizer proprietary cuts. In contrast, Figs. 8 and 9 show the performances of the 7 formulations when enabling Gurobi Optimizer proprietary cuts. The gaps are expressed in percentage and the performances are (i) represented by means of box and whiskers plots and (ii) shown in function of 9 data-sets: **FalkU**$_{(120,150)}$,

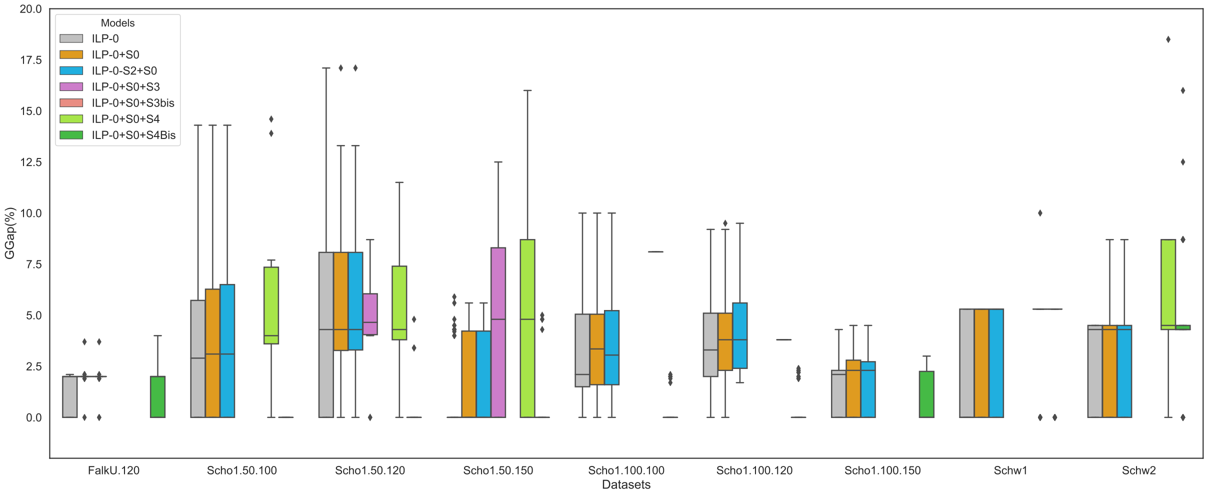


Fig. 6: Comparison of the gurobi gaps (%) of formulations `ILP-0`, `ILP-0+S0`, `ILP-0-S2+S0` and `ILP-0+S0+Si` for i in $\{3, 3Bis, 4, 4Bis\}$ on the data-sets from BPPLIB, when disabling Gurobi Optimizer proprietary cuts

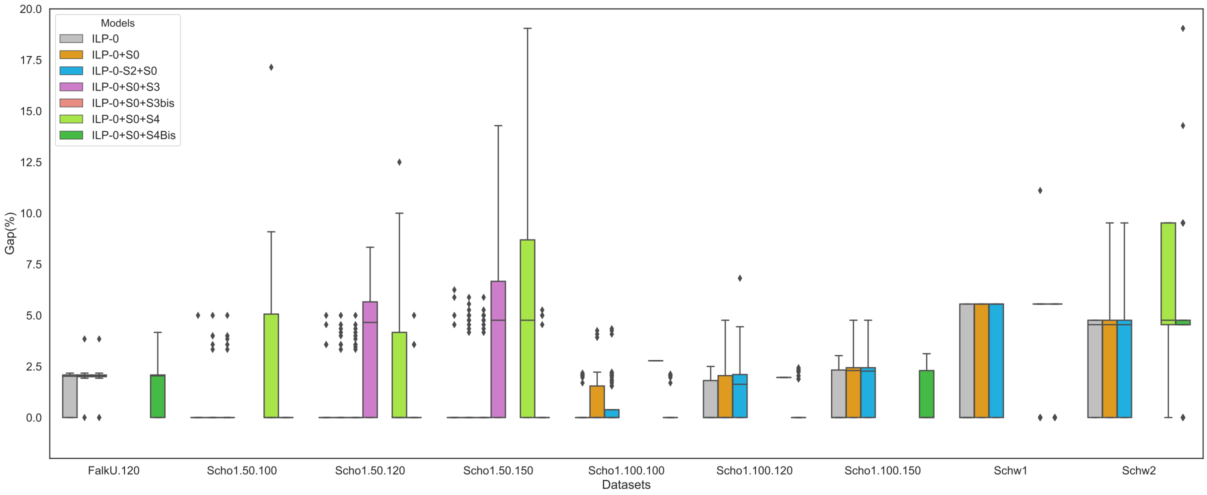


Fig. 7: Comparison of the gaps (%) of formulations `ILP-0`, `ILP-0+S0`, `ILP-0-S2+S0` and `ILP-0+S0+Si` for i in $\{3, 3Bis, 4, 4Bis\}$ on the data-sets from BPPLIB, when disabling Gurobi Optimizer proprietary cuts

Scho1$_{(50,C)}$, for C in $\{100, 120, 150\}$), **Scho1**$_{(100,C)}$, for C in $\{100, 120, 150\}$), **Schw1**$_{(100,1000)}$ and **Schw2**$_{(120,1000)}$. For the rest of data-sets, the behavior of all formulations was generally similar, as shown in Table IV.

As shown in Figs. 6 and 8, respectively in Figs. 7 and 9, formulation `ILP-0+S0+S4Bis` provides generally the smallest median gurobi gaps `GGap` and gaps `Gap`, except in the case of data-sets **FalkU**$_{(120,150)}$ and **Scho1**$_{(100,150)}$, in which its results are too similar to those of `ILP-0`. This fact is consistent with the results discussed in the previous sections.

In the same way, Figs. 6 and 8 show that the activation of the Gurobi Optimizer proprietary cuts from one hand causes a general decrement of the median gaps related to the formulations but on the other hand does not change their general trends. In particular, we observed that the use of these proprietary strategies has a major impact on formulations `ILP-0+S0`, `ILP-0-S2+S0`, `ILP-0+S0+S3` and `ILP-0+S0+S4`, minor in formulation `ILP-0` and becomes negligible or absent in formulations `ILP-0+S0+S3Bis` and `ILP-0+S0+S4Bis`. The trend is more marked in datasets **Scho1**$_{(50,C)}$, for C in $\{100, 120\}$) and **Scho1**$_{(100,C)}$, for C in $\{100, 120, 150\}$) and less in the others, for which the improvements are marginal.

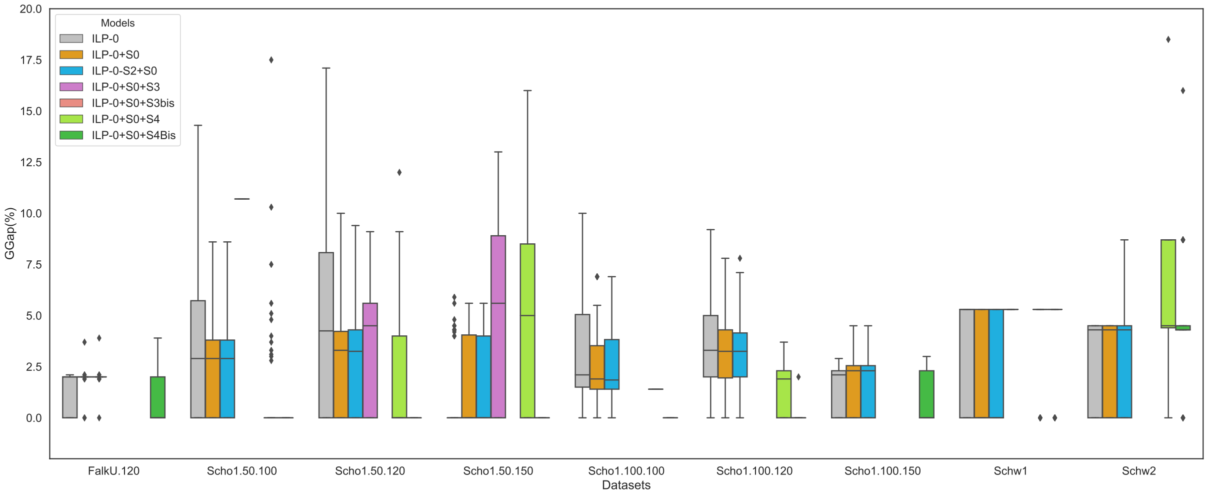


Fig. 8: Comparison of the gurobi gaps (%) of formulations `ILP-0`, `ILP-0+S0`, `ILP-0-S2+S0` and `ILP-0+S0+Si` for i in $\{3, 3Bis, 4, 4Bis\}$ on the data-sets from BPPLIB, when enabling Gurobi Optimizer proprietary cuts

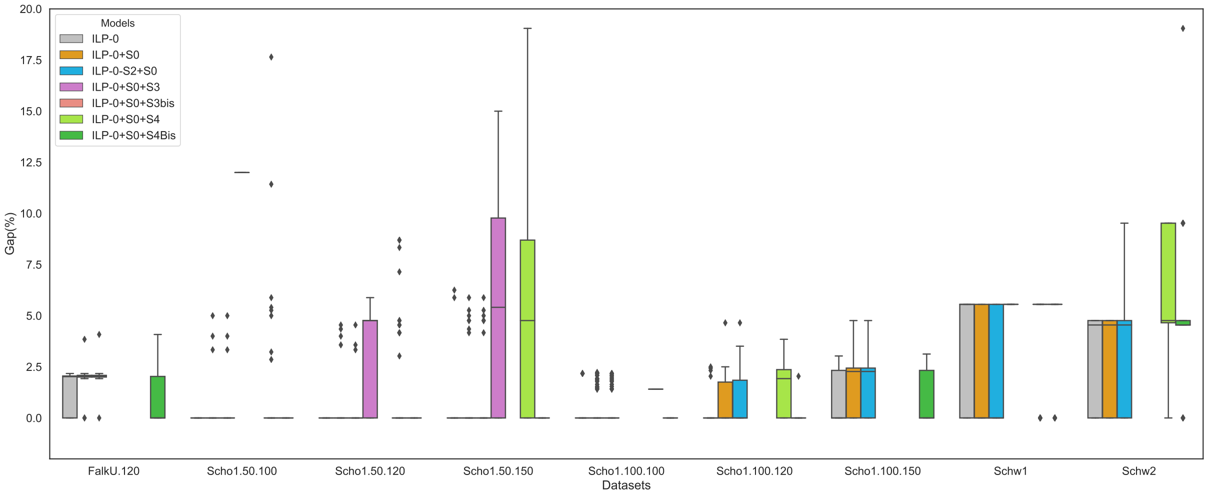


Fig. 9: Comparison of the gaps (%) of formulations `ILP-0`, `ILP-0+S0`, `ILP-0-S2+S0` and `ILP-0+S0+Si` for i in $\{3, 3Bis, 4, 4Bis\}$ on the data-sets from BPPLIB, when enabling Gurobi Optimizer proprietary cuts

Our study of the topic of symmetry breaking constraints for the classical formulation `ILP-0` has led us to the consideration of an alternative encoding, and thus an alternate symmetry-less formulation for the 1D-BPP. This new formulation encodes partitions directly, removing the need for variables to encode the use of bins. We refer to Hadj Salem and Kieffer (2020) [12] for further details.

IX. Conclusion and future work

We have presented a study of how symmetry breaking constraints can improve the resolution performance of integer linear formulations for the 1-dimensional bin packing problem. Our study includes a review of all known and/or symmetry breaking constraints.

One exciting perspective of this work would be to investigate the impact of reusing these inequalities to other optimization problem ILP formulations, e.g., BPP with Conflicts (BPC), Cutting Stock Problem, etc. Understand how symmetry breaking methods interact with other ILP features such as branching strategies and cutting plane methods, in the context of packing problems, may be also a new direction should be investigated.

Acknowledgements

We are indebted to Professor André ROSSI (Paris-Dauphine University - LAMSADE) for suggesting both symmetry breaking constraints (S1) (see Section V) and the alternate formulation `ILP-0-S2+S0` (see Section VI), as well as for many constructive comments, that led to a clearer presentation.

References

[1] A. Scholl, R. Klein and C. Jürgens, "Bison: A fast hybrid procedure for exactly solving the one-dimensional bin packing problem," *Computers & Operations Research,* vol. 24(7), 1997, pp. 627–645.

[2] B. T. Denton, A. J. Miller, H. J. Balasubramanian and T. R. Huschka, "Optimal allocation of surgery blocks to operating rooms under uncertainty," *Operations research,* vol. 58(4-part-1), 2010, pp. 802–816.

[3] D.M. Ryan and E.A. Foster, "An integer programming approach to scheduling," *Computer scheduling of public transport urban passenger vehicle and crew scheduling,* 1981, pp. 269–280.

[4] E. Falkenauer, "A hybrid grouping genetic algorithm for bin packing," *Journal of heuristics,* vol. 2(1), 1996, pp. 5–30.

[5] F. Margot, "Symmetry in integer linear programming," *50 Years of Integer Programming 1958-2008,* Springer, 2010, pp. 647–686.

[6] H. Cambazard and B. O'Sullivan, "Propagating the bin packing constraint using linear programming," *International Conference on Principles and Practice of Constraint Programming,* St. Andrews, Scotland, 2010, pp. 129–136.

[7] H. Dyckhoff, "A new linear programming approach to the cutting stock problem," *Operations Research,* vol. 29(6), 1981, pp. 1092–1104.

[8] H.D. Sherali and J.C. Smith, "Improving discrete model representations via symmetry considerations," *Management Science,* vol. 47(10), 2001, pp. 1396–1407.

[9] J. Lysgaard, A.N Letchford and R.W Eglese, "A new branch-and-cut algorithm for the capacitated vehicle routing problem," *Mathematical Programming,* vol. 100(2), 2004, pp. 423–445.

[10] J. Ostrowski, M.F. Anjos and A. Vannelli, "Symmetry in scheduling problems," ,Citeseer, 2010, pp. 59–70.

[11] J. V. De Carvalho, "LP models for bin packing and cutting stock problems," *European Journal of Operational Research,* vol. 141(2), 2002, pp. 253-273.

[12] K. Hadj Salem and Y. Kieffer, "New Symmetry-less ILP Formulation for the Classical One Dimensional Bin-Packing Problem," 26^{th} *International Computing and Combinatorics Conference,* Atlanta, GA, USA, 2020, pp. 423-434.

[13] L. Liberti, "Symmetry in mathematical programming," *Mixed Integer Nonlinear Programming,* Springer, New York, NY, 2012, pp. 263–283.

[14] M. Delorme, M. Manuel and S. Martello, "Bin packing and cutting stock problems: Mathematical models and exact algorithms," *European Journal of Operational Research,* vol. 255, 2016, pp. 1–20.

[15] M. Delorme, M. Manuel and S. Martello, "BPPLIB: a library for bin packing and cutting stock problems," *Optimization Letters,* vol. 12(2), 2018, pp. 235–250.

[16] M. Jünger, T. M. Liebling, D. Naddef, G. L. Nemhauser, W. R. Pulleyblank, G. Reinelt, G. Rinaldi and L. A. Wolsey, "50 Years of integer programming 1958-2008: From the early years to the state-of-the-art," *Springer Science & Business Media,* 2009, pp. 123-136.

[17] P. Schwerin G. and Wäscher, "The bin-packing problem: A problem generator and some numerical experiments with FFD packing and MTP," *International Transactions in Operational Research,* vol. 4(5-6), 1997, pp. 377–389.

[18] R. Jans, "Solving lot-sizing problems on parallel identical machines using symmetry breaking constraints," *INFORMS Journal on Computing,* vol. 21(1), 2009, pp. 123-136.

[19] S. Martello and P. Toth, "Knapsack problems: algorithms and computer implementations," *John Wiley & Sons, Inc.,* 1990, pp. 123-136.

[20] S. Martello and P. Toth, "Lower bounds and reduction procedures for the bin packing problem," *Discrete applied mathematics,* vol. 28(1), 1990, pp. 59–70.

[21] G. Wäscher and T. Gau, "Heuristics for the integer one-dimensional cutting stock problem: A computational study," *Operations-Research-Spektrum,* vol. 18(3), 1996, pp. 131–144.

DOI: 10.15439/2020F18
ISSN 2300-5963 ACSIS, Vol. 21

Bi-level Optimization Application for Urban Traffic Management

Krasimira Stoilova
Institute of Information and Communication Technologies – Bulgarian Academy of Sciences,
Acad. G. Bonchev str. bl.2, 1113 Sofia, Bulgaria
Email: k.stoilova@hsi.iccs.bas.bg

Todor Stoilov
Institute of Information and Communication Technologies – Bulgarian Academy of Sciences,
Acad. G. Bonchev str. bl.2, 1113 Sofia, Bulgaria
Email: todor@hsi.iccs.bas.bg

***Abstract*—A bi-level modeling for traffic lights optimization is presented in the paper. The bi-level modeling allows increasing the set of control influences, the number of constraints and applies two goal functions in hierarchical order. The bi-level formalism allows integration of small optimization problems in hierarchical order to a complex interconnected and complicated optimization problem. These features have been applied for optimal control of traffic lights in urban network. The bi-level problem formulation allows to minimize the queue lengths of vehicles and to maximize the outgoing flows from arterial directions. Both control influences of the green light durations and time cycles are evaluated as optimal bi-level control influences.**

I. Introduction

TRAFFIC congestion in urban networks is an everyday problem, which has negative influences on many human and society activities. The traffic congestions result in queuing of vehicles in urban crossroad sections and generate excessive delays, increasing pollution, degradation of infrastructure. The congestion events can be changed by modifying the urban infrastructure. But it is easy to conclude that such “off line” management of a transportation system will not give reasonable results in time.

That is why from control point of view the management of the traffic behavior with on-line strategies, adapting the control influences to dynamical urban behavior is the most appropriate solution. The application of traffic lights control at intersections is power solution to reduce the bottlenecks on the network. Such traffic lights control is a competitive strategy, which improves the traffic mobility [1]. This optimization allows decreasing the travel time, reducing the fuel consumptions, traffic emissions, and noise. The optimization of the traffic lights duration is regarded as power tool for regulation the traffic flows at the intersections.

The control influences on the traffic conditions are not so many: duration of the green lights and/or the split of the cycle; the cycle duration, which concerns the time for all lights including green, red and amber one; the phase between the traffic lights on sequential road intersections. These three types of control influences must be evaluated as solutions of appropriate control problem, which considers the current traffic state on urban intersections. The evaluation of the values of these control parameters can resolve problems like:

- Traffic congestion: reducing oversaturation at intersections;
- Traffic fluctuations: change of the current plan of the traffic lights to respond to considerable change of traffic intensity;
- Change of urban infrastructure due to road accident, closed street for human events, redirection of heavy vehicles.

A control problem, which optimizes only one isolated intersection cannot avoid the generation of bottlenecks. The congestions do not allow the transport to use the urban capacity because the overall traffic system degrades and the car motion is restricted on the overall urban network. Hence, the traffic light control has to be implemented on a network infrastructure and respectively to be applied control influences both with traffic lights, cycle durations and phase/time delays between the green lights achieving ‘green waves’ on important directions

Obviously, the definition of such complex optimization problem, where the transportation network comprises many intersections will generate high dimensions of the control problem and computational requirements will arise for real time solutions. Additionally, the increase of the control space will make the control problem also non-linear and hard for analytical definition.

The approach which is applied in this research is the integration and interconnection of small optimization problems in hierarchical order by means to define and to solve more complex control problem for traffic optimization. The small optimization problem has only one type of control influence: traffic lights green light duration or cycle duration. But making integration of these problems in hierarchical order the complicated one will have extended control space containing both types of control influences. This research targets the development of optimal control strategy, which implements simultaneously in optimal manner both types of control influences: the duration of the

This work is supported by Project KP06-H37/6 “Modelling and optimization of urban traffic in network of crossroads” with the Bulgarian Research Fund.

green lights and the duration of the traffic lights cycle. Such control problem has power to change the traffic behavior with extended set of control parameters, which is a benefit for these control processes. Such increase of the control space in this research is achieved by integration of control problem in a bi-level hierarchical procedure. The bi-level approach is implemented for the definition of a complex optimization problem [4, 18]. The problem is applied for the traffic optimization on arterial urban network in town of Sofia. The numerical simulations give advantages to the bi-level optimal control of the traffic lights and cycles in the urban transportation network in Sofia. A self-adaptive traffic signal control system adjusting the signal timing parameters in real time is considered.

II. Optimal Control of the Traffic Lights on Intersection

The design of the traffic lights stages at isolated intersection is standardized in industrial countries [2], [3]. For analytical overview about traffic signal control one can refer to [4]. The current practice of the application of the traffic engineering and traffic signals is presented in [5], [6]. A review of traffic control strategies, which are applied frequently in traffic systems are analyzed in [7], [8]. The optimization of the traffic signals under specific requirements is considered in [9]. A self-adaptive traffic control system, adjusting the signal timing parameters in real time is considered in [10]. This paper does not present formal definition of the optimization problem. Only the traffic lights are evaluated on procedural way. The cycle of timing is not considered as optimal problem solution.

Model predictive approach is also used in urban traffic management [11]. Store–and-forward modeling is applied and minimization of queue lengths in front of the intersection is performed. The duration of the green lights is estimated and controlled by simulation environment. The cycle duration and offset are not taken into consideration.

The store-and-forward modeling was applied for definition of optimal control problem in [12]. The solution of the problem gives plans only for the traffic lights.

Requirements towards public transport by optimization of traffic signals are considered in [13], [14]. Optimal signal settings are calculated only for the traffic lights. Simulation tools are also used for optimization of traffic lights [15].

Meta heuristics algorithms are also applied for intersection control [16].

All these methods try to solve optimization problem with objective value the traffic lights cycle duration or the green light duration that represent one-criterion optimization problem. But the formalism in optimization theory moves to the definition of more complex problems, based on hierarchical system theory. Hierarchically interconnected optimization problems allow to be extended the space of the control influences and parameters. Because the solution of such problems is quite complex, mainly the bi-level formalism is used for transportation problems.

III. Application of Bi-Level Optimization in Transportation

The bi-level approach is based on hierarchical integration of both optimization problems. The bi-level problems appeared firstly in game theory concerning the behavior and negotiations between leader and follower. Up to date bibliographical reviews about the bi-level and multilevel programming one can find in [17], [18]. The formal definition of a bi-level optimization problem is

$$\min_y F(x,y) \qquad (1)$$
$$x \equiv arg\begin{cases}\min_x f(x,y)\\ g(x,y) \le 0\end{cases}$$

$x \in X,\ y \in Y,\ F, f$ – scalar functions, g – set of constraints.

The solution of bi-level optimization is not an easy task, however it exists a tendency for its intensive application in different transportation problems.

In [19] goods must be transported and distributed from m sources to n destinations. But the transportation is divided hierarchically on several layers, which makes different priority to each destination. The bi-level objectives insist minimization of shipment time at each hierarchical level.

In [20] is defined a logistic bi-level problem. It concerns the distribution of a set of logistics centers and customer transportation costs. The lower optimization problem minimizes the customer costs together with satisfying their demands. The upper problem minimizes the cost of establishing the logistics centers.

A bi-level formalism is also used in public transportation [21]. It has been optimized the time interval between buses, considering the capacity of each bus. The user choice of routes is kept as low level problem. The bi-level transportation problem on upper level minimizes the travel costs, and on the lower level bus transportation scheme is considered [22].

Application of bi-level optimization in the domain of transportation policies one can find also in [23]; for network transportation - in [24]; for special kind of goods and their transportation in [25], [26]; for intermodal transport - [27] , [28] ; for locating of logistics in [29]. Timing considerations in transportation are addressed in [4], [7], [30], [31].

This analysis illustrates that the bi-level optimization formalism has entered in the transportation domain, giving benefits for the design, control, and decision making problems. The power of the bi-level formalism is based on the opportunity the optimization problem to accommodate more constraints, to extend the space of the optimization parameters, to apply in hierarchical order two goal functions. Short explanations of these features are given with the comparison between the bi-level problem (1) and the classical optimization problem (2):

$$\min_x f(x) \qquad (2)$$
$$g(x) \le 0\ .$$

The classical optimization problem (2) evaluates as optimal solutions the set of parameters **x.** The set of optimal solutions in the bi-level problem (1) is higher in comparison with (2) because (1) contains both sets **x** and **y**. The classical problem (2) takes in consideration small set of constraints, $g(x)$, while the bi-level problem considers bigger set of requirements, simultaneously $G(x,y)$ and $g(x,y)$. Finally, the bi-level problem keeps two criteria on their max/min values, $F(x,y)$ and $f(x,y)$, in comparison with the single criteria $f(x)$ for problem (2). Hence, the bi-level problem has bigger potential to consider more requirements in the optimization problem. This is one of the reasons the bi-level formalism to be applied intensively in different domains for optimization of resource allocation, control policies, design processes. That is why the described advantages are applied in this research for integration of optimization problems for increase of the control space of a transportation problem both with the traffic lights duration and the traffic light cycle.

IV. Transport Network Under Bi-Level Optimization

The urban network, which is considered, is one of the main arterial streets of Sofia town. This urban place very often generates transportation bottlenecks. Free cars, public transportation and pedestrian flows take place in this urban network. On close distances, five sequential intersections are controlled with traffic lights. After assessment of the behavior of the traffic flows, the intensities of inputs/outputs of vehicles, it has been identified the traffic light plans on this network, which currently keep constant parameters for the green light durations and cycle time. The goal of this research leads to evaluation of new values of the traffic lights parameters: green phases and cycle duration.

The topology of the transport network is presented in Figure 1. The main inputs of vehicles come from left and right direction of this arterial network. The crossing intersections reduce the main traffic direction and generate congestions inside of the arterial network. Such congestions make degradation for the main transport directions (left to right and opposite) and also increase the waiting vehicles, which cross the main street. Five regulated with traffic lights intersections have to support such plans for the green lights and cycle durations by means the queues and waiting vehicles into the arterial street to be prevented from congestions.

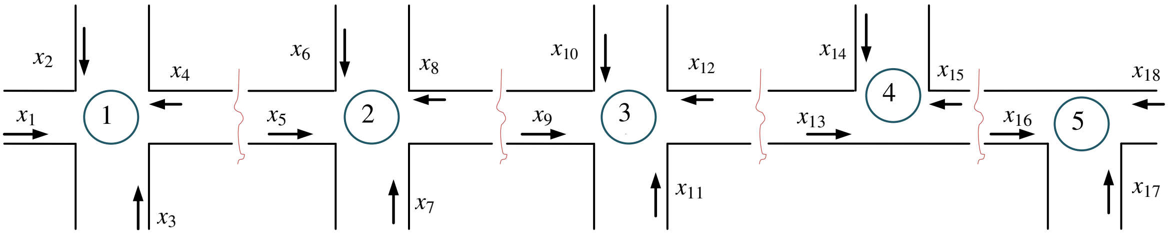


Fig.1 Transportation network under bi-level control

The bi-level optimization goal is to minimize the queue lengths in front of the crossroad sections and to keep such volumes of cars, which will not generate bottlenecks on the arterial direction of the network from left to right and opposite.

V. Definition of the Bi-Level Optimization Problem

The bi-level optimization problem is defined as integration if two optimization sub-problems, Fig.2. The low level sub-problem targets the minimization of the queue lengths **x** in front of the traffic lights of the intersections. This minimization is done by calculating the duration of the green lights **u** for all intersections.The lower level problem applies as predefined parameters the duration of the cycle durations **c.** The last are evaluated by the upper level optimization sub-problem. The upper-level optimization sub-problem takes as predefined parameters the queue lengths **x** and the green duration **u** and it evaluates optimal cycles **c** for maximization of the outflows of the arterial directions. Hence, the bi-level problem is constituted as two interconnected sub-problems.

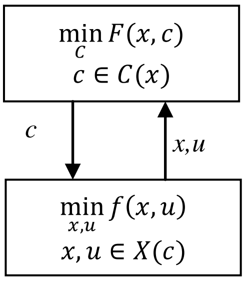


Fig.2. Hierarchical optimization of two optimization sub-problems

The solutions of the bi-level problem will give optimal values of the queue lengths, green light durations and the cycle durations for the traffic lights.

5.1. Definition of the low level sub-problem

The lower level problem aims to minimize the queue lengths. The common formulation of this problem is

$$\min_x f(x,u) \tag{3}$$

$$x,u \in X(c) \tag{4}$$

The goal function is chosen in a quadratic form

$$\min(x^2+u^2) \quad . \qquad (5)$$

The analytical description of the queue lengths of sub-problem (3)-(4) is based on the store and forward modeling. The general form of the store and forward modeling is given by relation (6)

$$\mathbf{x(k+1)=x(k) + x_{in} - x_{out}}, \qquad (6)$$

$$\mathbf{x_{out} = s^{(i)}\, u + \bar{s}^{(i)} u},$$

where k+1 is the current control cycle, **x** are the queue lengths in the previous k and the current $k+1$ control cycle; $\mathbf{x_{in}}$ are the inflows of vehicles at each intersection, $\mathbf{x_{out}}$ are the outgoing flows. The volume of outgoing vehicles $\mathbf{x_{out}}$ is managed by the duration of the green light of this intersection for direction i, where $\mathbf{s^{(i)}}$ is the saturation flow on this direction, $\mathbf{\bar{s}^{(i)}}$ is the saturation for the turning flows and **u** is the duration of the green light of the intersection.

We have to minimize all the queues in the network having in mind the store and forward model (6). The available traffic flows of the first crossroad section are graphically presented in Fig.3.

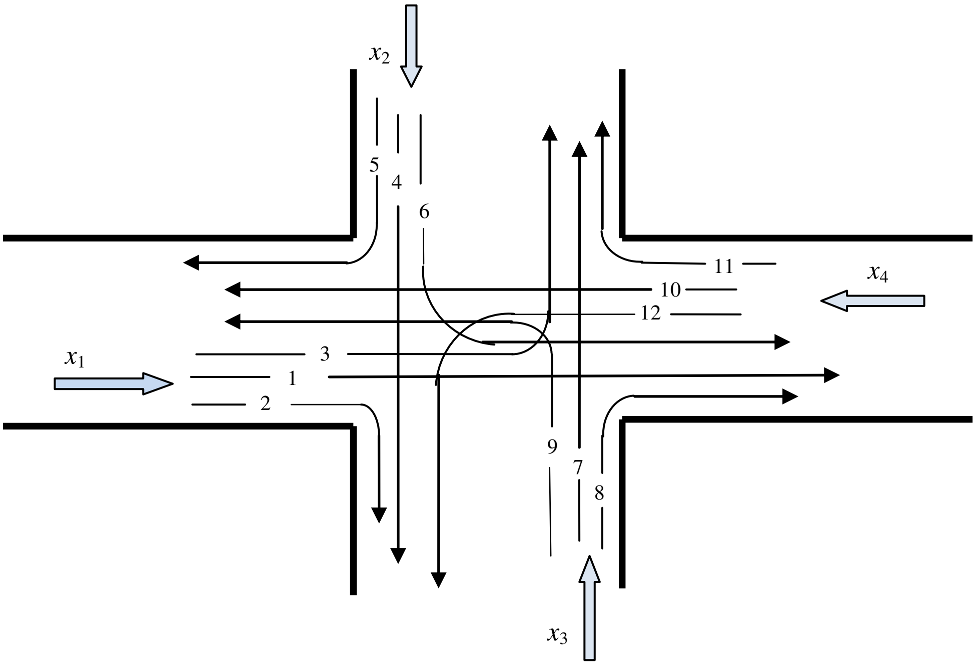


Fig.3. Traffic flows of the first crossroad section

For the first crossroad section (node) the duration of the green light in horizontal direction is noted as u_1 and in vertical direction u_2. The flow saturations in horizontal and vertical directions are s_1 and s_2. We suppose that the right curve has saturation $\bar{s_1}$ in horizontal and $\bar{s_2}$ in vertical direction. The levels of saturations for the left curves are respectively $\frac{1}{2}\bar{s_1}$ and $\frac{1}{2}\bar{s_2}$.

These levels of the saturation flows are defined according to the infrastructural dimensions. Following Fig.3, the first crossroad section manages twelve traffic flows, noted sequentially from 1 to 12 with corresponding outflows, computed according to the duration of the green lights **u** and the corresponding saturation flows in direction:

“1” → u_1s_1; “2” → $u_1\bar{s_1}$; “3” → $\frac{1}{2}u_1\bar{s_1}$;
“4” → u_2s_2; “5” → $u_2\bar{s_2}$; “6” → $\frac{1}{2}u_2\bar{s_2}$;
“7” → u_2s_2; “8” → $u_2\bar{s_2}$; “9” → $\frac{1}{2}u_2\bar{s_2}$;
“10” → u_1s_1; “11” → $u_1\bar{s_1}$; “12” → $\frac{1}{2}u_1\bar{s_1}$.

The same form of modelling is applied for the second crossroad section (node), where the green light duration in horizontal and vertical directions are noted respectively u_3 and u_4 and the horizontal and vertical levels of saturation flows are s_3 and s_4. The set of outflows are defined as:

“13” → u_3s_3; “14” → $u_3\bar{s_3}$; “15” → $\frac{1}{2}u_3\bar{s_3}$;
“16” → u_4s_4; “17” → $u_4\bar{s_4}$; “18” → $\frac{1}{2}u_4\bar{s_4}$;
“19” → u_4s_4; “20” → $u_4\bar{s_4}$; “21” → $\frac{1}{2}u_4\bar{s_4}$;
“22” → u_3s_3; “23” → $u_3\bar{s_3}$; “24” → $\frac{1}{2}u_3\bar{s_3}$.

For the third crossroad section (node) the corresponding parameters are u_5 and u_6 for the green phase durations and s_5 and s_6 for the saturation flows. The traffic outflows are noted as:

“25” → u_5s_5; “26” → $u_5\bar{s_5}$; “27” → $\frac{1}{2}u_5\bar{s_5}$;
“28” → u_6s_6; “29” → $u_6\bar{s_6}$; “30” → $\frac{1}{2}u_6\bar{s_6}$;
“31” → u_6s_6; “32” → $u_6\bar{s_6}$; “33” → $\frac{1}{2}u_6\bar{s_6}$;
“34” → u_5s_5; “35” → $u_5\bar{s_5}$; “36” → $\frac{1}{2}u_5\bar{s_5}$.

For the fourth node the notation used are (u_7 and u_8, s_7 and $\overline{s_8}$) and the outgoing traffic flows are:

“37” → u_7s_7; “38” → $\frac{1}{2}u_7\overline{s_7}$;
“39” → $u_8\overline{s_8}$; “40” → $\frac{1}{2}u_8\overline{s_8}$;
“41” → u_7s_7; “42” → $u_7\overline{s_7}$.

The last fifth node applies notations (u_9 and u_{10}, s_9 and s_{10}) with outgoing traffic flows :

“43” → u_9s_9; “44” → $u_9\overline{s_9}$;
“45” → $u_{10}\overline{s_{10}}$; “46” → $\frac{1}{2}u_{10}\overline{s_{10}}$;
“47” → u_9s_9; “48” → $\frac{1}{2}u_9\overline{s_9}$.

The level of the queue lengths (x) in front of the traffic lights is changed for each traffic light control cycle. Applying the store-and-forward model the levels of the queue lengths for the current control cycle depend from the residual queues from the previous control cycle (x_{io}), i=1,…,48, increased by the incoming flows (x_{in}), decreased by the outgoing flows on each direction (straight ahead, right and left curves). Applying the relations of the store-and forward modeling for the urban network from Fig.1, the relations of the queues according to (6) define a set of 18 inequalities, requiring control policy, which results in less queue lengths for the current control cycle:

$$\begin{array}{l}
x_1 \le x_{1o} + x_{1in} - u_1s_1 - u_1\overline{s_1} - \frac{1}{2}u_1\overline{s_1} \qquad (7)\\
x_2 \le x_{2o} + x_{2in} - u_2s_2 - 1.5u_2\overline{s_2}\\
x_3 \le x_{3o} + x_{3in} - u_2s_2 - 1.5u_2\overline{s_2}\\
x_4 \le x_{4o} + u_3s_3 + 1.5u_4\overline{s_4} - u_1s_1 - 1.5u_1\overline{s_1}\\
x_5 \le x_{5o} + u_1s_1 + 1.5u_2\overline{s_2} - u_3s_3 - 1.5u_4\overline{s_4}\\
x_6 \le x_{6o} + x_{6in} - u_4s_4 - 1.5u_4\overline{s_4}\\
x_7 \le x_{7o} + x_{7in} - u_4s_4 - 1.5u_4\overline{s_4}\\
x_8 \le x_{8o} + u_5s_5 + 1.5u_6\overline{s_6} - u_3s_3 - 1.5u_3\overline{s_3}\\
x_9 \le x_{9o} + u_3s_3 + 1.5u_4\overline{s_4} - u_5s_5 - 1.5u_5\overline{s_5}\\
x_{10} \le x_{10o} + x_{10in} - u_6s_6 - 1.5u_6\overline{s_6}\\
x_{11} \le x_{11o} + x_{11in} - u_6s_6 - 1.5u_6\overline{s_6}\\
x_{12} \le x_{12o} + u_7s_7 + u_8\overline{s_8} - u_5s_5 - 1.5u_5\overline{s_5}\\
x_{13} \le x_{13o} + u_5s_5 + 1.5u_6\overline{s_6} - u_7s_7 - 0.5u_7\overline{s_7}\\
x_{14} \le x_{14o} + x_{14in} - 1.5u_8\overline{s_8}\\
x_{15} \le x_{15o} + u_9s_9 + 0.5u_{10}\overline{s_{10}} - u_7s_7 - u_7\overline{s_7}\\
x_{16} \le x_{16o} + u_7s_7 + 0.5u_8\overline{s_8} - u_9s_9 - u_9\overline{s_9}\\
x_{17} \le x_{17o} + x_{17in} - 1.5u_{10}\overline{s_{10}}\\
x_{18} \le x_{18o} + x_{18in} - u_9s_9 + 0.5u_9\overline{s_9}
\end{array}$$

In matrix form, the set of inequalities (7) is written as

$$\mathbf{A_1x} + \mathbf{A_2u} \le \mathbf{B} \qquad (8)$$

The lower level sub-problem is defined in the form

$$\begin{array}{l}
\min(\mathbf{x}^{\mathrm{T}}\mathbf{x}+\mathbf{u}^{\mathrm{T}}\mathbf{u}) \qquad (9)\\
\mathbf{A_1x} + \mathbf{A_2u} \le \mathbf{B}\ ,
\end{array}$$

where minimization of the queues **x** is considered in the goal function, $\mathbf{A_1}$, $\mathbf{A_2}$, **B** are corresponding matrices, derived from (7).

For the topology of the transportation network of Figure 1, the relations (7) presented in form (8) give the set of constraints for the low level sub-problem as

$$\begin{array}{l}
x_1 + (s_1 + 1.5\overline{s_1}\ )u_1 \le x_{1o} + x_{1in} \qquad (10)\\
x_2 + (s_2 + 1.5\overline{s_2})u_2 \le x_{2o} + x_{2in}\\
x_3 + (s_2 + 1.5\overline{s_2})u_2 \le x_{3o} + x_{3in}\\
x_4 + (s_1 + 1.5\overline{s_1})u_1 - s_3u_3 - 1.5\overline{s_4}u_4 \le x_{4o}\\
x_5 - s_1u_1 - 1.5\overline{s_2}u_2 + s_3u_3 + 1.5\overline{s_4}u_4 \le x_{5o}\\
x_6 + (s_4 + 1.5\overline{s_4})u_4 \le x_{6o} + x_{6in}\\
x_7 + (s_4 + 1.5\overline{s_4})u_4 \le x_{7o} + x_{7in}\\
x_8 + (s_3 + 1.5\overline{s_3})u_3 - s_5u_5 - 1.5\overline{s_6}u_6 \le x_{8o}\\
x_9 - u_3s_3 - 1.5\overline{s_4}u_4 + (s_5 + 1.5\overline{s_5})u_5 \le x_{9o}\\
x_{10} + (s_6 + 1.5\overline{s_6})u_6 \le x_{10o} + x_{10in}\\
x_{11} + (s_6 + 1.5\overline{s_6})u_6 \le x_{11o} + x_{11in}\\
x_{12} + (s_5 + 1.5\overline{s_5})u_5 - s_7u_7 - \overline{s_8}u_8 \le x_{12o}\\
x_{13} - s_5u_5 - 1.5\overline{s_6}u_6 + (s_7 + 0.5\overline{s_7})u_7 \le x_{13o}\\
x_{14} + 1.5\overline{s_8}u_8 \le x_{14o} + x_{14in}\\
x_{15} + (s_7 - \overline{s_7})u_7 - s_9u_9 - 0.5\overline{s_{10}}u_{10} \le x_{15o}\\
x_{16} - s_7u_7 - 0.5\overline{s_8}u_8 + (s_9 + \overline{s_9})u_9 \le x_{16o}\\
x_{17} + 1.5\overline{s_{10}}u_{10} \le x_{17o} + x_{17in}\\
x_{18} + (s_9 + 0.5\overline{s_9})u_9 \le x_{18o} + x_{18in}\\
u_1 + u_2 = \alpha_1C_1\\
u_3 + u_4 = \alpha_2C_2\\
u_5 + u_6 = \alpha_3C_3\\
u_7 + u_8 = \alpha_4C_4\\
u_9 + u_{10} = \alpha_5C_5
\end{array}$$

5.2 *Definition of the Upper level sub-problem*

The upper level sub-problem is defined in a way that the solutions **x, u** of the lower problem participate as parameters in its constraints and/or goal function. The upper level sub-problem targets the maximization of the outflows from the arterial directions (left and right of Fig.1) by evaluating the duration of the time cycle of the traffic lights **c.** From practical and realistic requirements, the time cycle is constrained between upper $\mathbf{C_{max}}$ and lower $\mathbf{C_{min}}$ bounds.

The goal function is chosen in quadratic form to keep minimal cycle duration, which benefits the equilibrium distribution of waiting vehicles in overall network. The abnormal increase of the traffic cycles will generate congestion of the corresponding crossing sections.

Having the solutions of the lower level optimization problem x and u, the upper level optimization problem becomes analytically defined. The arguments of the problem are the traffic cycle durations c_i, i=1,…,5, denoted below as vector **y**:

$$\begin{array}{l}
\min_{\mathbf{y}}\mathbf{y}^{\mathbf{T}}\mathbf{y} \qquad (11)\\
\mathbf{C_{min}} \le \mathbf{y} \le \mathbf{C_{max}}\\
u_1 + u_2 = 0.9y_1\\
u_3 + u_4 = 0.9y_2\\
u_5 + u_6 = 0.9y_3\\
u_7 + u_8 = 0.9y_4\\
u_9 + u_{10} = 0.9y_5
\end{array}$$

It is assumed the 10% of the traffic cycle **c** or **y** is used for the amber light. The resulting 90% of the cycle duration is used by the green lights in the both crossing directions.

The solutions of the upper level optimization problem give the duration of the cycles **c**=(c_i, i=1,…,5). These values are given to the lower-level problem where they participate as parameters for the definition of problem (10) for the new control cycle of the traffic lights.

VI. Numerical Simulation of the Bi-Level Control

Problems (9) and (11) evaluate the duration of the green lights **u** and the cycles **c** for one control cycle. The sequentially solutions of these bi-level problems will give the dynamics of the control policy with the changes of the queue lengths and the outgoing flows. The numerical simulation uses as initial data the current estimates of the traffic behavior, which were performed for this urban network for peak time from 16:00-18:00 for three weeks. Comparisons have been done between the current applied plan of traffic lights and the numerically evaluations with the bi-level optimization.

The bi-level problem have been solved iteratively and on each iteration the obtained solutions of queue lengths, green light durations, and time cycles are used as input parameters for the next iteration. The numerical simulation was performed in MATLAB environment. Inside it has been install the bi-level toolbox YALMIP [32], which is a toolbox, implementing bi-level calculations. Particularly, the “solvebilevel” function was called in repeatedly way.

For the urban network it has been estimated values for the inflows as:

x_{1in} = 260/3600; x_{2in} = 54/3600; x_{3in} = 206/3600; x_{6in} = 50/3600; x_{7in} = 90/3600; x_{10in} = 96/3600
x_{14in} = 270/3600; x_{17in} = 320/3600; x_{18in} = 250/3600;
$x_{i\,o}$ = 2, i=1,…,18.

The current plan, applied for controlling the five intersections, has the following cycle durations
c_1 = 60; c_2 = 55; c_3 = 55; c_4 = 70; c_5 = 60;
$C_{min} = 30$; $C_{max} = 120$.
The green lights are fixed to values

$\mathbf{u}^{\mathbf{T}}$ = [27; 27; 25; 25; 25; 25; 32; 32; 27; 27].

The saturation flows are assessed according to the urban infrastructure with values

$\mathbf{s}^{\mathbf{T}}$=[500/3600; 160/3600; 550/3600; 300/3600; 550/3600; 160/3600; 550/3600; 420/3600; 500/3600; 420/3600];

$\bar{\mathbf{s}}^{\mathbf{T}}$=[200/3600; 160/3600; 157/3600; 148/3600; 250/3600; 150/3600; 490/3600; 240/3600; 240/3600; 20/3600].

The dynamical changes per time cycle of the 18 queue lengths **x** are presented graphically in Figures 4 till 22. The numerical simulations applied 10 control steps of the traffic network by solving the bi-level problem (9)-(11). It has been estimated that benefits of the total traffic behavior comes up to the 5-th step, because the majority of the vehicle queues vanish. That is the reason the illustrations of the results of the bi-level control in graphical mode to be up to 5-th control step. The horizontal axes of the figures below have meaning of sequence of the control steps.

A comparison between the queue lengths, resulting by bi-level control (in blue and solid line) and with the application of existing “fixed” time plans without optimization (red and dashed line) are presented in Figures 4 to 22.

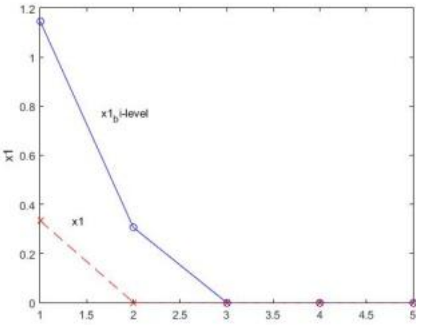


Fig.4 Queue x_1

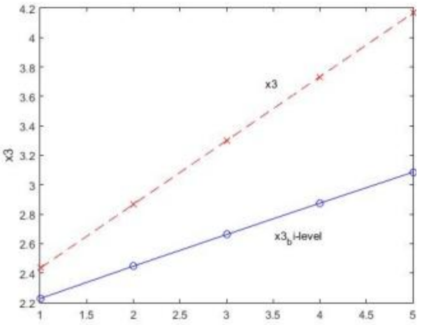


Fig.6 Queue x_3

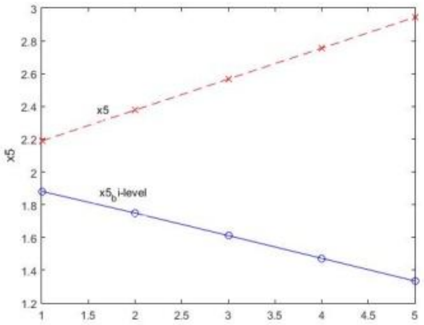


Fig.8 Queue x_5

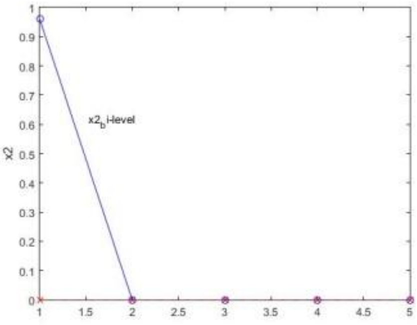


Fig.5 Queue x_2

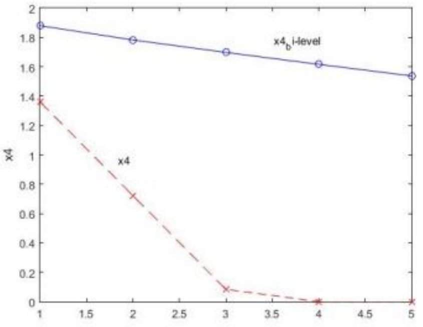


Fig.7 Queue x_4

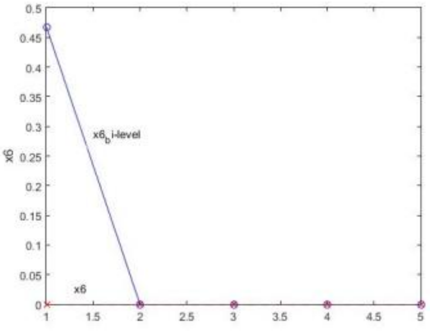


Fig.9 Queue x_6

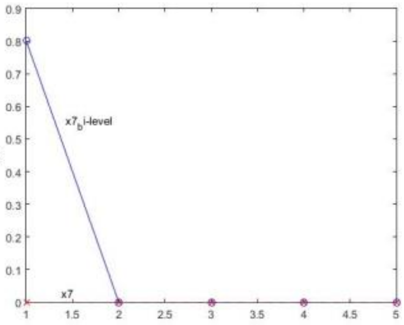


Fig.10 Queue x_7

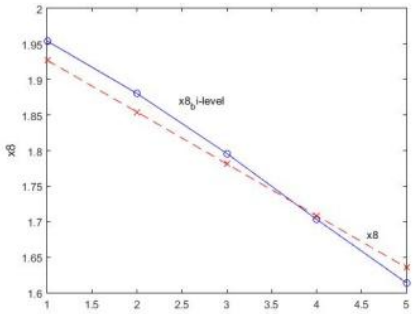


Fig.11 Queue x_8

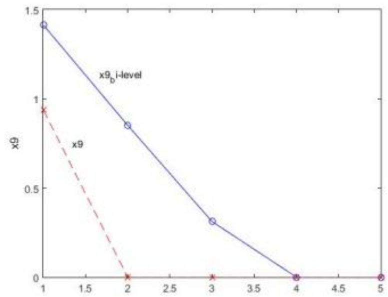


Fig.12 Queue x_9

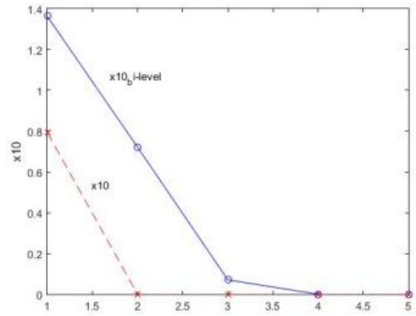


Fig.13 Queue x_{10}

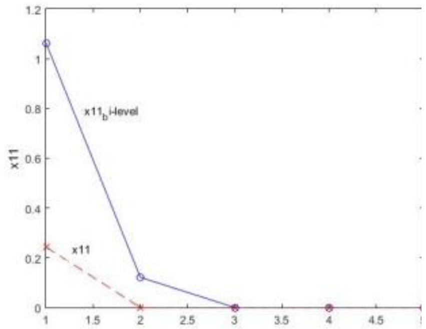


Fig.14 Queue x_{11}

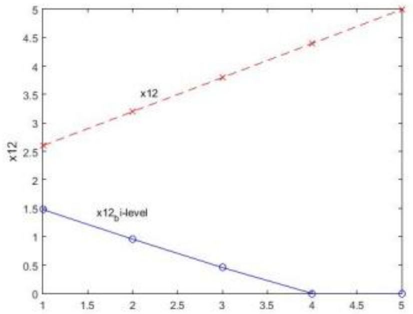


Fig.15 Queue x_{12}

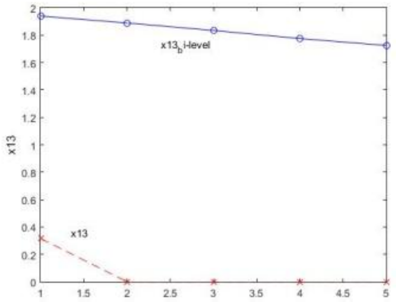


Fig.16 Queue x_{13}

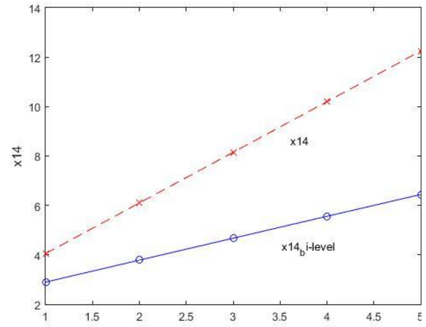


Fig.17 Queue x_{14}

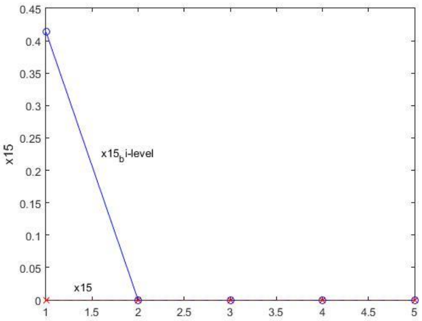


Fig.18 Queue x_{15}

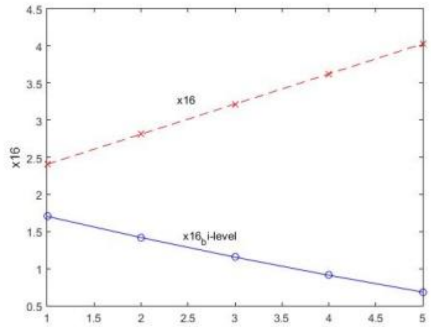


Fig.19 Queue x_{16}

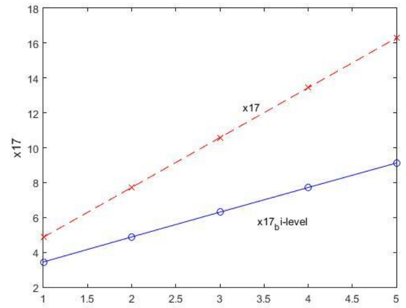


Fig.20 Queue x_{17}

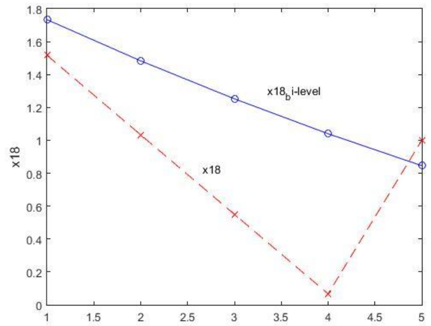


Fig.21 Queue x_{18}

The graphical assessment proves that the bi-level control optimization gives considerably better results in comparison with the existing "fixed" case of traffic control. All of the queue lengths keep decreasing behavior. It is obvious that the "fixed" control, currently applied on places, generates increasing character of the queue lengths, Figures 8, 9, 15 and 21. The corresponding queue lengths x_5 , x_{12}, x_{16} and x_{18} increase but with the application of the bi-level control the same queues have decreasing character. This is a benefit, which comes from this new bi-level formulation of the traffic control.

On Fig.22 it is presented an integral graphical comparison for the queue lengths for all queues with bi-level (blue solid lines) and the current "fixed" control (red dashed lines) strategies. The horizontal axis represents the traffic directions. It is evident that the "fixed" control strategy keeps higher levels of the queue lengths for the all 18 directions. The bi-level control maintains lower values of queues.

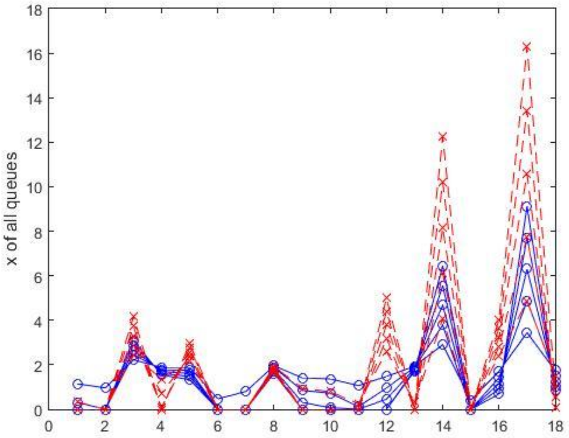


Fig.22 Comparison of all 18 queues x_i

Because the target of the bi-level control is to give preferences to the arterial directions (West to East and opposite) the next Figures 23 and 24 illustrate the case of the application of bi-level control. It is graphically proved that in both directions the queue lengths of the arterial directions have decreasing character and this prevents the occurrence of bottlenecks and congestions.

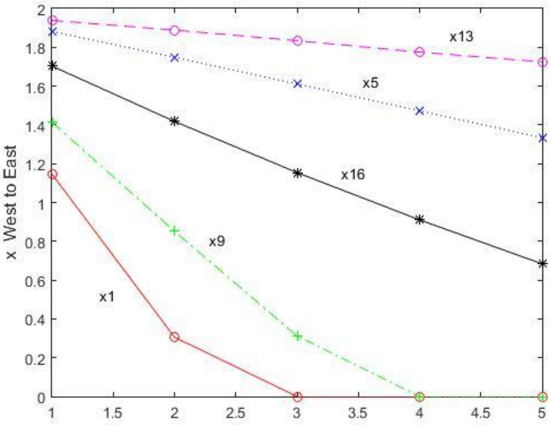


Fig.23 Queues x_i from West to East

An integral comparison between the bi-level control strategy and the "fixed" time one is given in Fig.25. All vehicles, which are waiting in queues of this urban network are calculated as a sum of each individual queues. The bi-level control strategy gives total value of waiting vehicles in blue solid line, while the "fixed" time strategy is in red dashed line. It is evident that the "fixed" strategy has rapid increasing character, which explains the current situation of frequent congestions and bottlenecks on the arterial directions. In opposite, the bi-level policy decreases the total amount of waiting vehicles, which benefit the arterial outflows of the network.

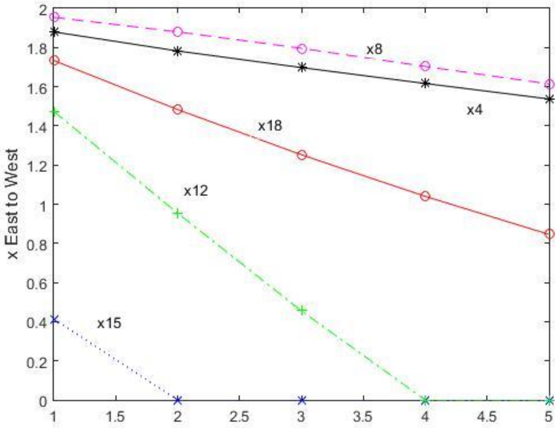


Fig.24 Queues x_i from East to West

.

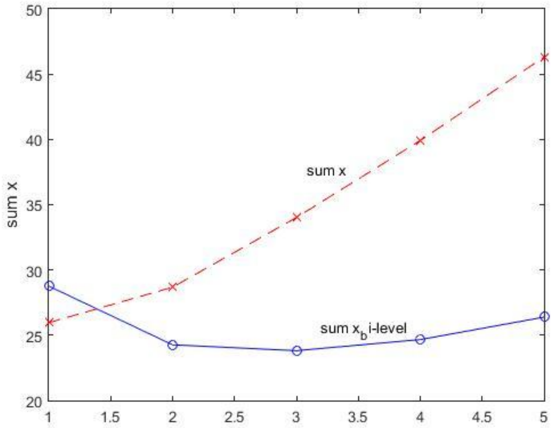


Fig.25 Comparison of the sum of all queues

For illustration purposes in Fig.26 are given the changes of the green lights durations, evaluated by the bi-level control. It is seen that they do not have constant values and the different crossroad sections apply different duration for the different control iterations. For the case of "fixed" time control the green lights per section are kept constant with values u_1=27; u_2=27; u_3=25; u_4=25; u_6=25; u_7=25; u_8=32; u_9=32; u_{10}=27; u_{11}=27. The bi-level control demonstrates sensibility and adaptive features for the evaluation of the green lights according to the current values of waiting vehicles. This is a benefit for the bi-level control.

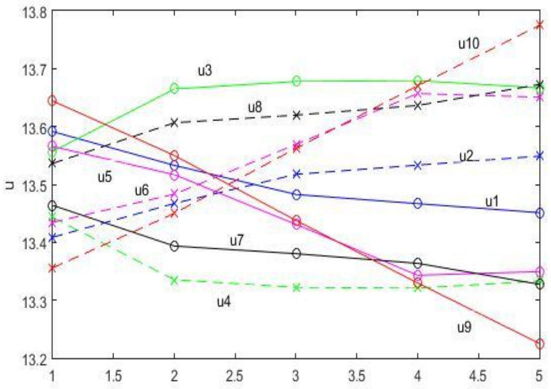


Fig.26 Optimization results of green light durations

VI. Conclusions

An arterial transportation network with five crossroad sections is modeled in order to improve the traffic behavior by bi-level formalism. This hierarchical modeling allows extending the control space not only with the green light durations but with the duration of the control cycle of the traffic lights. The increased set of control influences results in better behavior of the traffic state: considerable reduction of the queue lengths in the total transportation network as to give priority in arterial direction. Thus, the bi-level optimal control allows reducing the appearance of events as bottlenecks and traffic congestions.

The bi-level modeling allows by integration of smaller optimization problems to have simultaneously two important control influences to the traffic: green lights ant time cycles. This research applies the store-and-forward modeling for the definition of the low optimization problem. The upper problem takes care of the outgoing flows by optimizing the cycle of the traffic flows. Thus, the application of the hierarchical approach benefits the control optimization procedure by optimizing simultaneously two important traffic control influences. The presented numerical simulation results illustrate the bigger potential and positive results of the bi-level control strategy. The comparisons with the currently established "fixed" plan control prove the benefit of the bi-level optimization control.

References

[1] B. Park and J. D. Schneeberger, *Evaluation of Traffic Signal Timing Optimization Methods Using a Stochastic and Microscopic Simulation Program*, Virginia Transportation Research Council, 2003.

[2] W. H. Kraft, W. S. Homburger, and J. L. Pline, *Traffic Engineering Handbook*, Washington USA, Institute of Transportation Engineers, 2009.

[3] P. Koonce, L. Rodegerdts, K. Lee, S. Quayle, S. Beaird, C. Braud, J. Bonneson, P. Tarnoff, and T. Urbanik, *Traffic Signal Timing Manual.* Washington: Federal Highway Administration, 2008.

[4] K. Han, Y. Sun, H. Liu, T. L. Friesz, and T. Yao, "A bi-level model of dynamic traffic signal control with continuum approximation," *Transportation Research Part C*, vol.55, pp. 409-431, 2015, http://www.sciencedirect.com/science/article/pii/S0968090X15001266, https://www.academia.edu/12549962/A_bi-level_model_of_dynamic_traffic_signal_control_with_continuum_approximation

[5] L. Li, D. Wen, and D. Yao, "A survey of traffic control with vehicular communications," *IEEE Transactions on Intelligent Transportation Systems*, vol.15, no 1, pp. 425-432, 2014,. DOI: 10.1109/TITS.2013.2277737 https://www.researchgate.net/publication/260720276_A_Survey_of_Traffic_Control_With_Vehicular_Communications

[6] R. P. Roess, E. S. Prassas, and W. R. McShane, *Traffic Engineering*, 5th ed. Hoboken, NJ Pearson Education, 2019, ISBN-10:0-13-459971-3, ISBN-13:978-0-13-459971-7, https://www.pearsonhighered.com/assets/preface/0/1/3/4/0134599713.pdf

[7] H. Wei, G. Zheng, V. Gayah, and Z. Li, A survey on traffic signal control methods. Cornell University, 2020, https://arxiv.org/pdf/1904.08117.pdf,

[8] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang, Review of road traffic control strategies, in *Proc. IEEE 91*, 12, pp. 2043–2067, 2003.

[9] E. Eriskin, S. Karahancer, S. Terzi, and M. Saltan, Optimization of traffic signal timing at oversaturated intersections using elimination pairing system. 10th International Scientific Conference Transbaltica, Transportation Science and Technology, Procedia Engineering 187, pp. 295 – 300, 2017, doi: 10.1016/j.proeng.2017.04.378 , https://www.sciencedirect.com/science/article/pii/S1877705817319082

[10] Y. Wang , X. Yang, H. Liang , and Y. Liu, "A Review of the Self-Adaptive Traffic Signal Control System Based on Future Traffic Environment," *J. of Advanced Transportation,* vol. 2018, Article ID 1096123, 12 pages, https://doi.org/10.1155/2018/1096123

[11] T. Tettamanti, I. Varga, and T. Peni, "MPC in urban traffic management," *Model predictive control*, Ed.T. Zheng, IntechOpen, 2010 DOI: 10.5772/9922. Available from: https://www.intechopen.com/books/model-predictive-control/mpc-in-urban-traffic-management

[12] K. Aboudolas, M. Papageorgiou, and E. Kosmatopoulos, "Store-and-forward based methods for the signal control problem in large-scale congested urban road networks," *Transportation Research Part C*, vol.1, pp. 163–174, 2009, doi:10.1016/j.trc.2008.10.002

[13] R. Scheffle and M. Strehler, "Optimizing Traffic Signal Settings for Public Transport Priority," 17th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS), 2017. G. D'Angelo and T. Dollevoet; Eds, Article No. 9; pp. 9:1–9:15, DOI:10.4230/OASIcs.ATMOS.2017.9

[14] V. Ivanov, "Monitoring of urban road transport,".*Proc. of Intern conf Automatics and Informatics*, 2017, pp. 135-141, ISSN:1313-1850.

[15] K. N. Hewage and J. Y. Ruwanpura, „Optimization of traffic signal light timing using simulation", in *Proc. 2004 Winter Simulation Conference*, R. G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, Eds, 2004, pp.1428-1433, DOI: 10.1109/WSC.2004.1371482 ·

[16] A. Jamal, M. T. Rahman, H. M. Al-Ahmadi, I. Ullah, and M. Zahid. Intelligent intersection control for delay optimization: using meta-heuristic search algorithms. 2020, https://www.mdpi.com/2071-1050/12/5/1896/pdf

[17] L. N. Vicente and P. H. Calamai, "Bilevel and multilevel programming: A bibliography review," *J Glob Optim,* vol. 5, pp. 291–306, 1994, https://doi.org/10.1007/BF01096458

[18] B. Colson, P. Marcotte, and G. Savard, "An overview of bilevel optimization," *J. Ann Oper Res* vol. 153, pp. 235–256, 2007, DOI 10.1007/s10479-007-0176-2, https://www.iro.umontreal.ca/~marcotte/ARTIPS/AOR2007.pdf

[19] S. A. Khandelwal and M. C. Puri, "Bilevel time minimizing transportation problem," *J. Discrete Optimization*, volume 5, no 4, pp. 714-723, November 2008, https://doi.org/10.1016/j.disopt.2008.04.004

[20] H. Sun, Z. Gao, and J. Wu, "A bi-level programming model and solution algorithm for the location of logistics distribution centers," *J. Applied Mathematical Modelling*, vol. 32, no 4, pp. 610-616, April 2008, https://doi.org/10.1016/j.apm.2007.02.007

[21] A. Arizti, A. Mauttone, and M. E. Urquhart, "A bilevel approach to frequency optimization in public transportation systems," in *18th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS 2018)*, .65, pp. 7:1-7:13, ISBN 978-3-95977-096-5, ISSN 2190-6807,

DOI:10.4230/OASIcs.ATMOS.2018.7, http://drops.dagstuhl.de/opus/volltexte/2018/9712/

[22] J. Hao, X. Liu, X. Shen, and N. Feng, “Bilevel Programming Model of Urban Public Transport Network under Fairness Constraints,” in *Discrete Optimization for Dynamic Systems of Operations Management in Data-Driven Society*, 2019, https://doi.org/10.1155/2019 /2930502,

[23] M. Patriksson, “Robust bi-level optimization models in transportation science,” *Philosophical transactions of royal Society A*, vol. 366, no 1872, pp. 1931-1940, 2008, http://doi.org/10.1098/rsta.2008.0007

[24] R. Z. Farahania, E. Miandoabchib, W. Y. Szetoc, and H. Rashidid, “A review of urban transportation network design problems,” *European Journal of Operational Research*, vol. 229, no 2, September 2013, Pages 281-302, doi: 10.1016/j.ejor.2013.01.001

[25] X. Jia, R. He, C. Zhang, and H. Chai, "A Bi-Level Programming Model of Liquefied Petroleum Gas Transportation Operation for Urban Road Network by Period-Security," *Sustainability, MDPI, Open Access Journal*, vol. 10, no 12, pp. 1-20, December 2018, https://ideas.repec.org/a/gam/jsusta/v10y2018i12p4714-d189583.html

[26] K. Moad, J. François , J. P. Bourrières , L. Lebel, and M. Vuillermo, “A bi-level decision model for timber transport planning”, *6th Int conf Information systems, logistics and supply chain*, 2016 Bordeaux, http://ils2016conference.com/wp-content/uploads/2015/03/ILS2016_TD02_3.pdf

[27] C. Tawfik, S. Limbourg, “Bilevel optimization in the context of intermodal pricing: state of art,” *Transportation Research Procedia*, vol. 10, pp. 634 – 643, 2015, https://orbi.uliege.be/bitstream/2268 /185274/1/1-s2.0-S2352146515002045-main.pdf, doi: 10.1016/j.trpro .2015.09.017

[28] A. Sinha, P. Malo, and K. Deb, *Transportation Policy Formulation as a Multi-objective Bilevel Optimization Problem*, 2015, https://www.egr.msu.edu/~kdeb/papers/c2015009.pdf

[29] C. Lu, S. Yan, H. Ko and H. Chen, "A bilevel model with a solution algorithm for locating weigh-in-motion stations," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 380-389, Feb. 2018, https://ieeexplore.ieee.org/document/7922613

[30] R. G. Ródenas, M. L. L.García, M. T. S. Rico, and J. A. L. Gómez, “A bilevel approach to enhance prefixed traffic signal optimization,” *J. Engineering Applications of Artificial Intelligence*, vol. 84, pp. 51-65, September 2019, https://doi.org/10.1016/j.engappai.2019.05.017

[31] S. Goel, S. F. Bush, and C. Gershenson, Self-Organization in Traffic Lights: Evolution of Signal Control w ith Advances in Sensors and Communications, June 2017, https://www.researchgate.net/ publication/319271996_Self-Organization_in_Traffic_Lights_Evolution_of_Signal_Control_with_Advances_in_Sensors_and_Communications

[32] https://yalmip.github.io/

DOI: 10.15439/2020F110
ISSN 2300-5963 ACSIS, Vol. 21

A New Optimized Stochastic Approach for Multidimensional Integrals in Machine Learning

Venelin Todorov
Bulgarian Academy of Sciences
Institute of Mathematics and Informatics
ul. G. Bonchev 8, 1113 Sofia, Bulgaria
Bulgarian Academy of Sciences
Institute of Information and Communication Technologies
ul. G. Bonchev 25A, 1113 Sofia, Bulgaria
Email: vtodorov@math.bas.bg,venelin@parallel.bas.bg

Stoyan Apostolov
Sofia University
Faculty of Mathematics and Informatics
1764 Sofia, Bulgaria
Email: stoyanrapostolov@gmail.com

Ivan Dimov
Bulgarian Academy of Sciences
Institute of Information and Communication Technologies
ul. G. Bonchev 25A, 1113 Sofia, Bulgaria
Email: ivdimov@bas.bg

Stefka Fidanova
Bulgarian Academy of Sciences
Institute of Information and Communication Technologies
ul. G. Bonchev 25A, 1113 Sofia, Bulgaria
Email: stefka@parallel.bas.bg

***Abstract*—Stochastic techniques have been developed over many years in a range of different fields, but have only recently been applied to the problems in machine learning. A fundamental problem in this area is the accurate evaluation of multidimensional integrals. An introduction to the theory of the stochastic optimal generating vectors has been given. A new optimized lattice sequence with a special choice of the optimal generating vector has been applied to compute multidimensional integrals up to 30-dimensions. Clearly, the progress in the area of machine learning is closely related to the progress in reliable algorithms for multidimensional integration.**

I. Introduction

MONTE Carlo methods are suitable for mathematical modelling of multi-dimensional problems [10], since their computational complexity increases polynomially, but not exponentially with the dimensionality [2]. A general problem in neural networks and machine learning is the accurate evaluation of multidimensional integrals. In 2011 Shaowei Lin in his works [4], [5] consider the problem of evaluating multidimensional integrals in Bayesian statistics which are used in neural network and machine learning. We will primarily be interested in two kinds of integrals. The first has the form

$$\int_{\Omega} p_1^{u_1}(x)\dots p_s^{u_s}(x)dx, \tag{1}$$

where $\Omega \in \mathcal{R}^s$, $x = (x_1, \dots, x_s)$, $p_i(x)$ are polynomials and u_i are integers. The second kind of integrals has the form

$$\int_{\Omega} e^{-Nf(x)}\phi(x)dx, \tag{2}$$

where $f(x)$ and $\phi(x)$ are s-dimensional polynomials and N is a natural number. The asymptotics of such integrals is well understood for models in machine learning, but little was known for singular models until a breakthrough in 2001 [9].

II. QMC methods based on lattice rules

Lattice point sets are a special type of low-discrepancy stochastic sequences based on the use of deterministic sequences instead of random sequences [8]. The monographs of Sloan and Kachoyan [7] and Hua and Wang [3] provide comprehensive expositions on the theory of integration lattices.

In our study we will use the following a particular rank-1 lattice sequence [8]:

$$\mathbf{x}_k = \left\{ \frac{k}{N}\mathbf{z} \right\}, \ k = 1, \dots, N, \tag{3}$$

where N is an integer, $N \geq 2$, $\mathbf{z} = (z_1, z_2, \dots z_s)$ is an integer vector modulo N of dimensionality s called a generator of the set and $\{z\}$ denotes the fractional part of z. We denote by $P_N = \{x_1, x_2, \dots, x_N\}, x_i \in [0, 1)^s$ the integration nodes of the formula.

Definition 1: [7] We say that $f(x)$ belongs to the class of functions $E_s^{\alpha}(c)$ for $\alpha > 1$ and $c > 0$, if f is a periodic

Stoyan Apostolov is supported by the Bulgarian National Science Fund under Young Scientists Project KP-06-M32/2 - 17.12.2019 ”Advanced Stochastic and Deterministic Approaches for Large-Scale Problems of Computational Mathematics”. Venelin Todorov is supported by the National Scientific Program ”Information and Communication Technologies for a Single Digital Market in Science, Education and Security (ICT in SES)”, contract No DO1-205/23.11.2018, financed by the Ministry of Education and Science in Bulgaria and by the National Scientific Program for young scientists and post doctoral scientists of Ministry of Education and Science 2020-2021. The work is also supported by the Bulgarian National Science Fund under Project DN 12/5-2017 ”Efficient Stochastic Methods and Algorithms for Large-Scale Problems”.

function with period 1 for every of its components $x_i, i = 1, 2 \ldots, s$, defined over the unit cube $[0,1]^s$ and its Fourier coefficients satisfy the following inequalities:

$$|a(m)| < \frac{c}{(\overline{m}_1 \ldots \overline{m}_s)^{\alpha}}, \tag{4}$$

where

$$\overline{m} = \begin{cases} |m|, & |m| \neq 0, \\ 0, & m = 0, \end{cases}$$

and the constant c does not depend on $m_1, \ldots, m_s$.

The discrepancy and the "worst case" error are two important characteristics for the quality of the lattice sequences.

Definition 2: Consider the point set $X = \{x_i \mid i = 1, 2, \ldots N\}$ in $[0,1)^s$ and $N > 1$. Denote by $x_i = (x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(s)})$ and $J(v) = [0, v_1) \times [0, v_2) \times \ldots \times [0, v_s)$. Then the discrepancy of the set is defined as

$$D(N) := \sup_{0 \leq v_j \leq 1} \left| \frac{\#\{x_i \in J(v)\}}{N} - \prod_{j=1}^{s} v_j \right|.$$

Definition 3: For $f \in E_{\alpha}^{s}(c)$ the worst case error is defined as [8]

$$P_{\alpha}(z, N) = \sum_{z.a \equiv 0 \ (modN), a \neq 0} \frac{c}{(\overline{m}_1 \ldots \overline{m}_s)^{\alpha}}.$$

The quantity $P_{\alpha}(N, z)$ and the discrepancy are similar measures of the quality of the lattice point set.

In 1959 Bahvalov proved that [1] there exists an optimal choice of the generating vector $\mathbf{z}$, for which the error of integration satisfies

$$\left| \frac{1}{N} \sum_{k=0}^{N-1} f\left(\left\{ \frac{k}{N} \mathbf{z} \right\} \right) - \int\limits_{[0,1)^s} f(u) du \right| \leq cd \frac{(\log N)^{\beta(s,\alpha)}}{N^{\alpha}}, \tag{5}$$

for the function $f \in E_s^{\alpha}(c)$, where $\alpha > 1$ and $d(s, \alpha), \beta(s, \alpha)$ do not depend on N.

Moreover, if N is a prime number, then $\beta(s, \alpha) = \alpha(s-1)$. The generating vector $\mathbf{z}$, for which inequality (5) is satisfied, is an optimal generating vector and the point set P_N is a set of good integration points and the numerical integration method is called Good Lattice Point method (GLP). While the theoretical result establish the existence of optimal generating vectors, the difficulty of the development of GLPs is in the construction of the optimal vectors and this is especially difficult with increasing the dimensionality of the integral and dramatically increases the computational complexity.

The first generating vector that we are going to use is based on the generalized Fibonacci numbers of the corresponding dimension. Let $F_n^{(s)}$ is the n-th term of the corresponding generalized Fibonacci sequence [8] of dimensionality s. It's a sum of previous s terms from this sequence:

$$F_n^{(s)} = \sum_{i=n-s}^{n-1} F_i^{(s)}, \quad \text{where } n \text{ is an integer and } n \geq s \tag{6}$$

and the following initial conditions hold:

$$F_0^{(s)} = F_1^{(s)} = \ldots = F_{s-2}^{(s)} = 0, \ F_{s-1}^{(s)} = 1. \tag{7}$$

Consider the following generating vector [8]:

$$\mathbf{z} = (1, F_n^{(s)}(2), \ldots, F_n^{(s)}(s)), \tag{8}$$

where we use that

$$F_n^{(s)}(j) := F_{n+j-1}^{(s)} - \sum_{i=0}^{j-2} F_{n+i}^{(s)} \tag{9}$$

and $F_{n+l}^{(s)}$ $(l = 0, \ldots, j-1, j$ is an integer, $2 \leq j \leq s)$ is the corresponding term of the generalized Fibonacci sequence of dimensionality s.

The generating vector (8) is transformed into [3], [8]:

$$\mathbf{z} = (1, F_{n-1}^{(s)} + F_{n-2}^{(s)} + \ldots + F_{n-s+1}^{(s)}, \ldots, F_{n-1}^{(s)} + F_{n-2}^{(s)}, F_{n-1}^{(s)}). \tag{10}$$

Hua and Wang in 1981 [3] proved that the lattice point set with $N = F_n^{(s)}$ points obtained by using generating vector based on generalized Fibonacci numbers of corresponding dimensionality

$$\left(\left\{ \frac{1}{F_n^{(s)}} k \right\}, \left\{ \frac{F_n^{(s)}(2)}{F_n^{(s)}} k \right\}, \ldots, \left\{ \frac{F_n^{(s)}(s)}{F_n^{(s)}} k \right\} \right), \tag{11}$$

$1 \leq k \leq F_n^{(s)}$, has discrepancy

$$D^*_{F_n^{(s)}} = \mathcal{O}\left({F_n^{(s)}}^{-\frac{1}{2} - \frac{1}{2^{s+1}. \ln 2} - \frac{1}{2^{2s+3}}} \right)$$

and the worst case error is

$$P_{\alpha}(\mathbf{z}, F_n^{(s)}) = \mathcal{O}\left((F_n^{(s)})^{-\frac{\alpha}{2} - \frac{\alpha}{2^{s+1}. \log 2} - \frac{\alpha}{2^{2s+4}}} \right).$$

If we change the generating vector to be optimal in the way described in [6] we have improved the lattice sequence. The optimal generating vector that we are going to use is constructed recently by Dirk Nuyens [6]. This is a 600-dimensional base-2 generating vector of prime numbers for up to $2^{20} = 1048576$ points. The method is improved by generating the points from a lattice sequence in base 2 in gray coded radical inverse ordering. This generating vector is generated by the fast component-by-component algorithms, developed in his PhD thesis. The special choice of this optimal generating vector is better than the generating vector from generalized Fibonacci numbers for higher dimensions, which is only optimal for the two dimensional case [8].

III. Numerical examples

We considered three different examples of 4,7,10 and 30 dimensional integrals, respectively, for which we have computed their referent values.

Example 1. s = 4.

$$\int\limits_{[0,1]^4} x_1 x_2^2 e^{x_1 x_2} \sin(x_3) \cos(x_4) \approx 0.108975. \tag{12}$$

Example 2. s = 7.

$$\int\limits_{[0,1]^7} e^{1-\sum\limits_{i=1}^{3} sin(\frac{\pi}{2}.x_i)}.arcsin\left(sin(1)+\frac{\sum\limits_{j=1}^{7} x_j}{200}\right) \approx 0.7515. \tag{13}$$

Example 3. s = 10.

$$\int\limits_{[0,1]^{10}} \frac{4x_1x_3^2e^{2x_1x_3}}{(1+x_2+x_4)^2}e^{x_5+\cdots+x_{10}} \approx 14.808435. \tag{14}$$

Example 4. s= 30.

$$\int\limits_{[0,1]^{30}} \frac{4x_1x_3^2e^{2x_1x_3}}{(1+x_2+x_4)^2}e^{x_5+\cdots+x_{20}}x_{21}\ldots x_{30} \approx 3.244. \tag{15}$$

The results are given in the tables below. We make a comparison between plain Monte Carlo (CRUDE)optimized lattice sequence with an optimal generating vector (OPT), Fibonacci lattice sets (FIBO), Sobol sequence (SOBOL) and Matousek scrambling for Sobol sequence (SCR). Each Table contains information about the stochastic approach which is applied, the obtained relative error (RE), the needed CPU-time in seconds and the number of points. Note that when the FIBO method is tested, the number of sampled points are always Generalized Fibonacci numbers of the corresponding dimensionality.

Table I
ALGORITHM COMPARISON OF THE RE FOR THE 4-DIMENSIONAL INTEGRAL.

# of points	OPT	t,s	FIBO	t,s	SOBOL	t,s	SCR	t,s
1490	6.11e-4	0.002	1.01e-3	0.004	9.46e-4	0.43	3.78e-3	0.47
10671	2.13e-5	0.01	8.59e-5	0.02	5.28e-4	1.4	6.10e-4	1.59
20569	6.56e-6	0.02	3.89e-5	0.03	3.52e-5	4.32	1.97e-5	4.54
39648	9.14e-7	0.06	3.01e-5	0.07	2.68e-5	7.77	9.67e-6	8.26
147312	4.78e-7	0.15	3.71e-6	0.24	2.29e-6	23.7	1.40e-6	27.91

Table II
ALGORITHM COMPARISON OF THE RE FOR THE 4-DIMENSIONAL INTEGRAL

t, s	OPT	FIBO	SOBOL	SCR
1	5.66e-7	5.62e-6	7.54e-4	6.32e-4
5	3.12e-7	5.38e-7	3.26e-5	1.23e-5
10	5.14e-8	3.77e-7	1.50e-5	8.48e-6
20	3.18e-8	2.67e-8	3.55e-6	1.16e-6

Numerical results show essential advantage for the optimized lattice sets algorithm based on an optimal generating vector in comparison with Fibonacci generalized numbers and Sobol scramble sequence (1-2 orders). For lower dimensions FIBO and Sobol gives results of the same order-see Table I,II. For higher dimensions Scramble sequence SCR is better than FIBO and Sobol by at least 1 order. The results for relative errors corresponding to FIBO and Sobol are similar especially for higher sample number, see Tables III. If the computational time is fixed the advantage of Fibonacci lattice sets in terms of relative error in comparison with Sobol approach is clearly

Table III
ALGORITHM COMPARISON OF THE RE FOR THE 7-DIMENSIONAL INTEGRAL.

# of points	OPT	t,s	FIBO	t,s	SOBOL	t,s	SCR	t,s
2000	6.39e-4	0.14	2.81e-3	0.23	5.45e-3	1.04	2.51e-3	1.42
7936	3.23e-4	0.64	1.38e-3	0.87	1.28e-3	2.08	1.16e-3	3.08
15808	1.23e-5	0.95	9.19e-4	1.73	9.65e-4	3.26	7.58e-4	5.89
62725	3.15e-6	2.54	2.78e-5	3.41	5.18e-4	12.3	3.11e-4	15.64
124946	1.12e-6	6.48	6.87e-5	6.90	1.09e-4	25.4	8.22e-5	31.41

Table IV
ALGORITHM COMPARISON OF THE RE FOR THE 7-DIMENSIONAL INTEGRAL

t, s	OPT	FIBO	SOBOL	SCR
0.1	7.38e-4	2.38e-3	8.85e-3	8.37e-3
1	1.17e-5	6.19e-4	5.85e-3	1.37e-3
5	2.32e-6	8.81e-5	1.79e-3	8.38e-4
10	9.11e-7	1.88e-5	7.36e-4	4.78e-4
20	7.43e-7	3.87e-6	1.96e-4	9.87e-5

Table V
ALGORITHM COMPARISON OF THE RE FOR THE 10-DIMENSIONAL INTEGRAL.

# of points	OPT	t,s	FIBO	t,s	SOBOL	t,s	SCR	t,s
1597	3.14e-4	0.002	4.39e-3	0.003	6.31e-3	0.02	1.46e-3	0.05
17711	6.21e-5	0.02	1.81e-3	0.04	5.31e-4	0.11	1.83e-4	0.21
121393	4.34e-6	0.15	1.20e-3	0.16	1.78e-4	1.21	3.12e-5	1.47
832040	4.11e-7	0.75	1.19e-5	0.70	3.24e-5	12.1	8.25e-6	14.41
3524578	5.32e-8	6.35	2.63e-6	6.45	4.57e-6	121.5	7.71e-7	139.1

Table VI
ALGORITHM COMPARISON OF THE RE FOR THE 10-DIMENSIONAL INTEGRAL

t, s	OPT	FIBO	SOBOL	SCR
0.1	4.95e-6	9.19e-6	5.31e-4	4.19e-4
1	8.10e-7	5.63e-6	1.81e-4	1.21e-4
5	3.56e-8	2.15e-6	8.07e-5	7.21e-5
10	4.31e-8	1.79e-6	4.77e-5	3.51e-5
20	9.13e-9	8.61e-7	8.42e-6	7.09e-6

Table VII
ALGORITHM COMPARISON OF THE RE FOR THE 30-DIMENSIONAL INTEGRAL.

# of points	OPT	t,s	SCR	t,s	SOBOL	t,s	FIBO	t,s
1024	1.21e-2	0.02	5.78e-2	0.53	1.18e-1	0.42	8.81e-1	0.02
16384	4.11e-3	0.16	1.53e-2	5.69	8.40e-2	4.5	6.19e-1	0.14
131072	5.24e-4	1.34	1.35e-3	42.1	1.18e-2	30.2	2.78e-1	1.16
1048576	8.81e-5	9.02	6.78e-4	243.9	9.20e-3	168	9.86e-2	8.61

Table VIII
ALGORITHM COMPARISON OF THE RE FOR THE 30-DIMENSIONAL INTEGRAL

time,sec	OPT	SCR	SOBOL	FIBO
1	3.48e-3	2.38e-2	1.01e-1	2.38e-1
5	4.23e-4	5.46e-3	7.76e-2	1.81e-1
10	8.91e-5	1.25e-3	5.71e-2	9.48e-2
20	2.33e-5	6.11e-4	1.28e-2	7.87e-2

seen, see Tables IV. In general scrambling procedure improves the relative error of the unscrambled nets as it is the case for Sobol sequence and its scrambled version by Matousek linear scrambling as can be seen form the results in Tables V,VI. For very high dimensions the optimized lattice rule outperforms not only the scramble sequence, but also the FIBO method and Sobol sequence by at least 2 orders - see Table VII,VIII. The experiments show that the optimized lattice sequence with a special choice of the optimal generating vector is the best method in terms of lower relative errors with increasing the dimensionality of the integral. The optimized lattice sequence gives the best results compared to the other stochastic approaches also for a fixed computational times which show that the presented algorithm is the most computationally efficient.

IV. CONCLUSION

In this paper an optimized lattice rule has been presented and tested on multidimensional integrals used in machine learning. A comprehensive experimental study of optimized lattice rule, Fibonacci lattice sets, Sobol sequence and Matousek scrambling for Sobol sequence has been done on some case test functions. This approaches are the only possible algorithms for high dimensional integrals because the deterministic algorithms need an huge amount of time for the evaluation of the integral. The numerical tests show that the optimized lattice rule is the most efficient for multidimensional integration and especially for computing high dimensional integrals. It is an important element since this may be crucial in order to achieve a more reliable interpretation of the results in Bayesian statistics which is foundational in artificial intelligence and machine learning.

REFERENCES

[1] N. Bahvalov (1959) On the approximate calculation of multiple integrals, Journal of Complexity, Volume 31, Issue 4, 2015, Pages 502-516, ISSN 0885-064X, https://doi.org/10.1016/j.jco.2014.12.003.

[2] Dimov I., Monte Carlo Methods for Applied Scientists, New Jersey, London, Singapore, World Scientific, 2008, 291p.

[3] Hua, L.K. and Wang, Y., *Applications of Number Theory to Numerical analysis*, 1981.

[4] Lin S., "Algebraic Methods for Evaluating Integrals in Bayesian Statistics," Ph.D. dissertation, UC Berkeley, May 2011.

[5] Lin, S., Sturmfels B., Xu Z.: Marginal Likelihood Integrals for Mixtures of Independence Models, Journal of Machine Learning Research, Vol. 10, pp. 1611-1631, 2009, https://doi/10.5555/1577069.1755838.

[6] Kuo, F.Y., Nuyens, D. Application of Quasi-Monte Carlo Methods to Elliptic PDEs with Random Diffusion Coefficients: A Survey of Analysis and Implementation. Found Comput Math 16, 1631–1696 (2016). https://doi.org/10.1007/s10208-016-9329-5.

[7] Sloan I.H. and Kachoyan P.J., Lattice methods for multiple integration: Theory, error analysis and examples, SIAM J. Numer. Anal. 24, pp. 116–128, 1987, https://doi.org/10.1137/0724010.

[8] Wang Y., Hickernell F.J. (2002) An Historical Overview of Lattice Point Sets. In: Fang KT., Niederreiter H., Hickernell F.J. (eds) Monte Carlo and Quasi-Monte Carlo Methods 2000. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-56046-0_10.

[9] Watanabe S., Algebraic analysis for nonidentifiable learning machines. NeuralComput.(13), pp. 899—933, April 2001, https://doi.org/10.1162/089976601300014402.

[10] Zheleva, I., Georgiev, I., Filipova, M., & Menseidov, D. (2017, October). Mathematical modeling of the heat transfer during pyrolysis process used for end-of-life tires treatment. In AIP Conference Proceedings (Vol. 1895, No. 1, p. 030008). AIP Publishing LLC, https://doi.org/10.1063/1.5007367.

DOI: 10.15439/2020F111
ISSN 2300-5963 ACSIS, Vol. 21

A New Optimized Adaptive Approach for Estimation of the Wigner Kernel

Venelin Todorov
Bulgarian Academy of Sciences
Institute of Mathematics and Informatics
ul. G. Bonchev 8, 1113 Sofia, Bulgaria
Bulgarian Academy of Sciences
Institute of Information and Communication Technologies
ul. G. Bonchev 25A, 1113 Sofia, Bulgaria
Email: vtodorov@math.bas.bg,venelin@parallel.bas.bg

Stefka Fidanova
Bulgarian Academy of Sciences
Institute of Information and Communication Technologies
ul. G. Bonchev 25A, 1113 Sofia, Bulgaria
Email: stefka@parallel.bas.bg

Ivan Dimov
Bulgarian Academy of Sciences
Institute of Information and Communication Technologies
ul. G. Bonchev 25A, 1113 Sofia, Bulgaria
Email: ivdimov@bas.bg

Stoyan Poryazov
Bulgarian Academy of Sciences
Institute of Mathematics and Informatics
ul. G. Bonchev 8, 1113 Sofia, Bulgaria
Email: stoyan@math.bas.bg

***Abstract*—In this paper we study numerically an optimized Adaptive Monte Carlo algorithm for the Wigner kernel - an important problem in quantum mechanics represented by difficult multidimensional integrals. We will show the advantages of the optimized Adaptive MC algorithm and compare the results with the Adaptive approach from our previous work [4] and other stochastic approaches for computing the Wigner kernel in 3,6,9-dimensional case. The 12-dimensional case will be considered for the first time. A comprehensive study and an analysis of the computational complexity of the optimized Adaptive MC algorithm under consideration has also been presented.**

I. Introduction

The Monte Carlo (MC) methods are widely used in solving different multidimensional problems by performing realizations of random processes or random variables. One of the best known physicist Richard Feynman formulated the problem of finding an effective and fast algorithm with linear or polynomial computational complexity for computing multidimensional integrals that represent Wigner kernel [2]. More information about the signed particle formulation of a single-body and many-body system can be found in [3]. So far the Wigner kernel is calculated with deterministic methods which suffer from the „curse of dimensionality" and this means computational times growing exponentially with the problem dimension. Meanwhile stochastic MC methods are not affected by the „curse of dimensionality".

II. Description of the optimized Adaptive approach

Adaptive approach [1] is well known method for evaluation of multidimensional integrals, especially when the integrand function has peculiarities and peaks. Let p_j and I_{Ω_j} are the following expressions: $p_j = \int_{\Omega_j} p(\mathbf{x})\,\mathrm{d}\mathbf{x}$ and $I_{\Omega_j} = \int_{\Omega_j} f(\mathbf{x})p(\mathbf{x})\,\mathrm{d}\mathbf{x}$. Consider now a random point $\xi^{(j)} \in \Omega_j$ with a density function $p(\mathbf{x})/p_j$. In this case $I_{\Omega_j} = \mathbf{E}\left[\frac{p_j}{N}\sum_{i=1}^{N} f(\xi_i^{(j)})\right] = \mathbf{E}\theta_N$. This adaptive algorithm gives an approximation with an error $\varepsilon \leq c\ N^{-1/2}$, where $c \leq 0.6745\sigma(\theta)$ ($\sigma(\theta)$ is the standard deviation).

The optimized adaptive algorithm has higher accuracy than the original Adaptive Monte Carlo algorithm as can be seen from the tables below. The increase of the constant for the initial number of taken subregions $M = 4$ improves the relative error compared with the previous choice $M = 2$ in [4]. The optimized adaptive algorithm is described below.

Algorithm

```
1.   Input data: total number of points $N1$, constant $M = 4$(the initial number of subregions taken), constant $\varepsilon$ (max value of the variance in each subregion), constant $\delta$ (maximal admissible number of subregions), d-dimensionality of the initial region/domain, $f$ - the function of interest.
     1.1. Calculate the number of points to be taken in each subregion $N = N1/\delta$.
```

Venelin Todorov is supported by the Bulgarian National Science Fund under Young Scientists Project KP-06-M32/2 - 17.12.2019 "Advanced Stochastic and Deterministic Approaches for Large-Scale Problems of Computational Mathematics" and by the National Scientific Program "Information and Communication Technologies for a Single Digital Market in Science, Education and Security (ICT in SES)", contract No DO1-205/23.11.2018, financed by the Ministry of Education and Science in Bulgaria. Stoyan Poryazov is supported by the joint research project Symbolic-Numerical Decision Methods for Algebraic Systems of Equations in Perspective Telecommunication Tasks of IMI-BAS, Bulgaria and JINR, Dubna, Russia. The work is also supported by the Bulgarian National Science Fund under Project DN 12/5-2017 "Efficient Stochastic Methods and Algorithms for Large-Scale Problems".

```
2.  For $j = 1,\ M^d$:
    2.1.  Calculate the approximation of $I_{\Omega_j}$ and the variance $\mathbf{D}_{\Omega_j}$ in subdomain $\Omega_j$ based on $N$ independent realizations of random variable $\theta_N$;
    2.2.  If $(\mathbf{D}_{\Omega_j} \geq \varepsilon)$ then
          2.2.1.  Choose the axis direction on which the partition will perform,
          2.2.2.  Divide the current domain into two $(G_{j_1}, G_{j_2})$ along the chosen direction,
          2.2.3.  If the length of obtained subinterval is less than $\delta$ then go to step 2.2.1 else $j = j_1$ $G_{j_1}$ is the current domain right and go to step 2.1;
    2.3.  Else if $(\mathbf{D}_{\Omega_j} < \varepsilon)$ but an approximation of $I_{G_{j_2}}$ has not been calculated yet, then $j = j_2$ $G_{j_2}$ is the current domain along the corresponding direction right and go to step 2.1;
    2.4.  Else if $(\mathbf{D}_{\Omega_j} < \varepsilon)$ but there are subdomains along the other axis directions, then go to step 2.1;
    2.5.  Else Accumulation in the approximation $I_N$ of $I$.
```

For the simple case when we have the two dimensional case ($N = 2$) and on the first step in the optimized adaptive approach we have $M = 4$ subdomains in our optimized Adaptive approach and

$$\hat{\theta}_N = \frac{1}{N_1}\sum_{i=1}^{N_1}\theta_i + \frac{1}{N_2}\sum_{i=1}^{N_2}\theta_i + \frac{1}{N_3}\sum_{i=1}^{N_3}\theta_i + \frac{1}{N_4}\sum_{i=1}^{N_4}\theta_i,$$

where $N_1 + N_2 + N_3 + N_4 = N$, so we have the same number of operations as the Crude Monte Carlo, which computational complexity is linear, to evaluate an approximation of I_{G_j}.

So we choose only $\mathcal{O}(1)$ subdomains where the variance is greater than the parameter ε and this is independent of N. When we divide the domain on every step adaptiveness is not in all subdomains, but only in $\mathcal{O}(1)$ subdomains. At the beginning we have to choose $\frac{N}{k_0}$ random points. After that when dividing the domain into 2^N subdomains, we choose only $\mathcal{O}(1)$ subdomains, this choice is again independent of N. In these subdomains we choose $\frac{N}{k_1}$ points. On the j^{th} step of the Adaptive approach we choose $\mathcal{O}(1)$ subdomains with $\frac{N}{k_j}$ points. We have that $\sum\limits_{j=0}^{i} \frac{1}{k_j} = 1$. Therefore for the computational complexity we obtain

$$\frac{N}{k_0} + \mathcal{O}(1)\frac{N}{k_1} + \cdots + \mathcal{O}(1)\frac{N}{k_i} =$$

$$= N\mathcal{O}(1)\left(\sum_{j=0}^{i}\frac{1}{k_j}\right) = N\mathcal{O}(1) = \mathcal{O}(N).$$

In this way we can conclude that the computational complexity of the optimized Adaptive algorithm is linear.

III. Numerical examples

A new formulation of quantum mechanics in terms of signed classical field-less particles is presented in [3]. Just for completeness we give here the three postulates which completely define the new mathematical formulation of quantum mechanics taken from [3].

Postulate I. Physical systems can be described by means of (virtual) Newtonian particles, i.e. provided with a position $\mathbf{x}$ and a momentum $\mathbf{p}$ simultaneously, which carry a sign which can be positive or negative.

Postulate II. A signed particle, evolving in a potential $V = V(x)$, behaves as a field-less classical point-particle which, during the time interval $\mathrm{d}t$, creates a new pair of signed particles with a probability $\gamma(\mathbf{x}(t))\mathrm{d}t$, where

$$\gamma(\mathbf{x}) = \int_{-\infty}^{+\infty} \mathrm{D}\mathbf{p}' V_W^+(\mathbf{x};\mathbf{p}') \equiv \lim_{\triangle\mathbf{p}'\to 0^+} \sum_{\mathbf{M}=-\infty}^{+\infty} V_W^+(\mathbf{x};\mathbf{M}\triangle\mathbf{p}'),$$

where $\hbar = \frac{h}{2\pi}$ is the reduced Planck constant (h) or Dirac constant, $\mathbf{M} = (M_1, M_2, \ldots, M_d)$ is a set of d integers and $V_W^+(\mathbf{x};\mathbf{p})$ is the positive part of the quantity

$$V_W(\mathbf{x};\mathbf{p}) = \frac{i}{\pi^d \hbar^{d+1}} \int_{-\infty}^{+\infty} \mathrm{d}\mathbf{x}' e^{-\frac{2i}{\hbar}\mathbf{x}'\mathbf{p}} [V(\mathbf{x}+\mathbf{x}') - V(\mathbf{x}-\mathbf{x}')], \tag{1}$$

known as the Wigner kernel (in a d-dimensional space) [5]. If, at the moment of creation, the parent particle has sign s, position $\mathbf{x}$ and momentum $\mathbf{p}$, the new particles are both located in $\mathbf{x}$, have signs $+s$ and $-s$, and momentum $\mathbf{p} + \mathbf{p}'$ and $\mathbf{p} - \mathbf{p}'$ respectively, with $\mathbf{p}'$ chosen randomly according to the (normalized) probability $\frac{V_W^+(\mathbf{x};\mathbf{p})}{\gamma(\mathbf{x})}$.

Postulate III. Two particles with opposite sign and same phase-space coordinates $(\mathbf{x}, \mathbf{p})$ annihilate.

The infinite domain of integration can be mapped into the s-dimensional unit hypercube using the following transformation $\frac{1}{2} + \frac{1}{\pi}\arctan(x)$ which maps $(-\infty, \infty)$ to $(0, 1)$. We want to compute (1) in the $3, 6, 9$ and for the first time in 12-dimensional case,

$$Vw(x,p) = \int e^{\left(\frac{-i2\sum\limits_{k=1}^{n} x'_k p_k}{\hbar}\right)} \times$$

$$[V(x_1+x'_1, \ldots x_n+x'_n) - V(x_1-x'_1, \ldots x_n-x'_n)]dx_1^{'} \ldots dx_n^{'},$$

where the Wigner potential is $V = V(x) = \{x_1 \ldots x_n,\ \ x', x, p, x + x', x - x' \ \in \ [0,1]\}$. It is well known that Wigner kernel has real values [5].

First, we will make a comparison with deterministic method of mid rectangulars, and after that with the well known stochastic approaches of Sobol QMC and Fibonacci based lattice rule FIBO, see [4].

In Table I it can be seen that the optimized stochastic approach gives better results and lower relative errors than the adaptive approach used in our previous study [4]. It can be

Table I
RELATIVE ERROR OF THE OPTIMIZED ADAPTIVE APPROACH, ADAPTIVE APPROACH AND THE DETERMINISTIC MID RECTANGULAR METHOD

s	N	determ.	t (s)	OptAdapt	t (s)	Adapt	t (s)
3	$32^2 \times 50$	8.51e-03	0.2	1.47e-03	0.1	2.71e-03	0.1
	$32^2 \times 100$	8.21e-03	0.5	1.14e-04	0.21	3.42e-04	0.2
	$64^2 \times 50$	5.76e-03	1	5.12e-0	0.6	7.52e-05	0.55
	$64^2 \times 100$	4.89e-03	1.9	7.11e-06	1.4	1.21e-05	1.3
6	$8^4 \times 50^2$	1.16e-02	41.2	8.64e-05	19.5	9.09e-04	18.1
	$8^4 \times 100^2$	9.75e-03	160.6	5.21e-06	59.4	1.52e-05	57.9
	$16^4 \times 50^2$	7.84e-03	635.2	3.21e-05	321	4.37e-04	311.5
	$16^4 \times 100^2$	2.12e-03	2469.1	2.13e-05	1001.6	3.80e-04	987.1
9	$6^6 \times 16^3$	1.75e-03	835.5	5.11e-05	345	7.62e-05	330.5
	$6^6 \times 32^3$	1.35e-03	5544.1	1.41e-05	2133.6	2.73e-05	2225.1
	$6^6 \times 40^3$	1.12e-03	10684.4	1.67e-06	4531.5	8.12e-06	4491.5

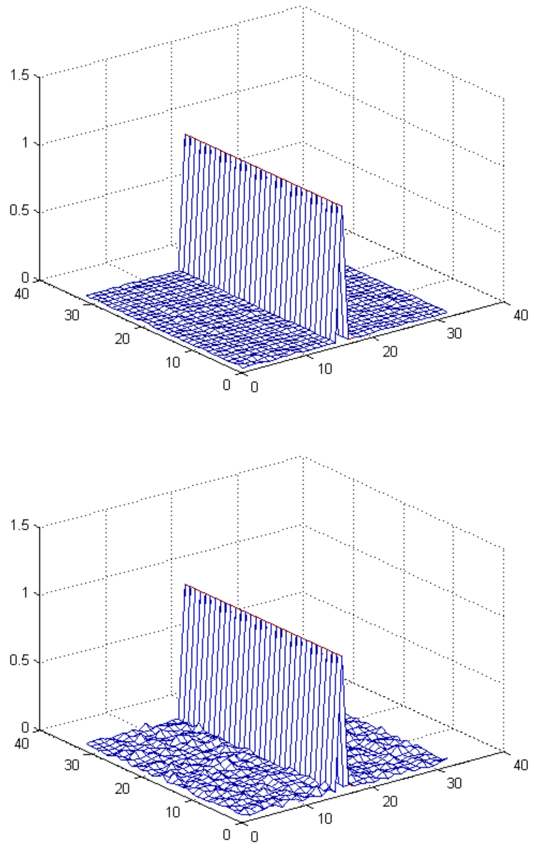


Figure 1. The Wigner kernel with optimized adaptive and standard adaptive method

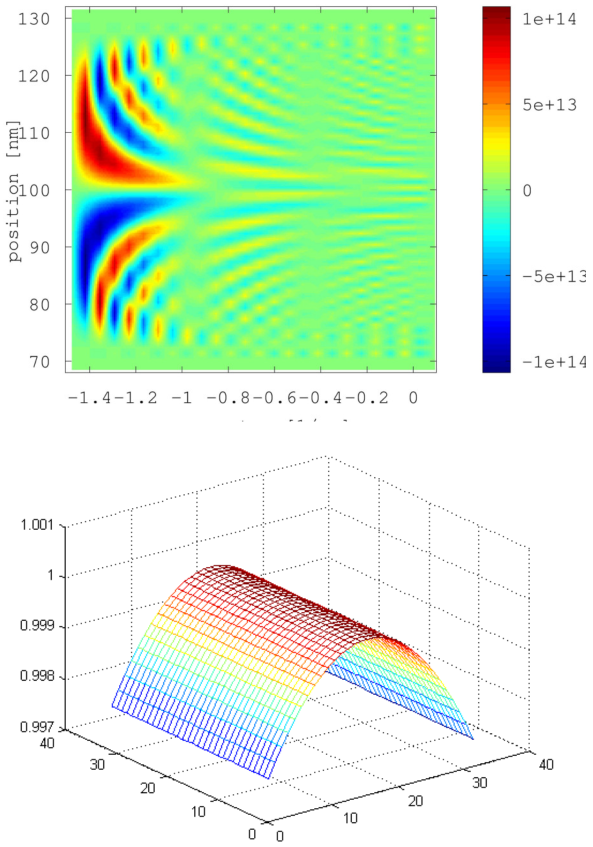


Figure 2. The position and the peak of the Wigner kernel with optimizes adaptive approach

Table II
RELATIVE ERROR FOR 3 DIMENSION

N	Adapt	t,s	OptAdapt	t,s	FIBO	t,s	Sobol	t,s
10^3	5.36e-03	0.3	6.75e-04	0.4	3.72e-02	0.02	1.07e-02	0.05
10^4	4.84e-04	2.9	8.15e-05	3.3	7.06e-03	0.07	8.77e-03	0.54
10^5	2.51e-05	29	5.01e-06	32.6	3.40e-03	0.43	8.57e-04	5.74
10^6	1.76e-05	287	4.38e-07	302	1.01e-03	4.4	6.73e-04	51.6
10^7	6.26e-06	2535	8.02e-08	2708	1.80e-04	49.7	5.98e-05	499

Table III
RELATIVE ERROR FOR 3 DIMENSION

,s	Adapt	OptAdapt	FIBO	Sobol
0.1	6.74e-03	8.73e-04	8.12e-03	1.01e-02
1	8.73e-04	4.05e-05	5.42e-03	7.27e-03
10	5.62e-05	9.12e-06	2.11e-03	7.83e-04
100	3.43e-06	8.18e-07	9.50e-04	2.18e-04

seen that the computational time for the optimized Adaptive MC approach is better than the deterministic method when the dimensionality increases. The advantage of the optimized adaptive algorithm in comparison with the previously used adaptive algorithm is shown on Figure 1, and the computation of the position of the signs and the peak are given in Figure 2. The numerical results including relative errors and

Table IV
RELATIVE ERROR FOR 6 DIMENSION

N	Adapt	t,s	OptAdapt	t,s	FIBO	t,s	Sobol	t,s
10^3	6.72e-03	0.41	2.23e-04	0.5	7.82e-03	0.01	2.42e-02	0.09
10^4	9.10e-04	3.5	4.74e-05	4.1	5.01e-03	0.07	5.02e-03	0.78
10^5	5.26e-05	33	5.43e-06	37	6.88e-03	0.43	4.60e-04	7.19
10^6	2.70e-06	315	5.04e-07	351	7.68e-04	5.97	3.59e-04	73
10^7	1.03e-06	2438	8.12e-08	2841	4.12e-04	48	8.11e-05	590

Table V
RELATIVE ERROR FOR 6 DIMENSION

t,s	Adapt	OptAdapt	FIBO	Sobol
0.1	9.25e-04	6.81e-04	8.11e-03	2.13e-02
1	4.51e-04	9.09e-05	9.25e-04	3.31e-03
10	2.57e-05	8.13e-06	5.11e-04	9.34e-04
100	2.72e-06	5.08e-07	1.05e-04	1.27e-04

Table VI
RELATIVE ERROR FOR 9 DIMENSION

N	Adapt	t,s	OptAdapt	t,s	FIBO	t,s	Sobol	t,s
10^3	4.92e-02	0.4	8.23e-04	0.5	2.03e-02	0.06	5.42e-02	0.11
10^4	9.09e-04	3.9	2.02e-05	4.7	2.02e-03	0.07	6.02e-03	0.88
10^5	3.32e-05	35	1.08e-06	40	9.16e-04	0.53	3.57e-03	7.56
10^6	6.46e-06	367	4.14e-07	381	7.13e-04	3.7	8.02e-04	72
10^7	1.21e-06	2742	8.91e-08	2912	4.84e-04	40	5.19e-04	621

Table VII
RELATIVE ERROR FOR 9 DIMENSION

t,s	Adapt	OptAdapt	FIBO	Sobol
0.1	9.24e-03	1.73e-03	1.35e-03	5.42e-02
1	1.23e-03	8.05e-05	8.72e-04	5.59e-03
10	3.82e-05	6.32e-06	6.51e-04	5.84e-03
100	3.09e-06	7.58e-07	3.70e-04	6.39e-04

Table VIII
RELATIVE ERROR FOR 12 DIMENSION

N	Adapt	t,s	OptAdapt	t,s	FIBO	t,s	Sobol	t,s
10^3	3.91e-03	0.7	3.21e-04	0.9	1.33e-02	0.09	2.85e-02	0.2
10^4	5.04e-04	4.5	1.08e-05	6.1	1.34e-03	0.1	4.04e-03	1.34
10^5	2.76e-04	48	5.04e-06	56	5.51e-04	0.68	1.77e-03	9.8
10^6	4.14e-05	415	2.72e-07	432	4.43e-04	4.7	4.07e-04	82
10^7	2.31e-06	3351	4.87e-08	3467	2.5684e-04	60	2.7e-04	700

Table IX
RELATIVE ERROR FOR 12 DIMENSION

t,s	Adapt	OptAdapt	FIBO	Sobol
0.1	4.66e-03	6.22e-04	6.56e-04	2.67e-02
1	3.25e-04	4.51e-05	4.45e-04	2.98e-03
10	3.24e-05	3.56e-06	3.56e-04	2.45e-03
100	1.31e-05	4.16e-07	1.87e-04	3.21e-04

computational times corresponding to the algorithms under consideration are presented, and the algorithms efficiency is discussed. A numerical comparison for a given number of samples between the adaptive approach (Adapt) used in [4], the Sobol (Sob) and the Lattice sequences FIBO described in [4] and the new optimized Adaptive approach (OptAdapt) has been given in Tables II,IV,VI. From the all experiments it can be clearly seen that the optimized adaptive approach gives relative errors with at least 1 or 2 orders better than those produced by the adaptive approach for the cost of slightly bigger computational times, because of the increased number of subregions taken in every subdomain M. The adaptive approach itself gives superior results to the other two stochastic approaches as it is completely described in our previous study [4]. The optimized adaptive MC algorithm is the slowest, but it requires smaller number of random points to achieve better accuracy even for higher dimensions and for a fixed computational time it gives the best relative error by at least 1 order, as can be seen from Tables III,V,VII. The optimized Adaptive MC approach outperforms the other two approaches FIBO and Sobol QMC by at least $3-5$ even for 12 dimensional case, see Table VIII,IX. The efficiency of the optimized adaptive MC algorithm is clearly shown in the case of Wigner kernel, where the integrand have computational specialty in the local subarea of the integration domain - see Figure 1 and how the peak is approximated by the optimized adaptive approach, see Figure 2.

IV. CONCLUSIONS

The optimized adaptive Monte Carlo algorithm under consideration gives the most accurate results in computing the Wigner kernel by a stochastic approach and it has lower computational complexity than the existing deterministic approaches. This means that the proposed optimized stochastic approach is of great importance for the problems in quantum mechanics with high dimensions. Therefore, the presented optimized adaptive MC algorithm is one new successful solution (in terms of robustness and reliability) of Richard Feynman's problem for Wigner kernel evaluation.

REFERENCES

[1] Berntsen J., Espelid T.O., Genz A. (1991) An adaptive algorithm for the approximate calculation of multiple integrals, ACM Trans. Math. Softw. 17: 437–451, https://doi.org/10.1145/210232.210233.

[2] Feynman R.P. (1948) Space-time approach to non-relativistic quantum mechanics, Rev. Mod. Phys. 20, https://doi.org/10.1103/RevModPhys.20.367.

[3] Sellier J.M., Nedjalkov M., Dimov I. (2015) An introduction to applied quantum mechanics in the Wigner Monte Carlo formalism, Physics Reports Volume 577: 1–34, https://doi.org/10.1016/j.physrep.2015.03.001.

[4] Todorov, V., Dimov, I., Georgieva, R., & Dimitrov, S. Adaptive Monte Carlo algorithm for Wigner kernel evaluation. Neural Comput & Applic 32, 9953-9964 (2020). https://doi.org/10.1007/s00521-019-04519-9.

[5] Wigner E. (1932) On the quantum correction for thermodynamic equilibrium, Phys. Rev. 40: 749, https://doi.org/10.1103/PhysRev.40.749.

Intuitionistic Fuzzy Hamiltonian Cycle by Index Matrices

Velichka Traneva
"Prof. Asen Zlatarov" University
"Prof. Yakimov" Blvd, Burgas 8000, Bulgaria
Email: veleka13@gmail.com

Stoyan Tranev
"Prof. Asen Zlatarov" University
"Prof. Yakimov" Blvd, Burgas 8000, Bulgaria
Email: tranev@abv.bg

***Abstract*—In this paper, the algorithm for finding a Hamiltonian cycle in an intuitionistic fuzzy graph (IFG) is proposed, based on the theories of intuitionistic fuzzy sets (IFSs) and of index matrices (IMs). The aim of the paper is to extend the algorithm to find a fuzzy Hamiltonian cycle (FHC) in an IFG to the intuitionistic fuzzy (IFHC) using the IFSs and IMs concepts. An intuitionistic fuzzy graph example about network of Wizz air airlines is modeled by the extended IM to illustrate the proposed algorithm. In the paper also are introduced for the first time three index-type operations over IMs.**

I. Introduction

A HAMILTONIAN cycle is a cycle through a graph that visits each node exactly once (see [21]). Determining if a graph is Hamiltonian is well known to be NP-complete [20]. Dirac (1952, [6]) described some relations between the degree of the nodes in a graph and the lengths of the circuits contained in it. Ore, Chvatal and Fan have provided the sufficient conditions for a graph to be Hamiltonian (see [7], [19], [23]). Zhao, in 2007, gave better conditions for the existence of Hamiltonian paths in a graph (see [17]).

Nowadays, some parameters of the graph problem may be uncertain due to uncontrollable factors. The fuzzy sets (FSs) of Zadeh appeared in 1963 [18] to deal with this environment. The first idea of a fuzzy graph was described by Kaufman [2]. Rosenfeld [3] developed the theory of fuzzy graphs in 1975. Mordeson and Nair have also proposed another concepts in fuzzy graphs [8]. An algorithm for fuzzy Hamiltonian cycle in a network using adjacency matrix was proposed by Gani and Latha [1]. In 1983, Atanassov proposed the IFSs ([9], [11]), which is an extension of the FSs. The major advantage of IFS over FS is that IFS separates the degree of membership and non-membership of an element.

In this paper, it is proposed for the first time two algorithms for finding Hamiltonian cycle in an intuitionistic fuzzy graph (IFG), based on the concepts of IFSs and of IMs (see [10], [12]). The first algorithm is illustrated with an IFG example about a network of Wiz Air [28].

The rest of this paper is structured as follows: Section 2 describes the related concepts of the IMs, IFSs and IFGs. In the section 2 also are introduced for the first time three index-type operations over IMs. In Section 3, we propose an algorithm for determining a Hamiltonian cycle in an IFG, based on the fuzzy algorithm [1], by using the concepts of IMs and IFSs. The effectiveness of the proposed method is demonstrated by an example in Section 4. Section 5 outlines the conclusion and some directions for future research.

II. Basic definitions of IMs, intuitionistic fuzzy logic and IFG

This section presents some definitions on intuitionistic fuzzy pairs (IFPs) from (see [5], [11], [15], [25]), on IMs concept from (see [12], [27]) and on IFG (see [4], [12]).

A. Remarks on Intuitionistic Fuzzy (IF) Logic

The **IFP** is an object in the form of an ordered pair $\langle a,b\rangle = \langle\mu(p),\nu(p)\rangle$, where $a,b\in[0,1]$ and $a+b\leq 1$, that is used as an evaluation of a proposition p (see [15]). $\mu(p)$ and $\nu(p)$ respectively determine the "truth degree" (degree of membership) and "falsity degree" (degree of non-membership). With two IFPs $x=\langle a,b\rangle$ and $y=\langle c,d\rangle$ were defined some basic operations and relations with IFPs

$$\begin{array}{r}
x\wedge_1 y=\langle\min(a,c),\max(b,d)\rangle;\\
x\vee_1 y=\langle\max(a,c)),\min(b,d)\rangle;\\
x\wedge_2 y=x+y=\langle a+c-a.c,b.d\rangle;\\
x\vee_2 y=x.y=\langle a.c,b+d-b.d\rangle;\\
\neg x=\langle b,a\rangle;\alpha.x=\langle 1-(1-a)^{\alpha},b^{\alpha}\rangle(\alpha\in R);\\
x-y=\langle\max(0,a-c),\min(1,b+d,1-a+c)\rangle
\end{array}\qquad(1)$$

and relations with IFPs

$$\begin{array}{cc}
x\geq y \text{ iff } a\geq c \text{ and } b\leq d; & x\leq y \text{ iff } a\leq c \text{ and } b\geq d;\\
x\geq_{\Box} y \text{ iff } a\geq c; & x\leq_{\Box} y \text{ iff } a\leq c;\\
x\geq_{\Diamond} y \text{ iff } b\leq d; & x\leq_{\Diamond} y \text{ iff } b\geq d;\\
x=y & \text{iff } a=c \text{ and } b=d;\\
x\geq_R y & \text{iff } R_{\langle a,b\rangle}\leq R_{\langle c,d\rangle},
\end{array}\qquad(2)$$

where

$$R_{\langle a,b\rangle}=0.5(2-a-b)0.5(|1-a|+|b|+|1-a-b|)\quad[5].$$

B. Definition, Operations and Relations over Extended Intuitionistic Fuzzy Index Matrices

Let $\mathscr{I}$ be a fixed set. The definition of two-dimensional extended intuitionistic fuzzy IM (2-D EIFIM) $[K^*,L^*,\{\langle\mu_{k_i,l_j},\nu_{k_i,l_j}\rangle\}]$ with sets K and L $(K,L\subset\mathscr{I})$ is [12]:

The work is supported by the Ministry of Education and Science under the Programme "Young scientists and postdoctoral students", approved by DCM # 577/17.08.2018.

$$\begin{array}{c|ccc}
 & l_1,\langle\alpha_1^l,\beta_1^l\rangle & \dots & l_n,\langle\alpha_n^l,\beta_n^l\rangle \\
\hline
k_1,\langle\alpha_1^k,\beta_1^k\rangle & \langle\mu_{k_1,l_1},\nu_{k_1,l_1}\rangle & \dots & \langle\mu_{k_1,l_n},\nu_{k_1,l_n}\rangle \\
\vdots & \vdots & \dots & \vdots \\
k_i,\langle\alpha_i^k,\beta_i^k\rangle & \langle\mu_{k_i,l_1},\nu_{k_i,l_1}\rangle & \dots & \langle\mu_{k_i,l_n},\nu_{k_i,l_n}\rangle \\
\vdots & \vdots & \dots & \vdots \\
k_m,\langle\alpha_m^k,\beta_m^k\rangle & \langle\mu_{k_m,l_1},\nu_{k_m,l_1}\rangle & \dots & \langle\mu_{k_m,l_n},\nu_{k_m,l_n}\rangle
\end{array},$$

where for every $1 \le i \le m, 1 \le j \le n$: $\mu_{k_i,l_j}, \nu_{k_i,l_j}, \mu_{k_i,l_j} + \nu_{k_i,l_j} \in [0,1]$; $\alpha_i^k, \beta_i^k, \alpha_i^k + \beta_i^k \in [0,1]$; $\alpha_j^l, \beta_j^l, \alpha_j^l + \beta_j^l \in [0,1]$ and $K^* = \{\langle k_i, \alpha_i^k, \beta_i^k\rangle | k_i \in K\} = \{\langle k_i, \alpha_i^k, \beta_i^k\rangle | 1 \le i \le m\}$,
$L^* = \{\langle l_j, \alpha_j^l, \beta_j^l\rangle | l_j \in L\} = \{\langle l_j, \alpha_j^l, \beta_j^l\rangle | 1 \le j \le n\}$.

In [12] are defined the following operations over two EIFIMs $A = [K^*, L^*, \{\langle\mu_{k_i,l_j}, \nu_{k_i,l_j}\rangle\}]$ and $B = [P^*, Q^*, \{\langle\rho_{p_r,q_s}, \sigma_{p_r,q_s}\rangle\}]$: negation, addition-$(\circ,*)$, termwise subtraction-(max,min), termwise multiplication-$(\min,\max)$, transposition, reduction, projection and substitution. We recall only index type and aggregation operations, and internal subtraction with IMs.

Index type operations [27]: $Index_{\{(\min_R/\max_R)\}(\not\perp),k_i}(A)$ finds the indices of the minimum/ maximum IFFP of the k_i-th row of A with no empty value in accordance with the relations (2).

$Index_{(\max\mu(\nu)),k_i}(A)$ finds the indices of the IFFP of the k_i-th row of A, for which $\mu(\nu)_{k_i,l_{v_x}}$ is maximum.

$AGIndex_{\{(\min_R/\max_R)\}(\notin F)(\not\perp)(\not\langle 0,1\rangle)(\mu\neq 0)}(A)$ finds the indices of the minimum/ maximum element between the elements of A, whose indices respectively $\notin F$ or with no empty value, or not equal to $\langle 0,1\rangle$, or with non-zero degree of membership in accordance with the relations (2).

Let us define the following new index type operations:

$Index_{(\langle m,n\rangle)}(A) = \{\langle k_{w_1}, l_{v_1}\rangle, \dots, \langle k_{w_y}, l_{v_y}\rangle, \dots, \langle k_{w_W}, l_{v_V}\rangle\}$, where $\langle k_{w_y}, l_{v_y}\rangle$ (for $1 \le i \le m$) are the indices of the elements equal to the IFP $\langle m,n\rangle$ of A.

$AGIndex^1_{\{(\min_R/\max_R)\}(\notin F)/(\not\perp)/(\neq\langle 0,1\rangle)}(A)$ determines the index of the minimum/ maximum element between the elements $\langle\alpha_i^k, \beta_i^k\rangle$ of K^* (the first dimension of A), whose indices respectively $\notin F$ or with no empty value, or not equal to $\langle 0,1\rangle$ in accordance with the relations (2).

$AGIndex^2_{\{(\min_R/\max_R)\}(\notin F)/(\not\perp)/(\neq\langle 0,1\rangle)}(A)$ determines the index of the minimum/ maximum element between the elements $\langle\alpha_j^l, \beta_j^l\rangle$ of L^* (the second dimension of A), whose indices respectively $\notin F$ or with no empty value, or not equal to $\langle 0,1\rangle$ in accordance with the relations (2).

Aggregation operations over EIFIMs

We use the operations $\#_q, (q \le i \le 3)$ in aggregation evaluations [26] over IFPs $x = \langle a,b\rangle$ and $y = \langle c,d\rangle$:

$x\#_1 y = \langle min(a,c), max(b,d)\rangle$;

$x\#_2 y = \langle average(a,c), average(b,d)\rangle$;

$x\#_3 y = \langle max(a,c), min(b,d)\rangle$.

Let $k_0 \notin K^*$ be a fixed index. Following [12], [26], another form of the defined aggregation operation in [12] by K is:

$$\alpha_{K,\#_q}(A,k_0) = \begin{array}{c|ccc}
 & l_1,\langle\alpha_1^l,\beta_1^l\rangle & \dots & l_n,\langle\alpha_n^l,\beta_n^l\rangle \\
\hline
k_0, \mathop{\#_q}\limits_{i=1}^{m} \langle\alpha_i^k,\beta_i^k\rangle & \mathop{\#_q}\limits_{i=1}^{m} \langle\mu_{k_i,l_1},\nu_{k_i,l_1}\rangle & \dots & \mathop{\#_q}\limits_{i=1}^{m} \langle\mu_{k_i,l_n},\nu_{k_i,l_n}\rangle
\end{array}.$$

Aggregate global internal operation: $AGIO_{\oplus_{(\max,\min)}}(A)$.

Internal subtraction of IMs' components ([24], [25], [27]):

$IO_{-_{(\max,\min)}}(\langle k_i, l_j, A\rangle, \langle p_r, q_s, B\rangle) = [K, L, \{\langle\gamma_{t_u,v_w}, \delta_{t_u,v_w}\rangle\}]$, $\langle\gamma_{t_u,v_w}, \delta_{t_u,v_w}\rangle$

$$= \begin{cases} \langle\mu_{t_u,v_w}, \nu_{t_u,v_w}\rangle, & \text{if } t_u \neq k_i \in K, \\ & v_w \neq l_j \in L; \\ \langle\max(0, \mu_{k_i,l_j} - \rho_{p_r,q_s}), & \text{if } t_u = k_i \in K, \\ \min(1, \nu_{k_i,l_j} + \sigma_{p_r,q_s}, 1 - \mu_{k_i,l_j} + \rho_{p_r,q_s})\rangle & v_w = l_j \in L \end{cases}$$

C. Intuitionistic Fuzzy Graphs (IFGs)

Let A be an IFS over E_1 and B – over E_2. In [11], [22] are defined six versions of the Cartesian products of two IFSs.

The concept of the IFG was introduced in 1994 in [4]. Let us have a fixed set of vertices $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$. An $(\circ)$-IFG G (over $\mathcal{V}$) is the ordered pair $G = (V^*, A^*)$, where

$$V \subset \mathcal{V}, V^* = \{\langle v, \mu_V(v), \nu_V(v)\rangle | v \in V\},$$

$$A \subset V \times V, A^* = \{\langle\langle x,y\rangle, \mu_A(x,y), \nu_A(x,y)\rangle | \langle x,y\rangle \in V \times V\}$$

and functions $\mu_V : \mathcal{V} \to [0,1]$ and $\nu_V : \mathcal{V} \to [0,1]$ define the degree of membership (existence) and the degree of non-membership (non-existence), respectively, of the element $v \in \mathcal{V}$ to the set V; functions $\mu_A : E_1 \times E_2 \to [0,1]$ and $\nu_A : E_1 \times E_2 \to [0,1]$ define the degree of membership and the degree of non-membership, respectively, of the element $\langle x,y\rangle \in E_1 \times E_2$ to the set $A \subseteq E_1 \times E_2$; these functions have the forms of the corresponding components of the $\circ$-Cartesian product over IFSs from [11], [22] and for all $\langle x,y\rangle \in E_1 \times E_2$,

$$0 \le \mu_V(x) + \nu_V(x) \le 1, 0 \le \mu_A(x,y) + \nu_A(x,y) \le 1.$$

The all parameters of the IFG G are IFPs. The expert approach described in detail in [11] may be used to determine the distances between any two vertices and the existence of the vertices of the graph in the form of IFPs.

Now, for the graph $G = (V,A)$ was constructed the Extended Intuitionistic Fuzzy Graph (EIFG) $G^* = (V^*, A^*)$ [12]. It has the following IM-representation as adjacency EIFIM C:

$$[V^*, V^*, \{\langle\mu(v_i,v_j), \nu(v_i,v_j)\rangle\}] \qquad (3)$$

$$= \begin{array}{c|ccc}
 & v_1,\langle\alpha(v_1),\beta(v_1)\rangle & \dots & v_n,\langle\alpha(v_n),\beta(v_n)\rangle \\
\hline
v_1,\langle\alpha(v_1),\beta(v_1)\rangle & \langle\mu_{v_1,v_1},\nu_{v_1,v_1}\rangle & \dots & \langle\mu_{v_1,v_n},\nu_{v_1,v_n}\rangle \\
\vdots & \vdots & \dots & \vdots \\
v_i,\langle\alpha(v_i),\beta(v_i)\rangle & \langle\mu_{v_i,v_1},\nu_{v_i,v_1}\rangle & \dots & \langle\mu_{v_i,v_n},\nu_{v_i,v_n}\rangle \\
\vdots & \vdots & \dots & \vdots \\
v_n,\langle\alpha(v_n),\beta(v_n)\rangle & \langle\mu_{v_n,v_1},\nu_{v_n,v_1}\rangle & \dots & \langle\mu_{v_n,v_n},\nu_{v_n,v_n}\rangle
\end{array},$$

where for every $1 \le i \le n, 1 \le j \le n$: $C_{v_i,v_j} = \langle\mu_{v_i,v_j}, \nu_{v_i,v_j}\rangle$ and $\langle\alpha(v_i), \beta(v_i)\rangle$ are IFPs.

Proposition 1 In an IFG, if every vertex has exactly two adjacent vertices, then there exists a Hamiltonian cycle.

The proof of the proposition is analogous to that of [1].

III. Algorithms for Hamiltonian cycle in an IFG

Let us be given EIFG $G^* = (V^*, A^*)$ with the IM-representation as EIFIM C with a structure (3). The purpose is to find the Hamiltonian cycle in G^*. Let us extend the algorithms for Hamiltonian cycle in a fuzzy graph from [1] to intuitionistic fuzzy ones, based on IFSs and IMs concepts.

Algorithm 1: Minimum edge degree algorithm to find intuitionistic fuzzy Hamiltonian cycle

Let us define $X[V^*, V^*, \{\langle \rho(v_i, v_j), \sigma(v_i, v_j)\rangle\}]$

$$= \begin{array}{c|ccc} & v_1, \langle \alpha(v_1), \beta(v_1)\rangle & \dots & v_n, \langle \alpha(v_n), \beta(v_n)\rangle \\ \hline v_1, \langle \alpha(v_1), \beta(v_1)\rangle & \langle \rho_{v_1,v_1}, \sigma_{v_1,v_1}\rangle & \dots & \langle \rho_{v_1,v_n}, \sigma_{v_1,v_n}\rangle \\ \vdots & \vdots & \dots & \vdots \\ v_n, \langle \alpha(v_n), \beta(v_n)\rangle & \langle \rho_{v_n,v_1}, \sigma_{v_n,v_1}\rangle & \dots & \langle \rho_{v_n,v_n}, \sigma_{v_n,v_n}\rangle \end{array}$$

where for i and j: x_{v_i,v_j} and $\langle \alpha(v_i), \beta(v_i)\rangle$ are IFPs.

Let us we create the following auxiliary IMs:

1) $S = [V^*, V^*, \{s_{k_i,l_j}\}]$, such that $S = C$ i.e.
$(s_{k_i,l_j} = c_{k_i,l_j} \ \forall k_i \in V^*, \forall l_j \in V^*)$;

2)

$$RC[V^*, e_0] = \begin{array}{c|c} & e_0 \\ \hline k_1 & rc_{k_1,e_0} \\ \vdots & \vdots \\ k_n & rc_{k_n,e_0} \end{array},$$

where for $1 \le i \le n$: $rc_{k_i,e_0} = \{0,1\}$ depending on whether the k_i-th vertex of the matrix S is crossed out (introduced in the Hamiltonian path) or not. When the algorithm starts, $rc_{k_i,e_0} = 0, x_{v_i,v_j} = \langle 0,1\rangle$ $(\forall k_i \in V*, \forall l_j \in V*)$.

We will propose the algorithm for determining a Hamiltonian path in G^*, interpreted with the tools of IMs and IFPs extending the fuzzy algorithm from [1]:

Step 1. Construct the EIFIM C for the given IFG G^* and create EIFIM S such that $S := C$; Check the condition of the proposition 1. for the existence of a Hamiltonian cycle in G^* : for $i = 1$ to n then $\{AGIndex_{\{(\min_R / \max_R)\}(\mu \neq 0)}(S) = \{\langle k_{w_i}, l_{v_1}\rangle, \dots, \langle k_{w_i}, l_{v_y}\rangle, \dots, \langle k_{w_i}, l_{v_V}\rangle\}\}$.
If $V < 2$ then {the Hamiltonian path does not exist and the algorithm **Stop**}
else Go to *Step 2*}.

Step 2. Search for a minimum IFP in the S with non-zero membership degree in accordance with the relations (2). If there are several such elements, then we choose any one.
$AGIndex_{\{(\min_R)\}(\mu \neq 0)}(S) = \langle k_z, l_z\rangle$.
If $x_{k_z,l_z} = \langle 0,1\rangle$, then $\{rc[k_z, e_0] = 1;\ S_{(k_i,\perp)}\}$
$x_{k_z,l_z} = \langle 1,0\rangle$ $\{rc[k_i, e_0] = 1;\ S_{(k_i,\perp)}\}$; Go to *Step 3*.

Step 3. If the minimum IFP s_{k_z,l_z} in the IM S does not allow for a fuzzy Hamiltonian path, then the next higher minimum IFP is selected.
$AGIndex_{\{(\min_R)\}(\mu \neq 0)}(S) = \langle k_x, l_x\rangle$; Go to *Step 4*.

Step 4. Identify the k_x-th row and l_x-th column in the S, where the minimum IFP appears. We add to Hamiltonian the vertex l_x i.e. from k_x it reaches l_x. $rc_{k_x,e_0} = rc_{l_x,e_0} = 1$, $x_{k_x,l_x} = \langle 1,0\rangle$ and S is reduced by $S_{(\perp,l_x)}$; Go to *Step 5*.

Step 5. Search for a minimum IFP of the l_x-th row, such that: it forms a fuzzy Hamiltonian path; if the minimum value occurs more than once, then an IFP is selected for a Hamiltonian path.
The operation is: $AGIndex_{\{(\min_R)\}(\mu \neq 0)}\left(pr_{l_x,V^*}S\right) = \langle l_x, l_u\rangle$;
Then $rc_{l_x,e_0} = rc_{l_u,e_0} = 1$, $x_{k_x,l_u} = \langle 1,0\rangle$ and the IM S is reduced by $S_{(l_x,l_u)}$.

Step 6. Repeat *Step 3* through *Step 4* row-wise until $|Index_{(1)}(RC)| = n$. With $|A|$ let us we denote the number of elements of the A, where A is an IM.
If $|Index_{(1)}(RC)| = n$ then an intuitionistic fuzzy Hamiltonian path with all n vertices of G^* is found and go to *Step 7*,
else there is no intuitionistic fuzzy Hamiltonian path and repeat *Step 2* or *Step 3* as required.

Step 7. If intuitionistic fuzzy Hamiltonian path exists, then only one row k_l will be left out in the IM S. Select an IFP with non-zero degree of membership from that row to form a fuzzy Hamiltonian cycle, if exists. If we can select the IFP s_{k_l,k_z} with non-zero membership degree then {the Hamiltonian cycle exists, $x_{k_l,k_z} = \langle 1,0\rangle$ and go to *Step 8*}
else the algoritm **Stop.**

Step 8. If $|Index_{(\langle m,n\rangle)}(X)| = n$, then the optimal Hamiltonian path is obtained with minimum intuitionistic fuzzy length according to (2). The IF length of the path is:
$AGIO1_{\oplus_{(\max,\min)})}\left(C \otimes_{(\min,\max)} X\right)$ or $AGIO2_{\oplus_{(\vee_2)})}\left(C \otimes_{(\wedge_2)} X\right)$,
where $\vee_2$ and $\wedge_2$ are the operations from (1).

For an IFG with n vertices the algorithm visits all the permutations of the vertices, so the complexity is $O(n!)$.

Algorithm 2 Minimum vertex degree algorithm to find an intuitionistic fuzzy Hamiltonian cycle

Step 1. We select a vertex $v_{min,i}$, whose $\langle \alpha(v_{min,i}), \beta(v_{min,i})\rangle$ is the minimum IFPs $\langle \alpha(v_i), \beta(v_i)\rangle (1 \le i \le n)$. (If there is more than one vertex with same IFP, then choose any one).

Step 2. Select a vertex with IFP whose is next higher to the minimum IFP $\langle \alpha(v_{min,i}), \beta(v_{min,i})\rangle$.

Step 3. Identify the adjacent vertices of the vertex with minimum IFP selected.

Step 4. We select the unvisited adjacent vertex of the minimum IFP $\langle \alpha(v_i), \beta(v_i)\rangle (1 \le i \le n)$. (If more than one adjacent vertex has the same minimum IFP, then choose any one vertex).

Step 5. *Step 3 – Step 4* are repeated until an IF Hamiltonian cycle is found else go to *Step 1.* or *Step 2.* as required.

This **algorithm 2** can be presented with similar IMs operations to that of **algorithm 1**, but the operation $AGIndex^1_{\{(\min_R / \max_R)\}(\mu \neq 0)}(S)$ will be used instead $AGIndex_{\{(\min_R / \max_R)\}(\mu \neq 0)}(S)$. In the algorithm 2, *step 3* and *step 4* need to be repeated starting with each vertex of IFG G^* to find all possible IF Hamiltonian cycles. We can identify the minimum length of IF Hamiltonian cycle(s).

IV. An example for Hamiltonian cycle in IFG

The part of the airline network of the Wizz air is modeled by an IFG $G^* = (V^*, A^*)$. The vertices of IFG are {Sofia airport (S), Dortmund airport (D), Viena airport (V), Brussels South

Charleroi Airport (Br), Barcelona airport (Ba)}. The IFG has the following IM-representation as EIFIM $C[V^*, V^*]$:

$$\begin{array}{c|ccccc}
 & S,\langle 0.6;0.20\rangle & D,\langle 0.75;0.20\rangle & V,\langle 0.70;0.1\rangle & Br,\langle 0.80;0.10\rangle & Ba,\langle 0.83;0.15\rangle \\
\hline
S,\langle 0.6;0.20\rangle & \langle 0;1\rangle & \langle 0.80;0.18\rangle & \langle 0.40;0.50\rangle & \langle 0.80;0.10\rangle & \langle 0.90;0.05\rangle \\
D,\langle 0.75;0.20\rangle & \langle 0.80;0.18\rangle & \langle 0;1\rangle & \langle 0.35;0.60\rangle & \langle 0.10;0.90\rangle & \langle 0.50;0.48\rangle \\
V,\langle 0.70;0.1\rangle & \langle 0,40;0.50\rangle & \langle 0,35;0.60\rangle & \langle 0;1\rangle & \langle 0.38;0.60\rangle & \langle 0.75;0.20\rangle \\
Br,\langle 0.80;0.10\rangle & \langle 0,80;0.10\rangle & \langle 0,10;0.90\rangle & \langle 0.38;0.60\rangle & \langle 0;1\rangle & \langle 0.45;0.50\rangle \\
Ba,\langle 0.83;0.15\rangle & \langle 0.90;0.05\rangle & \langle 0.50;0.48\rangle & \langle 0.75;0.20\rangle & \langle 0.45;0.50\rangle & \langle 0;1\rangle
\end{array}.$$

The IFPs, presented the vertices and the edges of G^* are calculated using the expert approach [11]. Each of the experts is asked to evaluate the degree of membership of IFP, corresponding to every vertex using the ratio of the air distance to all destinations from it to the total air distance of all the air roads. The minimum degree of membership proposed by the experts for the respective vertex is the degree of membership for this vertex. The degree of non-membership of the vertex is calculted as 1-maximum degree of membership, proposed by the experts. The approach to calculating the IFP for the length of each edge is similar. Each expert estimates the degree of membership of IFP, corresponding to every edge using the ratio of the air distance between two cities to the total air distance of all the air roads in the map of Wizz air. Let us find a Hamiltonian cycle in the G^* using the algorithm 1:

Step 1. Let us create EIFIM C for the given IFG G^*. Then we create IM S such that $S := C$. The condition of the proposition 1 for the existence of a Hamiltonian cycle in G^* is met.

Step 2. The minimum IFP in the S with non-zero membersheep degree is $s_{D,V} = \langle 0.35; 0.60\rangle$.

Step 3. The minimum IFP in the S represents that from Dortmund airport it reaches Vienna airport.

Step 4. In the row "V" the minimum IFP is equal to $\langle 0.38; 0.60\rangle = s_{V,Br}$. Therefore from Vienna airport it goes to Brussels South Charleroi Airport.

Step 5. Repeat *Steps 3-4*, the path obtained is: D-V-Br-Ba-S.

Step 6. From the row "S", select the element $s_{S,D} = \langle 0.80; 0.18\rangle$ with non-zero membership degree to get an IF Hamiltonian cycle "D-V-Br-Ba-S-D." The IF length of the path is: $AGIO1_{\oplus_{(\max,\min)}}\left(C \otimes_{(\min,\max)} X\right) = \langle 0.9; 0.1\rangle$.

V. Conclusion

In this paper it was developed new methods to obtain the IF Hamiltonian cycle in an IFG, using the IFSs and IMs concepts. The main contribution of our approach lies in its ability to find a Hamiltonian cycle not only in a clear but also in an uncertain environment. The proposed algorithms can be generalized to multidimensional intuitionistic fuzzy data [13]. The efficiency of the first algorithm was demonstrated by a real data example from the selected network map of Wizz air. In the paper also was defined three index type operations over EIFIMs. In the future, we will extend this algorithm for an application to the interval-valued intuitionistic fuzzy graphs [14].

References

[1] A. Gani, S. Latha, "A new algorithm to find fuzzy Hamilton cycle in a fuzzy network using adjacency matrix and minimum vertex degree," *SpringerPlus, International Journal of Pure and Applied Mathematics*, vol. 5, 2016, 1854.

[2] A. Kauffman, *Introduction a la Theorie des Sous-emsembles Flous,* Paris: Masson et Cie Editeurs, 1973.

[3] A. Rosenfeld, "Fuzzy graphs," *in: L.A. Zadeh, K.S. Fu, M. Shimura (Eds.), Fuzzy Sets and Their Applications,* Academic Press, New York, 1975, pp. 77–95.

[4] A. Shannon, K. Atanassov, "A first step to a theory of the intuitionistic fuzzy graph," *in D. lakov, ed., Proc. of the First workshop on Fuzzy based expert systems,* Sofia, 1994, pp. 59-61.

[5] E. Szmidt, J. Kacprzyk, "Amount of information and its reliability in the ranking of Atanassov's intuitionistic fuzzy alternatives," *in: Rakus-Andersson and etc. (eds.),* Recent Advances in Decision Making, SCI, Springer, Heidelberg, vol. 222, 2009, pp. 7–19.

[6] GA. Dirac, "Some theorems on abstract graphs," *Proc Lond Math Soc,* vol. 3 (1), 1952, pp. 69–81.

[7] G. Fan, "New sufficient conditions for cycles in graphs," *J Comb Theory Ser B,* vol. 37, 1984, pp. 221–227.

[8] JN. Mordeson, PS. Nai, "Cycles and co-cycles of fuzzy graphs," *Inf Sci,* vol. 90, 1996, pp. 39–49.

[9] K. Atanassov, "Intuitionistic Fuzzy Sets," VII ITKR Session, Sofia, 20-23 June 1983.

[10] K. Atanassov, "Generalized index matrices," *Comptes rendus de l'Academie Bulgare des Sciences,* vol. 40(11), 1987, pp. 15-18.

[11] K. Atanassov, *On Intuitionistic Fuzzy Sets Theory,* STUDFUZZ. Springer, Heidelberg, vol. 283; 2012.

[12] K. Atanassov, *Index Matrices: Towards an Augmented Matrix Calculus. Studies in Computational Intelligence*, Springer, Cham, vol. 573; 2014.

[13] K. Atanassov, "n-Dimensional extended IMs Part 1," *Advanced Studies in Contemporary Mathematics*, vol. 28 (2), 2018, pp. 245-259.

[14] K. Atanassov, *Interval-valued intuitionistic fuzzy sets*, Studies in Fuzziness and Soft Computing, vol. 388; 2020.

[15] K. Atanassov, E. Szmidt, J. Kacprzyk, "On intuitionistic fuzzy pairs," *Notes on Intuitionistic Fuzzy Sets,* vol. 19 (3), 2013, pp. 1-13.

[16] K. Kathirvel, K. Balamurugan, "Method for solving fuzzy transportation problem using trapezoidal fuzzy numbers," *International Journal of Engineering Research and Applications,* vol. 2 (5), 2012, pp. 2154-2158.

[17] K. Zhao, Lai Hong-Jian, Shao Yehang, "New sufficient condition for Hamiltonian graphs," *Appl Math Lett,* vol. 20, 2007, 116–122.

[18] L. Zadeh, *Fuzzy Sets,* Information and Control, vol. 8 (3), 338-353; 1965.

[19] O. Ore, "Note on Hamiltonian circuits." *Am Math Mon,* vol. 67, 1960, pp. 55.

[20] RM. Karp, "Reducibility among combinatorial problems," *in: Miller RE, Thatcher JW (eds),* Complexity of computations, Plemem Press, New York, 1972, pp. 85–103.

[21] S. Skiena, *Hamiltonian cycles. Implementing discrete mathematics: combinatorics and graph theory with mathematica reading,* Addison Wesley, New York; 1990, pp 196–198.

[22] V. Andonov, "On some properties of one Cartesian product over Intuitionistic fuzzy sets." *Notes on Intuitionistic Fuzzy Sets,* vol. 14 (1), 2008,12–19.

[23] V. Chvatal, "On Hamilton's ideals," *J Combin Theory Ser B,* vol. 12, 1972, pp. 63–168.

[24] V. Traneva, "Internal operations over 3-dimensional extended index matrices," *Proceedings of the Jangjeon Mathematical Society,* vol. 18 (4), 2015, pp. 547-569.

[25] V. Traneva, S. Tranev, V. Atanassova, "An Intuitionistic Fuzzy Approach to the Hungarian Algorithm," *in: G. Nikolov et al. (Eds.): NMA 2018,* LNCS 11189, Springer Nature Switzerland, AG, 2019, pp. 1–9.

[26] V. Traneva, S. Tranev, M. Stoenchev, K. Atanassov, " Scaled aggregation operations over two- and three-dimensional index matrices," *Soft computing,* vol. 22, 2019, pp. 5115-5120.

[27] V. Traneva, S. Tranev, *Index Matrices as a Tool for Managerial Decision Making,* Publ. House of the Union of Scientists, Bulgaria; 2017

[28] http://wizz.air-bg.com, last accessed 29 june 2020.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 349–358 DOI: 10.15439/2020F61 ISSN 2300-5963 ACSIS, Vol. 21

Intuitionistic Fuzzy Transportation Problem by Zero Point Method

Velichka Traneva
“Prof. Asen Zlatarov" University
“Prof. Yakimov" Blvd, Burgas 8000, Bulgaria
Email: veleka13@gmail.com

Stoyan Tranev
“Prof. Asen Zlatarov" University
“Prof. Yakimov" Blvd, Burgas 8000, Bulgaria
Email: tranev@abv.bg

***Abstract*—The transportation problems (TPs) support the optimal management of the transport deliveries. In classical TPs the decision maker has information about the crisp values of the transportation costs, availability and demand of the products. Sometimes in the parameters of TPs in real life there is ambiguity and vagueness caused by uncontrollable market factors.**

Uncertain values can be represented by fuzzy sets (FSs) of Zadeh. The FSs have the degrees of membership and non-membership. The concept of intuitionistic fuzzy sets (IFSs) originated in 1983 as an extension of FSs. Atanasov’s IFSs also have a degree of hesitansy to representing the obscure environment.

In this paper we formulate the TP, in which the transportation costs, supply and demand values are intuitionistic fuzzy pairs (IFPs), depending on the diesel prices, road condition, weather and other factors. Additional constraints are included in the problem: limits for the transportation costs. Its main objective is to determine the quantities of delivery from producers to buyers to maintain the supply and demand requirements at the cheapest transportation costs. The aim of the paper is to extend the fuzzy zero point method (FZPM [35]) to the intuitionistic FZPM (IFZPM) to find an optimal solution of the intuitionistic fuzzy TP (IFTP) using the IFSs and index matrix (IM) concepts, proposed by Atanassov. The solution algorithm is demonstrated by a numerical example. Its optimal solution is compared with that obtained by the intuitionistic fuzzy zero suffix method (IFZSM).

I. Introduction

The TP originally proposed by Hitchcock in 1941 [12]. Dantzig, in 1951, used simplex method to the TP [13]. The first overall, finished method for solving TP (“method of potentials") is developed by Kantorovich in 1949 [26].

In classical TP the decision maker has information about the values of the transportation costs, the demanded and offered quantities of the product. In real-life transportation problems, some of its parameters are uncertain due to climatic, road conditions or other market conditions. The costs are fuzzy in the absence of information or in uncertain environment. Zadeh proposed the fuzzy set (FS) theory [27] in 1963 to deal with uncertainty. In 1983, Atanassov proposed the IFSs [17], which is an extension of FSs of Zadeh. The main difference between FSs and IFSs is that the IFSs have a degree of hesitancy.

The following is a brief theoretical overview in the field of fuzzy (FTPs) or intuitionistic FTPs (IFTPs). Chanas et al., in 1984, has proposed a fuzzy linear programming model for solving TPs with clear transportation costs, fuzzy supply and demand values [39]. Gen et al. have given a genetic algorithm for finding an optimal solution of a bicriteria solid TP with fuzzy numbers (FNs) [28]. Jimenez and Verdegay, in 1999, researched fuzzy Solid TP with trapezoidal FNs and presented a genetic approach for solving FTP [11]. Liu and Kao [41] demostrated a method, based on Zadeh’s extension principle, to find the optimal solution of the trapezoidal FTPs. Dinagar and Palanivel [9] have described fuzzy Vogel’s approximation method and modified distribution method for determining an initial solution of trapezoidal FTPs. Pandian and Natarajan, in 2010, studied zero point method for solution for FTP with trapezoidal fuzzy parameters [35]. Improved zero point methods were described in (see [1], [2], [43]) for solving trapezoidal and triangular FTP.

Kaur and Kumar, in 2012, introduced fuzzy least cost method, fuzzy north west corner rule and fuzzy Vogel approximation method for determining of an optimal solution of FTP [5]. Basirzadeh [16] has found a fuzzy optimal solution of fully FTPs by transforming the fuzzy parameters into the crisp parameters using classical algorithms. Gani et al. [3] used Arsham and Khan’s simplex algorithm [15] to find a fuzzy optimal solution of FTPs with trapezoidal fuzzy parameters. A comparative analysis on the FTPs [42] was made and the conclusion has given that the zero point method is better than both the modified distribution method and Vogel’s Approximation method. Patil and Chandgude, in 2012, performed “Fuzzy Hungarian approach” for TP with trapezoidal FNs [7]. Aggarwal and Gupta, in 2013, described an procedure for solving intuitionistic fuzzy TP (IFTP) with trapezoidal IFNs via ranking method [14]. Jahihussain and Jayaraman, in 2013, presented a zero suffix method for obtaining an optimal solution for FTPs with triangular and trapezoidal FNs (see [37], [38]). Zero suffix method to solve FTP after its converting into the crisp problem was applied in [32] and [44]. A fuzzified version of zero suffix method was performed and applied in [29], in 2018, to FTPs. Shanmugasundari and Ganesan, in 2013, proposed a fuzzy modified distribution algorithm and a fuzzy approximation method of Vogel to solve FTP with FNs [30]. Gani and Abbas, in 2014 [4], and Kathirvel, and

This work on Sect. I and Sect. II is supported by the project of Asen Zlatarov University under Ref. No. NIX-423/2019 “Innovative methods for extracting knowledge management” The work on Sect. III and Sect. IV is supported by the Ministry of Education and Science under the Programme “Young scientists and postdoctoral students”, approved by DCM # 577/17.08.2018.

Balamurugun, in 2012 (see [24], [25]), proposed a method for solving TP in which the quantities demanded and offered are represented in the form of the trapezoidal intuitionistic FNs (IFNs). Antony et al. used Vogel's approximation method for solving triangular IFTP in 2014 [36]. "PSK method" for finding an optimal solution to IFTPs was presented by Kumar and Hussain in 2015 [33]. Fully FTPs was resolved in [40], in 2017, using a new method, based on the Hungarian and MODI algorithm. Two new methods for finding a fuzzy optimal solution of TPs with the LR flat fuzzy numbers were proposed by Kaur, Kacprzyk and Kumar [6], based on the tabular representation and on the fuzzy linear programming formulation. In [49], we have proposed for the first time the IFZSM to determine an optimal solution of the IFTP, interpreted by the IFSs and IMs [18] concepts.

Here, we proposed for the first time intuitionistic fuzzy zero point method (IFZPM) to solve optimally a type of TP, in which the transportation costs, supply and demand quantities are IFPs, depending on the climatic, road conditions and economic factors. The constraints are formulated to the problem additionally: limits to the transportation costs. The optimal solution algorithm is demonstrated with a numerical example. The optimal solutions, respectively obtained after the application of the intuitionistic fuzzy zero suffix method (IFZSM) and IFZPM, are compared. The two methods for finding an optimal solution for IFTPs are free from the problem of degeneracy. The optimal transportation cost of the studied TP, obtained by the IFZPM is better than or equal to that after the application of the IFZSM. The advantages of the algorithm are that it can be easy generalized for an application to multidimensional data and can be applied to both the TP with clear or known parameters, and with intuitionistic fuzzy ones. The structure of this paper is as follows: Section 2 recalls some remarks of the theories of the IMs and the IFPs. In Section 3, we propose an algorithm for IFTP extending the fuzzy zero point method [35] and using the concepts of IMs and IFSs. The reliability of the proposed approach is demonstrated by an example in Section 4 and the results are compared with those obtained after application of IFZSM. Section 5 outlines the conclusion and some directions for future research.

II. Introduction to IMs and intuitionistic fuzzy logic

In this section we recall some basic definitions on intuitionistic fuzzy pairs from (see [10], [19], [21], [23], [46]) and on index matrix apparatus from (see [20], [48]).

2.1. Short Remarks on Intuitionistic Fuzzy (IF) Logic

The **IFP** has the form of an ordered pair $\langle a,b\rangle = \langle \mu(p), \nu(p)\rangle$, where $a,b \in [0,1]$ and $a+b \leq 1$, that is used as an evaluation of a proposition p (see [21], [23]). $\mu(p)$ and $\nu(p)$ respectively determine the "truth degree" (degree of membership) and "falsity degree" (degree of non-membership).

Let us recall some basic operations as "negation", "addition", "subtraction", "multiplication" over two IFPs $x = \langle a,b\rangle$ and $y = \langle c,d\rangle$.

$$\begin{gathered}
\neg x = \langle b,a\rangle;\\
x \wedge_1 y = \langle \min(a,c), \max(b,d)\rangle;\\
x \vee_1 y = \langle \max(a,c), \min(b,d)\rangle;\\
x \wedge_2 y = x+y = \langle a+c-a.c, b.d\rangle;\\
x \vee_2 y = x.y = \langle a.c, b+d-b.d\rangle;\\
\alpha.x = \langle 1-(1-a)^{\alpha}, b^{\alpha}\rangle (\alpha \in R);\\
x-y = \langle \max(0,a-c), \min(1,b+d,1-a+c)\rangle.
\end{gathered} \tag{1}$$

The forms of the relations with IFPs are the following

$$\begin{array}{cc}
x \geq y \text{ iff } a \geq c \text{ and } b \leq d; & x \leq y \text{ iff } a \leq c \text{ and } b \geq d;\\
x \geq_{\square} y \text{ iff } a \geq c; & x \leq_{\square} y \text{ iff } a \leq c;\\
x \geq_{\diamond} y \text{ iff } b \leq d; & x \leq_{\diamond} y \text{ iff } b \geq d;\\
x = y & \text{iff } a = c \text{ and } b = d\\
x \geq_R y & \text{iff } R_{\langle a,b\rangle} \leq R_{\langle c,d\rangle},
\end{array} \tag{2}$$

where

$$R_{\langle a,b\rangle} = 0.5(2-a-b)0.5(|1-a|+|b|+|1-a-b|) \text{ [10].}$$

The IFP x is an **"intuitionistic fuzzy false pair" (IFFP)** if and only if $a \leq b$, while x is a **"false pair" (FP)** iff $a = 0, b = 1$.

Let a set E be fixed. An **"intuitionistic fuzzy set" (IFS)** A in E is an object of the following form (see [19]):

$$A = \{\langle x, \mu_A(x), \nu_A(x)\rangle | x \in E\},$$

where $\mu_A : E \to [0,1]$ and $\nu_A : E \to [0,1]$ define the degrees of membership and non-membership of the $x \in E$, respectively, and $0 \leq \mu_A(x) + \nu_A(x) \leq 1$ for every $x \in E$:

2.2. Definition, Operations and Relations over Intuitionistic Fuzzy Index Matrices

Let $\mathscr{I}$ be a fixed set. The definition of two-dimensional intuitionistic fuzzy index matrix (2-D IFIM) with index sets K and L ($K, L \subset \mathscr{I}$) is the following:

$$[K, L, \{\langle \mu_{k_i,l_j}, \nu_{k_i,l_j}\rangle\}]$$

$$\equiv \begin{array}{c|ccccc}
 & l_1 & \dots & l_j & \dots & l_n\\
\hline
k_1 & \langle \mu_{k_1,l_1}, \nu_{k_1,l_1}\rangle & \dots & \langle \mu_{k_1,l_j}, \nu_{k_1,l_j}\rangle & \dots & \langle \mu_{k_1,l_n}, \nu_{k_1,l_n}\rangle\\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots\\
k_m & \langle \mu_{k_m,l_1}, \nu_{k_m,l_1}\rangle & \dots & \langle \mu_{k_m,l_j}, \nu_{k_m,l_j}\rangle & \dots & \langle \mu_{k_m,l_n}, \nu_{k_m,l_n}\rangle
\end{array},$$

where for $i = 1,...,m; j = 1,...,n$:

$$0 \leq \mu_{k_i,l_j}, \nu_{k_i,l_j}, \mu_{k_i,l_j} + \nu_{k_i,l_j} \leq 1.$$

The basic operations over two IMs

$$A = [K, L, \{\langle \mu_{k_i,l_j}, \nu_{k_i,l_j}\rangle\}]$$

and

$$B = [P, Q, \{\langle \rho_{p_r,q_s}, \sigma_{p_r,q_s}\rangle\}]$$

are as follows [20]:

Negation: $\neg A = [K, L, \{\langle \nu_{k_i,l_j}, \mu_{k_i,l_j} \rangle\}]$.
Addition-$(\circ,*)$**:** $A \oplus_{(\circ,*)} B = [K \cup P, L \cup Q, \{\langle \phi_{t_u,v_w}, \psi_{t_u,v_w} \rangle\}]$, where $\langle \phi_{t_u,v_w}, \psi_{t_u,v_w} \rangle$

$$= \begin{cases} \langle \mu_{k_i,l_j}, \nu_{k_i,l_j} \rangle, & \text{if } t_u = k_i \in K \text{ and } v_w = l_j \in L - Q \\ & \text{or } t_u = k_i \in K - P \text{ and } v_w = l_j \in L; \\ \langle \rho_{p_r,q_s}, \sigma_{p_r,q_s} \rangle, & \text{if } t_u = p_r \in P \text{ and } v_w = q_s \in Q - L \\ & \text{or } t_u = p_r \in P - K \\ & \text{and } v_w = q_s \in Q; \\ \langle \circ(\mu_{k_i,l_j}, \rho_{p_r,q_s}), & \text{if } t_u = k_i = p_r \in K \cap P \\ *(\nu_{k_i,l_j}, \sigma_{p_r,q_s}) \rangle, & \text{and } v_w = l_j = q_s \in L \cap Q; \\ \langle 0, 1 \rangle, & \text{otherwise.} \end{cases}$$

where $\langle \circ, * \rangle \in \{\langle \max, \min \rangle, \langle \min, \max \rangle, \langle \text{ average,average} \rangle\}$.
Termwise subtraction-(max,min):

$$A -_{(\max,\min)} B = A \oplus_{(\max,\min)} \neg B.$$

Termwise multiplication-$(\min, \max)$:

$$A \otimes_{(\min,\max)} B = [K \cap P, L \cap Q, \{\langle \phi_{t_u,v_w}, \psi_{t_u,v_w} \rangle\}],$$

where

$$\langle \phi_{t_u,v_w}, \psi_{t_u,v_w} \rangle = \langle \min(\mu_{k_i,l_j}, \rho_{p_r,q_s}), \max(\nu_{k_i,l_j}, \sigma_{p_r,q_s}) \rangle.$$

Transposition: A' is the transposed IM of A.
Reduction: The symbol "$\perp$" denotes the lack of some component in the definitions. The operation $(k, \perp)$-reduction of the IM A is defined by:

$$A_{(k,\perp)} = [K - \{k\}, L, \{c_{t_u,v_w}\}],$$

where $c_{t_u,v_w} = a_{k_i,l_j}$ for $t_u = k_i \in K - \{k\}$ and $v_w = l_j \in L$.
Projection: Let $M \subseteq K$ and $N \subseteq L$. Then,

$$pr_{M,N} A = [M, N, \{b_{k_i,l_j}\}],$$

where for each $k_i \in M$ and each $l_j \in N$, $b_{k_i,l_j} = a_{k_i,l_j}$.
Substitution: Let IM $A = [K, L, \{a_{k,l}\}]$ be given. The some forms of the substitution over A are defined for the couples of indices (p,k) and/or (q,l), respectively, by

$$\left[\frac{p}{k}; \perp\right] A = \left[(K - \{k\}) \cup \{p\}, L, \{a_{k,l}\}\right],$$

$$\left[\perp; \frac{q}{l}\right] A = \left[K, (L - \{l\}) \cup \{q\}, \{a_{k,l}\}\right].$$

Index type operations:

$$AGIndex_{\{(\min/\max)/(\min_{\square}/\max_{\square})/(\min_{\diamond}/\max_{\diamond})(\min_R/\max_R)\}(\not\perp)}(A)$$

$$= \langle k_i, l_j \rangle$$

finds the index of the minimum/ maximum element of A with no empty value in accordance with the relations (2).

$$AGIndex_{\{(\min/\max)/(\min_{\square}/\max_{\square})/(\min_{\diamond}/\max_{\diamond})(\min_R/\max_R)\}(\not\perp)(\notin F)}$$

$$(A) = \langle k_i, l_j \rangle$$

presents the index of the minimum/ maximum element between the elements of A, whose indexes $\notin F$, with no empty value in accordance with the relations (2).

$$Index_{\{(\min/\max)/(\min_{\square}/\max_{\square})/(\min_{\diamond}/\max_{\diamond})(\min_R/\max_R)\}(\not\perp),k_i}(A)$$
$$= \{\langle k_i, l_{v_1} \rangle, \ldots, \langle k_i, l_{v_x} \rangle, \ldots, \langle k_i, l_{v_V} \rangle\},$$

where $\langle k_i, l_{v_x} \rangle$ (for $i = 1, ..., m; j = 1, ..., n; x = 1, ..., V$) are the indices of the minimum/ maximum IFFP of k_i-th row of A with no empty value in accordance with the relations (2).

$$Index_{(\not\perp)}(A) = \{\langle k_1, l_{v_1} \rangle, \ldots, \langle k_i, l_{v_i} \rangle, \ldots, \langle k_m, l_{v_m} \rangle\},$$

where $\langle k_i, l_{v_i} \rangle$ (for $1 \leq i \leq m$) are the indices of the element of A, whose cell is full.

$$Index_{(\max \mu(\nu)),k_i}(A) = \{\langle k_i, l_{v_1} \rangle, \ldots, \langle k_i, l_{v_x} \rangle, \ldots, \langle k_i, l_{v_V} \rangle\},$$

where $\langle k_i, l_{v_x} \rangle$ (for $1 \leq i \leq V, 1 \leq x \leq n$) is the indices of the IFFP of k_i-th row of A, for which $\mu(\nu)_{k_i,l_{v_x}}$ is maximum.

$$Index_{(\max \mu(\nu)),l_j}(A) = \{\langle k_{w_1}, l_j \rangle, \ldots, \langle k_{w_y}, l_j \rangle, \ldots, \langle k_{w_W}, l_j \rangle\},$$

where $\langle k_{w_y}, l_j \rangle$ (for $1 \leq y \leq W, 1 \leq j \leq n$) are the indices of the IFFP of l_j-th column of A, for which $\mu(\nu)_{k_{w_y},l_j}$ is maximum.
Aggregation operations
Let us use the operations $\#_q, (q \leq i \leq 3)$ from [47] for scaling aggregation operations over two IFPs $x = \langle a, b \rangle$ and $y = \langle c, d \rangle$:
$x\#_1 y = \langle min(a,c), max(b,d) \rangle$;
$x\#_2 y = \langle average(a,c), average(b,d) \rangle$;
$x\#_3 y = \langle max(a,c), min(b,d) \rangle$.
The following inequality holds:

$$x\#_1 y \leq x\#_2 y \leq x\#_3 y \quad [47].$$

Let $k_0 \notin K$ be a fixed index. The definition of the aggregation operation by the dimension K is [20], [47]: is:

$$\alpha_{K,\#_q}(A, k_0)$$

$$= \begin{array}{c|ccc} & l_1 & \ldots & l_n \\ \hline k_0 & \underset{i=1}{\overset{m}{\#_q}} \langle \mu_{k_i,l_1}, \nu_{k_i,l_1} \rangle & \ldots & \underset{i=1}{\overset{m}{\#_q}} \langle \mu_{k_i,l_n}, \nu_{k_i,l_n} \rangle \end{array},$$

where $1 \leq q \leq 3$.
Aggregate global internal operation: $AGIO_{\oplus_{(\max,\min)}}(A)$.
This operation finds the addition of all elements of A.
Internal subtraction of the components of the IM A **([45], [46], [48]):**

$$IO_{-_{(\max,\min)}}(\langle k_i, l_j, A \rangle, \langle p_r, q_s, B \rangle) = [K, L, \{\langle \gamma_{t_u,v_w}, \delta_{t_u,v_w} \rangle\}]$$

$$\langle \gamma_{t_u,v_w}, \delta_{t_u,v_w} \rangle$$

$$= \begin{cases} \langle \mu_{t_u,v_w}, \nu_{t_u,v_w} \rangle, & \text{if } t_u \neq k_i \in K, \\ & v_w \neq l_j \in L; \\ \langle \max(0, \mu_{k_i,l_j} - \rho_{p_r,q_s}), & \text{if } t_u = k_i \in K, \\ \min(1, \nu_{k_i,l_j} + \sigma_{p_r,q_s}, 1 - \mu_{k_i,l_j} + \rho_{p_r,q_s}) \rangle & v_w = l_j \in L \end{cases}$$

where $k_i \in K$, $l_j \in L$; $p_r \in P$, $q_s \in Q$.
The non-strict relation "inclusion about value" The form of this type of relations between two IMs A and B is as follows:
$A \subseteq_v B$ iff $(K = P)$ & $(L = Q)$ & $(\forall k \in K)(\forall l \in L)(a_{k,l} \leq b_{k,l})$.

III. Intuitionistic fuzzy zero point approach to the IFTP

Let us extend the IFTP from [49]: A trader supplies a product to n different companies (consumers) $\{l_1,\dots,l_j,\dots,l_n\}$ after delivery of that product from different m manifacturers (producers) $\{k_1,\dots,k_i,\dots,k_m\}$ in quantities $c_{k_i,R}$ (for $1 \le i \le m$). Let the consumers (destinations) need this product in quantities of c_{Q,l_j} (for $1 \le j \le n$).

Let c_{k_i,l_j} be the intuitionistic fuzzy cost for transporting one unit quantity of the product from the k_i-th producer to the l_j-th consumer; x_{k_i,l_j} - the number of units of the product, transported from k_i-th source to l_j-th destination and c_{pl,l_j} (for $1 \le j \le n$) are limits to the transportation costs of the delivery a product from the k_i-th manifacturer to the l_j-th destination under form of IFPs.

All parameters, involved in the problem, are IFPs. For estimating the transportation costs in the form of IFPs, we can use the expert approach described in detail in [19]. Each expert needs to evaluate at least a part of the alternatives in terms of their performance with respect to each defined criterion. The experts is not sure about the transportation costs due the climatic and traffic conditions, or economic factors. He hesitates in prediction of the transportation cost due to changes in some uncontrollable factors. The transportation costs are evaluated as intuitionistic fuzzy numbers after a thorough discussion, interpreted by the intuitionistic fuzzy concept: these numbers express a “positive” and a “negative” evaluations, respectively. The reliability of the expert assessment (confidence in her/his evaluation with respect to each criterion) may be involved in the evaluation process. The purpose of the trader is how to satisfy the requests of the users so that the intuitionistic fuzzy transportation cost is minimum according to (2).

Let us formulate the mathematical model of the above problem:

$$\begin{aligned}
&\text{An objective function: minimize } \sum_{i=1}^{m}\sum_{j=1}^{n} c_{k_i,l_j} x_{k_i,l_j}\\
&\text{Subject to: } \sum_{j=1}^{n} x_{k_i,l_j} = c_{k_i,R}, \qquad i = 1,2,\dots,m \qquad\qquad (3)\\
&\sum_{i=1}^{m} x_{k_i,l_j} = c_{Q,l_j}, \qquad j = 1,2,\dots,n
\end{aligned}$$

We add the constraint to the problem (3): c_{pl,l_j}, for $1 \le j \le n$ – an intuitionistic fuzzy upper limit to the corresponding transportation cost of delivery a particular product from the k_i-th source to the l_j-th destination.

Note: The operations “addition” and “multiplication”, used in the problem (3) are those for IFPs, defined in Sect. II.

The transportation costs of the problem (3) for delivery from a given manufacturer to a given user are entered in the cost IM C:

$$C[K,L]$$

$$= \begin{array}{c|ccccc}
 & l_1 & \dots & l_n & R & pu \\ \hline
k_1 & \langle \mu_{k_1,l_1}, \nu_{k_1,l_1} \rangle & \dots & \langle \mu_{k_1,l_n}, \nu_{k_1,l_n} \rangle & \langle \mu_{k_1,R}, \nu_{k_1,R} \rangle & \langle \mu_{k_1,pu}, \nu_{k_1,pu} \rangle \\
\vdots & \vdots & \ddots & \vdots & \vdots & \\
k_m & \langle \mu_{k_m,l_1}, \nu_{k_m,l_1} \rangle & \dots & \langle \mu_{k_m,l_n}, \nu_{k_m,l_n} \rangle & \langle \mu_{k_m,R}, \nu_{k_m,R} \rangle & \langle \mu_{k_m,pu}, \nu_{k_m,pu} \rangle \\
Q & \langle \mu_{Q,l_1}, \nu_{Q,l_1} \rangle & \dots & \langle \mu_{Q,l_n}, \nu_{Q,l_n} \rangle & \langle \mu_{Q,R}, \nu_{Q,R} \rangle & \langle \mu_{Q,pu}, \nu_{Q,pu} \rangle \\
pl & \langle \mu_{pl,l_1}, \nu_{pl,l_1} \rangle & \dots & \langle \mu_{pl,l_n}, \nu_{pl,l_n} \rangle & \langle \mu_{pl,R}, \nu_{pl,R} \rangle & \langle \mu_{pl,pu}, \nu_{pl,pu} \rangle \\
pu_1 & \langle \mu_{pu_1,l_1}, \nu_{pu_1,l_1} \rangle & \dots & \langle \mu_{pu_1,l_n}, \nu_{pu_1,l_n} \rangle & \langle \mu_{pu_1,R}, \nu_{pu_1,R} \rangle & \langle \mu_{pu_1,pu}, \nu_{pu_1,pu} \rangle
\end{array},$$

where $K = \{k_1,k_2,\dots,k_m,Q,pl,pu_1\}$, $L = \{l_1,l_2,\dots,l_n,R,pu\}$ and for $1 \le i \le m$, $1 \le j \le n$, $\{c_{k_i,l_j}, c_{k_i,R}, c_{k_i,pu}, c_{pl,l_j}, c_{pl,R}, c_{pl,pu}, c_{Q,l_j}, c_{Q,R}, c_{Q,pu}, c_{pu_1,l_j}, c_{pu_1,R}, c_{pu_1,pu}\}$ are IFPs.

Let we denote by $|K| = m+3$ the number of elements of the set K; then $|L| = n+2$. We also define the IM

$$X[K*,L*] = \begin{array}{c|ccccc}
 & l_1 & \dots & l_j & \dots & l_n \\ \hline
k_1 & x_{k_1,l_1} & \cdots & x_{k_1,l_j} & \cdots & x_{k_1,l_n} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
k_m & x_{k_m,l_1} & \dots & x_{k_m,l_j} & \dots & x_{k_m,l_n}
\end{array},$$

$K* = \{k_1,k_2,\dots,k_m\}$, $L* = \{l_1,l_2,\dots,l_n\}$, and for $1 \le i \le m$, $1 \le j \le n$: $x_{k_i,l_j} = \langle \rho_{k_i,l_j}, \sigma_{k_i,l_j} \rangle$.

For the needs of the algorithm, let us we create the following auxiliary index matrices:

1) $S = [K,L,\{s_{k_i,l_j}\}]$, such that $S = C$ i.e. ($s_{k_i,l_j} = c_{k_i,l_j}$ $\forall k_i \in K, \forall l_j \in L$);

2)

$$D[K*,L*] = \begin{array}{c|ccccc}
 & l_1 & \dots & l_j & \dots & l_n \\ \hline
k_1 & d_{k_1,l_1} & \cdots & d_{k_1,l_j} & \cdots & d_{k_1,l_n} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
k_m & d_{k_m,l_1} & \dots & d_{k_m,l_j} & \dots & d_{k_m,l_n}
\end{array},$$

where $K* = \{k_1,k_2,\dots,k_m\}$, $L* = \{l_1,l_2,\dots,l_n\}$, and for $i = 1,...,m$; $j = 1,...,n$: $d_{k_i,l_j} = \{1 \text{ or } 2\}$ depending on whether the elements s_{k_i,l_j} of S are crossed out with 1 or 2 lines.

3)

$$RC[K*,e_0] = \begin{array}{c|c}
 & e_0 \\ \hline
k_1 & rc_{k_1,e_0} \\
\vdots & \vdots \\
k_m & rc_{k_m,e_0}
\end{array},$$

where $K* = \{k_1,k_2,\dots,k_m\}$ and for $1 \le i \le m$: $rc_{k_i,l_j} = \{0 \text{ or } 1\}$ depending on whether the k_i-th row of the matrix S is crossed out or not.

4)

$$CC[r_0,L*] = \begin{array}{c|ccccc}
 & l_1 & \dots & l_j & \dots & l_n \\ \hline
r_0 & cc_{r_0,l_1} & \cdots & cc_{r_0,l_j} & \cdots & cc_{r_0,l_n}
\end{array},$$

where $L* = \{l_1,l_2,\dots,l_n\}$, and for $1 \le j \le n$: $cc_{k_i,l_j} = \{0 \text{ or } 1\}$ depending on whether the l_j-th row of the matrix S is crossed out or not.

5)

$$RM[K/\{Q,pl,pu_1\},R] = pr_{K/\{Q,pl,pu_1\},R}C$$

and

$$CM[pu_1,L/\{R,pu\}] = pr_{pu_1,L/\{R,pu\}}C;$$

6) $U[K*,L*,\{u_{k_i,l_j}\}]$ and for $1 \leq i \leq m$, $1 \leq j \leq n$:

$$u_{k_i,l_j} = \begin{cases} 1, & \text{if } c_{k_i,l_j} < c_{pl,l_j} \\ \perp, & \text{otherwise} \end{cases} ;$$

When starting the algorithm, $rm_{k_i,R} = rc_{k_i,e_0} = cc_{r_0,l_j} = cm_{pu_1,l_j} = 0, u_{k_i,l_j} = \perp, x_{k_i,l_j} = \langle 0,1 \rangle$ $(\forall k_i \in K*, \forall l_j \in L*)$.

We will propose for the first time a new intuitionistic fuzzy approach for determining the optimal solution of the TP with intuitionistic fuzzy costs, demand and supply extending the zero point method ([2], [35], [34], [43]) and using the concepts of IMs and IFPs. In the program code was used a part of Microsoft Visual Studio.NET 2010 C project's.

Step 1. Let us create the IFIM C for the given problem and then, convert it into a balanced one ($\sum_{i=1}^{m} c_{k_i,R} = \sum_{j=1}^{n} c_{Q,l_j}$), if it is not.

The program executes the following operations:

– We define $2-D$ IMs as follows:

$$S_1[Q,L/\{R,pu\}] = pr_{Q,L/\{R,pu\}}C;$$

$$S_2[K/\{Q,pl,pu_1\},R] = pr_{K/\{Q,pu_1\},R}C$$

and let $\{k_{m+1},l_{n+1}\} \notin K \cup L$.

By $L/\{R,pu\}$ let us denote the index set L without the indices R, pu.

– If $\alpha_{K,\#_q}(S_1,l_{n+1}) \supset_v \left[\frac{Q}{R};\perp\right](\alpha_{L,\#_q}(S_2,l_{n+1}))'$ (i.e. $\sum_{i=1}^{m} c_{k_i,R} > \sum_{j=1}^{n} c_{Q,l_j}$), then

introduce dummy column l_{n+1} having all its costs as $\langle 0,1 \rangle$ and execute operations for finding the demand at this dummy destination: $c_{Q,l_{n+1}} = \sum_{i=1}^{m} c_{k_i,R} - \sum_{j=1}^{n} c_{Q,l_j}$;

{Let us define $2-D$ IMs S_3, S_4, S_5 such that

$$S_3 = \alpha_{K,\#_q}(S_1,l_{n+1}) -_{(\max,\min))} \alpha_{L,\#_q}(\left[\frac{Q}{R};\perp\right](S_2,l_{n+1}))';$$

$$S_4 = [K/\{Q,pl,pu_1\},\{l_{n+1}\},\{\langle 0,1 \rangle\}];$$

$$S_5 = [K,\{l_{n+1}\},\{c_{k_i,l_{n+1}}\}] = S_3 \oplus_{(\max,\min))} S_4;$$

The new matrix of costs is obtained by carrying out the operation “matrix addition":

$C := C \oplus_{(\max,\min))} S_5$, go to *Step 2.* }

– If $\left[\perp;\frac{R}{Q}\right]\alpha_{K,\#_q}(S_1,k_{m+1}))' \subset_v \alpha_{L,\#_q}(S_2,k_{m+1}))'$ (i.e. $\sum_{i=1}^{m} c_{k_i,R} < \sum_{j=1}^{n} c_{Q,l_j}$), then

introduce dummy row k_{m+1} having all its costs as $\langle 0,1 \rangle$ and execute operations for finding the demand at this dummy destination: $c_{k_{m+1},R} = \sum_{i=1}^{m} c_{k_i,R} - \sum_{j=1}^{n} c_{Q,l_j}$.

{Let us define $2-D$ IMs S_3, S_4, S_5 such that

$$S_3 = \alpha_{K,\#_q}(C_2,k_{n+1}) -_{(\max,\min))} \left[\perp;\frac{R}{Q}\right]\alpha_{L,\#_q}(C_1,k_{m+1}))';$$

$$S_4[\{k_{m+1}\},L/\{Q,pu\},\{\langle 0,1 \rangle\}];$$

$$S_5 = [k_{m+1},L,\{c_{k_{m+1},l_j}\}] = S_3 \oplus_{(\max,\min))} S_4;$$

$C := C \oplus_{(\max,\min))} S_5$, go to *Step 2.* }

Step 2. Checking the conditions for limiting the transportation costs

for (int $i = 1; i < m; i++$)
for (int $j = 1; j < n; j++$)

$$\{\text{If}\left(\left[\frac{k_i}{pl};\perp\right]pr_{pl,l_j}C\right) \supset_v pr_{k_i,l_j}C, \text{ then} u_{k_i,l_j} = 1.\}$$

$$EG = Index_{(\perp)}(U)$$

$$= \{\langle k_{i_1},l_{j_1}\rangle, \langle k_{i_2},l_{j_2}\rangle, \ldots, \langle\langle k_{i_\phi},l_{j_\phi}\rangle\};$$

for each $\langle k_i,l_j \rangle \in EG$, let us the element s_{k_i,l_j} of S is equal to $\langle 1,0 \rangle$ [31];

Go to *Step 3.*

Step 3. Determination of zero membership value – row level For each row of the matrix S, the smallest element is found in accordance with the relations (2) and is saved to the right of the row, in the column pu. The code uses the operation *AGIO* for finding the indexes of the minimum elements of the row:

for (int $i = 1; i < m; i++$)
for (int $j = 1; j < n; j++$)

$$\{AGIndex_{\{(\min)/(\min_\square)/(\min_\diamond)/(\min_R)\}}\left(pr_{k_i,L/\{R,pu\}}S\right) = \langle k_i,l_{v_j}\rangle;$$

If $pr_{k_i,l_{v_j}}S \subseteq_v \left(\left[\frac{k_i}{pl};\perp\right]pr_{pl,l_{v_j}}S\right)$, then

$$S_6[k_i,l_{v_j}] = pr_{k_i,l_{v_j}}S; S_7 = \left[\perp;\frac{pu}{l_{v_j}}\right]S_6;$$

$$S := S \oplus_{(\max,\min)} S_7.\}$$

Then from each element of the matrix S, subtract the smallest element in the same row:

for (int $i = 0; i < m; i++$)
for (int $j = 0; j < n; j++$)
$\{IO_{-(\max,\min)}\left(\langle k_i,l_j,S\rangle, \langle k_i,pu,pr_{K/\{Q,pl,pu_1\}}S\rangle\right)\}$;
Go to *Step 4.*

Step 4. Determination of zero membership value – column level For each column of the matrix S, the smallest element is found in accordance with the relations (2). It is saved at the bottom of the column, in line pu_1:

for (int $j = 1; j < n; j++$)

$$\{AGIndex_{\{(\min)/(\min_\square)/(\min_\diamond)/(\min_R)\}}\left(pr_{K/\{Q,pl,pu_1\},l_j}S\right)$$

$$= \langle k_{w_i},l_j\rangle;$$

Let us create two 2-D IMs S_6 and S_7 :

$$S_6[k_{w_i},l_j] = pr_{k_{w_i},l_j}S; S_7 = \left[\frac{pu_1}{k_{w_i}};\perp\right]S_6;$$

$$S := S \oplus_{(\max,\min)} S_7.\}$$

for (int $j = 1; j < n; j++$)
for (int $i = 1; i < m; i++$)

$$\{IO_{-(\max,\min)}\left(\langle k_i, l_j, S\rangle, \langle pu_1, l_j, pr_{pu_1, L/\{R,pu\}} S\rangle\right)\};$$

Go to *Step 5.*

Step 5. Optimality criterion

1) Check if each quantity offered is less than or equal to the total quantity offered, whose reduced costs are with zero membership degrees.
for (int $i = 1; i < m; i++$)

$$\{Index_{(\min\mu),k_i}(A) = \{\langle k_i, l_{v_1}\rangle, \dots, \langle k_i, l_{v_x}\rangle, \dots, \langle k_i, l_{v_V}\rangle;$$

We create $2-D$ IMs as follows:

$$G_{v_1}[k_i, l_{v_1}] = pr_{k_i, l_{v_1}} C, \dots, G_{v_V}[k_i, l_{v_V}] = pr_{k_i, l_{v_V}} C,$$

$$\text{and } G[k_i, R] = pr_{k_i, R} C;$$

If

$$G[k_i, R] \subseteq_v G_{v_1} +_{(\max,\min)} \dots +_{(\max,\min)} G_{v_x} + \dots +_{(\max,\min)} G_{v_V},$$

then go to *Step 5.2.*
else $\{RM[k_i, R] = 1$ and go to *Step 6.*$\}$
}

2) Check if each required quantity is less than or equal to the total required quantity, whose reduced costs have zero membership degrees.
for (int $j = 1; j < n; j++$)

$$\{Index_{(\min\mu),l_j}(A) = \{\langle k_{w_1}, l_j\rangle, \dots, \langle k_{w_y}, l_j\rangle, \dots, \langle k_{w_W}, l_j\rangle\};$$

We define $2-D$ IMs as follows:

$$G_{w_1}[k_{w_1}, l_j] = pr_{k_{w_1}, l_j} C, \dots, G_{w_W}[k_{w_W}, l_j] = pr_{k_{w_W}, l_j} C,$$

$$\text{and } G[pu_1, l_j] = pr_{pu_1, l_j} C;$$

If

$$G[pu_1, l_j]$$

$$\subseteq_v G_{w_1} +_{(\max,\min)} \dots +_{(\max,\min)} G_{w_y} + \dots +_{(\max,\min)} G_{w_W},$$

then go to *Step 8.*
else $\{CM[pu_1, l_j] = 1$ go to *Step 6.* $\}$
}

Step 6. Revise the cost IM All elements $\langle 0, 1\rangle$ in the S are crossed out with minimum number of lines (horizontal, vertical or both). If there is no element $\langle 0, 1\rangle$ in a given row or column, then the element with the minimum degree of membership is crossed out from that row or column in the cost IM S obtained in *s*tep 4. (omitting the unsatisfied supply and demand of 5.1 and 5.2.

This step introduces IM $D[K*, L*]$, which has the same dimensions as the X matrix. We use it to mark whether an element in the S is crossed out with a horizontal or vertical line, or both.
If

$$d_{k_i, l_j} = 1,$$

s_{k_i, l_j} is crossed out with 1 line;
If

$$d_{k_i, l_j} = 2,$$

the s_{k_i, l_j} element is covered with 2 lines.
We create two matrices $CC[r_0, L*]$ and $RC[K*, e_0]$, in which it is recorded that the element is covered by a line in a row or column in the S matrix.
for (int $i = 1; i < m; i++$)
for (int $j = 1; j < n; j++$)
– If $s_{k_i, l_j} = \langle 0, 1\rangle$ (or $\langle k_i, l_j\rangle \in Index_{(\min\mu),k_i}(S), rm_{k_i, R} = 0$ and $d_{k_i, l_j} = 0$,
then {

$$rc[k_i, e_0] = 1; d_{k_i, l_j} = 1 \ \forall l_j; S_{(k_i, \perp)}$$

}
– If $\{s_{k_i, l_j} = \langle 0, 1\rangle$ (or $\langle k_i, l_j\rangle \in Index_{(\min\mu),k_i}(S), cm_{pu_1, l_j} = 0$ and $d_{k_i, l_j} = 1\}$,
then {

$$d_{k_i, l_j} = 2; cc_{r_0, l_j} = 1; d_{k_i, l_j} = 1 \ \forall k_i; S_{(\perp, l_j)}$$

}.

Step 7. Develop the new revised cost IM We select the minimum IF cost of the S using the relations (2), that is not crossed by the lines in *Step 6*, and subtract it from each of its uncovered elements, and we add it to each of its elements that is covered by two lines. We return to *Step 5.*

$$AGIndex_{(\min,\max)}(S) = \langle k_x, l_y\rangle;$$

(that finds the smallest element index among the elements of the S matrix.)

Subtract S_{k_x, l_y} uncrossed each element of the matrix with reduced prices:

$$IO_{-(\max,\min)}\left(\langle S\rangle, \langle k_x, l_y, S\rangle\right).$$

We add it to each element of S, which is crossed out by two lines, i.e. $d[k_i, l_j] = 2$:
for (int $i = 1; i < m; i++$)
for (int $j = 1; j < n; j++$)
{if $d_{k_i, l_j} = 2$ then create

$$S_1 = pr_{k_x, l_y} C; S_2 = pr_{k_i, l_j} C \oplus_{(\max,\min)} \left[\frac{k_i}{k_x}; \frac{l_j}{l_y}\right] S_1;$$

$$S := S \oplus_{(\max,\min)} S_2;$$

if $d_{k_i, l_j} = 1$ then

$$S := S \oplus_{(+)} pr_{k_i, l_j} C\}.$$

Go to *Step 5.*

Step 8. Determination of a cell for allocation

1) Use relations (2) to select the largest IF cost in the IM

S. If a tie exists, use any arbitrary tie-breaking choice. Let us denote this cell as $c_{k_{i*},l_{j*}}$.

$$AGIndex_{(\max,\min)}(S) = \langle k_{x*}, l_{y*}\rangle;$$

2) Select a single cost with zero degree of membership for allocation corresponding to k_{i*}-th row and/or l_{j*}-th column if exists and assigns the most possible to that cost cell and strike the satisfied IF supply or IF demand.
Let us $s_{k_e,l_g} = min(s_{Index_{(\min\mu),k_{x*}(A)}}, s_{Index_{(\min\mu),l_{j*}}})(A)$.
Then the minimum of the required and offered quantity is assigned to the corresponding s_{k_e,l_g} cell and delete the row/column with exhausted required or offered quantity. So we find the reduced IM S.
We find minimum of $s_{k_e,R}$ and s_{Q,l_g} by the operations:
We create the IMs $S_8[k_e,R] = pr_{k_e,R}S$ and $S_9[Q,l_g] = pr_{Q,l_g}S$;
If $S_8 \subseteq_v \left[\frac{k_e}{l_g};\frac{R}{Q}\right](S_9)'$ (i.e. $\min(s_{k_e,R}, s_{Q,l_g}) = s_{k_e,R}$), then

$$\{X := X \oplus_{(\max,\min)} \left[\perp;\frac{l_g}{R}\right] S_8;$$

We obtain a new matrix with dimensions
$(m+2)\times(n+2)$ by deleting the k_e-th row of the S using the operation “reduction" $S_{(k_e,\perp)}$.
Let us create IM S_{10} as follows:

$$S_{10}[Q,l_g] = S_9 -_{(\max,\min)} \left[\frac{Q}{R};\frac{l_g}{k_e}\right](S_8)';$$

Then $S := S \oplus_{(\max,\min)} S_{10}$; }
If $S_8 \supseteq_v \left[\frac{k_i}{l_j};\frac{R}{Q}\right](S_9)'$ (i.e. $\min(s_{k_i,R}, s_{Q,l_j}) = s_{Q,l_j}$), then {the IM X changes with: $X := X \oplus_{(\max,\min)} \left[\frac{k_e}{Q};\perp\right] S_9$.
We obtain a new matrix with dimensions $(m+3)\times(n+1)$ by reduction of the l_g-th column of S. Let us construct IM S_{11} as follows:

$$S_{11}[k_e,R] = S_8 -_{(\max,\min)} \left[\frac{k_e}{l_g};\frac{R}{Q}\right](S_9)';$$

$$S := S \oplus_{(\max,\min)} S_{11};\}$$

Repeat *Steps 8* until $|S| = 6$ (all the required quantities are satisfied and all the offered quantities are exhausted), i.e. S is reduced to the form

$$S[K^r,L^r] = \begin{array}{c|cc} & R & pu \\ \hline Q & \langle\mu_{Q,R},\nu_{Q,R}\rangle & \langle\mu_{Q,pu},\nu_{Q,pu}\rangle \\ & & \\ pl & \langle\mu_{pl,R},\nu_{pl,R}\rangle & \langle\mu_{pl,pu},\nu_{pl,pu}\rangle \\ pu_1 & \langle\mu_{pu_1,R},\nu_{pu_1,R}\rangle & \langle\mu_{pu_1,pu},\nu_{pu_1,pu}\rangle \end{array};$$

Go to *Step 9.*

Step 9.

$$D = Index_{\not\perp}X$$
$$= \{\langle k_{i*_1}, l_{j*_1}\rangle, \ldots, \langle k_{i*_f}, l_{j*_f}\rangle, \ldots, \langle k_{i*_\varphi}, l_{j*_\varphi}\rangle\}.$$

If the intuitionistic fuzzy feasible solution is degenerated (it contains less than $m+n-1$ (the total number of producers and consumers decreased by 1) occupied cells in the X i.e. $|D| < m+n-1$) [8] then increase the basic cells x_{k_i,l_j} with one to which the minimum transportation cost corresponds. Let us the recorded delivery of this cell is $\langle 0,1\rangle$. The IMs operations are:
If

$$|D| < m+n-1,$$

then
$\{AGIndex_{\{(\min/\max)/(\min_\Box/\max_\Box)/(\min_\Diamond/\max_\Diamond)(\min_R/\max_R)\}(\not\perp)(\notin D)}(C)$

$$= \langle k_\alpha, l_\beta\rangle;$$

$x_{k_{al},l_\beta} = \langle 0,1\rangle\}$.
Go to *Step 10.*

Step 10.
for (int $i = 1; i < m; i++$)
for (int $j = 1; j < n; j++$)
If $x_{k_i,l_j} \neq \langle\perp,\perp\rangle$ and $\langle k_i,l_j\rangle \in EG$ then the problem has not solution [8] and the algorithm stop else
{all the required and offered quantities are exhausted and the algorithm stop. The optimal basic solution $X_{opt}[K*,L*,\{x_{k_i,l_j}\}]$ is obtained.}
for (int $i = 1; i < m; i++$)
for (int $j = 1; j < n; j++$)
If $x_{k_i,l_j} = \langle\perp,\perp\rangle$ then $x_{k_i,l_j} = \langle 0,1\rangle$.
The optimal intuitionistic fuzzy transportation cost is:

$$AGIO^1_{\oplus_{(\max,\min)})}\left(C_{(\{Q,pl,pu_1\},\{R,pu\})} \otimes_{(\min,\max)} X_{opt}\right)$$

or

$$AGIO^2_{\oplus_{(\vee_2)})}\left(C_{(\{Q,pl,pu_1\},\{R,pu\})} \otimes_{(\wedge_2)} X_{opt}\right),$$

where $\vee_2$ and $\wedge_2$ are the operations from (1).

IV. An example of the IFTP

Let us extend the IFTP from [49]: A trader supplies a product to 4 different companies $\{l_1,l_2,l_3,l_4\}$. Let a product be produced at the manufacturers $\{k_1,k_2,k_3\}$ in quantities $c_{k_i,R}$ (for $1\le i\le 3$). Let the companies ($\{l_1,l_2,l_3,l_4\}$) demand this product in an quantity of c_{Q,l_j} (for $1\le j\le 4$) and c_{pl,l_j} (for $1\le j\le 4$) are intuitionistic fuzzy limits to the transportation costs of delivery a particular product from the k_i-th source to the l_j-th destination. The trader is not certain about the transportation costs, the demanded and supplied quantities due to several uncertainties. Let the cost c_{k_i,l_j} for transporting one unit quantity of the product from the k_i-th producer to the l_j-th user is an IFP and is an element of IFIM $C[K,L]$

$$C[K,L] = \left\{ \begin{array}{c|cccccc} & l_1 & l_2 & l_3 & l_4 & R & pu \\ \hline k_1 & \langle 0.6,0.2\rangle & \langle 0.7,0.1\rangle & \langle 0.3,0.1\rangle & \langle 0.8,0.1\rangle & \langle 0.5,0.2\rangle & \langle\perp,\perp\rangle \\ k_2 & \langle 0.5,0.3\rangle & \langle 0.4,0.1\rangle & \langle 0.5,0.1\rangle & \langle 0.3,0.2\rangle & \langle 0.7,0.1\rangle & \langle\perp,\perp\rangle \\ k_3 & \langle 0.4,0.2\rangle & \langle 0.3,0.2\rangle & \langle 0.6,0.1\rangle & \langle 0.7,0.2\rangle & \langle 0.4,0.5\rangle & \langle\perp,\perp\rangle \\ Q & \langle 0.4,0.2\rangle & \langle 0.5,0.3\rangle & \langle 0.6,0.2\rangle & \langle 0.06,0.02\rangle & \langle\perp,\perp\rangle & \langle\perp,\perp\rangle \\ pl & \langle 0.55,0.3\rangle & \langle 0.6,0.4\rangle & \langle 0.75,0.2\rangle & \langle\perp,\perp\rangle & \langle\perp,\perp\rangle & \langle\perp,\perp\rangle \\ pu_1 & \langle\perp,\perp\rangle & \langle\perp,\perp\rangle & \langle\perp,\perp\rangle & \langle\perp,\perp\rangle & \langle\perp,\perp\rangle & \langle\perp,\perp\rangle \end{array}\right.$$

Let x_{k_i,l_j} is the number of units of the product, transported from the k_i-th producer to l_j-th destination (for $1 \leq i \leq 3$ and $1 \leq j \leq 4$) and is an element of IFIM X with initial elements $\langle \bot, \bot \rangle$. The trader wants to satisfy the required quantities of the users so that the intuitionistic fuzzy transportation cost is minimum.

Solution of the problem:

Step 1. The problem is balanced.

Step 2. Checking the conditions for limiting the transportation costs

for (int $i = 1; i < m; i++$)

for (int $j = 1; j < n; j++$)

{If

$$\left(\left[\frac{k_i}{pl}; \bot \right] pr_{pl,l_j} C \right) \subset_v pr_{k_i,l_j} C,$$

then

$$u_{k_i,l_j} = 1$$

}.

The IM C is transformed in:

$$C[K,L] = \left\{ \begin{array}{c|cccccc} & l_1 & l_2 & l_3 & l_4 & R & pu \\ \hline k_1 & \langle 0.6,0.2 \rangle & \langle 1,0 \rangle & \langle 0.3,0.1 \rangle & \langle 1,0 \rangle & \langle 0.5,0.2 \rangle & \langle \bot,\bot \rangle \\ k_2 & \langle 0.5,0.3 \rangle & \langle 0.4,0.1 \rangle & \langle 0.5,0.1 \rangle & \langle 0.3,0.2 \rangle & \langle 0.7,0.1 \rangle & \langle \bot,\bot \rangle \\ k_3 & \langle 0.4,0.2 \rangle & \langle 0.3,0.2 \rangle & \langle 0.6,0.1 \rangle & \langle 0.7,0.2 \rangle & \langle 0.4,0.5 \rangle & \langle \bot,\bot \rangle \\ Q & \langle 0.4,0.2 \rangle & \langle 0.5,0.3 \rangle & \langle 0.6,0.2 \rangle & \langle 0.06,0.02 \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \\ pl & \langle 0.55,0.3 \rangle & \langle 0.6,0.4 \rangle & \langle 0.75,0.2 \rangle & \langle 0.65,0.3 \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \\ pu_1 & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \end{array} \right.$$

Let us define IM $S = [K, L, \{s_{k_i,l_j}\}]$ such that $S = C$.

Step 3. Determination of zero membership value – row level In each row of the $S[K,L]$, the smallest element is found in accordance with the relation (2):

$$\langle a,b \rangle \ \leq_R \ \langle c,d \rangle \text{ iff } R_{\langle a,b \rangle} \geq R_{\langle c,d \rangle}$$

and it is subtracted from all elements in the row and go to *Step 4.*

$$S = \left\{ \begin{array}{c|cccccc} & l_1 & l_2 & l_3 & l_4 & R & pu \\ \hline k_1 & \langle 0.3,0.3 \rangle & \langle 0.7,0.1 \rangle & \langle 0,0.2 \rangle & \langle 0.7,0.1 \rangle & \langle 0.5,0.2 \rangle & \langle 0.3,0.1 \rangle \\ k_2 & \langle 0.2,0.5 \rangle & \langle 0.1,0.3 \rangle & \langle 0.2,0.3 \rangle & \langle 0,0.4 \rangle & \langle 0.7,0.1 \rangle & \langle 0.3,0.2 \rangle \\ k_3 & \langle 0.1,0.4 \rangle & \langle 0,0.4 \rangle & \langle 0.3,0.3 \rangle & \langle 0.4,0.4 \rangle & \langle 0.4,0.5 \rangle & \langle 0.3,0.2 \rangle \\ Q & \langle 0.4,0.2 \rangle & \langle 0.5,0.3 \rangle & \langle 0.6,0.2 \rangle & \langle 0.06,0.02 \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \\ pl & \langle 0.55,0.3 \rangle & \langle 0.6,0.4 \rangle & \langle 0.75,0.2 \rangle & \langle 0.65,0.3 \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \\ pu_1 & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \end{array} \right.$$

Step 4. Determination of zero membership value – column level The smallest element is found for each column of the matrix $S[K,L]$ in accordance with the relation from (2)

$$\langle a,b \rangle \ \leq_R \ \langle c,d \rangle \text{ iff } R_{\langle a,b \rangle} \geq R_{\langle c,d \rangle}$$

and it is subtracted from all elements in the corresponding column and go to *Step 5.*

$$S = \left\{ \begin{array}{c|cccccc} & l_1 & l_2 & l_3 & l_4 & R & pu \\ \hline k_1 & \langle 0.2,0.7 \rangle & \langle 0.7,0.3 \rangle & \langle 0,0.4 \rangle & \langle 0.7,0.3 \rangle & \langle 0.5,0.2 \rangle & \langle 0.3,0.1 \rangle \\ k_2 & \langle 0.1,0.9 \rangle & \langle 0.1,0.7 \rangle & \langle 0.2,0.5 \rangle & \langle 0,0.8 \rangle & \langle 0.7,0.1 \rangle & \langle 0.3,0.2 \rangle \\ k_3 & \langle 0,0.8 \rangle & \langle 0,0.8 \rangle & \langle 0.3,0.5 \rangle & \langle 0.4,0.6 \rangle & \langle 0.4,0.5 \rangle & \langle 0.3,0.2 \rangle \\ Q & \langle 0.4,0.2 \rangle & \langle 0.5,0.3 \rangle & \langle 0.6,0.2 \rangle & \langle 0.06,0.02 \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \\ pl & \langle 0.55,0.3 \rangle & \langle 0.6,0.4 \rangle & \langle 0.75,0.2 \rangle & \langle 0.65,0.3 \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \\ pu_1 & \langle 0.1,0.4 \rangle & \langle 0,0.4 \rangle & \langle 0,0.2 \rangle & \langle 0,04 \rangle & \langle \bot,\bot \rangle & \langle \bot,\bot \rangle \end{array} \right.$$

Step 5. Optimality criterion

1) Check if each required quantity is less than or equal to the total required quantity, whose reduced costs are with zero membership degrees.

2) Check id each quantity offered is less than or equal to the total quantity offered, whose reduced costs have zero membership degrees.

3) If 5.1 and 5.2 are satisfied then go to *Step 8.* else go to *Step 6.*

Step 6. Revise the cost IM Minimum number of lines (horizontal, vertical or both) are drawn to cover all elements $\langle 0,1 \rangle$ in the S. If there is no element $\langle 0,1 \rangle$ in a given row or column, then the element with the minimum degree of membership is crossed out from that row or column in the cost IM S obtained in *Step 4.*

Step 7. Develop the new revised cost IM We select the minimum IF cost of the S that is not crossed by the lines in *Step 6.*, and subtract it from each of its uncovered elements, and we add it to each of its elements that is covered by two lines. We return to *Step 5.*

The *Steps 5., 6. and 7.* are executed twice and then proceeds to *Step 8.* IM S takes the following form after these steps:

$$S = \left\{ \begin{array}{c|cccc} & l_1 & l_2 & l_3 & \dots \\ \hline k_1 & \langle 0,1\rangle & \langle 0.5,0.5\rangle & \langle 0,0.4\rangle & \dots \\ k_2 & \langle 0,1\rangle & \langle 0,1\rangle & \langle 0.28,0.45\rangle & \dots \\ k_3 & \langle 0,0.8\rangle & \langle 0,0.8\rangle & \langle 0.43,0.41\rangle & \dots \\ Q & \langle 0.4,0.2\rangle & \langle 0.5,0.3\rangle & \langle 0.6,0.2\rangle & \dots \\ pl & \langle 0.55,0.3\rangle & \langle 0.6,0.4\rangle & \langle 0.75,0.2\rangle & \dots \\ pu_1 & \langle 0.1,0.4\rangle & \langle 0,0.4\rangle & \langle 0,0.2\rangle & \dots \end{array} \right.$$

$$\left\{ \begin{array}{cccc} \dots & l_4 & R & pu \\ \hline \dots & \langle 0.6,0.4\rangle & \langle 0.5,0.2\rangle & \langle 0.3,0.1\rangle \\ \dots & \langle 0,0.8\rangle & \langle 0.7,0.1\rangle & \langle 0.3,0.2\rangle \\ \dots & \langle 0.46,0.54\rangle & \langle 0.4,0.5\rangle & \langle 0.3,0.2\rangle \\ \dots & \langle 0.06,0.02\rangle & \langle \perp,\perp\rangle & \langle \perp,\perp\rangle \\ \dots & \langle 0.65,0.3\rangle & \langle \perp,\perp\rangle & \langle \perp,\perp\rangle \\ \dots & \langle 0,04\rangle & \langle \perp,\perp\rangle & \langle \perp,\perp\rangle \end{array} \right.$$

Step 8.

1) Use relations from (2) to select the largest IF cost in the IM S. Let us denote this cell as $c_{k_{i*},l_{j*}}$.

2) Select a single cost with zero degree of membership for allocation corresponding to k_{i*}-th row and/or l_{j*}-th column if exists and determine the most possible to that cost cell and strike the satisfied IF supply or IF demand.

Steps 8. is repeated three times until $|S| = 6$ (all the demands are satisfied and all the supplies are exhausted).

Step 9. The intuitionistic fuzzy optimal solution, presented by the IM X_{opt} is non-degenerated, it includes 6 occupied cells. The IM X_{opt} has the following form:

$$X_{opt} = \left\{ \begin{array}{c|cccc} & l_1 & l_2 & l_3 & l_4 \\ \hline k_1 & \langle 0,1\rangle & \langle 0,1\rangle & \langle 0.5,0.2\rangle & \langle 0,1\rangle \\ k_2 & \langle 0.4,0.2\rangle & \langle 0.1,0.8\rangle & \langle 0.1,0.4\rangle & \langle 0.06,0.02\rangle \\ k_3 & \langle 0,1\rangle & \langle 0.4,0.5\rangle & \langle 0,1\rangle & \langle 0,1\rangle \end{array} \right. . \tag{4}$$

Step 10. The optimal intuitionistic fuzzy optimal solution $X_{opt}[K*, L*, \{x_{k_i,l_j}\}]$ is obtained. The optimal intuitionistic fuzzy transportation cost is:

$$AGIO^1_{\oplus_{(\max,\min))}}\left(C_{(\{Q,pl,pu_1\},\{R,pu\})}\otimes_{(\min,\max)}X_{opt}\right) = \langle 0.4,0.2\rangle \tag{5}$$

or

$$AGIO^2_{\oplus_{(\vee_2))}}\left(C_{(\{Q,pl,pu_1\},\{R,pu\})}\otimes_{(\wedge_2)}X_{opt}\right) = \langle 0.464,0.006\rangle. \tag{6}$$

The degree of membership (acceptance) of this optimal solution is equal to 0.4 (or 0.464) and the its degree of non-membership (non-acceptance) is equal to 0.2 (or 0.006).

Let us compare the results, obtained after application of IFZSMA [49] and IFZPM over IFTP, presented in the section IV. The optimal solution IM $X_{opt}[K*, L*]$, obtained after application of IFZSM is as follows [49]:

$$X_{opt} = \left\{ \begin{array}{c|cccc} & l_1 & l_2 & l_3 & l_4 \\ \hline k_1 & \langle 0,1\rangle & \langle 0,1\rangle & \langle 0.5,0.2\rangle & \langle 0,1\rangle \\ k_2 & \langle 0.4,0.2\rangle & \langle 0.2,0.6\rangle & \langle 0.1,0.4\rangle & \langle 0.03,0.02\rangle \\ k_3 & \langle 0,1\rangle & \langle 0.4,0.5\rangle & \langle 0,1\rangle & \langle 0,1\rangle \end{array} \right. . \tag{7}$$

The optimal intuitionistic fuzzy cost of the IFTP is [49]:

$$AGIO^1_{\oplus_{(\max,\min))}}\left(C_{(\{Q,pl,pu_1\},\{R,pu\})}\otimes_{(\min,\max)}X_{opt}\right) = \langle 0.4,0.2\rangle$$

or

$$AGIO^2_{\oplus_{(\vee_2))}}\left(C_{(\{Q,pl,pu_1\},\{R,pu\})}\otimes_{(\wedge_2)}X_{opt}\right) = \langle 0.475,0.005\rangle.$$

The optimal solutions (4) and (7), obtained respectively by the IFZSM and the IFZPM, coincide.

The ranking function R, defined in (2), we can use to rank alternatives of decision-making process. For the obtained optimal solutions of IFZSM and IFZPM $R_{\langle 0.4;0.2\rangle} = 0.42, R_{\langle 0.475;0.005\rangle} = 0.39$, and $R_{\langle 0.464;0.006\rangle} = 0.41$. When we use the pairs of operations $\langle \max,\min\rangle$ and $\langle \min,\max\rangle$ in (5), the optmal transportation cost after IFZSM and IFZPM coincide. When we use the pairs of operations $\vee_2$ and $\langle \wedge_2\rangle$ in (6), the optmal transportation cost after IFZPM is less than the optimal transportation cost after IFZPM.

The example illustrates the reliability of the proposed IFZPM.

V. Conclusion

In this paper it is proposed for the first time to extend the FZPM [2] to IFZPM for determining an optimal solution of a type of IFTP using the concepts of the IMs anf IFSs. The formulated IFTP has additional constraints: upper limits to the transportation costs. The proposed algorithm for solution of the IFTP is illustrated with a numerical example. The optimal solution of the problem in the example is compared with that obtained by the intuitionistic fuzzy zero suffix method (IFZSM). The advantages of the proposed algorithm is that it can be easy generalized to the multidimensional intuitionistic fuzzy TPs [22] and also can be applied to both the TP with crisp parameters and with intuitionistic fuzzy ones.

In the future, we will extend IFZPM to the multidimensional intuitionistic fuzzy TPs [22] and will apply the proposed approach for the TPs in different areas.

References

[1] A. Edwuard Samuel, “Improved zero point method,” *Applied mathematical sciences,* vol. 6 (109), 2012, pp. 5421–5426.

[2] A. Edwuard Samuel, M. Venkatachalapathy, “Improved zero point method for unbalanced FTPs,” *International Journal of Pure and Applied Mathematics,* vol. 94 (3), 2014, pp. 419–424.

[3] A. Gani, A. Samuel, D. Anuradha, “Simplex type algorithm for solving fuzzy transportation problem,” *Tamsui Oxf. J. Inf. Math. Sci.,* vol. 27, 2011, pp. 89–98.

[4] A. Gani, S. Abbas, “A new average method for solving intuitionistic fuzzy transportation problem,” *International Journal of Pure and Applied Mathematics,* vol. 93 (4), 2014, pp. 491-499.

[5] A. Kaur, A. Kumar, “A new approach for solving fuzzy transportation problems using generalized trapezoidal fuzzy numbers,” *Applied Soft Computing,* vol. 12 (3), 2012, pp. 1201-1213.

[6] A. Kaur, J. Kacprzyk and A. Kumar, *Fuzzy transportation and transshipment problems,* Studies in fuziness and soft computing, vol. 385, 2020.

[7] A. Patil, S. Chandgude, “Fuzzy Hungarian Approach for Transportation Model,” *International Journal of Mechanical and Industrial Engineering,* vol. 2 (1), pp. 77-80, 2012.

[8] B. Atanassov, *Quantitative methods in business management,* Publ. houseTedIna, Varna; 1994. (in Bulgarian)

[9] D. Dinagar, K. Palanivel, "On trapezoidal membership functions in solving transportation problem under fuzzy environment," *Int. J. Comput. Phys. Sci.,* vol. 1, 2009, pp. 1–12.

[10] E. Szmidt, J. Kacprzyk, "Amount of information and its reliability in the ranking of Atanassov's intuitionistic fuzzy alternatives," *in: Rakus-Andersson, E., Yager, R., Ichalkaranje, N., Jain, L.C. (eds.),* Recent Advances in Decision Making, SCI, Springer, Heidelberg, vol. 222, DOI: 10.1007/978-3-642-02187-9_2, 2009, pp. 7–19.

[11] F. Jimenez, J. Verdegay, "Solving fuzzy solid transportation problems by an evolutionary algorithm based parametric approach," *European Journal of Operational Research,* vol. 117 (3),1999, pp. 485-510.

[12] F. Hitchcock, "The distribution of a product from several sources to numerous localities," *Journal of Mathematical Physics,* vol. 20, 1941, pp. 224-230.

[13] G. Dantzig, *Application of the simplex method to a transportation problem,* Chapter XXIII, Activity analysis of production and allocation, New York, Wiley, Cowles Commision Monograph, vol. 13, 359-373; 1951.

[14] G. Gupta, A. Kumar, M. Sharma, "A Note on A New Method for Solving Fuzzy Linear Programming Problems Based on the Fuzzy Linear Complementary Problem (FLCP)," *International Journal of Fuzzy Systems,* 2016, pp. 1-5.

[15] H. Arsham, A. Khan, "A simplex type algorithm for general transportation problems-An alternative to stepping stone," *Journal of Operational Research Society,* vol. 40 (6), 2017, pp. 581-590.

[16] H. Basirzadeh, "An approach for solving fuzzy transportation problem, " *Appl. Math. Sci.,* vol. 5, 2011, pp. 1549–1566.

[17] K. Atanassov, "Intuitionistic Fuzzy Sets," VII ITKR Session, Sofia, 20-23 June 1983 (Deposed in Centr. Sci.-Techn. Library of the Bulg. Acad. of Sci., 1697/84) (in Bulgarian). Reprinted: *Int. J. Bioautomation,* vol. 20(S1), 2016, pp. S1-S6.

[18] K. Atanassov, "Generalized index matrices," *Comptes rendus de l'Academie Bulgare des Sciences,* vol. 40(11), 1987, pp. 15-18.

[19] K. Atanassov, *On Intuitionistic Fuzzy Sets Theory,* STUDFUZZ. Springer, Heidelberg, vol. 283; DOI:10.1007/978-3-642-29127-2, 2012.

[20] K. Atanassov, *Index Matrices: Towards an Augmented Matrix Calculus. Studies in Computational Intelligence*, Springer, Cham, vol. 573; DOI: 10.1007/978-3-319-10945-9, 2014.

[21] K. Atanassov, "Intuitionistic Fuzzy Logics," *Studies in Fuzziness and Soft Computing,* Springer, vol. 351, DOI:10.1007/978-3-319-48953-7, 2017.

[22] K. Atanassov, "n-Dimensional extended index matrices Part 1," *Advanced Studies in Contemporary Mathematics*, vol. 28 (2), 2018, pp. 245-259.

[23] K. Atanassov, E. Szmidt, J. Kacprzyk, "On intuitionistic fuzzy pairs," *Notes on Intuitionistic Fuzzy Sets,* vol. 19 (3), 2013, pp. 1-13.

[24] K. Kathirvel, K. Balamurugan, "Method for solving fuzzy transportation problem using trapezoidal fuzzy numbers," *International Journal of Engineering Research and Applications,* vol. 2 (5), 2012, pp. 2154-2158.

[25] K. Kathirvel, K. Balamurugan, "Method for solving unbalanced transportation problems using trapezoidal fuzzy numbers," *International Journal of Engineering Research and Applications,* vol. 3 (4), 2013, pp. 2591-2596.

[26] L. Kantorovich, M. Gavyrin, *Application of mathematical methods in the analysis of cargo flows,* Coll. of articles Problems of increasing the efficiency of transport, M.: Publ. house AHSSSR, 110-138; 1949. (in Russian)

[27] L. Zadeh, *Fuzzy Sets,* Information and Control, vol. 8 (3), 338-353; 1965.

[28] M. Gen, K. Ida, Y. Li, E. Kubota, "Solving bicriteria solid transportation problem with fuzzy numbers by a genetic algorithm," *Computers & Industrial Engineering,* vol. 29 (1), 1995, pp. 537-541.

[29] M. Purushothkumar, M. Ananthanarayanan, S. Dhanasekar, "Fuzzy zero suffix Algorithm to solve Fully Fuzzy Transportation Problems," *International Journal of Pure and Applied Mathematics,* vol. 119 (9), 2018, pp. 79-88.

[30] M. Shanmugasundari, K. Ganesan, "A novel approach for the fuzzy optimal solution of fuzzy transportation problem," *International journal of Engineering research and applications,* vol. 3 (1), 2013, pp. 1416-1424.

[31] N. Lalova, L. Ilieva, S. Borisova, L. Lukov, V. Mirianov, *A guide to mathematical programming,* Science and Art Publishing House, Sofia; 1980 (in Bulgarian)

[32] P. Jayaraman, R. Jahirhussain, "Fuzzy optimal transportation problem by improved zero suffix method via Robust Ranking technique," *International Journal of Fuzzy Mathematics and systems,* vol. 3 (4), 2013, pp. 303-311.

[33] P. Kumar, R. Hussain, "A method for solving unbalanced intuitionistic fuzzy transportation problems," *Notes on Intuitionistic Fuzzy Sets,* vol. 21 (3), 2015, pp. 54-65.

[34] P. Ngastiti, B. Surarso, B. Sutimin, "Zero point and zero suffix methods with robust ranking for solving fully fuzzy transportation problems," *Journal of Physics: Conference Series,* vol. 1022, 2018, pp. 1-10.

[35] P. Pandian, G. Natarajan, "A new algorithm for finding a fuzzy optimal solution for fuzzy transportation problems," *Applied Mathematical Sciences,* vol. 4, 2010, pp. 79- 90.

[36] R. Antony, S. Savarimuthu, T. Pathinathan, "Method for solving the transportation problem using triangular intuitionistic fuzzy number," *International Journal of Computing Algorithm,* vol. 03, 2014, pp. 590-605.

[37] R. Jahirhussain, P. Jayaraman, "Fuzzy optimal transportation problem by improved zero suffix method via robust rank techniques," *International Journal of Fuzzy Mathematics and Systems (IJFMS),* vol. 3, 2013, pp. 303-311.

[38] R. Jahihussain , P. Jayaraman, "A new method for obtaining an optinal solution for fuzzy transportation problems," *International Journal of Mathematical Archive,* vol. 4 (11), 2013, pp. 256-263.

[39] S. Chanas, W. Kolodziejckzy, A. Machaj, "A fuzzy approach to the transportation problem," *Fuzzy Sets and Systems,* vol. 13, 1984, pp. 211-221.

[40] S. Dhanasekar, S. Hariharan, P. Sekar, "Fuzzy Hungarian MODI Algorithm to solve fully fuzzy transportation problems," *Int. J. Fuzzy Syst.,* vol. 19 (5), 2017, pp. 1479-1491.

[41] S. Liu, C. Kao, "Solving fuzzy transportation problems based on extension principle," *Eur. J. Oper. Res.,* vol. 153, 2004, pp. 661–674.

[42] S. Mohideen, P. Kumar, "A Comparative Study on Transportation Problem in Fuzzy Environment," *International Journal of Mathematics Research,* vol. 2 (1), 2010, pp. 151-158.

[43] T. Karthy, K. Ganesan, "Revised improved zero point method for the trapezoidal fuzzy transportation problems," *AIP Conference Proceedings*, 2112, 020063, 2019, pp. 1-8.

[44] V. Sudhagar, V. Navaneethakumar, "Solving the Multiobjective two stage fuzzy transportation problem by zero suffix method," *Journal of Mathematics Research,*vol. 2 (4), 2010, pp. 135-140.

[45] V. Traneva, "Internal operations over 3-dimensional extended index matrices," *Proceedings of the Jangjeon Mathematical Society,* vol. 18 (4), 2015, pp. 547-569.

[46] V. Traneva, S. Tranev, V. Atanassova, "An Intuitionistic Fuzzy Approach to the Hungarian Algorithm," *in: G. Nikolov et al. (Eds.): NMA 2018,* LNCS 11189, Springer Nature Switzerland, AG, 2019, pp. 1–9, DOI: 10.1007/978-3-030-10692-8_19.

[47] V. Traneva, S. Tranev, M. Stoenchev, K. Atanassov, " Scaled aggregation operations over two- and three-dimensional index matrices," *Soft computing,* vol. 22, 2019, pp. 5115-5120, DOI: 10.1007/s00500-018-3315-6.

[48] V. Traneva, S. Tranev, *Index Matrices as a Tool for Managerial Decision Making,* Publ. House of the Union of Scientists, Bulgaria; 2017 (in Bulgarian)

[49] V. Traneva, S. Tranev, "An Intuitionistic fuzzy zero suffix method for solving the transportation problem," *in: Dimov I., Fidanova S. (eds) Advances in High Performance Computing. HPC 2019,* Studies in computational intelligence, Springer, Cham, vol. 902, DOI: 10.1007/978-3-030-55347-0_7, 2020.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 359–367 DOI: 10.15439/2020F197 ISSN 2300-5963 ACSIS, Vol. 21

Feasibility of computerized adaptive testing evaluated by Monte-Carlo and post-hoc simulations

Lubomír Štěpánek
Institute of Biophysics and Informatics
First Faculty of Medicine, Charles University
Salmovská 1, Praha 2
lubomir.stepanek@lf1.cuni.cz

Patrícia Martinková
Institute of Computer Science of the Czech Academy of Sciences
Pod Vodárenskou věží 2, Praha 8
Faculty of Education, Charles University, Myslíkova 7, Praha 1
martinkova@cs.cas.cz

***Abstract*—Computerized adaptive testing (CAT) is a modern alternative to classical paper and pencil testing. CAT is based on an automated selection of optimal item corresponding to current estimate of test-taker's ability, which is in contrast to fixed predefined items assigned in linear test. Advantages of CAT include lowered test anxiety and shortened test length, increased precision of estimates of test-takers' abilities, and lowered level of item exposure thus better security. Challenges are high technical demands on the whole test work-flow and need of large item banks.**

In this study, we analyze feasibility and advantages of computerized adaptive testing using a Monte-Carlo simulation and post-hoc analysis based on a real linear admission test administrated at a medical college. We compare various settings of the adaptive test in terms of precision of ability estimates and test length.

We find out that with adaptive item selection, the test length can be reduced to 40 out of 100 items while keeping the precision of ability estimates within the prescribed range and obtaining ability estimates highly correlated to estimates based on complete linear test (Pearson's $\rho \doteq 0.96$). We also demonstrate positive effect of content balancing and item exposure rate control on item composition.

I. Introduction

MULTI-ITEM assessment instruments find their use in number of areas including admission or other educational tests, psychological measurement, health-related questionnaires, and other behavioral measurements. A usual way to perform achievement testing is by assigning a fixed set of items which are supposed to measure construct of interest, such as knowledge of biology, level of depression, fatigue, or respondent's quality of life.

Given that the abilities may greatly differ across test-takers, the respondents with higher levels of ability may be bored by easier items, while those with lower levels of ability might experience inconvenient stress. An effective and appropriate selection of items which suit the best the test-takers of a given ability can thus be more convenient for respondents, may save time and moreover provide estimates of better precision than fixed tests of the same length.

Adaptive tests [1], [2] have been an alternative to linear tests for decades. The most complex version of adaptive tests is the one in which the item selection is done after each item administration depending on the current estimate of test-taker's ability which is iteratively updated. Multistage tests [3] on the other hand involve assigning blocks of items adaptively depending on the ability estimate from the previous test section.

Basic principles of computerized adaptive test are presented in Figure 1.

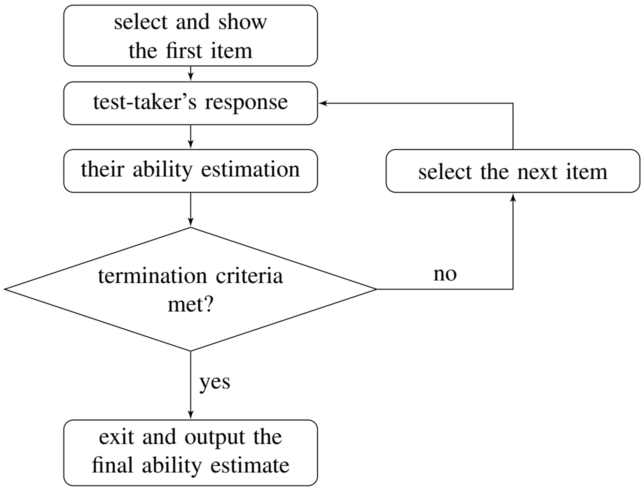

select and show the first item
test-taker's response
their ability estimation
select the next item
termination criteria met?
no
yes
exit and output the final ability estimate

Fig. 1. Computerized adaptive testing flowchart

An adaptive test is initialized by the selection and administration of the first item. The first item can be selected randomly or based on prior ability estimate of the respondent. Average ability can be used as an uninformed estimate, alternatively, initial estimate may be based on respondent's answers to one or a small number of pre-test items.

Depending on the answer to the first item, the test-taker ability estimate is updated. If the termination criterion (such as number of administered items or precision of the estimate) is not met, the updated ability estimate is used to select the next optimal item. This cycle is repeated until the a priori specified termination criterion is met; then, eventually, the test is stopped and final estimate of the test-taker ability is provided as an output.

A. Comparison of linear and adaptive testing

Both the linear and adaptive test scenario have their advantages and disadvantages, respectively. Advantages of adaptive tests have been demonstrated in areas of educational testing [4], testing of psychological distress [5], [6], as well as health-related measurements such as in mobility surveys [7], and testing general disability [8]. While the adaptive tests are usually shorter in terms of number of items and overall time needed to complete the test, they also enable to estimate test-taker's ability with better precision than linear tests of similar length. The lower level of item exposure usually implies also better security when items are administered adaptively.

However, since the adaptive testing is more complex it requires higher technical facility and support of trained experts. The initial setting of the adaptive test may provide number of options which may have crucial impact on functioning of the adaptive test. Therefore, feasibility and optimal setting of CAT with respect to the given item bank and population of test takers need to be analyzed in order to apply the adaptive test effectively and profitably.

In this work, we use Monte Carlo simulations and post-hoc analysis based on real data of admission test administrated at a medical college with the aim to derive the optimal setting of adaptive test. We also compare the precision of different settings and estimate the correlation between the adaptively estimated ability and estimates based on answers to complete set of 100 items. We discuss results for different levels of precision, and various test termination criteria. We also implement content balancing and item exposure rate control to see how it affects performance and properties of the adaptive test. We discuss the findings in context of the admission testing and other educational testing at medical faculties.

The paper proceeds as follows. We firstly describe the data and introduce all necessary background theory, including underlying models, settings of adaptive tests and design of the simulation studies in the *Research Methodology* section. We then present results of the post-hoc analysis and Monte Carlo simulation in Section *Results*. Finally, discussion and final remarks are provided in *Conclusion* section.

II. Research Methodology

A. Data and item calibration

We used data from a real fixed admission test administrated to 2372 test-takers (applicants) at First Faculty of Medicine, Charles University, Prague in 2015 [9], also see [10]. Interactive presentation of psychometric properties of the admission test is available in R package ShinyItemAnalysis [11].

The test consisted of 100 dichotomously scored items covering different Biology topics. For the purpose of this analysis, items were classified into three general domains – genetics, taxonomy, and human biology, respectively. The mutual proportions of these three domains were of nearly equal size.

To evaluate psychometric properties of the items, unidimensional two-parameter logistic (2PL) item-response theory (IRT) model was fitted to describe the probability of a correct answer given applicant's ability [12],

$$\begin{aligned} p_i\left(\theta_p\right) = \Pr\left(U_{pi}=1|\theta_p,\boldsymbol{\xi}_i\right) &= \Psi\left[a_i\left(\theta_p-b_i\right)\right] \\ &= \frac{\exp\left[a_i\left(\theta_p-b_i\right)\right]}{1+\exp\left[a_i\left(\theta_p-b_i\right)\right]}, \end{aligned} \tag{1}$$

where θ_p is the ability of subject $p \in \{1,2,\dots,N\}$, vector $\boldsymbol{\xi}_i = (a_i, b_i)^T$ stands for set of item parameters (discrimination and difficulty, respectively) for item $i \in \{1,2,\dots,I\}$, and $\Psi(\bullet)$ is the logistic function.

In the item calibration phase, the item parameters $(a_i, b_i)^T$ were estimated. To estimate the item parameters, we used marginal maximum likelihood (MML) as follows [13]. Let us assume local independence, i. e. independence of item responses for the same subject given their ability θ_p (within subject). Then the probability $\Pr\left(\mathbf{u}_p|\theta_p,\boldsymbol{\xi}\right)$ of response pattern $\mathbf{u}_p$ of subject p follows the form

$$\Pr\left(\mathbf{u}_p|\theta_p,\boldsymbol{\xi}\right) = \prod_{i=1}^{I} \Pr\left(U_{pi}=u_{pi}|\theta_p,\boldsymbol{\xi}_i\right). \tag{2}$$

Supposing there is no cooperation between subjects, we can also assume independence between subjects (in-between). Let's further denote $\boldsymbol{\xi} = (\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_I)$ the matrix of parameters for all items i. Then the marginal likelihood function takes form

$$L\left(\boldsymbol{\xi},\mu,\sigma;\mathbf{U}\right) = \prod_{p=1}^{N} \Pr\left(\mathbf{u}_p|\boldsymbol{\xi},\mu,\sigma\right) \tag{3}$$

with

$$\begin{aligned} &\Pr\left(\mathbf{u}_p|\boldsymbol{\xi},\mu,\sigma\right) = \\ &= \int\dots\int \Pr\left(\mathbf{u}_p|\theta_p,\boldsymbol{\xi}\right) g\left(\theta_p|\mu,\sigma\right) \mathrm{d}\theta_p, \end{aligned}$$

where μ and σ are the expected value and the variance of respondent ability θ_p. With this approach, abilities θ_p are treated as stochastic variables with normal distribution, $\theta_p \sim \mathcal{N}\left(\mu,\sigma\right)$ and are integrated out [14].

The first-order derivatives with respect to ability parameters θ_p result into the likelihood equations [15] that could be numerically estimated using Expectation-Maximization (EM) algorithm [16], producing the desired estimates of item parameters $\hat{\boldsymbol{\xi}} = \left(\hat{\boldsymbol{\xi}}_1,\dots,\hat{\boldsymbol{\xi}}_I\right)$.

B. Settings of adaptive tests

Initialization. Initial item was selected as the one maximizing observed Fisher information at ability $\theta_0 = 0$, see [17].

Ability estimation. Test-taker's ability is iteratively updated whenever the respondent answers to a given item and the answer is collected. Beginning with the equation (2), the likelihood is as follows

$$L\left(\theta;\mathbf{u}_p\right) = \prod_{i=1}^{I} P\left(U_{pi}=u_{pi}|\theta,\boldsymbol{\xi}_i\right) \tag{4}$$

and is maximized with respect to θ. Then, the first-order and second-order partial derivatives are needed to compute the maximum likelihood estimates (MLE) and their standard errors [18].

While we used MLE to estimate ability in most results shown here, other methods are available. In case the ability is only unidimensional, a popular approach is the weighted likelihood estimator (WLE) [19]; which maximizes equation (4) weighted by a function $w(\theta)$, thus

$$L\left(\theta;\mathbf{u}_p\right) = w(\theta)L\left(\theta|\mathbf{u}_p\right). \tag{5}$$

Finally, Bayesian ability estimation [20] specifies prior ability distribution $p\left(\theta_p|\mu,\sigma\right)$ and maximizes posterior distribution of θ_p given $\mathbf{u}_p$ of the following form:

$$p\left(\theta_p|\mathbf{u}_p,\boldsymbol{\xi},\mu,\sigma\right) = \frac{\Pr\left(\mathbf{u}_p|\theta_p,\boldsymbol{\xi}\right)p\left(\theta_p|\mu,\sigma\right)}{\int\dots\int\Pr\left(\mathbf{u}_p|\theta_p,\boldsymbol{\xi}\right)p\left(\theta_p|\mu,\sigma\right)\mathrm{d}\theta_p}.$$

Item selection. We used likelihood-based item selection [17], i. e. in each step, the next (k-th) item was selected to maximize the observed Fisher information

$$i_k \equiv \arg\max_j\left\{I_{\mathbf{U}_{k-1},U_j}\left(\hat{\theta}_{k-1}\right)\right\} \tag{6}$$

at $\theta_p = \hat{\theta}_{p,k-1}$ given a subject p, where $\mathbf{U}_{k-1}$ is an answer pattern up to the $(k-1)$-th item [17]. This rule is also known as the maximum-information rule in adaptive testing.

Other item selection procedures include naive approach such as Urry's criterion picking always an item with difficulty closest to the current ability estimate [21]. In Bayesian framework, the posterior distribution of θ_p after the preceding item serves as the prior distribution for the selection of the next item. If the posterior distribution after $k-1$ items has density $p\left(\theta|\mathbf{u}_{k-1}\right)$, then the k-th item is selected such that the posterior distribution

$$p\left(\theta|\mathbf{u}_{k-1},U_{i_k}\right) \propto p\left(\theta|\mathbf{u}_{k-1}\right)p\left(U_{i_k}=u_{i_k}|\theta\right)$$

is optimized in some sense [20].

Termination criteria. In our simulation studies, we used ability estimate precision as a stopping rule. Assuming the $I_{\mathbf{U}_{p,k-1}}(\hat{\theta}_{p,k-1})$ is observed Fisher information [17] at $\hat{\theta}_{p,k-1}$ where $\mathbf{U}_{k-1}$ is an answer pattern until the $(k-1)$-th item (inclusively) given a subject p, then standard error of ability $\hat{\theta}_{p,k-1}$ is

$$\mathrm{SE}(\hat{\theta}_{p,k-1}) = \frac{1}{\sqrt{I_{\mathbf{U}_{p,k-1}}\left(\hat{\theta}_{p,k-1}\right)}}. \tag{7}$$

For the adaptive test, we specified the maximal allowed standard error $\mathrm{SE}(\theta)_{\max}$ of the ability estimate based on the distribution of standard errors of the ability estimates from the full 100-item test. For subject p, the test was terminated just after administration of the k-th item if $\mathrm{SE}(\hat{\theta}_{p,k}) \leq \mathrm{SE}(\theta)_{\max}$ and $\mathrm{SE}(\hat{\theta}_{p,k-1}) > \mathrm{SE}(\theta)_{\max}$. Otherwise, the test was stopped if the length of 100 items was reached and all available items were used.

Whenever the termination (stopping) criterion is met, the adaptive test is ended and final estimate of test-taker's ability is provided.

Content balancing. Balancing of an adaptive test content is usually treated as a combinatorial constrained optimization problem [22]. Alternatively, it is based on a shadow-test approach by projection of rest of the test at the current moment (after $k-1$ items are administrated), which is a nonlinear program using maximum-information rule and constrained by domain attributes and other conditions [22].

In the post-hoc analysis described in this paper, we used one of the combinatorial designs, where we initially set desired proportions of expected administration rate to each of the three domains (genetics, taxonomy, human biology). The items were selected in a way to minimize differences between the currently observed and initially set proportions.

Item exposure rate control. The rates of how many times each item is administrated to one or more of test-takers throughout one adaptive test session may be controlled to minimize their unwanted leakage outside the tested population. Hetter-Sympson experiment is commonly applied to face this problem and was also used in our simulation study [23]. The algorithm was run before the optimally selected item was administrated, output of which was a decision either to administer the item, or to pass and select the next best item at the current estimate of ability $\hat{\theta}_{p,k}$. The administered items were removed from the item pool. Hetter-Symspon experiment is based on evaluation of joint conditional probabilities of item administration; thus cumbersome and usually must be numerically simulated.

There are also some alternatives – an experiment determining which items are eligible for subjects and which not [24]. If an item is eligible, it remains in the pool for the subject p; otherwise it is removed. This works as a principle of "self-adjustment"; when an item was highly exposed within previous $p-1$ subjects, it is likely not to be eligible for the p-th subject.

C. Post-hoc analysis

In post-hoc analysis, the item parameters and the response patterns of the respondents were used to rerun the test under adaptive conditions. By doing this, the properties of the adaptive test (such as the test length, precision of estimated abilities etc.) were "post-hoc" evaluated and compared to the original linear test.

Considering the dataset of test-takers taking the real test, we varied the maximal allowed standard error of the ability estimates and ran the adaptive version of the test for each of the test-takers investigating how many items were needed to complete the test. The pseudocode of this simulation is provided in Algorithm 1.

Similarly, we calculated the z-score for each subject using the test scores from the real test,

$$z\text{-score} = \frac{x_p - \bar{x}}{s_x},$$

where x_p is a test score of a subject p, $\bar{x}$ is an average test score and s_x is a standard deviation of all test scores. All test-takers having their z-scores in the interval of $|z - z^*| \leq \delta$, where

Algorithm 1: Investigation of adaptive test length depending on the precision of ability estimate

```
Data: data of the real test
Result: boxplots of test lengths for CAT with different
        standard errors of ability estimates
$\{\mathcal{S}\}$   // set or subset of respondents;
        // of the real test;
$\{\mathcal{A}\} = \emptyset$  // list of vectors of lengths;
           // for different standard;
          // errors;
$\{\mathcal{E}\} = \emptyset$  // list of standard errors;
for $j = 1 : 7$ do
    $\mathrm{SE} = 0.15 + 0.05 \cdot j$;
    $\{\mathcal{E}\} = \{\mathcal{E}\} \cup \{\mathrm{SE}\}$;
    $\{D\} = \emptyset$ ;
    for $p \in \mathcal{S}$ do
        run an adaptive test for subject $p$ with stopping
         criterion $\mathrm{SE}(\theta)_{\max} = \mathrm{SE}$ and save its length
         as $d$;
        $\{D\} = \{D\} \cup \{d\}$;
    end
    $\{\mathcal{A}\} = \{\mathcal{A}\} \cup \{D\}$;
end
make a boxplot of $\{\mathcal{A}\}$ vs. $\{\mathcal{E}\}$ ;
```

Algorithm 2: Investigation of average adaptive test length depending on the z-score from the original linear test

```
Data: data of the real test
Result: boxplots of average test lengths for groups
        based on $z$-scores from the original linear test
$\delta = 0.05$   // neighbourhood around $z^*$;
$\{\mathcal{A}\} = \emptyset$  // list of vectors of lengths;
         // for different $z$-scores;
$\{\mathcal{Z}\}$      // list of original $z$-scores;
$\{\mathcal{Z}^*\} = \emptyset$  // list of $z^*$-scores;
for $j = 1 : 17$ do
    $z^* = -2.00 + 0.25 \cdot j$;
    $\{\mathcal{Z}^*\} = \{\mathcal{Z}^*\} \cup \{z^*\}$;
    $\{D\} = \emptyset$ ;
    for all subjects with $z \in \mathcal{Z}$ such that $|z - z^*| \leq \delta$
     do
        run an adaptive test for the subject with
         stopping criterion $\mathrm{SE}(\theta)_{\max} = 0.30$ and save
         its length as $d$;
        $\{D\} = \{D\} \cup \{d\}$;
    end
    $\{\mathcal{A}\} = \{\mathcal{A}\} \cup \{D\}$;
end
make a boxplot of $\{\mathcal{A}\}$ vs. $\{\mathcal{Z}^*\}$ ;
```

$\delta = 0.05$ and $z^* \in \{-2.00, -1.75, -1.50, \ldots, +1.75, +2.00\}$ were supposed to virtually take the adaptive test, keeping the $\mathrm{SE}(\theta)_{\max} = 0.30$ for equation (7) constant. The z^* neighbourhood $\delta = 0.05$ was chosen empirically, but consequently, one can realize that $\delta = 0.125$ would cover continuously the entire range of all z-scores. For each z^*, a vector of all the adaptive tests' lengths was displayed in the final boxplot. The schema of the simulation is provided in Algorithm 2.

Similarly, the effect of content balancing and item exposure rate control was analyzed. When an adaptive test was administered to each test-taker from a randomly selected subset, we counted how many times individual items occur in the tests. Absolute numbers of the items' occurrences were then counted up for different scenarios – besides the situation when neither the content balancing nor the item exposure was applied, the case of (only) the content balancing and (only) the item exposure rate controlling was taken into account. Eventually, using the fact, the items were classified into three domains (Genetics, Taxonomy, Human Biology), their counts could be clearly plotted using boxes in a boxplot.

Finally, to study the impact of adaptive test with different settings on the admission process, we enumerated the admission mismatch rate between linear and adaptive tests. We assumed the best fifth of all the applicants would be admitted and we calculated the mismatch rate as the ratio of students who would be admitted based on their score in the linear test but not based on the score in adaptive test and vice versa. We then compared the admission mismatch rate for adaptive tests with stopping criteria $\mathrm{SE}(\theta)_{\max} \in \{0.20, 0.30, 0.40, 0.50\}$. While the best fifth for the linear test was calculated using the z-scores, the MML ability estimates were used for the adaptive test.

D. Monte Carlo simulation studies

Whereas the post-hoc analysis requires real data from an administrated test, the Monte-Carlo simulation study starts from the scratch – it generates abilities of "virtual" test-takers usually following normal distribution and responses based on selected model (e. g. the 2PL IRT model) with given item parameters. We first simulated the linear test, then, based on the simulated answers, the adaptive scenario was simulated. We then correlated ability estimates from the adaptive test with the true ability values. Finally, we displayed lengths of adaptive test and we correlated the ability estimate with the true ability. Other comparisons and analyses are possible (length with respect to the true ability score, etc.), but not presented here. The algorithm of the simulation is technically described in Algorithm 3.

Analyses were performed in R programming language and environment [25] using the package mirtCAT [26].

III. Results

All items of the linear test were calibrated using the 2PL IRT model as described by equation (1). Item characteristic curves and item information curves are plotted in Fig. 2 and Fig. 3. All items have positive discrimination $a_i > 0$ for

Algorithm 3: Investigation of ability estimates based on adaptive tests using a Monte-Carlo simulation

```
Data: generated abilities following $\mathcal{N}(0, 1^2)$,
  item parameters (estimated from real data), adaptive
  test's stopping criterion $\mathrm{SE}(\theta)_{\max} = 0.30$, item
  selection using maximum-information rule
Result: a list of ability estimates based on adaptive test
$n = 300$      // number of generated;
               // abilities;
$\{\mathcal{S}\}$          // list of $n$ generated;
               // abilities;
               // following $\mathcal{N}(0, 1^2)$;
$\{\mathcal{A}\} = \emptyset$    // list of adaptive-based;
               // ability estimates;
$\{\mathcal{D}\} = \emptyset$    // list of lengths;
               // of adaptive tests;
for $p = 1 : n$ do
    apply 2PL IRT model on $p$-th ability of $\{\mathcal{S}\}$ and
      simulate an answer pattern ;
    use the answer pattern and run an adaptive test for
      $p$-th ability and save its length as $d$ and ability
      estimate as $\hat{\theta}_p$ ;
    $\{\mathcal{A}\} = \{\mathcal{A}\} \cup \hat{\theta}_p$;
    $\{\mathcal{D}\} = \{\mathcal{D}\} \cup \{d\}$;
end
make a boxplot of $\{\mathcal{D}\}$ ;
make a scatterplot of $\{\mathcal{A}\}$ vs. $\{\mathcal{S}\}$, calculate
  a correlation of $\{\mathcal{A}\}$ and $\{\mathcal{S}\}$ ;
```

$\forall i \in \{1, 2, \ldots, 100\}$, resulting in a spectrum of the item information curves.

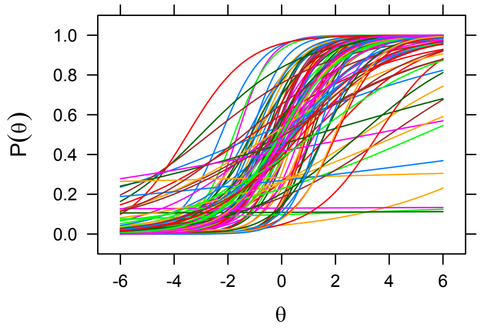


Fig. 2. Item characteristic curves of the linear test estimated using 2PL IRT model.

When applying the 2PL IRT model on the data from the linear test, we get, besides other, also standard errors of the ability estimates for each test-taker. Histogram of these standard errors in in Fig. 4. Range of the standard errors of the ability estimates is between 0.20 to 0.50, with majority of values within the interval $\langle 0.20, 0.30 \rangle$.

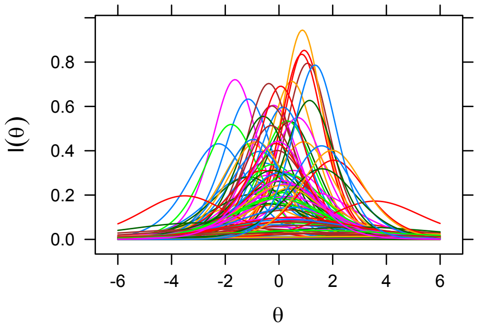


Fig. 3. Item information curves of the linear test estimated using 2PL IRT model.

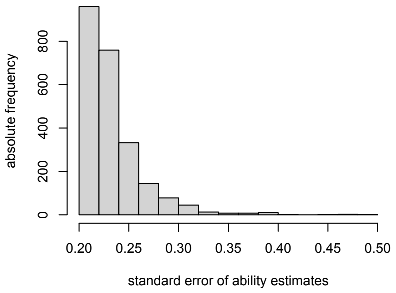


Fig. 4. Histogram of standard errors of the ability estimates.

A. *Post-hoc analysis*

Post-hoc analysis used the real test-takers data to simulate the results under scenario of an adaptive test with selected parameters. As an example, Fig. 5 demonstrates iteratively estimated ability estimates and order of items in which they would be administered to the 1-st subject under adaptive scenario with terminating criterion $\mathrm{SE}(\theta)_{\max} = 0.30$. We can see that the initial item would be item number 81, the last item would be item number 70. The width of the grey belt stands for precision of the ability estimate at each step k, equal to two standard errors $2\mathrm{SE}(\hat{\theta})_{p,k}$ of the ability estimate of person p. The belt becomes more narrow as the test-taker answers more and more items. Note that the standard error after 18 administered items is $\mathrm{SE}(\hat{\theta})_{1,18} \leq 0.30$ while after 17 administered items it is $\mathrm{SE}(\hat{\theta})_{1,17} > 0.30$.

As a result of Algorithm 1, Fig. 6 presents how the number of items needed to stop the adaptive test depends on the termination criterion. We can see that the higher the maximal standard error is applied as the termination criterion, the lower the number of items is needed to terminate the adaptive test.

As a result of simulation described with Algorithm 2, Fig. 7 illustrates how the respondent ability (estimated with a z-score) affects the number of items needed to stop the adaptive test. The size of maximal allowed standard error of the ability estimates as the stopping criterion was set to $\mathrm{SE}(\theta)_{\max} = 0.30$ based on the distribution of the standard

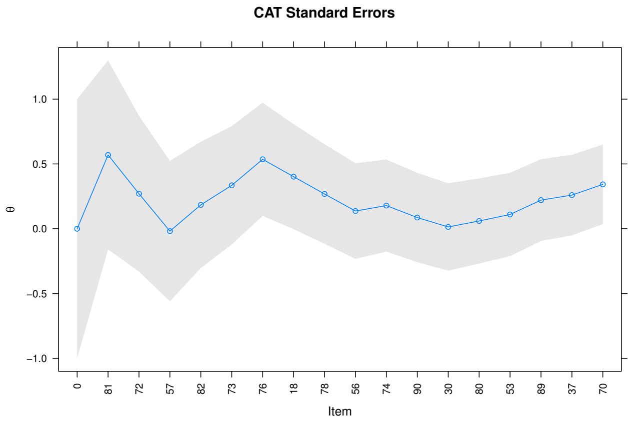


Fig. 5. A plot of progress of 1-st subject in an adaptive test with the terminating criterion set to maximal allowable standard error of the ability estimates of $\mathrm{SE}(\theta)_{\max} = 0.30$.

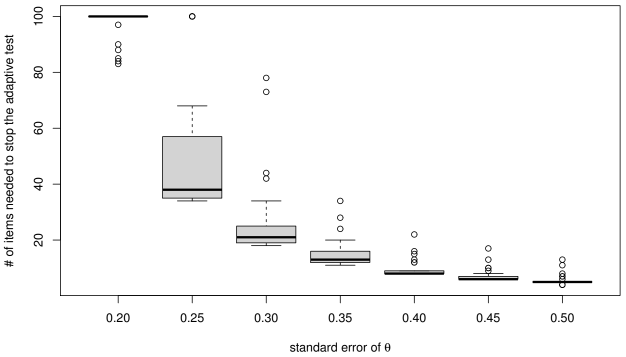


Fig. 6. Number of items needed to stop the adaptive test versus a size of standard error of the ability estimate as the stopping criterion.

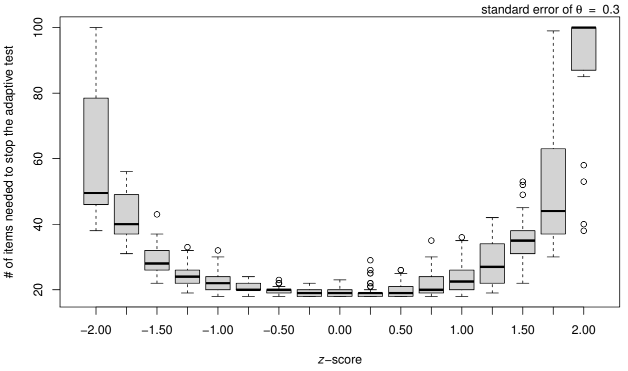


Fig. 7. Number of items needed to stop the adaptive test in respondents of different ability levels.

errors, displayed in Fig. 4. We can see that the closer the z-score is to zero, the lower number of items is needed to complete the adaptive test while meeting the required ability estimate precision defined by $\mathrm{SE}(\theta)_{\max} = 0.30$. This corresponds to the fact that the information functions for majority of items have the maxima for ability around zero as demonstrated in Fig. 3. Contrary, for z-scores far from zero, the observed Fisher information is small for most of the items, thus a larger number of items is needed to meet the stopping criterion, and often not even meeting it using all 100 items available.

In Fig. 8, we plot numbers of occurrences of items in all individual adaptive tests for randomly selected 50 test-takers, considering that each item belongs to one of the following three domains – either to genetics, taxonomy, or human biology, respectively. While the proportions of the three domains of items as they were administrated vary a lot in Fig. 8 where neither the content balancing nor the item exposure rate control is applied, these numbers are near equal when the content balancing is applied. When the item exposure rate control is employed, there is no visible change in comparison to no application of the exposure control.

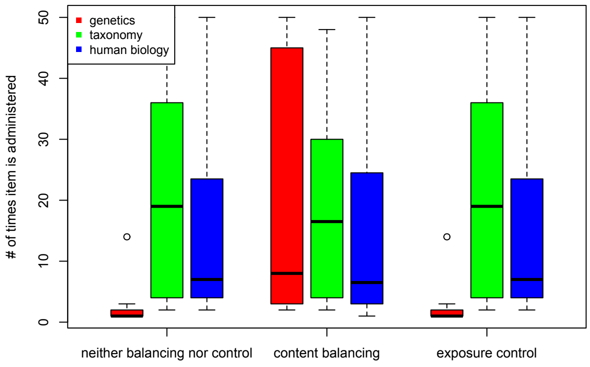


Fig. 8. Number of items belonging to the domains *genetics*, *taxonomy*, *human biology*, respectively, as were administrated with application of neither content balancing nor item exposure control, with application of content balancing, and with application of item exposure rate control only.

Table I provides mismatch matrices for linear and adaptive tests with different stopping criteria $\mathrm{SE}(\theta)_{\max}$. As expected, the mismatch rate increases with increased allowed standard error applied as a stopping criterion in the adaptive test. The mismatch rate is 0.036, 0.083, 0.102 and 0.118 for adaptive tests with stopping rules $\mathrm{SE}(\theta)_{\max} = 0.20$, 0.30, 0.40 and 0.50, respectively.

B. Monte-Carlo simulation study

As a result of the Monte-Carlo simulation study described by Algorithm 3, Fig. 9 provides a boxplot illustrating the mean length of the adaptive test for the set of test-takers with the generated abilities. While each test-taker has to answer to all (100) items within the linear fashion, they would only have to answer about 25 % of items to finish the simulated adaptive test with the termination criterion $\mathrm{SE}(\theta)_{\max} = 0.30$. The length of the test using this adaptive scenario provides

TABLE I
MISMATCH MATRICES OF ADMITTED TEST-TAKERS BY LINEAR AND ADAPTIVE TEST WITH STOPPING CRITERION $\mathrm{SE}(\theta)_{\max} \in \{0.20, 0.30, 0.40, 0.50\}$.

$\mathrm{SE}(\theta)_{\max} = 0.20$		admitted by adaptive test	
		no	yes
admitted by linear test	no	1842	38
	yes	48	435
$\mathrm{SE}(\theta)_{\max} = 0.30$		admitted by adaptive test	
		no	yes
admitted by linear test	no	1787	93
	yes	103	380
$\mathrm{SE}(\theta)_{\max} = 0.40$		admitted by adaptive test	
		no	yes
admitted by linear test	no	1762	118
	yes	123	360
$\mathrm{SE}(\theta)_{\max} = 0.50$		admitted by adaptive test	
		no	yes
admitted by linear test	no	1737	143
	yes	136	347

75% shortening as compared to the linear test, while keeping the same precision of ability estimates for most respondents.

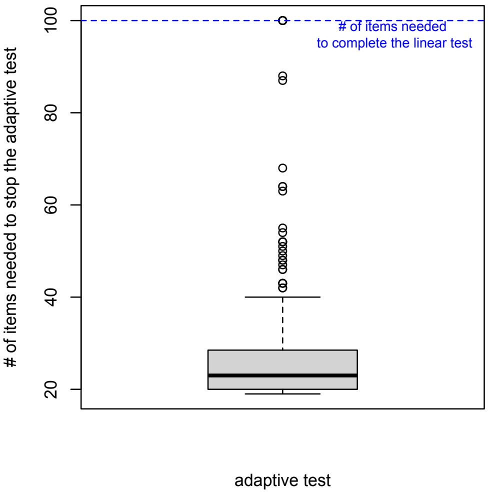


Fig. 9. A boxplot of number of items needed to be answered to complete the adaptive test based on Monte-Carlo simulated test-takers' abilities. The blue dashed line shows a length of the linear test (100 items).

Pearson's correlation between the generated abilities and their estimates based on the adaptive tests is about $\rho \doteq 0.960$, which is depicted also in Fig. 10.

IV. CONCLUSION

Both the post-hoc analysis and Monte-Carlo simulation study showed that average test lengths can be shortened with adaptive tests, while keeping the standard error of the ability estimates at the same level for most of the respondents. The shortening of the test within the adaptive test with $\mathrm{SE}(\hat{\theta})_{\max} = 0.30$ was by about 75 % percent, i. e. while the original linear test had 100 items, the adaptive one was ended on average after answering 25 items only.

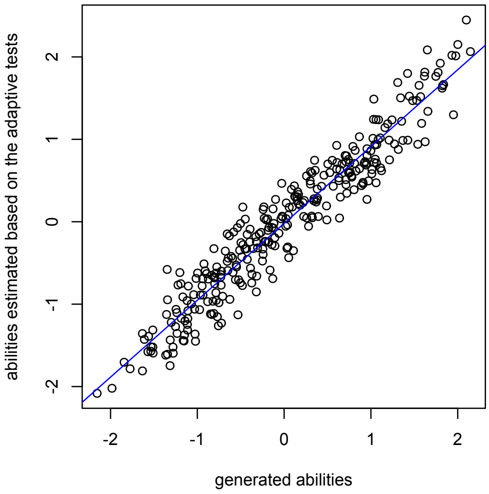


Fig. 10. A scatterplot of the generated abilities and their estimates based on the adaptive tests. The blue line stands for an axis of the first quadrant of the plot.

When even larger standard errors of the ability estimates are tolerated, the length of the test could be reduced even more, e. g. to only 10 items per one test, as was shown in the post-hoc analysis of the average adaptive test length with varying stopping criterion.

The post-hoc simulation also demonstrated that an average length for adaptive tests is shorter for average ability levels.

While the content balancing with the combinatorial approach showed a significant improvement in test domain equalizing, an effect of the item exposure rate did not seem to be so eminent under our setting.

The lower the tolerated standard error as a stopping criterion of the adaptive test is, the lower is the mismatch error rate when using an adaptive test instead of the linear one. The mismatch rate was less than 10% for adaptive test with stopping criterion of $\mathrm{SE}(\hat{\theta})_{\max} = 0.30$.

The Monte-Carlo simulation study also indicated that ability estimates provided by the adaptive tests can be tightly correlated with their true (generated) values; thus, although the shortened length, the adaptive test can provide precise estimates of the respondent abilities.

To conclude, usage of adaptive testing seems to be a promising alternative to classic linear tests and offers many advantages as showed by the simulations.

ACKNOWLEDGMENT

Research was supported by Charles University grant PRIMUS/17/HUM/11.

REFERENCES

[1] Wim J Linden, Wim J van der Linden, and Cees AW Glas. *Computerized adaptive testing: Theory and practice*. Springer, 2000.

[2] Howard Wainer, Neil J Dorans, Ronald Flaugher, et al. *Computerized adaptive testing: A primer*. Routledge, 2000.

[3] David Magis, Duanli Yan, and Alina A Von Davier. *Computerized adaptive and multistage testing with R: Using packages catr and mstr*. Springer, 2017.

[4] David J Weiss and G Gage Kingsbury. "Application of computerized adaptive testing to educational problems". In: *Journal of Educational Measurement* 21.4 (1984), pp. 361–375.

[5] Jan Stochl, Jan R Böhnke, Kate E Pickett, et al. "Computerized adaptive testing of population psychological distress: simulation-based evaluation of GHQ-30". In: *Social psychiatry and psychiatric epidemiology* 51.6 (2016), pp. 895–906.

[6] Jan Stochl, Jan R Böhnke, Kate E Pickett, et al. "An evaluation of computerized adaptive testing for general psychological distress: combining GHQ-12 and Affectometer-2 in an item bank for public mental health research". In: *BMC medical research methodology* 16.1 (2016), p. 58.

[7] Dagmar Amtmann, Alyssa M Bamer, Jiseon Kim, et al. "A comparison of computerized adaptive testing and fixed-length short forms for the Prosthetic Limb Users Survey of Mobility (PLUS-MTM)". In: *Prosthetics and orthotics international* 42.5 (2018), pp. 476–482.

[8] Karon F Cook, Seung W Choi, Paul K Crane, et al. "Letting the CAT out of the bag: comparing computer adaptive tests and an eleven-item short form of the Roland-Morris Disability Questionnaire". In: *Spine* 33.12 (2008), p. 1378.

[9] Patricia Martinková, Lubomír Štěpánek, Adéla Drabinová, et al. "Semi-real-time analyses of item characteristics for medical school admission tests". In: *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*. Ed. by M. Ganzha, L. Maciaszek, and M. Paprzycki. Vol. 11. Annals of Computer Science and Information Systems. IEEE, 2017, pp. 189–194. DOI: 10.15439/2017F380. URL: http://dx.doi.org/10.15439/2017F380.

[10] Čestmír Štuka, Patrícia Martinková, Karel Zvára, et al. "The prediction and probability for successful completion in medical study based on tests and pre-admission grades". In: *New Educational Review* 28 (2012), pp. 138–52.

[11] Patrícia Martinková and Adéla Drabinová. "ShinyItemAnalysis for Teaching Psychometrics and to Enforce Routine Analysis of Educational Tests." In: *R Journal* 10.2 (2018).

[12] Wim J. van der Linden and Cees A.W. Glas. "25 Statistical Aspects of Adaptive Testing". In: *Handbook of Statistics*. Elsevier, 2006, pp. 801–838. DOI: 10.1016/s0169-7161(06)26025-5. URL: https://doi.org/10.1016/s0169-7161(06)26025-5.

[13] R. Darrell Bock and Murray Aitkin. "Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm". In: *Psychometrika* 46.4 (Dec. 1981), pp. 443–459. DOI: 10.1007/bf02293801. URL: https://doi.org/10.1007/bf02293801.

[14] Yoshio Takane and Jan de Leeuw. "On the relationship between item response theory and factor analysis of discretized variables". In: *Psychometrika* 52.3 (Sept. 1987), pp. 393–408. DOI: 10.1007/bf02294363. URL: https://doi.org/10.1007/bf02294363.

[15] Cees A. W. Glas. "Modification indices for the 2-PL and the nominal response model". In: *Psychometrika* 64.3 (Sept. 1999), pp. 273–294. DOI: 10.1007/bf02294296. URL: https://doi.org/10.1007/bf02294296.

[16] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society, Series B* 39.1 (1977), pp. 1–38.

[17] Hua-Hua Chang and Zhiliang Ying. "Nonlinear sequential designs for logistic item response theory models with applications to computerized adaptive tests". In: *The Annals of Statistics* 37.3 (June 2009), pp. 1466–1488. DOI: 10.1214/08-aos614. URL: https://doi.org/10.1214/08-aos614.

[18] Daniel O. Segall. "Multidimensional adaptive testing". In: *Psychometrika* 61.2 (June 1996), pp. 331–354. DOI: 10.1007/bf02294343. URL: https://doi.org/10.1007/bf02294343.

[19] Thomas A. Warm. "Weighted likelihood estimation of ability in item response theory". In: *Psychometrika* 54.3 (Sept. 1989), pp. 427–450. DOI: 10.1007/bf02294627. URL: https://doi.org/10.1007/bf02294627.

[20] Frederic Lord. *Applications of item response theory to practical testing problems*. Hillsdale, N.J: L. Erlbaum Associates, 1980. ISBN: 978-0898590067.

[21] Frank L. Schmidt, John E. Hunter, and Vern W. Urry. "Statistical power in criterion-related validation studies." In: *Journal of Applied Psychology* 61.4 (1976), pp. 473–485. DOI: 10.1037/0021-9010.61.4.473. URL: https://doi.org/10.1037/0021-9010.61.4.473.

[22] Wim J. van der Linden and Richard M. Luecht. "Observed-score equating as a test assembly problem". In: *Psychometrika* 63.4 (Dec. 1998), pp. 401–418. DOI: 10.1007/bf02294862. URL: https://doi.org/10.1007/bf02294862.

[23] Rebecca D. Hetter and J. Bradford Sympson. "Item exposure control in CAT-ASVAB." In: *Computerized adaptive testing: From inquiry to operation.* American Psychological Association, 1997, pp. 141–144. DOI: 10.1037/10244-014. URL: https://doi.org/10.1037/10244-014.

[24] Martha L. Stocking and Charles Lewis. "Controlling Item Exposure Conditional on Ability in Computerized Adaptive Testing". In: *Journal of Educational and Behavioral Statistics* 23.1 (1998), p. 57. DOI: 10.2307/1165348. URL: https://doi.org/10.2307/1165348.

[25] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2017. URL: https://www.R-project.org/.

[26] R. Philip Chalmers. “Generating Adaptive and Non-Adaptive Test Interfaces for Multidimensional Item Response Theory Applications”. In: *Journal of Statistical Software* 71.5 (2016), pp. 1–39. DOI: 10.18637/jss.v071.i05.

Advances in Computer Science & Systems

ACSS is welcoming presentations of the scientific aspects related to applied sciences. The session is oriented on the research where the computer science meets the real world problems, real constraints, model objectives, etc. However the scope is not limited to applications, we all know that all of them were born from the innovative theory developed in laboratory. We want to show the fusion of these two worlds. Therefore one of the goals for the session is to show how the idea is transformed into application, since the history of modern science show that most of successful research experiments had their continuation in real world. ACSS session is going to give an international panel where researchers will have a chance to promote their recent advances in applied computer science both from theoretical and practical side.

Scope:

- Applied Artificial Intelligence
- Applied Parallel Computing
- Applied methods of multimodal, constrained and heuristic optimization
- Applied computer systems in technology, medicine, ecology, environment, economy, etc.
- Theoretical models of the above computer sciences developed into the practical use

Technical Session Chairs

- **Woźniak, Marcin,** Institute of Mathematics, Silesian University of Technology, Poland
- **Dimov, Ivan,** Bulgarian Academy of Sciences, Institute of Information and Communication Technologies, Bulgaria

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 371–380 DOI: 10.15439/2020F188 ISSN 2300-5963 ACSIS, Vol. 21

BiLSTM with Data Augmentation using Interpolation Methods to Improve Early Detection of Parkinson Disease

Olusola O. Abayomi-Alli,
Robertas Damaševičius
Department of Software Engineering, Kaunas University of Technology, Kaunas, Lithuania
olusola.abayomi-alli@ktu.edu,
robertas.damasevicius@ktu.lt

Rytis Maskeliūnas
Department of Applied Informatics, Vytautas Magnus University, Kaunas, Lithuania
rytis.maskeliunas@vdu.lt

Adebayo Abayomi-Alli
Department of Computer Science, Federal University of Agriculture, Abeokuta, Nigeria
abayomiallia@funaab.edu.ng

***Abstract*—The lack of dopamine in the human brain is the cause of Parkinson disease (PD) which is a degenerative disorder common globally to older citizens. However, late detection of this disease before the first clinical diagnosis has led to increased mortality rate. Research effort towards the early detection of PD has encountered challenges such as: small dataset size, class imbalance, overfitting, high false detection rate, model complexity, etc. This paper aims to improve early detection of PD using machine learning through data augmentation for very small datasets. We propose using Spline interpolation and Piecewise Cubic Hermite Interpolating Polynomial (Pchip) interpolation methods to generate synthetic data instances. We further investigate on reducing dimensionality of features for effective and real-time classification while considering computational complexity of implementation on real-life mobile phones. For classification we use Bidirectional LSTM (BiLSTM) deep learning network and compare the results with traditional machine learning algorithms like Support Vector Machine (SVM), Decision Tree, Logistic regression, KNN and Ensemble bagged tree. For experimental validation we use the Oxford Parkinson disease dataset with 195 data samples, which we have augmented with 571 synthetic data samples. The results for BiLSTM shows that even with a holdout of 90%, the model was still able to effectively recognize PD with an average accuracy for ten rounds experiment using 22 features as 82.86%, 97.1%, and 96.37% for original, augmented (Spline) and augmented (Pchip) datasets, respectively. Our results show that proposed data augmentation schemes have significantly ($p < 0.001$) improved the accuracy of PD recognition on a small dataset using both classical machine learning models and BiLSTM.**

***Index Terms*—speech impairment, Parkinson's, voice analysis; deep learning, data augmentation, interpolation, small data.**

I Introduction

PARKINSON Disease (PD) is a degenerative disorder of the central nervous system with major damage affecting the motor system in the brain cells [1]. This disease is among the most common and fastest growing neurodegenerative disorders affecting close to 7 to 10 million people globally [2-3]. It is majorly caused by the lack of dopamine (neurotransmitter) in the human brain [4] and its effect can be categorized into motor and non-motor symptoms such as voice/speech impairment, dementia, depression, slow thinking, rigidity, tremor, bradykinesia, and other cognitive disabilities [4-5]. From 60% to 90% of PD patients suffer from speech impairment such as slurred, mumbled or slow speech [6-7], among other symptoms.

Quite a number of research progress have been accounted in previous studies but the need to further explore more sophisticated algorithms of artificial intelligence (AI) methods is still ongoing with the aim of improving the health of the aged citizen through early detection of the PD disease and other diseases with similar symptoms. Several databases have been created for easing research output in the detection of neurodegenerative disorder and these databases presented in existing literature for detection of PD include dataset for detecting speech impairment (dysphonia) [1], drawing movement [8], Volatile Organic Compounds (VOCs) in blood [9], cognitive impairment [10], electroencephalohraphy (EEG) and electromyography (EMG) bio-signals [11], images such as magnetic resonance imaging (MRI), functional MRI (fMRI), positron emission tomography (PET) [12], etc. In majority of cases, once the symptoms of the neurodegenerative disorder such as PD have been validated by a medical expert, the chances of disease progress in patients becomes higher due to late detection [13]. Therefore, further research endeavors in early diagnosis of PD before it progresses any further making any medical assistance and treatment ineffective are very important [14].

The traditional methods require a lot of monitoring of living activities, motor skills, and other neurological parameters to determine the PD progress in a patient [5]. Recent advancement in AI methods have increased research focus towards adopting algorithms to enhance diagnostics of PD among patients. Existing research contributions include the implementation of mobile applications for PD diagnosis and monitoring [14-18]. The contribution of this paper is:

- Effective interpolation-based data augmentation techniques to generate synthetic data samples for training of machine learning models.
- To explore dimensionality reduction with the aim of identifying the best set of features for classification.
- Finally, to investigate and compare the performance of BiLSTM deep learning models and traditional machine learning algorithms in early detection of PD using the original (Oxford Parkinson) and augmented datasets.

The rest of the paper is organized as follows: Section II discusses in details the related works with highlights on the shortcomings of existing solutions. Section III presents the methods used in this study with an emphasis on the proposed methods and data used with introductory explanation of neural network models. Section IV describes the implementation details and the results achieved from our proposed models and presents a comparison of our results with existing studies using the same dataset. The paper concludes in Section V with future research recommendations.

II Literature Review

This section discuss the various studies tailored towards detecting and classifying PD with a focus on previous work based on speech impairment. A typical wave form variance of a healthy person and an individual suffering from speech impairment is depicted in Fig. 1.

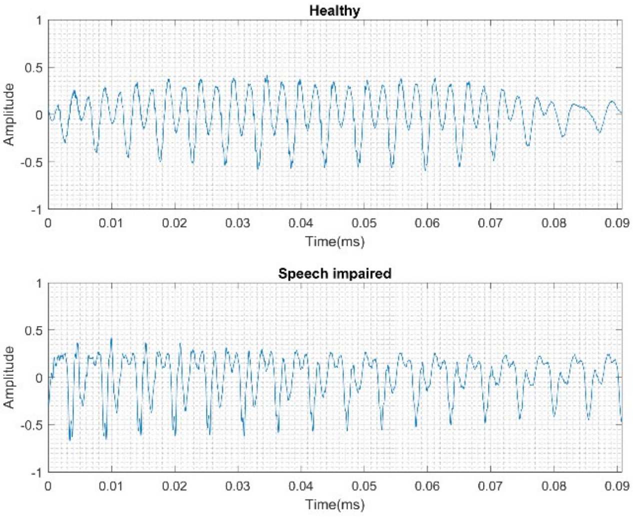


Fig. 1. A typical wave form variance of a healthy person and an individual suffering from speech impairment (data taken from the dataset described in [20,21])

A Related Studies on Speech Impairment

Previous study on early diagnosis of PD include [19], which presented an ensemble classifier based on Deep Belief Network (DBN) and Self-Organizing Map (SOM) for remote tracking of PD progress. Recent studies [20, 21] proposed a hybrid model based on bidirectional LSTM (Bi-LSTM) neural network and wavelet scattering transform (WST) and SVM classifier to detect speech impairments. Authors experimented on 15 subjects and 7 diseased subjects making up for 339 voice samples. The results showed that the proposed based on WST and SVM outperform Bi-LSTM and is expected to improve the decision systems for speech impairment detection with accuracy of 96.3%. Similar study was conducted in [22] using online handwriting dynamic signals for detection of PD. The authors investigated the impact of transfer learning and data augmentation methods, and evaluated the classification approach based on CNN-BLSTM and SVM. The study concluded that integrating data augmentation helps to achieve favourable results.

Authors in [23] introduced a bio-inspired algorithm for decision support system to evaluate voice challenges. The study compared the performance of different mathematical solution such as Fourier and Gabor transformation with bio-inspired algorithms. The result of the proposed voice analysis system using heuristic and spiking neural network gave a promising results when compared with the state-of–the–art methods with the average effectiveness as 87%. Author in [13] presented four machine learning (ML) methods for detection of PD from sustained phonation and speech signals. Authors applied eighteen feature extraction approach obtained from acoustic cardioid and smartphone recording on four ML algorithms KNN, MLP, optimum-path forest (OPF) and SVM. Authors in [1] proposed an extreme learning machine (ELM) for predicting PD. Authors in [5] presented a Principal Component Analysis (PCA) algorithm on original feature sets and other non-linear classifiers. The study gave an impressing performance of random forest accuracy as 96.87%. Authors claimed that reducing dimensionality plays an important role in improving overall classification of PD. Authors in [7] introduced the combination of Gaussian processes and automatic relevance determination for detecting PD. The study was conducted on two PD dataset and the focus was based on the using small amount of relevant acoustic features for detection. Authors in [24] presented an automatic analyses of PD using Mel-Frequency Cepstral Coefficients (MFCC), combined with Gaussian Mixture Models (GMM). In addition, authors in [25] incorporated MFCC and Intrinsic Mode Functions (IMF) for detection of PD. Authors in [26] also proposed using MFCC and glottal pulse for early detection of PD.

A number of ML algorithms have been implemented by researchers in detection of PD such as the application of supervised classification algorithms was presented in [27]. The classification result gave a peak accuracy of 85% while it is promising when compared to diagnosis accuracy of non-experts and specialists. Multiple learner for PD detection was investigated by authors in [25, 2]. Authors in [28] presented an ensemble classification methods based on random subspace classifier using kNN. While the latter [2] utilized ensemble bagging with genetic algorithm for detection of PD. Similarly, the study [29] presented a hybrid approach based on Synthetic Minority Over-Sampling Techniques (SMOTE) and Random Forest (RF) classifier for classifying PD. The overall classification result showed significant improvement with accuracy of 94.89% with the 10-fold validation test.

Furthermore, deep learning methods were applied in [30] for the diagnosis of PD. The study applied Multilayer Feedforward Neural Network (MLFNN) with Back-propagation (BP) algorithm for early detection of PD. Their experimental result gave a low specificity of 63.6% and a fair accuracy of 80% compared with other studies. Despite, the application of various techniques and methods on detecting PD, it is concluded these methods are still far from obtaining desired result in terms of accurate identification of PD [1].

TABLE I. SUMMARY OF RELATED WORKS

References	Methods		Contributions	Limitations	Type of Data
	Classification	Data Augmentation			
[31]	Convolutional Neural Network (CNN)	Jittering, scaling, rotating, permutating, magnitude warping, time-warping methods	Application of data augmentation improved generalization performance	Fluctuations in misclassification due to noisy labels.	Wearable Sensor Data (Motor State)
[32]	long-short-term memory recurrent neural network (LSTM-RNN)	doubling the number of datapoints for non-zero; converted all non-zero tremor scores to a single value (positive)	Balancing of training data	No significant improvement in accuracy; Issues with overfitting	Motion data (Tremor)
[33]	CNN	Image interpolation	Best detection rate was achieved based on sentence segments	Authors did not compare the performance with existing studies	Speech Data
[34]	Different regression models	Random resize and crop, random horizontal flip, and color jitter	Pitch-related features perform better than alternative features	The accuracy of interference need to be improved considerable.	Speech Data
[18]	CNN	magnitude perturbation, temporal perturbation, and random rotation	achieved satisfactory performance	overenrolled tremor-dominant PD subjects	wearable sensor devices

The summary of some related work that applied data augmentation techniques for PD detection is presented in Table I. Some of the challenges affecting research efforts and the performance of learning are still centered on insufficient data, noisy labels, and large intra-class variability [31]. However, there is still a need for more efficient and reliable data augmentation techniques to improve accuracy and the reliability of the detection thus reducing error rate [5].

B Data Augmentation

Data Augmentation have been successful applied in many classification application due to the fact that it leverages on small data by transforming existing samples and generating new ones [31,35]. The application of data augmenting have improved generalization of deep learning models and prevented overfitting of trained data. Some of its application areas include in face recognition system [36], motion detection system [37], etc. In addition, it also enhance deep learning model performance and overall stability of training results. The latter is especially relevant for the so called "small data problem" [38], when only little data is available for training of machine learning models.

Some application of the commonly used data augmentation techniques in image processing or signal processing, are geometric transformation which include scaling, shifting, rotation/ reflection, time wrapping and addition of noise. Recent studies in detection of PD have applied variety of data augmentation methods to accelerometer and gyroscope recordings such as magnitude scaling, rotation and magnitude scaling [18], cropping methods, window slicing, jittering, etc. [31,35]. Some of the drawbacks affecting the application of data augmentation include the need to maintain correct annotations/ labels which mostly requires expert knowledge.

Based on these, we present an effective data augmentation technique based on interpolation methods (spline and pchip) for generating synthetic values for further classification analysis. We further investigate the impact of data augmentation and feature reduction of the performance of the neural network model for detection of PD from speech data.

III METHODS AND MATERIALS

We discus the various steps involved in our proposed model as depicted in the functional block diagram in Fig. 2.

A Data Source

For this study, we used the Oxford Parkinson Disease dataset [39], which comprises of biomedical voice measurement from 31 individuals with 23 individuals suffering from PD. The dataset description is summarized in Table II and it consists of 195 voice recordings (147 PD and 48 healthy voice recordings), 22 real-value features.

Each recording was subjected to different measurements, consisting of vocal fundamental frequency (average, maximum and minimum) measured in Hertz, Multi-Dimensional Voice Program (MDVP) for percentage measurement of variations of frequency (Jitter) and amplitude (Shimmer), harmonicity measurement (HNR and NHR) and records of non-linear dynamics (NLD) features namely: correlation dimension (D2), Period Density Entropy (RPDE), Detrended Fluctuation Analysis (DFA) and frequency variation measurement which include spread1, spread2 and Pitch Period Entropy (PPE). The data is divided randomly using the ratio of training and testing as 70:30, respectively.

The training dataset comprises of 103 PD and 34 Healthy which we further used in the generation of synthetic dataset. A total number of 571 synthetic data samples was generated (consisting of 320 PD and 251 Healthy) and the overall data used for training our the deep network model is 708. The testing data consist of 58 instances from the original data.

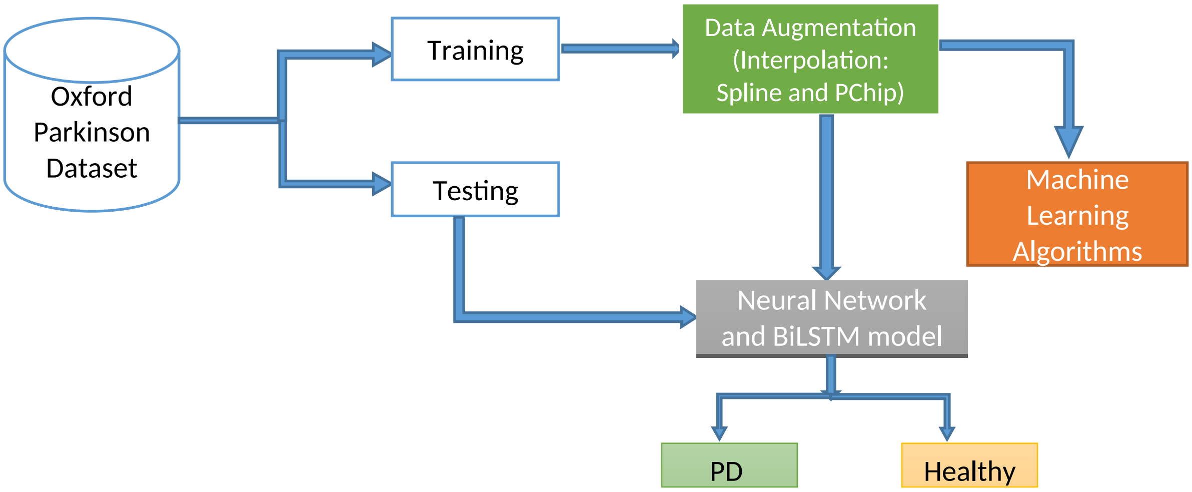

Oxford Parkinson Dataset
Training
Testing
Data Augmentation (Interpolation: Spline and PChip)
Machine Learning Algorithms
Neural Network and BiLSTM model
PD
Healthy

Fig. 2. Block diagram of our proposed model

B Data augmerntation using interpolation

Interpolation can be described as the method for calculating unknown values from a specified values or input with the goal of identifying analytic functions that moves through a given points to interpolate for any arbitrary point. Some of the most commonly used interpolation techniques in literature include, but are not limited to linear, polynomial, spline, pchip, nearest neighbor, multi-dimensional etc. Take an unknown function $f(x)$ assuming we are given exact values at $(n+1)$ distinct points $x_0 < x_1 < \ldots < x_n$ such that the values of $f(x_0), f(x_1), \ldots, f(x_n)$ are already known.

Interpolation generates a function $Q(x)$ that moves through the known points thus identifying the function with the aim of satisfying interpolation requirements (see Fig. 3) given in Eq. (1).

$$Q(x_j)=f(x_j), 0 \le j \le n, \qquad (1)$$

For spline interpolation, we use a common cubic spline function. The cubic spline is a third degree derivative polynomial using a continuity conditions of spline interpolation. Therefore, spline interpolation is referred to as finding a polynomial on subintervals that are connected in a smooth manner. A spline of degree k is said to have a knots assuming we pick points of $(n+1)$ at $t_0 < t_1 < \ldots < t_n$. Therefore, a spline of degree k having $t_0, t_{1,} \ldots, t_n$ is a function of $s(x)$ which satisfies the two major properties:

- On $((t_{i-1}, t_i), s(x)$ is a polynomial of degree $\le k$, where $s(x)$ is a polynomial on every subinterval defined by the knots.
- Smoothness: $s(x)$ has a continuous $(k-1)$-th derivative on the interval $[t_0, t_n]$.

TABLE II. SUMMARY OF FEATURES (OXFORD PD DATASET [37])

Categories	Features
Vocal Fundamental Frequency	Average: MDVP:Fo(Hz), Maximum: MDVP:Fhi(Hz), Minimum:MDVP:Flo(Hz),
Frequency Parameters	MDVP:Jitter(%), MDVP:Jitter(Abs), MDVP:RAP, MDVP:PPQ, Jitter:DDP
Amplitude Parameters	MDVP:Shimmer, MDVP:Shimmer(dB), Shimmer:APQ3, Shimmer:APQ5, MDVP:APQ, Shimmer:DDA
Harmonicity Parameters	Noise-to-Harmonic (NHR), Harmonic-to-Noise (HNR)
Other Parameters	RPDE, D2: (Non-linear dynamical complexity measures), DFA: (Signal fractal scaling exponent), spread1, spread2, PPE: (Three nonlinear measures of fundamental frequency variation)

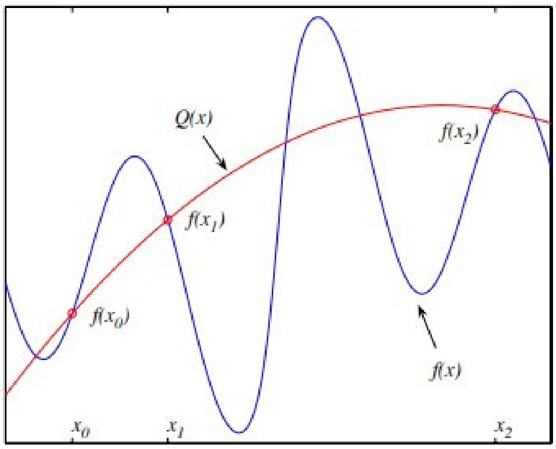


Fig. 3. A typical function $f(x)$ showing the interpolation points x_0, x_1, $x_2$and the interpolating polynomial $Q(x)$ (adopted from [40])

C Neural Network Model

Neural network models are biological inspired method which defines a function as an input (set of observations) to produces an output or decision. The elements of a neural network include the input layer (X_t) with each input layer having a neuron and the weight (Ʊ), a hidden layer (H_t) and an output (Y_t). The input layer accepts signals of examination measurements which varies from ($X_{n=iton}$) while the hidden layer processes the input signals and passes them forward to the output layer for classification.

The deep learning model used in this study is a variant of recurrent neural network (RNN) known as bi-directional LSTM (BiLSTM) model (see Fig. 4). The BiLSTM model was used with the training options: Adam optimizer, maxepoch size of "250" and gradient threshold "1", initial learning rate of 0.005. We also used a verbose of 0, piecewise learning rate schedule, and the learning rate drop period, and drop factor values are 125 and 0.2, respectively.

D Performance Metrics

The experimental result was evaluated using Accuracy (percentage of true correctness of both PD patient and healthy patients), Sensitivity (percentage of PD test for PD patients), and Specificity (percentage of healthy test for healhy patients).

IV Results and Discussion

The proposed model was implemented on Matlab R2019 (MathWorks Inc., USA) using some specific toolboxes such as classification learner for analysis on supervised ML algorithms. The result of our findings is divided into two subsection and also the comparison of our work with existing study using the same dataset is presented as well.

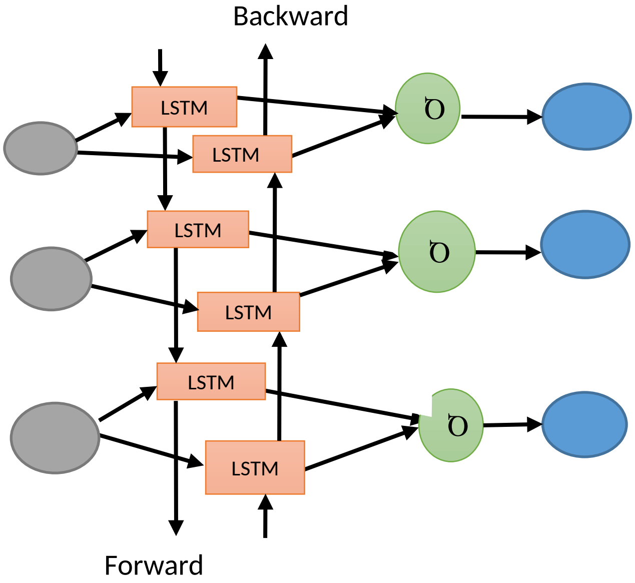


Fig. 4. The BiLSTM model used for classification

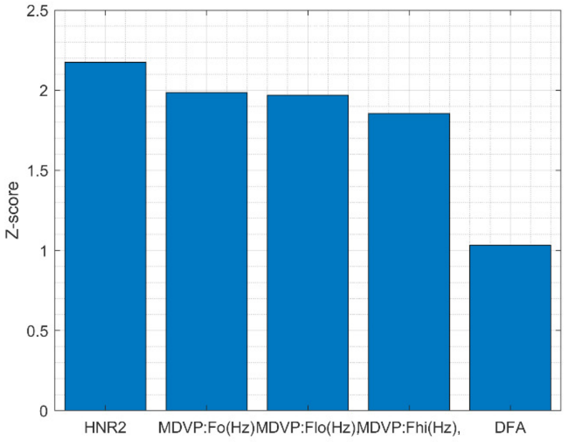


Fig. 5. The most informative features from Oxford Parkinson dataset identified using feature ranking based on non-parameteric (Wilcoxon) criterion

A Result Based on Machine Learning Algorithms

In this study, we evaluated the performance of different supervised ML algorithms such as Decision Tree, Linear Discriminant, logistic regression, SVM, KNN, and other ensemble algorithms to identify the best classifier. We investigated the performance using 5-fold cross validation. The experimental results based on the 22 features for original, augmented (spline) and augmented (pchip) datasets respectively is summarized in Table II.

Furthermore, we explored reducing feature dimensionality reduction using feature ranking based on non-parameteric (Wilcoxon) criterion and removing the highly correlated features to obtain the most relevant features. The best 5 features captured in our experiments for dimensionality reduction are: HNR2, MDVP:Fo(Hz), MDVP:Flo(Hz), MDVP:Fhi(Hz), and DFA. The results of feature ranking are presented in Fig. 5. The performance of these 5 features on traditional machine learning algorithms and BiLSTM network is presented in Table III.

Our results show that both spline and pchip augmentation was effective in allowing increasing the accuracy of classification by 4.45% ($p < 0.01$) and 4.11% ($p < 0.05$) for the 22 feature dataset, and by 4.80% ($p < 0.01$) and 1.93% (not significant) for the 5 feature dataset, when using spline and pchip augmentation, respectively (an average increase of accuracy calculated over 8 different machine learning methods). For evaluation of statistical significance, Student's two-sample t-test was used, which assumes that data are independent random samples from normal distributions.

B Results Based on BiLSTM Model

This subsection discussed the result obtained from our proposed BiLSTM model as depicted in Fig. 4. For our BiLSTM model, we used 10 % of training samples and 90% for testing for the three datasets. We selected such a small number of training samples, which is not commonly used, in order to validate our approach for solving the small dataset problem. We also analyzed with 20 hidden units and our best performance was achiedved on 250 epoch. The summary of experiment on 20 hidden neurons and 250 epochs for holdout of 90% is presented in Fig. 6. To estimate statistical confidence limits, all experiments were repeated 10 times. Our experimental results on the three data sets show a mean accuracy for 22 features as 83.49±2.33%, 96.59±1.18% and 96.2±0.75% for original, augmented with spline interpolation (Spline) and augmented with Pchip interpolation (Pchip) datasets, respectively (assuming 95% confidence level). The performance of the BiLSTM model on spline interpolated dataset gave better results when compared with overall performance on the original and augmented (Spline) dataset.

When considering different levels of holdout, the proposed data augmentation techniques allowed to improve accuracy both in case of high holdout values when little data is available for training, and in case or small holdout, when accuracy is reduced by overfitting. The results, presented in Fig. 7 show the benefit of using data augmentation techniques for both high and low holdout values.

Our results for all machine learning models trained using an augmented (spline) dataset are summarized in Fig. 8. The results show that the best results were achieved using Weighted KNN at 98%, however, the BiLSTM also achieved comparatively good results at 97.1%. In addition, the confusion matrix the best BiLSTM model with spline augmentation using 22 features) is depicted on Fig. 9.

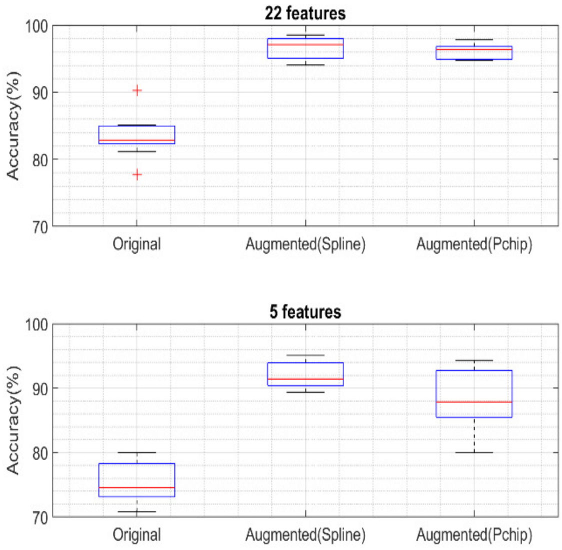


Fig. 6. Classification accuracy using original and augmented datasets and BiLSTM model with 22 and 5 dataset features

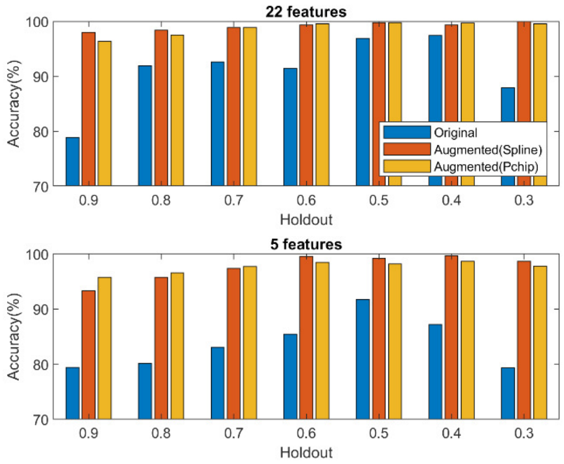


Fig. 7. Accuracy of the BiLSTM models trained with different holdout values for original and augmented datasets

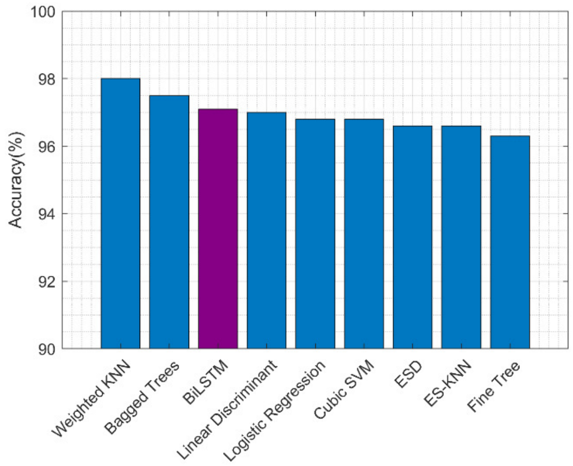


Fig. 8. Comparison of results of machine learning methods and BiL-STM model on augmented (spline) dataset

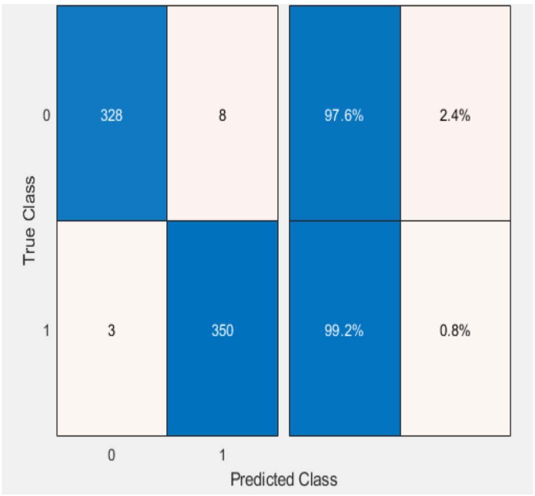


Fig. 9. Confusion matrix for best classification model using BiLSTM with 22-feature dataset augmented by spline interpolation

TABLE II.

CLASSIFICATION RESULTS ON OXFORD PARKINSON DATASET (22 FEATURES) . BEST RESULTS ARE SHOWN IN BOLD.

Algorithms	Original Data			Spline			Pchip		
	Acc (%)	Sp (%)	Sen (%)	Acc (%)	Sp (%)	Sen (%)	Acc (%)	Sp (%)	Sen (%)
Fine Tree	89.7	92.3	94.4	96.3	97.40	95.20	97.6	**97.73**	97.46
Linear Discriminant	87.2	96.3	86.5	97.0	96.37	97.74	97.0	96.37	97.74
Logistic Regression	86.2	94.4	97.7	96.8	96.36	97.18	95.3	91.47	**100**
Cubic SVM	96.6	97.8	97.8	96.8	95.85	97.74	95.1	94.43	95.76
Weighted KNN	96.6	98.8	95.5	**98.0**	**97.49**	98.59	97.3	96.92	97.74
Bagged Trees	94.0	95.6	96.6	97.5	97.46	97.46	**97.6**	97.2	98.0
Ensembled Subspace Discriminant (ESD)	92.3	93.5	96.6	96.6	94.60	**98.87**	96.5	95.57	97.46
Ensemble Subspace KNN (ES-KNN)	**97.4**	**98.9**	**97.8**	96.6	95.58	97.74	96.5	95.32	97.74

TABLE III.

CLASSIFICATION RESULTS WITH A REDUCED SET OF 5 FEATURES. BEST RESULTS ARE SHOWN IN BOLD.

Algorithms	Original Data			Spline			Pchip		
	Acc (%)	Sp (%)	Sen (%)	Acc (%)	Sp (%)	Sen (%)	Acc (%)	Sp (%)	Sen (%)
Fine Tree	94.9	96.6	96.6	97	96.9	97.12	97.6	**98.28**	96.89
Linear Discriminant	86.3	94.4	88.4	95.2	91.23	**100**	95.2	91.23	**100**
Logistic Regression	87.2	95.5	88.5	94.2	91.95	96.89	87.6	94.03	80.22
Cubic SVM	91.5	94.4	94.4	95.3	91.68	99.72	80	98.18	61.07
Weighted KNN	97.4	98.9	97.8	95.6	94.49	96.89	95.9	94.04	98.02
Bagged Trees	**94.9**	**96.6**	**96.6**	**97.5**	**97.20**	97.74	94.2	93.83	94.63
Ensembled Subspace Discriminant (ESD)	88.0	88.7	96.6	95.2	91.23	**100**	**98**	98.02	98.02
Ensemble Subspace KNN (ES-KNN)	88	87.6	96.3	96.6	95.33	98.02	95.2	91.23	**100**

TABLE IV.
COMPARISON OF CLASSIFICATION RESULTS WITH KNOWN STUDIES

Reference	Methodology	Validation Method	Accuracy (%)	Specificity (%)	Sensitivity (%)
[5]	PCA with Random Forest (RF)	-	96.87	99.85	99.75
[7]	Gaussian Process+ 5 features	10-fold CV	96.92	99.29	90
[2]	Ensemble bagging +Genetic Algorithm (GA)	-	98.28	-	-
[27]	Multilayer Feedforward Neural Network (MLFNN) with Back-propagation (BP)	10-fold CV	80.0	63.6	83.3
[39]	Kernel support vector machine	bootstrap with 50 replicates	91.4	-	-
[41]	Rough set theory	Split validation	95.0	94.0	95.0
[42]	Hybrid Relief prior and Bacterial Foraging Optimization SVM (RF-BFO-SVM)	5-fold CV	97.42	91.50	99.29
[43]	Artificial neural networks (ANN)	10-fold CV	96.88	100	95.74
[44]	Linear kernel SVM	10-fold CV	65.12	-	-
[45]	Hybrid kernel extreme learning machine approach	average 10-fold CV	95.97	91.11	97.27
[46]	Deep Autoencoder Neural Network	-	96.11	89.78	98.15
[47]	k-NN and PCA using the created ParkDet 2.0	10-fold CV	99.1	-	-
[48]	Stability Selection method using Random Forest and Logistic Regression algorithms	5 fold CV	94.36	-	-
[49]	Complex-Valued Neural Networks and mRMR Feature Selection Algorithm	10-fold CV	98.12	98.96	99.24
Our Model	BiLSTM with Original Data	Holdout	82.86	90.5	87.97
	BiLSTM with Augmentation (Spline)	Holdout	97.1	98.78	95.57
	BiLSTM with Augmentation (Pchip)	Holdout	96.37	97.94	93.14

C Statistical analysis

To compare the performance of proposed data augmentation schemes and to assess the statistical significance of the results, we have adopted the non-parametric Friedman test and post-hoc Nemenyi test, which compare the mean ranks of the methods across multiple classification runs. The results of the Nemenyi test regarding original, augmented (Spline) and Augmented (Pchip) datasets (see Fig. 10) show that the differences between mean ranks of the methods are statistically significant (Friedman's $p < 0.001$). The Critical Difference (CD) shows the smallest difference in mean ranks, where the difference is not statistically significant. Note that for the 22-feature dataset both Spline and Pchip augmentation schemes work equally well, but significantly better than using the original dataset without augmentations. The same observation is confirmed for the 5-feature dataset: both Spline and Pchip augmentation schemes allow to achieve significantly better results as compared to the results without augmentation. In the latter case, the Spline-based augmentation works better than the Pchip augmentation, but the difference is not significant (the difference between mean ranks is smaller than the CD value).

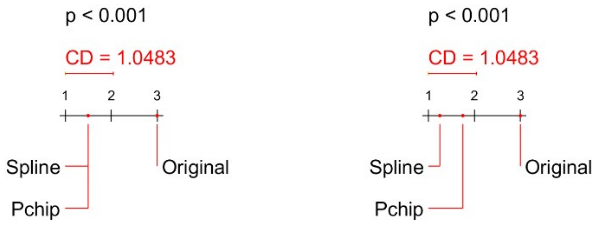


Fig. 10. Critical distance diagrams for full dataset (22 features) (left) and reduced feature dataset (5 features) (right). CD – critical distance.

D Comparison with Existing Work

For the purpose of validating our proposed method, we compared our results with previous work as shown in Table IV. Considering that the PD dataset utilized in this study has 22 features, we summarized and compared the result based on the original dataset as well as with the features obtained using feature selection. The comparison table shows the various algorithm proposed in literature with our proposed model. Our experimental results using BiLSTM with Augmentation (Spline) achieved a significant improvement in accuracy, specificity and sensitivity. However, some existing methods such as proposed in [2],[42],[47],[49] achieved accuracy between 97.42-99.1% and thereby outperformed our method. Some of limitations of this state-of-the-art methods is increasing computational complexity. Therefore, we can argue that our proposed model reveals a simple and effective method for detecting PD.

One key limitations of our data augmentation method (Interpolation) is generating noisy (out of range) values. This play a major role in affecting the performance of our model. Thus, further study is to consider more diversity among data augmentation techniques with the aim of reducing noise and error rates, and improving performances.

E Evaluation and discussion

Data augmentation using the interpolation methods allowed us to increase the accuracy of PD recognition using voice data. The application of interpolation effectively increases the resolution of captured signal, which allows to recover some of the information lost due to the microphone sampling rate that is lower than needed to solve this task. In the datased we used (Oxford Parkinson [39]), voice data was captured using a microphone with a sampling rate of 44.1 kHz. However, the study [50] concluded that a sampling rate of 96 kHz is preferred for effective PD recognition, which makes the interpolation techniques an attractive method for dealing with low resolution voice data.

Another advantage of data augmentation is the ability to increase data volume for model training. Effective training of neural networks, especially deep learning models, usually require having large amounts of data. However, in case of niche applications, such as diagnosing rare diseases, the datasets are usually small. Generation of the synthetic (surrogate) data for training allows to obtain better models, thus increasing the accuracy of classification, as also was demonstrated in this paper.

V Conclusion

The need to increase available data for classification when using small datasets with the aim of improving recognition of Parkinson disease (PD) cannot be over-emphasized. This paper effectively applied the interpolation (spline and pchip) methods for the generation of synthetic data instances thus increasing the learning samples available for training of machine learning models and improving the classification performance. This study was able to effectively address the problem of class imbalance by augmenting the original data samples using the interpolation method. Two interpolation techniques (spline and pchip) were used to generate synthetic data. A total number of 571samples was generated by each technique consisting of 320 Healthy and 251 Parkinson disease samples.

This paper investigated the performance of traditional machine learning algorithms and BiLSTM model in classifying the three categories of data samples. Our results showed that for an efficient and simple data augmentation technique based on spline and pchip interpolation have proven to be effective in the detection of PD. The analysis results for BiLSTM shows that even with a holdout of 90% for testing, the model was still able to effectively classify PD on three datasets (original Oxford Parkinson dataset, original dataset augmented using spline interpolation and original dataset using pchip intermolation) with an average accuracy of 82.86%, 97.1%, and 96.37% for the original, spline and pchip datasets, respectively (all 22 features were used). Further experiments was carried out for feature dimensionality reduction and the best results were obtained on 5 features and the the average accuracy on the 90% holdout was 74.14%, 91.44%, and 87.88% for for the original, spline and pchip datasets, respectively. The experimental results using spline augmentation have shown statistically significant ($p < 0.001$ using Friedman's test) consistency in impoving the accuracy for both 22 feature and 5 feature datasets.

The comparison of our results with the existing studies shows that the application data augmentation did not only improved accuracy, but was also able to reduce overfitting and improve the overall performance. This study was able to apply a simple BiLSTM model to effective classify speech impairment which will efficiently enhance the early detection of PD. Our proposed model based on using data augmentation techniques for small datasets showed a significant improvement in accuracy, when only a small amount of data is available for training. Note that we simulated a small dataset using an extreme value of 90% holdout for training data, which has not been used by other authors before.

Future recommendation is to explore other data augmentation methods based on different AI methods and architectural frameworks with the aim of developing an intelligent model for speech recognition for small datasets.

References

[1] Agarwal, A., Chandrayan, S. and Sahu, S.S., 2016. Prediction of Parkinson's disease using speech signal with Extreme Learning Machine. In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) (pp. 3776-3779). IEEE. Doi: 10.1109/ICEEOT.2016.7755419

[2] Fayyazifar, N. and Samadiani, N., 2017. Parkinson's disease detection using ensemble techniques and genetic algorithm. In 2017 Artificial Intelligence and Signal Processing Conference (AISP) (pp. 162-165). IEEE. Doi: 10.1109/AISP.2017.8324074

[3] Dorsey, E.R., Elbaz, A., Nichols, E., Abd-Allah, F., Abdelalim, A., Adsuar, J.C., Ansha, M.G., Brayne, C., Choi, J.Y.J., Collado-Mateo, D. and Dahodwala, N., 2018. Global, regional, and national burden of Parkinson's disease, 1990–2016: a systematic analysis for the Global Burden of Disease Study 2016. The Lancet Neurology, 17(11), 939-953. doi:10.1016/S1474-4422(18)30295-3

[4] Saikia, A., Majhi, V., Hussain, M. and Paul, S., 2019. A Systematic review on Application based Parkinson's disease Detection Systems. International Journal on Emerging Technologies 10(3): 166-173.

[5] Aich, S., Younga, K., Hui, K.L., Al-Absi, A.A. and Sain, M., 2018, February. A nonlinear decision tree based classification approach to predict the Parkinson's disease using different feature sets of voice data. In 2018 20th International Conference on Advanced Communication Technology (ICACT) (pp. 638-642). IEEE. Doi: 10.23919/ICACT.2018.8323864

[6] Chan, M.Y., Chu, S.Y., Ahmad, K. and Ibrahim, N.M., 2019. Voice therapy for Parkinson's disease via smartphone videoconference in Malaysia: A preliminary study. Journal of telemedicine and telecare, doi:10.1177/1357633X19870913

[7] Despotovic, V., Skovranek, T. and Schommer, C., 2020. Speech Based Estimation of Parkinson's Disease Using Gaussian Processes and Automatic Relevance Determination. Neurocomputing, , 401, 173–181.
doi:10.1016/j.neucom.2020.03.058

[8] Gil-Martín, M., Montero, J.M. and San-Segundo, R., 2019. Parkinson's disease detection from drawing movements using convolutional neural networks. Electronics, 8(8), p.907. doi:10.3390/electronics8080907

[9] Lavner, Y., Khatib, S., Artoul, F. and Vaya, J., 2014, December. An algorithm for processing and analysis of Gas Chromatography-Mass Spectrometry (GC-MS) signals for early detection of Parkinson's disease. In 2014 IEEE 28th Convention of Electrical & Electronics Engineers in Israel (IEEEI) (pp. 1-5). IEEE. Doi: 10.1109/EEEI.2014.7005772

[10] Liu, H.J., Li, X.Y., Chen, H., Yu, H.L., Tao, Q.Q. and Wu, Z.Y., 2020. Identification of susceptibility loci for cognitive impairment in a cohort of Han Chinese patients with Parkinson's disease. Neuroscience Letters, 135034. Doi: doi:10.1016/j.neulet.2020.135034

[11] Saikia, A., Hussain, M., Barua, A.R. and Paul, S., 2019. EEG-EMG correlation for parkinson's disease. International Journal of Engineering and Advanced Technology, 8(6), pp.1179-85. Doi: 10.35940/ijeat.F8360.088619

[12] Rumman, M., Tasneem, A.N., Farzana, S., Pavel, M.I. and Alam, M.A., 2018. Early detection of Parkinson's disease using image processing and artificial neural network. In 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR) (pp. 256-261). IEEE. Doi: 10.1109/ICIEV.2018.8641081

[13] Almeida, J.S., Rebouças Filho, P.P., Carneiro, T., Wei, W., Damaševičius, R., Maskeliūnas, R. and de Albuquerque, V.H.C., 2019. Detecting Parkinson's disease with sustained phonation and speech signals using machine learning techniques. Pattern Recognition Letters, 125, pp. 55-62. doi:10.1016/j.patrec.2019.04.005

[14] Lauraitis, A., Maskeliūnas, R., Damaševičius, R., Połap, D. and Woźniak, M., 2019. A smartphone application for automated decision support in cognitive task based evaluation of central nervous system motor disorders. IEEE journal of biomedical and health informatics, 23(5), pp. 1865-1876. Doi: 10.1109/JBHI.2019.2891729

[15] Gatsios, D., Antonini, A., Gentile, G., Marcante, A., Pellicano, C., Macchiusi, L., Assogna, F., Spalletta, G., Gage, H., Touray, M. and Timotijevic, L., 2020. Mhealth for remote monitoring and management of Parkinson's disease: determinants of compliance and validation of a tremor evaluation method. JMIR mHealth and uHealth.

[16] Linares-Del Rey, M., Vela-Desojo, L. and Cano-de la Cuerda, R., 2019. Mobile phone applications in Parkinson's disease: a systematic review. Neurología (English Edition), 34(1), pp. 38-54. doi:10.1016/j.nrleng.2018.12.002

[17] Zhang, H., Song, C., Rathore, A.S., Huang, M., Zhang, Y. and Xu, W., 2020. mHealth Technologies towards Parkinson's Disease Detection and Monitoring in Daily Life: A Comprehensive Review. IEEE Reviews in Biomedical Engineering. DOI: 10.1109/RBME.2020.2991813

[18] Zhang, H., Deng, K., Li, H., Albin, R.L. and Guan, Y., 2020. Deep Learning Identifies Digital Biomarkers for Self-Reported Parkinson's Disease. Patterns, 100042. doi:10.1016/j.patter.2020.100042

[19] Nilashi, M., Ahmadi, H., Sheikhtaheri, A., Naemi, R., Alotaibi, R., Alarood, A.A., Munshi, A., Rashid, T.A. and Zhao, J., 2020. Remote Tracking of Parkinson's Disease Progression Using Ensembles of Deep Belief Network and Self-Organizing Map. Expert Systems with Applications, 113562. doi:10.1016/j.eswa.2020.113562

[20] Lauraitis, A., Maskeliūnas, R., Damaševičius, R. and Krilavičius, T., 2020. Detection of Speech Impairments Using Cepstrum, Auditory Spectrogram and Wavelet Time Scattering Domain Features. IEEE Access, 8, 96162 – 96172. Doi: 10.1109/ACCESS.2020.2995737

[21] Lauraitis, A., Maskeliūnas, R., Damaševičius, R. and Krilavičius, T., 2020. A Mobile Application for Smart Computer-Aided Self-Administered Testing of Cognition, Speech, and Motor Impairment. Sensors, 20, 3236. doi:10.3390/s20113236

[22] Taleb, C., Likforman-Sulem, L. and Mokbel, C., 2019. Improving Deep Learning Parkinson's Disease Detection Through Data Augmentation Training. In Mediterranean Conference on Pattern Recognition and Artificial Intelligence (pp. 79-93). Springer, Cham. Doi:

[23] Połap, D., Woźniak, M., Damaševičius, R. and Maskeliūnas, R., 2019. Bio-inspired voice evaluation mechanism. Applied Soft Computing, 80, pp. 342-357. doi:10.1016/j.asoc.2019.04.006

[24] Jeancolas, L., Benali, H., Benkelfat, B.E., Mangone, G., Corvol, J.C., Vidailhet, M., Lehericy, S. and Petrovska-Delacrétaz, D., 2017. Automatic detection of early stages of Parkinson's disease through acoustic voice analysis with mel-frequency cepstral coefficients. In

2017 International Conference on Advanced Technologies for Signal and Image Processing (ATSIP) (pp. 1-6). IEEE. Doi: 10.1109/ATSIP.2017.8075567

[25] Rueda, A. and Krishnan, S., 2017. Feature analysis of dysphonia speech for monitoring Parkinson's disease. In 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 2308-2311). IEEE. Doi: 10.1109/EMBC.2017.8037317

[26] Vikas, and Sharma, R.K. 2014, May. Early detection of Parkinson's disease through Voice. In 2014 International Conference on Advances in Engineering and Technology (ICAET) (pp. 1-5). IEEE. Doi: 10.1109/ICAET.2014.7105237

[27] Wroge, T.J., Özkanca, Y., Demiroglu, C., Si, D., Atkins, D.C. and Ghomi, R.H., 2018, December. Parkinson's disease diagnosis using machine learning and voice. In 2018 IEEE Signal Processing in Medicine and Biology Symposium (SPMB) (pp. 1-7). IEEE. Doi: 10.1109/SPMB.2018.8615607

[28] Eskıdere, Ö., Karatutlu, A. and Ünal, C., 2015, September. Detection of Parkinson's disease from vocal features using random subspace classifier ensemble. In 2015 Twelve International Conference on Electronics Computer and Computation (ICECCO) (pp. 1-4). IEEE. Doi: 10.1109/ICECCO.2015.7416886

[29] Polat, K., 2019. A Hybrid Approach to Parkinson Disease Classification Using Speech Signal: The Combination of SMOTE and Random Forests. In 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT) (pp. 1-3). IEEE. Doi: 10.1109/EBBT.2019.8741725

[30] Olanrewaju, R.F., Sahari, N.S., Musa, A.A. and Hakiem, N., 2014. Application of neural networks in early detection and diagnosis of Parkinson's disease. In 2014 International Conference on Cyber and IT Service Management (CITSM) (pp. 78-82). IEEE. Doi: 10.1109/CITSM.2014.7042180

[31] Um, T.T., Pfister, F.M., Pichler, D., Endo, S., Lang, M., Hirche, S., Fietzek, U. and Kulić, D., 2017. Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks. In Proceedings of the 19th ACM International Conference on Multimodal Interaction (pp. 216-220). doi:10.1145/3136755.3136817

[32] Bourdillon, A., Sawhney, K., Mehra, R., O'Grady, P. and Liu, T., Extracting kinetic features from wearable tech for clinical symptoms of Parkinsons Disease.

[33] Vaiciukynas, E., Gelzinis, A., Verikas, A. and Bacauskiene, M., 2017. Parkinson's disease detection from speech using convolutional neural networks. In International Conference on Smart Objects and Technologies for Social Good (pp. 206-215). Springer, Cham.

[34] Bayestehtashk, A., Asgari, M., Shafran, I. and McNames, J., 2015. Fully automated assessment of the severity of Parkinson's disease from speech. Computer speech & language, 29(1), pp.172-185. Doi: 10.1016/j.csl.2013.12.001

[35] Pan Q., Li, X., and Fang L. 2020. Data Augmentation of Deep learning-based on ECG Analysis. Feature Engineering and Computational Intelligence in ECG Monitoring, 91-111. Springer Nature Singapore Pte. Doi: 10.1007/978-981-15-3824-7_6

[36] Kutlugün, M.A., Sirin, Y. and Karakaya, M., 2019. The Effects of Augmented Training Dataset on Performance of Convolutional Neural Networks in Face Recognition System. In 2019 Federated Conference on Computer Science and Information Systems (FedCSIS) (pp. 929-932). IEEE. Doi: 10.15439/2019F181

[37] Lee, J.W., Nam, D.W., Yoo, W.Y., Kim, Y., Jeong, M. and Kim, C., 2018. Soccer object motion recognition based on 3D convolutional neural networks. In FedCSIS (Communication Papers) (pp. 129-134). Doi: 10.15439/2018F48

[38] Li, Z., Yao, H., and Ma, F. (2020). Learning with Small Data. Proceedings of the 13th International Conference on Web Search and Data Mining, WSDM '20. doi:10.1145/3336191.3371874

[39] Little, M. A., McSharry, P. E., Hunter, E. J., Spielman, J., and Ramig, L. O. 2009. Suitability of dysphonia measurements for telemonitoring of Parkinson's disease. IEEE Transactions on bio-medical engineering, 56(4), 1015. doi:10.1038/npre.2008.2298.1

[40] Levy, D., 2010. Introduction to numerical analysis. Department of Mathematics and Center for Scientific Computation and Mathematical Modeling (CSCAMM) University of Maryland, pp.2-2.

[41] Revett, Kenneth, Florin Gorunescu, and Abdel-Badeeh Mohamed Salem. "Feature selection in Parkinson's disease: A rough sets approach." In 2009 International Multiconference on Computer Science and Information Technology, pp. 425-428. IEEE, 2009. Doi: 10.1109/IMCSIT.2009.5352688

[42] Cai, Z., Gu, J. and Chen, H.L., 2017. A new hybrid intelligent framework for predicting Parkinson's disease. IEEE Access, 5, pp.17188-17200. Doi: 10.1109/ACCESS.2017.2741521.

[43] Wang, X., 2014. Data Mining Analysis of the Parkinson's Disease. Masters thesis Submitted to the College of Arts and Sciences, Georgia State University.

[44] Bhattacharya, I. and Bhatia, M.P.S., 2010. SVM classification to distinguish Parkinson disease patients. In Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India (pp. 1-6). Doi: 10.1145/1858378.1858392

[45] Chen, H.L., Wang, G., Ma, C., Cai, Z.N., Liu, W.B. and Wang, S.J., 2016. An efficient hybrid kernel extreme learning machine approach for early diagnosis of Parkinson′ s disease. Neurocomputing, 184, pp.131-144. doi:10.1016/j.neucom.2015.07.138

[46] Kose, U., Deperlioglu, O., Alzubi, J. and Patrut, B., Diagnosing Parkinson by Using Deep Autoencoder Neural Network. In Deep Learning for Medical Decision Support Systems (pp. 73-93). Springer, Singapore. doi:10.1007/978-981-15-6325-6_5

[47] Ozkan, H., 2016. A comparison of classification methods for telediagnosis of Parkinson's disease. Entropy, 18(4), p.115. Doi: 10.3390/e18040115

[48] Akyol, K., Bayir, Ş. and Baha, Ş.E.N., Importance of Attribute Selection for Parkinson Disease. Akademik Platform Mühendislik ve Fen Bilimleri Dergisi, 8(1), pp.175-180. Doi: 10.21541/apjes.541637

[49] Peker, M., Sen, B. and Delen, D., 2015. Computer-aided diagnosis of Parkinson's disease using complex-valued neural networks and mRMR feature selection algorithm. Journal of healthcare engineering, 6. Doi: 10.1260/2040-2295.6.3.281

[50] Wu, K., Zhang, D., Lu, G., and Guo, Z. 2018. Influence of sampling rate on voice analysis for assessment of Parkinson's disease. The Journal of the Acoustical Society of America, 144(3), 1416–1423. doi:10.1121/1.5053681

DOI: 10.15439/2020F130
ISSN 2300-5963 ACSIS, Vol. 21

Parallel implementation of a PIC simulation algorithm using OpenMP

Alin Suciu, Anca Hangan, Anca Marginean, Marius Joldos
Technical University of Cluj-Napoca, Romania
Computer Science Department
Email: {alin.suciu, anca.hangan, anca.marginean, marius.joldos}@cs.utcluj.ro

Gabriel Voitcu, Marius Echim
Institute of Space Science, Magurele, Romania
Email: {gabi, echim}@spacescience.ro

***Abstract*—Particle-in-cell (PIC) simulations are focusing on the individual trajectories of a very large number of particles in self-consistent and external electric and magnetic fields; they are widely used in the study of plasma jets, for example. The main disadvantage of PIC simulations is the large simulation runtime, which often requires a parallel implementation of the algorithm. The current paper focuses on a PIC1d3v simulation algorithm [1][2] and describes the successful implementation of a parallel version of it on a multicore architecture, using OpenMP, with very promising experimental and theoretical results.**

I. Introduction

PARTICLE-IN-CELL (PIC) simulations are extremely useful to model self-consistently plasma phenomena at kinetic scales [3][4]. Such kind of simulations focus on the individual trajectories of a very large number of particles in self-consistent and external electric and magnetic fields. One class of important phenomena that can be modeled with PIC simulations are the high-speed plasma jets observed within Earth's magnetosheath (see, for instance, [5] and references therein). This topic is highly relevant for the geomagnetic environment (e.g. [6][7]), but it is also of great importance in other astrophysical and space science contexts, like, for instance, the interaction of the planetary/magnetospheric plasmas with solar and stellar winds or the propagation of astrophysical relativistic jets (e.g.[8][9]). More generally, the topic of high-speed jets is relevant to transport phenomena, kinetic processes and discontinuities in collisionless magnetized plasmas.

The interaction of high-speed plasma jets with non-uniform magnetic fields has been investigated over time with multiple fluid and kinetic approaches (see, for instance, [10] for a review on this topic). Nevertheless, the electromagnetic PIC approach is the most suitable tool to address this issue since it allows for the simultaneous investigation of key physical effects as self-polarization, finite Larmor radius effects and electromagnetic processes. Indeed, [11][12][13] used for the first-time such kind of simulations in a 3D geometry to investigate the interaction of high-speed plasma jets with non-uniform magnetic fields in a simplified magnetopause-like configuration typical for a northward interplanetary magnetic field.

The electromagnetic PIC approach provides a fully kinetic description of plasma by considering both self-consistent electrostatic and electromagnetic effects at microscopic level. The time-step and grid spacing used in electromagnetic PIC simulations must fulfill very restrictive stability conditions in order to avoid undesired numerical effects that could arise due to the discretization of space and time [3]. Thus, the very fine spatial and temporal resolution resolves even the smallest scales, i.e. Debye length and plasma frequency, but also leads to large simulation runtimes.

To surpass this limitation, parallelization is often used for implementing PIC simulation algorithms, and parallel algorithms for PIC simulations are present throughout the scientific literature. The solution presented in [14] uses a distributed memory model to simulate large scale systems. In their approach, parallelism is achieved through geometrical domain decomposition, each process being responsible for evolving its own sub-domain. The authors of [15] have a similar approach of domain decomposition, while investigating load-balancing strategies for both distributed memory and shared memory models.

Finally, the authors of [16] achieve parallelism by dividing particles among threads according to their positions, on a shared memory machine, while taking advantage of cache reusability. A hybrid approach is presented in [17], in which the author uses MPI for communication between processes and OpenMP to parallelize the loops inside the processes. This way, the implementation takes advantage of the fact that the processing nodes have a multi-core architecture. To obtain better performance, in some approaches [18][19], parallel implementations are optimized using techniques that take advantage of the physical parallel machine characteristics.

In this paper, we use an explicit and relativistic 1d3v electromagnetic PIC algorithm developed for one-dimensional kinetic simulation of fully-ionized collisionless magnetized plasmas. This approach considers simulation geometries having a single dimension in the configuration space and all three dimensions in the velocity space. Such an algorithm can be used to study, for instance, the formation, structure and evolution of one-dimensional tangential discontinuities, a topic of great importance for understanding the physics of high-speed jets in space plasmas (e.g. [20][21]). Given the sequential algorithm, we describe how we developed its parallel version, while discussing the parallelization strategy. To the best of our knowledge, no other parallelization of the discussed algorithm can be found in the literature.

IEEE Catalog Number: CFP2085N-ART ©2020, PTI

Algorithm 1 Algorithm PIC1d3v

```
Begin simulation
// read input parameters
0. read(m, np, nt,...);
// initialize data
1. init(...);
// execute main loop
for i=1 to nt do
    Begin
        // half-advance of magnetic field
        2. bfield(...);
        // push particles (one time-step)
        3. mover(...);
        // half-advance of magnetic field
        4. bfield(...);
        // current density computation
        5. current(...);
        // full-advance of electric field
        6. efield(...);
        // apply periodicity for particles
        7. period(...);
        // compute energy density
        8. energy(...);
        // write data to files (optional)
        9. write_files(...);
    End loop
End simulation
```

The rest of this paper is organized as follows. Section II describes the sequential PIC1d3v algorithm. Section III describes the parallelization strategy, by first analyzing the execution time of each step and investigating the formula for the overall speedup. Section IV presents the experimental results. Section V concludes the paper.

II. The PIC1d3v algorithm

The PIC1d3v algorithm that we are focusing on is based on the proposed algorithm from [1][2] and raises a series of challenges that will be detailed below, following the pseudocode of the algorithm, step by step. (see Algorithm 1 below). Moreover, the flow of the simulation is visually described in Fig. 1.

Step 0– Read input parameters

During this step of the simulation the parameters are read from the input file. The most important parameters, those who control the size of the simulation are:

1) m – the number of cells; particles are distributed randomly across these cells at the beginning of the simulation.
2) np – the number of particles, meaning that np electrons and np ions are used in the simulation.
3) nt – the number of iterations; the main loop is executed nt times.

For the range of problems studied in our research group, concrete values for m range from $1,000$ to $10,000$ cells, values for np range from $100,000$ to $1,000,000$ particles and the values for nt range from 1 to $10,000,000$, but technically these values could be increased if needed. Let us focus a bit on the largest possible simulation consisting of $10,000,000$ iterations for $1,000,000$ particles spread across $10,000$ cells. The internal memory requirements for such a simulation are not difficult to fulfill for a common serial computer available today in any research lab. However, the execution time required for such a simulation (for a common serial computer) is impressive: approximately 450 days (that is 1.267 years)! The main goal of this paper, as stated above, is to reduce this execution time as much as possible, using a multicore CPU architecture, which is very commonplace nowadays.

Step 1 – Initialize data

During this step of the simulation the position, velocity, fields, currents and energy data is initialized.

Steps 2 to 9 – The main loop

The main focus of the algorithm is the main loop, which is also clearly the most time consuming part, and therefore the focus of the parallel optimization. We will shortly describe each step of the main loop below.

Steps 2 and 4 – Half-advance of the magnetic field

In these steps, the components of the magnetic field (according to Faraday's law) are computed, at a given time, by knowing their values at the previous time-step. At each step, the magnetic field advances only for a half time-step.

Step 3 – Push particles over one time-step

This step of the algorithm computes the electric and magnetic fields in the actual position of each particle. Then, it moves the particle to its new position after one time-step and recomputes its velocity.

Step 5 – Current density computation

At this step, current density is computed for particles, while applying periodic boundary conditions and a smoothing procedure.

Step 6 – Full-advance of the electric field

This step of the algorithm computes the components of the electric field from Ampere's law, at a given time, by knowing their values at the previous time-step.

Step 7 – Apply periodicity for particles

This step of the algorithm applies periodicity for each particle.

Step 8 – Compute energy density

This step of the algorithm computes the energy density of the system and does not interfere with the simulation. This step is used for diagnosis purposes throughout the simulation.

Step 9 – Write data to files (optional) For each iteration, computed data can be saved in binary output files, if intermediate data is required by the user.

III. Parallel implementation of the PIC1d3v algorithm

When considering a parallel implementation, the first question that needs to be addressed refers to the targeted parallel hardware architecture (multicore, GPU, cluster, etc.). For the purpose of our simulation needs we considered a multicore architecture as our target, so the natural choice for the software platform, given that our serial implementation was done in C/C++, was OpenMP.

With regard to the parallelization strategy/methodology, a closer look at Fig. 1 and Algorithm 1 reveals data dependencies among the steps of the sequential algorithm (e.g. to compute the electric field we need to compute the magnetic field and the currents first) which prevent loop iterations to be executed in parallel; so the only hope that remains is to concentrate our efforts on parallelizing the individual steps, with a focus on the main loop. We will discuss below the parallel approach taken for each step of the main loop of the algorithm.

A. Parallelization analysis

In order to compute the speedup of the main loop, as each step of the computation could be inherently sequential or maybe have some degree of parallelism, we need to apply Amdahl's law. Amdahl's law fits perfectly our scenario of having a fixed size problem that we want to solve in as little time as possible (also, processors have the same architecture, frequency, cache size, etc.). Therefore, we need to know the fractions of the computation, each fraction corresponding to one step of the main loop (in order to be able to apply Amdahl's law). We have to compute f_2 to f_9, the fractions of the total sequential execution time spent by each step of the main loop. To do so, we need to measure the total sequential execution time of the main loop (denoted as t_{loop}) and the sequential execution time for each step of the loop: $t_2, t_3, ..., t_9$. We then compute:

$$f_i = \frac{t_i}{t_{loop}}, \text{for } i = \overline{2, 9}. \quad (1)$$

In the following, we analyze each step of the main loop, in the maximal simulation scenario ($m = 10.000$, $np = 1.000.000$), in order to identify the hotspots and the bottlenecks. We then concentrate on parallelizing the hotspots, as this will give us the highest overall speedup. We also provide concrete values for each fraction of the main loop, in this scenario, so that we can identify the most significant steps and concentrate the parallelization efforts there. The concrete values for each fraction mentioned below are taken from Table I below, to illustrate the impact of each step of the algorithm.

Steps 2 and 4 – Half-advance of the magnetic field

These steps account (each) for $f_2 = f_4 = 0.01\%$ of the execution time of the main loop, due to the fact that it mainly consists of a single “for" loop of m iterations. We did not parallelize these steps because the overhead induced by the parallelization (thread creation and management, synchronization) would only lead to a slowdown instead of a speedup of the execution.

Step 3 – Push particles over one time-step

This step accounts for $f_3 = 74.83\%$ of the execution time of the main loop, and we were able to parallelize this step with the help of the “parallel for” directives. No data dependencies were detected so the parallelization was straightforward.

Step 5 – Current density computation

This step accounts for $f_5 = 8.39\%$ of the execution time of the main loop. Unfortunately this step requires a lot of synchronization among threads, so only minor parts of it were parallelizable. The “parallel for” directive was used where possible.

Step 6 – Full-advance of the electric field

This step accounts for $f_6 = 0.04\%$ of the execution time of the main loop, due to the fact that it mainly consists of a single "for" loop of m iterations. We did not parallelize this step because the overhead induced by the parallelization (thread creation and management, synchronization) would only lead to a slowdown instead of a speedup of the execution.

Step 7 – Apply periodicity for particles

This step accounts for $f_7 = 0.54\%$ of the execution time of the main loop; although it accounts for a very small fraction of the main loop, it has an embarrassingly parallel structure, being in fact just one loop controlled by the np parameter, so we used a “parallel for” directive to parallelize it.

Step 8 – Compute energy density

This step accounts for $f_8 = 16.170\%$ of the execution time of the main loop, and we were able to parallelize this step with the help of the “parallel for” directives. No data dependencies were detected so the parallelization was straightforward.

Step 9 – Write data to files (optional)

This step is optional so we will ignore it for now, especially since it represents I/O time and therefore it cannot be improved by parallelism. However, experiments performed at the maximum size of the simulation show that the impact of this step is negligible.

B. Computing the speedup

Given that we know the fractions of the main loop, f_2 to f_8, and assuming that we can compute (for a certain scenario, on a certain architecture, with a certain number of processors) the maximum speedup for each fraction, s_2 to s_8, then the overall speedup for the main loop is given by the following formula, derived from Amdahl's law:

$$s_{loop} = \frac{1}{\sum_{i=2}^{8} \frac{f_i}{s_i}} \quad (2)$$

One can notice that for steps 2, 4 and 6 of the main loop, the speedup is 1, since these are the serial fractions of the loop; on the other hand, all the other fractions will have a speedup larger than 1, but we expect different speedups for different steps, due to different degrees of parallelization that are possible for each of them. If we take a look at the initialization steps (steps 0 and 1), we notice that step 1 is inherently sequential but step 2 has some potential for parallelization; let us assign fractions f_0 and f_1 to these steps too, and also the speedups $s_0 = 1$ and $s_1 > 1$ (s_1 will be determined experimentally). Then the total speedup in this scenario is, by a similar formula:

$$s_{tot} = \frac{1}{\frac{f_0}{s_0} + \frac{f_1}{s_1} + \frac{f_{loop}}{s_{loop}}} \tag{3}$$

The problem however is that these fractions are not constant, they depend upon the parameter nt; let us consider first $nt = 1$, then we can measure t_{01}, t_{11} and t_{loop1} respectively, the execution time for the first two steps and the loop, with one execution of the main loop ($nt = 1$). The corresponding fractions are then:

$$f_{01} = \frac{t_{01}}{t_{01} + t_{11} + t_{loop1}} \tag{4}$$

$$f_{11} = \frac{t_{11}}{t_{01} + t_{11} + t_{loop1}} \tag{5}$$

$$f_{loop1} = \frac{t_{loop1}}{t_{01} + t_{11} + t_{loop1}} \tag{6}$$

and in general for nt iterations of the main loop, we have:

$$f_{0nt} = \frac{t_{01}}{t_{01} + t_{11} + nt * t_{loop1}} \tag{7}$$

$$f_{1nt} = \frac{t_{11}}{t_{01} + t_{11} + nt * t_{loop1}} \tag{8}$$

$$f_{loopnt} = \frac{nt * t_{loop1}}{t_{01} + t_{11} + nt * t_{loop1}} \tag{9}$$

Therefore, to compute the respective fractions, we only need to measure the execution time of one iteration of the main loop, which is feasible in practice and quite fast; based on that, considering the impact of the initialization steps, we can compute the total speedup of the algorithm as:

$$s_{tot} = \frac{1}{\frac{f_{0nt}}{s_0} + \frac{f_{1nt}}{s_1} + \frac{f_{loopnt}}{s_{loop}}} \tag{10}$$

We may notice that if the parameter nt grows, the impact of the initialization steps, as expected, becomes more and more negligible, as f_{loopnt} approaches 1 and thus s_{tot} approaches s_{loop}.

IV. Experiments and results analysis

We performed a series of experiments on a multicore (quad core) computer server (Intel®Core2^{TM} Quad CPU - Q8400 @ 2.66GHz processor, 8GB RAM).

First, we tested the limits of the simulation with regards to the size of the problem, and we encountered no problems and no limitations whatsoever, the only limit being the execution time, which grows in direct proportion with the growth of the nt parameter. To estimate the achievable speedup we performed a test with the following parameters (maximum size simulation scenario): $m = 10,000$; $np = 1,000,000$; $nt = 10$.

We computed the average execution time for each step of the main loop both for the serial execution (1 processor, 1 thread) and the parallel execution with the maximum number of processors (4 processors, 4 threads).

Thus, we could compute the speedups for each of the parallelized steps, and we obtained the results presented in Table I. Following equation (2), we compute the main loop speedup as:

$$s_{loop} = \frac{1}{0.377496} = 2.649 \tag{11}$$

One can notice that the computed formula matches exactly the value from the lower right corner of Table 1, so the theoretical prediction perfectly matches the experimental result; the impact of the serial fractions of the problem is concentrated in step 5 (current) and we were able to obtain an overall speedup for the main loop of 2.649 on a quad core computer. Considering that the impact of the initialization steps is negligible for a very large number of iterations, we may conclude that the overall speedup is approximately equal to the loop speedup:

$$s_{tot} \approx s_{loop} = 2.649 \tag{12}$$

Since the number of processors used for the parallel execution was $n = 4$, we can also compute the overall efficiency as:

$$e_{tot} = \frac{s_{tot}}{n} = \frac{s_{tot}}{4} = 66.23\% \tag{13}$$

A simple computation shows that this will reduce the execution time of the maximal simulation from 450 days to 170 days on the tested architecture. However, the number of cores can be increased even further and thus, with no changes to the implementation the execution time will be reduced even further. Another advantage of this approach is that we know in advance how much time the parallel simulation will take, so we can decide to launch it or not, perhaps adjust the setup (e.g. increase the number of cores, decrease the umber of iterations, etc.), and then we can re-compute the speedup and so on.

An important issue that we considered is the precision of the obtained results, given the fact that the algorithm uses a lot of floating point computations, and it is well known that even addition is not associative in floating point arithmetic. Thus we performed a series of experiments where we measured the maximal difference between the serial values and the parallel values, and the experiments showed that parallelization did not have a significant impact on the final and the intermediary results. The maximal difference was of order 10^{-9}, which is acceptable for the considered PIC simulation scenarios.

V. Conclusion

PIC simulations play an important role when simulating the individual trajectories of a very large number of particles in self-consistent and external electric and magnetic fields. We used here an explicit and relativistic 1d3v electromagnetic PIC algorithm developed for one-dimensional kinetic simulation of fully-ionized collisionless magnetized plasmas.

Starting from a PIC1d3v serial algorithm whose execution, for the largest simulation scenario) takes over a year, we were

TABLE I
EXECUTION TIMES, FRACTIONS AND SPEEDUPS FOR THE MAIN LOOP

Step no. (name)	Serial time (ms)	Fraction(%)	Parallel time (ms)	Speedup
2 (bfield)	0.57	0.0147	0.57	1.000
3 (mover)	2904.26	74.8314	868.60	3.344
4 (bfield)	0.56	0.0144	0.56	1.000
5 (current)	325.68	8.3915	305.64	1.066
6 (efield)	1.58	0.0407	1.58	1.000
7 (period)	20.77	0.5352	10.04	2.069
8 (energy)	627.65	16.1721	278.10	2.257
LOOP	3881.07	100	1465.09	2.649

able to significantly reduce the execution time by using a multicore CPU architecture, which is common place nowadays, and the clever use of the OpenMP directives, with a minimal modification of the original serial C/C++ code.

We performed a thorough analysis of the sequential algorithm and identified each step, each fraction of the problem, each dependency, computed the speedup for each fraction and finally computed the overall speedup, which shows promising results and enables even more speedups to be obtained on a multicore architecture with more cores, with no changes to the parallel code. Based on the promising results of the PIC1d3v parallel simulation presented here, we plan to move to a full PIC3d parallel simulation, following the same fundamental ideas presented in this paper.

ACKNOWLEDGMENT

This work was supported by the Romanian Ministry of Research and Innovation via a PCCDI grant (PCCDI Project no. 18 PCCDI/2018). Gabriel Voitcu and Marius Echim acknowledge support from the Romanian Ministry of Education and Research through the Core Programme LAPLAS VI/2020.

REFERENCES

[1] Omura, Y., Matsumoto, H. “KEMPO1: Technical Guide to One-dimensional Electromagnetic Particle Code", in Computer Space Plasma Physics: Simulation Techniques and Software, edited by H. Matsumoto and Y. Omura, pp. 21-65, Terra Scientific Publishing Company, Tokyo, 1993.

[2] Voitcu, G. "Kinetic simulations of plasma dynamics across magnetic fields and applications to the physics of planetary magnetospheres”, PhD thesis, University of Bucharest, Romania, 2014.

[3] Birdsall, C. K., Langdon, A. B.“Plasma physics via computer simulation”, Boca Raton: CRC Press, 1991, doi:10.1201/9781315275048.

[4] Hockney, R. W., Eastwood, J. W. “Computer simulation using particles”, Boca Raton: CRC Press, 1988, doi: 10.1201/9780367806934.

[5] Plaschke, F., Hietala, H., Archer, M., Blanco-Cano, X., Kajdic, P., Karlsson, T., Lee, S. H., Omidi, N., Palmroth, M., Roytershteyn, V., Schmid, D., Sergeev, V., Sibeck, D. Jets Downstream of Collisionless Shocks, Space Science Reviews, 214, 81, 2018, doi:10.1007/s11214-018-0516-3.

[6] Hietala, H., Partamies, N., Laitinen, T. V., Clausen, L. B. N., Facsko, G., Vaivads, A., Koskinen, H. E. J., Dandouras, I., Reme, H., Lucek, E. A. “Supermagnetosonic subsolar magnetosheath jets and their effects: from the solar wind to the ionospheric convection”, Annales Geophysicae, 30, 33, 2012, doi:10.5194/angeo-30-33-2012.

[7] Archer, M. O., Hietala, H., Hartinger, M. D., Plaschke, F., Angelopoulos, V. “Direct observations of a surface eigenmode of the dayside magnetopause”, Nature Communications, 10:615, 2019, doi:10.1038/s41467-018-08134-5.

[8] Karlsson, T., Liljeblad, E., Kullen, A., Raines, J. M., Slavin, J. A., Sundberg, T. “Isolated magnetic field structures in Mercury’s magnetosheath as possible analogues for terrestrial magnetosheath plasmoids and jets”, Planetary and Space Science, 129, 61, 2016, doi:10.1016/j.pss.2016.06.002.

[9] Nishikawa, K.-I., Frederiksen, J. T., Nordlund, A., Mizuno, Y., Hardee, P. E., Niemiec, J., Gomez, J. L., Peer, A., Dutan, I., Meli, A., Sol, H., Pohl, M., Hartmann, D. H. “Evolution of global relativistic jets: collimations and expansion with kKHI and the Weibel instability”, The Astrophysical Journal, 820:94, 2016, doi: 10.3847/0004-637X/820/2/94.

[10] Echim, M. M., Lemaire, J. F. “Laboratory and numerical simulations of the impulsive penetration mechanism”, Space Science Reviews, 92, 565, 2000, doi:10.1023/A:1005264212972.

[11] Voitcu, G., Echim, M. “Transport and entry of plasma clouds/jets across transverse magnetic discontinuities: Three-dimensional electromagnetic particle-in-cell simulations”, Journal of Geophysical Research - Space Physics, 121, 5, 4343-4361, 2016, doi:10.1002/2015JA021973.

[12] Voitcu, G., Echim, M. „Tangential deflection and formation of counterstreaming flows at the impact of a plasma jet on a tangential discontinuity”, Geophysical Research Letters, 44, 12, 5920-5927, 2017, doi:10.1002/2017GL073763.

[13] Voitcu, G., Echim, M. „Crescent-shaped electron velocity distribution functions formed at the edges of plasma jets interacting with a tangential discontinuity”, Annales Geophysicae, 36, 1521-1535, 2018, doi:10.5194/angeo-36-1521-2018.

[14] Bart, G, Peltz, C, Bigaouette, N, Fennel, T, Brabec, T, Varin, C. Massively parallel microscopic particle-in-cell. Computer Physics Communications. 2017 Oct 1;219:269-85.

[15] Miller, KG, Lee, RP, Tableman, A, Helm, A, Fonseca, RA, Decyk, VK, Mori, WB. Dynamic load balancing with enhanced shared-memory parallelism for particle-in-cell codes. arXiv preprint arXiv:2003.10406. 2020 Mar 23.

[16] Shah, K, Phadnis, A, Shah, M, Chaudhury, B. Parallelization of the Particle-In-Cell Monte Carlo Collision (PIC-MCC) Algorithm for Plasma Simulation on Intel MIC Xeon Phi Architecture. In proceedings of International Conference for High Performance Computing 2017.

[17] Sáez, X., Soba, A., Cela, J.M., Sánchez, E., Castejón, F. Particle-In-Cell algorithms for Plasma simulations on heterogeneous architectures. In 2011 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing 2011 Feb 9 pp. 385-389, doi:10.1109/PDP.2011.42

[18] Marszałek, Z., Woźniak, M., Połap, D., Fully flexible parallel merge sort for multicore architectures. Complexity, 2018, doi: 10.1155/2018/8679579

[19] Palkowski, M., Bielecki, W., "Parallel cache-efficient code for computing the McCaskill partition functions," 2019 Federated Conference on Computer Science and Information Systems (FedCSIS), Leipzig, Germany, 2019, pp. 207-210, doi: 10.15439/2019F8.

[20] Roth, M., De Keyser, J., Kuznetsova, M. M. “Vlasov theory of the equilibrium structure of tangential discontinuities in space plasmas”, Space Science Reviews, 76, 251, 1996, doi:10.1007/BF00197842.

[21] Echim, M. M., Lemaire, J. F., Roth M. “Self-consistent solution for a collisionless plasma slab in motion across a magnetic field”, Physics of Plasmas, 12, 072904, 2005, doi: 10.1063/1.1943848.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 387–390 DOI: 10.15439/2020F82 ISSN 2300-5963 ACSIS, Vol. 21

Retrieving Sound Samples of Subjective Interest With User Interaction

Jan Jakubik
Department of Computational Intelligence
Wroclaw University of Science and Technology
jan.jakubik@pwr.edu.pl

***Abstract*—This paper concerns the retrieval of audio samples with a high degree of user interaction, motivated by a practical use case. We consider an open set recognition scenario in which the goal is to find all occurrences of a subjectively interesting sound selected by a user within a particular audio file. We use only a single starting example and maintain interaction through yes-no answers from the user, indicating whether any new retrieved sound matches the target pattern. We present a small dataset for this task and evaluate a baseline solution based on Nonnegative Matrix Factorization and greedy feature selection.**

***Index Terms*—music information retrieval, matrix decomposition, active learning**

I. Introduction

AUDIO retrieval is a well-established research area with multiple practical use cases. Between music audio [1], sound effect [2], and speech [3] analysis, numerous research problems have been established and tackled with a range of techniques from the areas of signal processing and machine learning. Most novel approaches developed in recent years have leveraged the success of deep learning [4] and more generally, machine learning methods have been deployed in the area for decades.

However, machine learning systems and the methodology of their evaluation can arise concerns about their practicality. The majority of ML systems are evaluated with the implicit assumption of availability of annotated data with a distribution identical to that of the real domain. Quite often, classification tasks are defined with the assumption that the set of classes that need to be recognized does not change over time [5]. Approaches which adress these problems in current ML literature are known as zero-shot and one-shot learning, and the tasks they solve can be described as open-set recognition [6]. The prior knowledge of the target problem is minimized at the time of training and the goal instead is to maximize the system's capability to tackle new problems with as little data as possible.

In this paper, we consider a sound sample retrieval system which requires an open-set recognition scenario. We are motivated by a practical task defined within an R&D project in cooperation with a game development company. Specifically, we attempt to search for sound samples of subjective interest in electronic music, without any prior knowledge of what distinguishing features these sounds may have. To guide the systems' decisions, we are using only a single positive sample and responses resulting from user interactions to narrow down the search. The goal of this system is to allow efficient creation of content synchronized with music audio and as such, it should minimize the user's effort while achieving maximal recall.

The contributions of this paper are as follows: we propose a well-defined, zero-shot active learning scenario, motivated by a practical use case. We provide an evaluation dataset focused on electronic music composed of repetitive samples. The dataset is annotated based on listeners' subjective notion of what constitutes a ”sound of interest”. Finally, we evaluate an approach that may serve as a simple non-deep learning baseline for this problem and a reference for future work.

II. Related Work

Sound effect retrieval has been considered in several contexts, however, it is usually not in an active learning scenario. The exsiting work [7] focuses on general sound effects against non-musical background. Recent papers have attempted zero-shot learning for music auto-tagging with good results [8].

Active learning techniques have been developed mainly with the goal of lowering annotation costs of full datasets. As such, two types of techniques are typically proposed [9]. The first group consists of approaches based on model uncertainty [10], in which the selection of a new sample is based on the ”difficulty” of training samples. A number of criteria for selecting difficult samples have been proposed. The second group of active learning approaches exploits the distribution of samples over the feature space or label space and aims to draw the most representative samples [11]. Approaches of this type can utilize clustering methods to find good representative samples for the entire dataset.

Zero-shot learning has been considered mostly in the area of deep learning, where the ability of artificial neural networks to learn meaningful features from a low-level representation of data can be leveraged [12]. A pre-trained deep network feature extractor allows comparison of samples that emphasizes semantic similarities. This type of extractor can be trained without prior knowledge of classes that need to be recognized.

III. Materials and Methods

In this section, data and methods employed in the study are described. We summarize the problem definition, the dataset gathered for validation of the developed methods and standard signal processing and machine learning approaches that can be employed to build a baseline system for this task.

A. Problem Definition

The task in question was defined as the retrieval of interesting sounds, with a focus on particular samples that may be used repeatedly in electronic music. The use scenario was described as follows: an end-user, programmer or game designer, should be able to mark an interesting excerpt within the audio file that contains a "sound of interest". The nature of such a sound is not well-defined. Other occurrences of the sound can be slightly altered. The sound can be easily recognized against a variety of audio backgrounds.

Since a machine learning system for such a task requires a training dataset of positive or negative samples, and the user should not be expected to supply annotations before the retrieval process, we opted for a solution that combines active learning and zero-shot learning approaches. The user only supplies a single positive example and then receives a new sample after each query which they can mark as either positive or negative. This process continues for a limited number of queries, given by a pre-defined budget. The scenario is consistent with the definition of active learning but differs in detail from typically considered AL scenarios. In particular, the standard AL approaches seek to maximize overall performance inprovements, and are considered for entire datasets. Our scenario on the other hand aims to minimize user interaction specifically with negative samples while retrieving all positive samples from a single audio file. This difference is meaningful because many AL approaches base their choice of data for annotation on an uncertainty criterion, i.e., the user would be shown samples that are "equally likely" to be positive or negative.

Algorithm 1 Active Retrieval Procedure

```
function RETRIEVE($t_0$, l, b)
    $P \leftarrow \{t_0\}$
    $N \leftarrow \emptyset$
    while $|N| < b$ do
        $newsample \leftarrow GetBestSamples(P, N, l)$
        if $UserResponse(newsample) = positive$ then
            $P \leftarrow P \cup \{newsample\}$
        else
            $N \leftarrow P \cup \{newsample\}$
        end if
    end while
    return $P$
end function
```

Formally, our input is a sequence of vectors $X = (x_1, x_2, ...x_n)$ which represents the sound file (detailed in subsections C and D), starting point t_0 and length l of the initial positive example, and an answer budget b, which represents the user's patience. We seek a function that given two sets of positive examples P and negative examples N returns the time points at which other occurences of the sound of interest start. The overall search procedure is given by Algorithm 1, in which $GetBestSamples$ is the method used for retrieval (described in subsections E and F), while the function $UserResponse$ is the true-false response from user interaction.

B. The Dataset

The dataset for this study was created based on songs available at sampleswap.com, a website offering a variety of creative commons licensed electronic music. Our goal was to represent the use case of searching for electronic samples of interest, in particular, characteristic and repeating sound effects. The key issue here is that the samples may be present with a variety of different musical backgrounds.

300 audio files were chosen from the sampleswap.com repository and annotated by three people - two trained musicians and one non-musician. The annotations were based on a subjective notion of an interesting sound, with the caveat that the sound must occur multiple times within the file.

Files from four genre categories were annotated - Dubstep, House, Downtempo and Drum'n'Bass. Most audio files in the data were approximately 2 minutes long and the length of "sounds of interests" ranged between 1 and 7 seconds.

The dataset is available on request.

C. Representations of Audio Data

Within the study, we compare three different representations of audio data. We employ the standard approach of representing the sound file as a series of real-space vectors.

Basic approaches in music analysis use spectrogram representations that capture the local distribution of the signal's power over frequency bins. In this study, we include standard Short-Term Fourier Transform (STFT) with linear frequency scale, as well as Constant-Q Transform (CQT), in which frequency bins are spaced logarythmically. The latter corresponds better to the psychoacoustic properties of human hearing, as well as standard western musical scales.

In genre analysis and sound classification, another significant method of audio representation are Mel-Frequency Cepstral Coefficients [13]. MFCC are especially well known for capturing the timbral properties of sound and were commonly used in all types of audio ML tasks before the dominance of deep learning. MFCC can be extracted from short frames of audio and used to create a vector sequence analogous to a spectrogram.

In recent work on music analysis, it is also very common to use learned representations extracted by a pre-trained deep neural network, usually convolutional (CNN). In our preliminary experiments we tested neural representations transferred from other tasks (e.g., CNN autotagging on IRMAS dataset, classification on GTZAN). However, we failed to find one that outperforms NMF on the aformentioned standard features.

D. Matrix Decomposition Approach

Matrix decomposition methods are a standard approach in audio signal processing. Through matrix factorization methods, every spectrogram frame can be expressed as a linear combination of a number of base vectors corresponding to commonly occurring ”sound components”. Our baseline approach uses a factorized representation to identify the key components of the sound of interest and search for other occurrences within the audio file. Unlike deep learning feature extractors, MF representation can be trained on the level of a single audio file and only separate components relevant to that particular file, which makes it a good baseline without the requirement of training on a large dataset.

A nonnegative matrix factorization (NMF) is a decomposition that represents a given matrix X as a product of two nonnegative matrices W and H. As an optimization problem, NMF is obtained by solving Eq. 1:

$$\arg \min_{W>0,H>0} \|X - WH\|_F^2 \tag{1}$$

Additional constraints can be imposed to induce sparsity on matrices W, H, or minimize their Frobenius norm. For our baseline, we use the implementation in the scikit-learn library that allows both Frobenius norm and L1 norm regularization. The exact optimization problem in the scikit-learn implementation is formulated as follows (Eq. 2):

$$\begin{aligned} \arg \min_{W>0,H>0} ||X - WH||_F^2 + \alpha l(||W||_1 + ||H||_1) + \\ +\alpha(1-l)(||W||_F^2 + ||H||_F^2) \end{aligned} \tag{2}$$

The NMF representation allows for better retrieval of sounds of interest when other background sounds are present. The simple model can separate different additive components of the data matrix, and our expectation is that this will help separate different sound sources, including one corresponding to the sample of interest.

E. Feature Selection

Our baseline idea is to use NMF to separate the sound of interest from its background. It is particularly useful to employ a selection mechanism that chooses only the NMF components that give the best separation between positive and negative samples. This selection is performed after every user decision. Formally, the criterion is defined in Eq. 3:

$$\arg\max_i \min_{x \in P, y \in N} \|f_i(x) - f_i(y)\|_2 \tag{3}$$

where $f_i(x)$ indicates the i-th feature value of the sample x. The feature selection is greedy and the size of the selected subset of features is a hyperparameter we tune experimentally.

F. Best Sample Retrieval

Feature selection is repeated after every user decision, and the actual retrieval procedure is then based on simple nearest neighboor. Using the selected features, we choose an excerpt of given length with the lowest distance from its closest neighboor set of positives that does not overlap with any of the already returned excerpts.

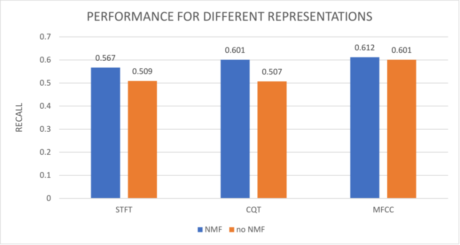


Fig. 1. Results of retrieval depending on audio representation

IV. Results

The experiments were performed using librosa [14] and scikit-learn [15] libraries. Implementations of standard STFT, CQT, and MFCC extraction in librosa were all used with default parameters. The data was then decomposed using a version of NMF provided in scikit-learn.

The classifier is a simple nearest neighbor method that returns the new sample that minimizes l2 distance to the closest positive sample while not overlapping with any of the samples in either positive or negative set. We use an answering budget of $b = 10$ and use recall as our main figure of merit. Since annotations are not perfectly timed due to human limitations, we consider every retrieved excerpt that overlaps in time with a ground truth excerpt for more than half of its duration as a true positive.

A. Comparison of Audio Representations

The first experiment compares the results achieved with different audio representations. MFCC, CQT, and STFT representations are compared in two variants: without NMF and after NMF. Results in terms of the recall are presented in Fig. 1.

There is a clear improvement resulting from the use of NMF to separate the basic components of the sound. In addition, the use of MFCC appears to be preferable to the use of standard frequency-domain transforms. The best recall of 61% is achieved with NMF of an MFCC representation.

B. Influence of NMF Hyperparameters

The purpose of this second experiment is to examine the influence of NMF hyperparameters on the recall of the active retrieval procedure. The regularization of NMF is of key importance in the methods’ capability to separate distinct sounds. In particular, the sparsity parameter encourages separation into a sparse dictionary of sounds, i.e., every frame of the source data can be expressed as a sum of only a few components. Additionally, the number of components itself is a hyperparameter that will affect the results significantly.

Fig. 2 shows the result of comparison of recall depending on the number of NMF components. There is a clear negative

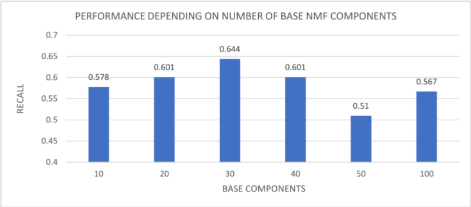


Fig. 2. Results of retrieval depending on number of NMF components.

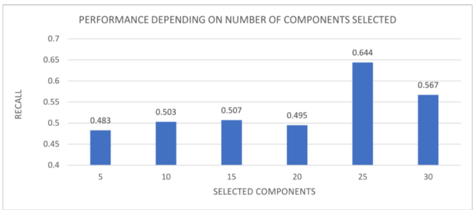


Fig. 3. Results for different number of selected features.

trend past 20 components, suggesting that number of components is sufficient to express relevant information about sounds within a single file.

We have also tested changes to L1 ratio and alpha parameter of regularization. Overall, the default parameters of NMF supplied in Librosa appear to give good results. The changes from parameter tuning offer a small gain over the default parameterization, and the use of regularization compared to lack thereof improves the recall by a margin of 1% at best. The use of L1 regularization, despite some intuitive basis for it, only worsens the results.

C. Influence of feature selection

In this experiment, we evaluate the influence of feature selection on the final result. The experiments are performed with the best parameters chosen from previous tests: 30 NMF components, $\alpha = 1$, and no L1 regularization. The results are presented in Fig. 3.

There is a visible positive effect of active feature selection on the results when the number of features selected is slightlly lower than the number of base components.

V. Conclusions and Future Work

We have defined a practically motivated sound retrieval task and presented a dataset and a simple baseline approach for its evaluation. Our task concerns retrieval of sounds of subjective interest within a single audio file based on user interaction in the form of simple yes-no answers. The presented solution uses Nonnegative Matrix Factorization to identify a base of audio components and feature selection to focus on components specific to the sound of interest. Simple nearest neighbor is then used to find the potential answers to the user's query.

The baseline demonstrates performance of 64% recall, which leaves significant room for improvement. The key area of future work is the potential use of feature learning, in particular deep network representations trained on sufficiently large unannotated data. It is also likely that the feature selection approach can improved beyond simple greedy selection based on the distance criterion.

VI. Acknowledgements

This research was carried out in cooperation with Scalac sp. z o.o. and Vixa Games game development team as part of the "Elaboration and implementation of audio modules (audio tracks indexing and analysis) and video modules (visualization of processed data) involving unique sound and graphic design use for the multimedia applications requirements" project co-funded by the European Union.

References

[1] Rainer Typke, Frans Wiering, and Remco Veltkamp. A survey of music information retrieval systems. pages 153–160, 01 2005.

[2] Douglas Turnbull, Luke Barrington, David Torres, and Gert Lanckriet. Semantic annotation and retrieval of music and sound effects. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):467–476, 2008.

[3] Moataz El Ayadi, Mohamed S. Kamel, and Fakhri Karray. Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recogn.*, 44(3):572–587, March 2011.

[4] Allen Huang and Raymond Wu. Deep learning for music. *arXiv preprint arXiv:1606.04930*, 2016.

[5] Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boult. Toward open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1757–1772, 2012.

[6] Wei Wang, Vincent W Zheng, Han Yu, and Chunyan Miao. A survey of zero-shot learning: Settings, methods, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–37, 2019.

[7] Zhao Shuyang, Toni Heittola, and Tuomas Virtanen. Active learning for sound event classification by clustering unlabeled data. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 751–755. IEEE, 2017.

[8] Jeong Choi, Jongpil Lee, Jiyoung Park, and Juhan Nam. Zero-shot learning for audio-based music classification and tagging. *arXiv preprint arXiv:1907.02670*, 2019.

[9] Yifan Fu, Xingquan Zhu, and Bin Li. A survey on instance selection for active learning. *Knowledge and information systems*, 35(2):249–283, 2013.

[10] A. Holub, P. Perona, and M. C. Burl. Entropy-based active learning for object recognition. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8, 2008.

[11] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

[12] Jeong Choi, Jongpil Lee, Jiyoung Park, and Juhan Nam. Zero-shot learning for audio-based music classification and tagging. *arXiv preprint arXiv:1907.02670*, 2019.

[13] Vibha Tiwari. Mfcc and its applications in speaker recognition. *International journal on emerging technologies*, 1(1):19–22, 2010.

[14] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, 2015.

[15] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 391–399 DOI: 10.15439/2020F178 ISSN 2300-5963 ACSIS, Vol. 21

StarCraft agent strategic training on a large human versus human game replay dataset

Štefan Krištofík, Peter Malík
Institute of Informatics, Slovak Academy of Sciences
Dúbravská cesta 9, 845 07 Bratislava, Slovakia
Email: stefan.kristofik@savba.sk

Matúš Kasáš, Štefan Neupauer
Faculty of Informatics and Information Technologies
Slovak University of Technology
Ilkovičova 2, 842 16 Bratislava, Slovakia

***Abstract*—Real-time strategy games are currently very popular as a testbed for AI research and education. StarCraft: Brood War (SC:BW) is one of such games. Recently, a new large, unlabeled human versus human SC:BW game replay dataset called STARDATA was published. This paper aims to prove that the player strategy diversity requirement of the dataset is met, i.e., that the diversity of player strategies in STARDATA replays is of sufficient quality. To this end, we built a competitive SC:BW agent from scratch and trained its strategic decision making process on STARDATA. The results show that in the current state of the competitive environment the agent is capable of keeping a stable rating and a decent win rate over a longer period of time. It also performs better than our other, simple rule-based agent. Therefore, we conclude that the strategy diversity requirement of STARDATA is met.**

I. Introduction

REAL-time strategy (RTS) games are considered a very hard challenge for AI today. In an RTS game, players gather resources which they then use to build production facilities and military units with the goal to attack the opponent and destroy all of their structures. Compared to turn-based board games such as Chess or Go, the main challenges in RTS are partial observability of the game state and a huge complexity. RTS AI agents have to tackle the problems of decision making under uncertainty, because the opponent's intents are not always visible and known. A player can only see parts of a map near his armies and buildings, the rest is obscured by the fog-of-war. Players have to actively scout for the opponent's activity or utilize the domain knowledge to make some kind of partially informed decisions. The state space and the number of possible actions at each decision cycle for a player is very large. This makes it impossible to directly apply the techniques used in board games to RTS [3].

Since the 2003 call for research in the field of RTS game AI [1] there have been notable advancements in this area. Multiple approaches were explored and tried in the context of RTS AI. Efficient solutions to the problems posed by the RTS game environment can help not only in the rapidly growing video gaming industry by providing players better, more challenging and more rewarding experience, but also in other AI disciplines and domains of our lives. Weather forecasts, road traffic and self-driving cars, finance, personal assistants, or robotics are some examples of such complex dynamic environments where fast real-time decisions with incomplete information are required from agents [3], [5]. This area has attracted not only individual enthusiasts or researchers, but also teams from some of the big commercial companies like Facebook, Microsoft and Google [4].

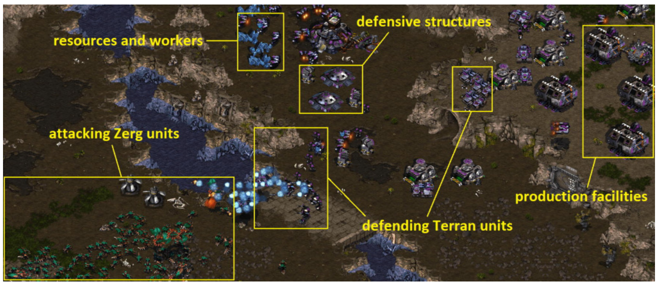


Fig. 1. StarCraft: Brood War

TABLE I
STARCRAFT: BROOD WAR RACES

Race	Characteristics	Advantages and Disadvantages
Zerg	cheap units fast production fast expansion	strong early game hard army micromanagement slow regeneration
Terran	can repair slow expansion can build anywhere	strong defense hard special ability micromanag. needs most space for buildings
Protoss	expensive units slow production cannot repair	strong units with shields strong unit abilities buildings can be disabled

Currently, the most popular RTS game in the context of AI research is StarCraft: Brood War (SC:BW). Fig. 1 shows a screenshot of the game. It is universally praised for a very good balance of all three playable races: Terran, Zerg and Protoss. See Table I for a brief comparison of races. Although the game is now fairly old, released in 1998, both the competitive and research scenes are still very active. Development of SC:BW agents is well supported by the API called BWAPI [1] first introduced in 2009 as well as other useful tools.

It should be noted that the sequel to SC:BW, StarCraft II, has also been gaining popularity in the AI research community

This work is supported by Slovak national project VEGA 2/0155/19.

[1] https://github.com/bwapi/bwapi

since the 2017 release of the game's API for AI research [2]. Although some very interesting results were achieved, e.g., by the Google Deep Mind team [5], the AI is not yet able to compete at a champion level [3].

Similarly to other AI disciplines such as image classification [9] or object detection [4], a number of challenges and tournaments are organized each year to compare the results of SC:BW agents, e.g., "Student StarCraft AI Tournament and Ladder (SSCAIT) [5], "BASIL Ladder" [6], "AIIDE StarCraft AI Competition" [7], "IEEE CoG StarCraft AI Competition" [8]. SC:BW AI agents are currently not yet able to reliably defeat expert level human players and even sometimes struggle against lower tier players as shown in a recent competition [4]. After beating humans in board games such as Chess and Go, overcoming human expert players in a genre of RTS games can be seen as the next goal for SC:BW AI agent research.

Throughout the game's long lifespan, a vast amount of SC:BW game replay data was accumulated and is available for players or AI agents to learn and improve. However, these data are scattered between many sources with various levels of quality. For a dataset to be a viable base for learning models for AI agents, it should be standardized so that it contains only valid replay files from the same game version, ideally containing games of a competitive nature between highly skilled players. Also it should contain replays of all 9 possible race match-ups, wide variety of competitive terrain maps, strategies and game lengths. An efficient usage of the replay data knowledge for training AI agents is still an open problem. Recently, a new large SC:BW replay dataset called STARDATA containing 65646 unlabeled human versus human replays was published by the Facebook research team [2]. The authors kept in mind many of the above mentioned requirements so this dataset is the first to be of sufficient size and quality for the task of AI agent training. It is the largest dataset of its kind to date. Prior work on compiling similar datasets resulted in much smaller sets of up to 7649 replays [2], [6], [7], [8]. Our goal in this work is to validate the strategy diversity requirement of STARDATA, which was not done yet. Also, to our knowledge, there were no attempts to use this dataset directly for learning.

The main contributions of this work are as follows. We validate the strategy diversity requirement of the STARDATA dataset containing SC:BW game replays. To this end, we build a competitive SC:BW Terran agent from scratch and train its strategic decision making process using solely the data extracted from the dataset. We test the agent's prowess in a continuous competitive environment (SSCAIT tournament) against other agents. The agent's win rate is evaluated. Also the ability to choose and execute a good counter-strategy to beat the opponent is inspected. Results show that the agent was able to maintain a stable competitive rating over a period of 4 months. This proves that an agent trained using strategy variety offered by STARDATA can be a strong contestant in the current state of the competition which in turn proves the strategy diversity requirement of the dataset is met. The results also show the Terran agent is performing better than our other, simple rule-based Zerg agent built from scratch trained using knowledge gained from a small machine versus machine replay dataset.

II. Related Work

Numerous attempts at strategy extraction or prediction from SC:BW replay datasets were conducted. In [6], the authors extracted 6 strategies for each race from a dataset containing 5493 replays. However, these strategies are not specifically targeted for different match-ups and are considered for all match-ups. Also, the extracted strategies are only considering low level units and only short games of up to 10 minutes. For strategy prediction, several machine learning algorithms were compared with promising results. In [7], the authors expand upon [6] by adding 570 new replays to the dataset. They also try to account for the fog-of-war information, i.e., the information about the areas that are currently not visible to the players. According to the results, the prediction model is not good for early game, but has better results for middle game predictions. In [8], the authors present a slightly larger dataset containing 7649 replays. Although some interesting statistical information was extracted from the dataset, the focus is rather on tactical aspects of the game, such as individual battle outcome prediction or battle detection.

In terms of dataset volume, STARDATA [2] is the largest one yet available for the research community. According to the results, it meets many requirements for a good base for learning models, such as match-up, map or game length diversity. However, the strategy diversity requirement is not addressed in detail. An attempt is also made at strategy extraction, but only few Protoss strategies were considered. Authors list several tasks that can be tackled by using their dataset. This work focuses on one of the tasks: strategy classification.

III. STARDATA training

A typical competitive SC:BW agent has 3 main modules:

- Macromanagement: involves long-term goals such as strategy, build order, unit compositions (see III-C2 for details) or expansion.
- Micromanagement: involves short-term goals such as combat effectiveness, unit positioning, targeting and behavior; also building placement.
- Scouting: involves gathering information about the opponent. It is often very important for the macromanagement module's decisions.

The quality of the agent depends on its ability to scout and identify the opponent's strategy and then to choose and effectively execute the proper counter-strategy to defeat them.

[2] https://news.blizzard.com/en-us/starcraft2/20944009/the-starcraft-ii-api-has-arrived

[3] https://www.nature.com/articles/d41586-019-03298-6

[4] https://www.kaggle.com/c/global-wheat-detection

[5] https://sscaitournament.com

[6] https://basil.bytekeeper.org

[7] http://www.cs.mun.ca/~dchurchill/starcraftaicomp/index.shtml

[8] https://cilab.gist.ac.kr/sc_competition

TABLE II
SC:BW STRATEGIES AND MATCH-UPS

Race	Terran	Protoss	Zerg	**In this work**
Terran	T (TvT)	TP (TvP)	TZ (TvZ)	yes
Protoss	PT (PvT)	P (PvP)	PZ (PvZ)	no
Zerg	ZT (ZvT)	ZP (ZvP)	Z (ZvZ)	

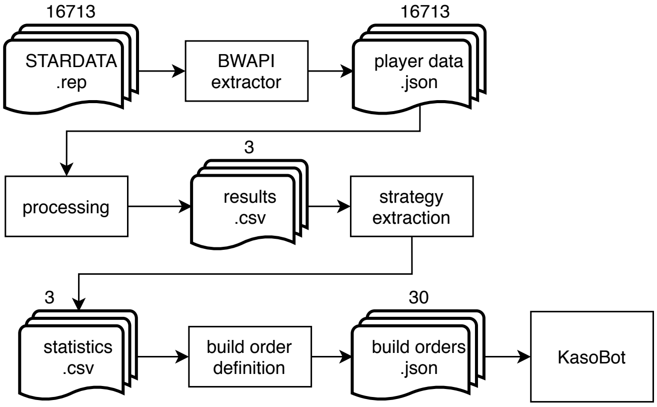


Fig. 2. Agent learning process

We built a competitive SC:BW Terran agent, nicknamed KasoBot, from scratch with the goal to prepare its strategic reasoning module before the game [9] by learning from unlabeled replay data from STARDATA [2]. We evaluate the agent from the strategic perspective, i.e., if it is able to select and execute the proper strategy in each game. We do not focus on the micromanagement tasks and only tackle this area partially (see III-F for details). Main goals of this work can be summarized as follows:

- Identify and categorize the most common player strategies from a subset of the STARDATA dataset.
- Label the strategies used by both players in this subset.
- Compare the strategies against each other in terms of win rate percentages.
- Build a competitive SC:BW agent able to execute the identified (extracted) strategies and, during a game, select the ones with a highest chance of winning based on the results of the comparison.
- Validate the strategy diversity requirement of STARDATA by evaluating the agent's results in a competitive environment.

Table II shows the strategy labels for each match-up which will be used in the remainder of this paper. In this work, we describe the extracted Terran strategies T, TP and TZ in TvT, TvP and TvZ match-ups, respectively. Our agent's learning process is shown in Fig. 2. The following sections will describe the individual steps.

TABLE III
DATASET

Replays	**Amount**
STARDATA Total	65646 [2]
Valid (BWAPI compatible)	65645
and competitive (2 players)	64550
and involving Terran	33741
and of length <15 minutes	**16713**
Versus Terran (TvT)	1468
Versus Protoss (TvP)	5014
Versus Zerg (TvZ)	10231

A. Dataset

We use STARDATA [2] for strategic learning. Several filters were applied before learning (Table III). Only replays that were executable in BWAPI, had exactly 2 active players, involved the Terran race on either side and had game length less than 15 minutes were considered. 16713 replays have passed these filters and were used for learning. The most numerous match-up was TvZ and the least numerous was the mirror match-up.

The length threshold was chosen because the diversity of late game strategies, i.e., strategies used in longer games, is worse than in early game. In other words, one can find less varied unit compositions used by players when inspecting later game stages than in early or middle stages. This is a consequence of the tech trees used in SC:BW. A player is required to first unlock the ability to build stronger, more expensive units, which takes some time. Therefore, these units typically appear in game only after a certain time has elapsed. We only want to learn early and middle game strategies. Late game strategies are easier to learn by manual inspection of some long games and analyzing the tech trees.

B. Raw data extraction

First, we extract player information from the original STARDATA replay files *(*.rep)*. For this purpose, we have implemented our own replay data extractor. This is a common practice in similar works [8], [2] due to the proprietary format of the original replay files that is hard to use directly. Each replay file is processed by running it in the game engine launched through the BWAPI interface together with our extractor. The following information is extracted from each replay file and stored in a separate *json* file:

- Basic information about the game: map name, length [10], player names and races, information needed to determine the winner (see III-C1 for details).
- For each building type used: name, ID [11], #built, #destroyed.
- For each building instance: timestamp [12] and current unit

[9] Other possible approaches are learning during the game or after the game [3]

[10] Measured in game frames. Competitive SC:BW games are played at 23.81 frames per second.

[11] BWAPI unique ID

[12] In frames

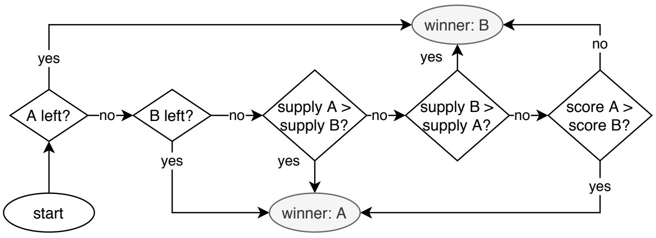


Fig. 3. Determining a winner of a SC:BW game. A, B - opposing players.

supply value [13] when it was built or destroyed.

- For each unit type used: name, ID, timestamp and current unit supply value when the first unit of that type was created, #built, #killed.
- For each finished tech or upgrade: name, ID, timestamp and current unit supply value when finished.

The above information is too generic and has to be processed further for strategy extraction.

C. Data processing

Next step is to process raw extracted player data into a more useful form as feature vectors. The results are stored in 3 *csv* files, one for each match-up that we are interested in: TvT, TvP and TvZ. The information about each TvP game is encoded on 1 row of the corresponding file. Same is true for TvZ games. However, the information about each TvT game is encoded on 2 rows (1 row from the perspective of each of the two players). After processing, the following information is available for each game:

- Basic information: similar to raw data (see III-B), but now also including the index of a winning player.
- For each building type used: timestamp and current supply value when the first instance of this type was constructed, maximum count of existing instances during the game.
- For each unit type used: #built.
- For each finished tech or upgrade: timestamp and current unit supply value when finished.
- For each of the selected building features from Table IV: building order (see III-C2 for details).
- For each of the selected unit features from Table IV: unit frequency (see III-C2 for details).

1) Determining a winner: To evaluate strategy win rates, it is necessary to determine the winning player in each game. This information is not explicitly available from the original replay files and must be gathered manually during processing. This process is illustrated in Fig. 3. If for any reason a player leaves the game, this action may be captured in the replay file as a player command and extracted. If such command is found, we always consider their opponent as a winner. Otherwise we consider the player with the largest unit supply value at the end of the game as a winner. This value is commonly used for this purpose [2]. If both players have the same unit supply values, we consider the player with the highest score as a winner. Score is an internal metric of SC:BW based on various player actions and achievements throughout the game.

2) Build orders and army compositions: SC:BW strategies are defined mainly by build orders and army compositions.

Build order refers to a specific sequence of building construction. As mentioned earlier, SC:BW uses tech trees. For example, to be able to create Marines, the player needs to construct Barracks, but Medics require both Barracks and Academy. Moreover, Academy can only be constructed if Barracks are already present. Therefore, the build order required to create Medics is: Barracks $\rightarrow$ Academy.

Army composition refers to a list of unit types which form the backbone of the player's army. In other words, army composition is defined by the most used unit types. Other, less frequently used unit types, should only be considered as support units for the main army composition. For example, one of the more popular Terran army compositions against Zerg opponents is Marines with Medics as the backbone with the support of few Siege Tanks and Science Vessels [14].

Our goal is to identify and categorize the most used strategies from STARDATA. Since our agent is Terran, we are interested in Terran strategies against all three races and Protoss and Zerg strategies against Terran. We have selected a set of the most important features for each race which will be used to distinguish between various strategies. The complete list is shown in Table IV. Other features not included in the table were deemed not as important for strategy characterization. We did not include mandatory buildings or units, i.e., used in every game because they are required to make any progress in the game whatsoever (e.g., Terran Barracks, Protoss Gateway, worker units). We also did not include some unpopular units which are built only very rarely (e.g., Protoss Scout, Zerg Devourer). We also did not include defensive structures, special non-combat units (e.g., Zerg Larva) and some late-game structures as well since we are interested in only early and middle game strategies.

To encode a build order into a feature vector, we assign each of the selected building features a number based on the order in which it was first constructed during a game. Buildings which were not built in a game are assigned $number\_of\_building\_features + 1$. In case of Terran, it is the value 9, since there are 8 building features. An example of a Terran build order is shown in Table V. The player has built Factory as first, followed by Machine Shop and has not built any Control Towers, Engineering Bays or Starports.

Based on the relative frequency of unit creation during a game, we assign each unit feature from Table IV a number representing how many units of this type were created relative to other types. We define the unit creation frequency as the number of units created per minute (1429 frames) starting since the frame when all the requirements for it were first satisfied during a game. Units created most frequently are

[13] Represents the current size of the player's army.

[14] E.g., https://www.youtube.com/watch?v=qyixL9J7-B8 at around 10 minute mark.

TABLE IV
SELECTED STRATEGY DEFINING FEATURES

Race	Buildings	Units
Terran	Academy Armory Command Center Control Tower Engineering Bay Factory Machine Shop Starport	Firebat Goliath Marine Medic Siege Tank Vulture Wraith
Protoss	Forge Nexus Robotics Facility Stargate Templar Archives	Archon Carrier Dark Templar Dragoon High Templar Zealot
Zerg	Hive Hydralisk Den Lair Spire	Hydralisk Lurker Mutalisk Scourge Ultralisk Zergling

TABLE V
EXAMPLE OF A TERRAN BUILD ORDER

Building	Build order	Building	Build order
Academy	5	Engineering Bay	9
Armory	4	Factory	1
Command Center	3	Machine Shop	2
Control Tower	9	Starport	9

assigned smaller numbers. Units which were not created in a game are assigned $number\_of\_unit\_features + 1$. In case of Terran, it is the value 8, since there are 7 unit features. This way a feature vector of unit frequencies is formed. An example of a Terran unit frequency is shown in Table VI. The player has built many Siege Tanks and Vultures and has not built any Goliaths, Medics or Firebats.

D. Strategy extraction

We extract Terran strategies (from both player perspectives) from TvT, both Terran and Protoss strategies from TvP and both Terran and Zerg strategies from TvZ. Terran strategies against different races may be very different, so we treat each match-up separately.

We treat the STARDATA replays as unlabeled data because the particular strategies used by both opponents are not known.

TABLE VI
EXAMPLE OF A TERRAN UNIT FREQUENCY ANALYSIS

Unit	Frequency rating	Unit	Frequency rating
Marine	4	Wraith	3
Vulture	2	Medic	8
Goliath	8	Firebat	8
Siege Tank	1	-	-

To search and find regularities in unlabeled data, we have chosen a cluster analysis method. In particular, we used the K-Means clustering algorithm for strategy extraction. This algorithm requires to know the desired number of clusters beforehand. Each cluster represents one distinct strategy. After few experiments with the number of clusters ranging from 6 up to 20, it was selected to be 10 for each match-up. Using this number of clusters guaranteed their sufficient diversity as well as the sufficient number of replays in each cluster. The algorithm produces clusters of different sizes. This is beneficial because popularity of strategies can vary. Therefore, more popular strategies will be represented by larger clusters than the less popular ones.

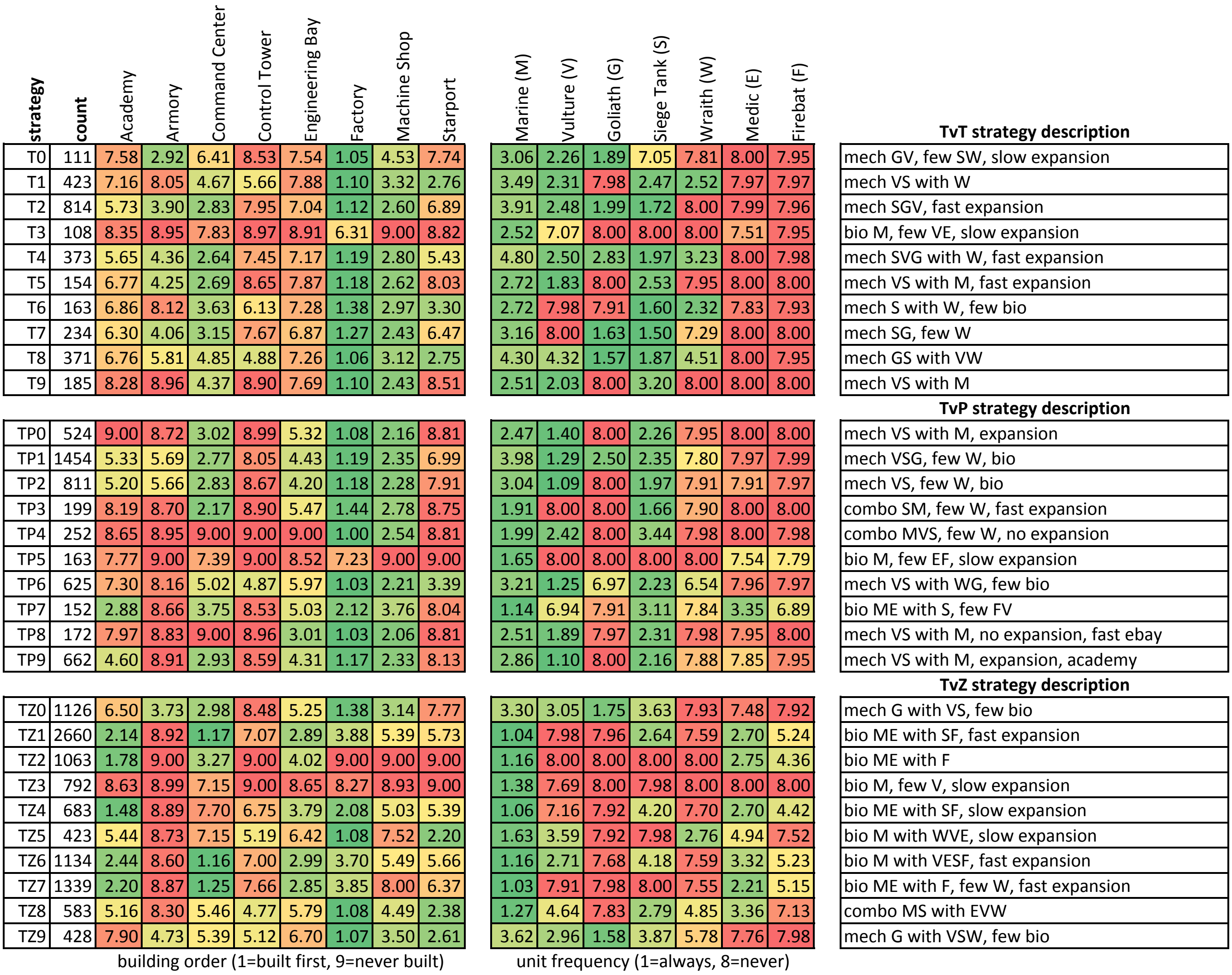
This resulted in a successful extraction of a total of 30 different Terran strategies, 10 for TvT, 10 for TvP and 10 for TvZ. The information is stored in 3 *csv* files, one for each match-up. Moreover, 10 Protoss strategies for PvT and 10 Zerg strategies for ZvT were also extracted. However, as can be seen from Table II, this work does not provide details on them and only focuses on Terran strategies.

E. Results

The summary of extracted Terran strategies against all three races is shown in Fig. 4. Clusters representing strategies for each match-up produced by K-Means were initially labeled by the algorithm as 0-9. We assigned more descriptive labels to strategies to clearly indicate the relevant match-up (see also Table II), e.g., cluster 4 from TvZ was assigned label TZ4. The table shows average building orders and average unit frequencies with respect to corresponding clusters. This means that, for example, strategy T6 can be characterized by the building order starting very often with Factory, then usually following with Machine Shop or Starport, and very often ending with Armory. This strategy is also characterized by the unit composition containing Siege Tanks very often, Wraiths and Marines often, too, and other units only very rarely. Strategy descriptions use abbreviated unit names (e.g., G=Goliath). The term *expansion* means the construction of Command Center, effectively establishing another base to boost the economy. The term *mech* refers to mechanical units (Vulture, Goliath, Siege Tank and Wraith), *bio* refers to biological units (Marine, Medic and Firebat) and *combo* refers to a combination of these. From the results, the following important conclusions can be drawn:

- Two main Terran army composition types are prevalent: bio-based and mech-based. The combination of both is rarely used.
- Bio units are often used against Zerg, but rarely against Terran or Protoss.
- Mech units are often used against Terran and Protoss, but not often against Zerg.
- Combo unit composition is rarely used against Protoss and Zerg and almost never against Terran.

The extracted strategies seem to offer a good variety of build orders and unit compositions overall. In some cases, the differences between particular strategies are negligible in unit compositions, but significant in build order (e.g., compare



strategy	count	Academy	Armory	Command Center	Control Tower	Engineering Bay	Factory	Machine Shop	Starport	Marine (M)	Vulture (V)	Goliath (G)	Siege Tank (S)	Wraith (W)	Medic (E)	Firebat (F)	TvT strategy description
T0	111	7.58	2.92	6.41	8.53	7.54	1.05	4.53	7.74	3.06	2.26	1.89	7.05	7.81	8.00	7.95	mech GV, few SW, slow expansion
T1	423	7.16	8.05	4.67	5.66	7.88	1.10	3.32	2.76	3.49	2.31	7.98	2.47	2.52	7.97	7.97	mech VS with W
T2	814	5.73	3.90	2.83	7.95	7.04	1.12	2.60	6.89	3.91	2.48	1.99	1.72	8.00	7.99	7.96	mech SGV, fast expansion
T3	108	8.35	8.95	7.83	8.97	8.91	6.31	9.00	8.82	2.52	7.07	8.00	8.00	8.00	7.51	7.95	bio M, few VE, slow expansion
T4	373	5.65	4.36	2.64	7.45	7.17	1.19	2.80	5.43	4.80	2.50	2.83	1.97	3.23	8.00	7.98	mech SVG with W, fast expansion
T5	154	6.77	4.25	2.69	8.65	7.87	1.18	2.62	8.03	2.72	1.83	8.00	2.53	7.95	8.00	8.00	mech VS with M, fast expansion
T6	163	6.86	8.12	3.63	6.13	7.28	1.38	2.97	3.30	2.72	7.98	7.91	1.60	2.32	7.83	7.93	mech S with W, few bio
T7	234	6.30	4.06	3.15	7.67	6.87	1.27	2.43	6.47	3.16	8.00	1.63	1.50	7.29	8.00	8.00	mech SG, few W
T8	371	6.76	5.81	4.85	4.88	7.26	1.06	3.12	2.75	4.30	4.32	1.57	1.87	4.51	8.00	7.95	mech GS with VW
T9	185	8.28	8.96	4.37	8.90	7.69	1.10	2.43	8.51	2.51	2.03	8.00	3.20	8.00	8.00	8.00	mech VS with M
																	TvP strategy description
TP0	524	9.00	8.72	3.02	8.99	5.32	1.08	2.16	8.81	2.47	1.40	8.00	2.26	7.95	8.00	8.00	mech VS with M, expansion
TP1	1454	5.33	5.69	2.77	8.05	4.43	1.19	2.35	6.99	3.98	1.29	2.50	2.35	7.80	7.97	7.99	mech VSG, few W, bio
TP2	811	5.20	5.66	2.83	8.67	4.20	1.18	2.28	7.91	3.04	1.09	8.00	1.97	7.91	7.91	7.97	mech VS, few W, bio
TP3	199	8.19	8.70	2.17	8.90	5.47	1.44	2.78	8.75	1.91	8.00	8.00	1.66	7.90	8.00	8.00	combo SM, few W, fast expansion
TP4	252	8.65	8.95	9.00	9.00	9.00	1.00	2.54	8.81	1.99	2.42	8.00	3.44	7.98	8.00	7.98	combo MVS, few W, no expansion
TP5	163	7.77	9.00	7.39	9.00	8.52	7.23	9.00	9.00	1.65	8.00	8.00	8.00	8.00	7.54	7.79	bio M, few EF, slow expansion
TP6	625	7.30	8.16	5.02	4.87	5.97	1.03	2.21	3.39	3.21	1.25	6.97	2.23	6.54	7.96	7.97	mech VS with WG, few bio
TP7	152	2.88	8.66	3.75	8.53	5.03	2.12	3.76	8.04	1.14	6.94	7.91	3.11	7.84	3.35	6.89	bio ME with S, few FV
TP8	172	7.97	8.83	9.00	8.96	3.01	1.03	2.06	8.81	2.51	1.89	7.97	2.31	7.98	7.95	8.00	mech VS with M, no expansion, fast ebay
TP9	662	4.60	8.91	2.93	8.59	4.31	1.17	2.33	8.13	2.86	1.10	8.00	2.16	7.88	7.85	7.95	mech VS with M, expansion, academy
																	TvZ strategy description
TZ0	1126	6.50	3.73	2.98	8.48	5.25	1.38	3.14	7.77	3.30	3.05	1.75	3.63	7.93	7.48	7.92	mech G with VS, few bio
TZ1	2660	2.14	8.92	1.17	7.07	2.89	3.88	5.39	5.73	1.04	7.98	7.96	2.64	7.59	2.70	5.24	bio ME with SF, fast expansion
TZ2	1063	1.78	9.00	3.27	9.00	4.02	9.00	9.00	9.00	1.16	8.00	8.00	8.00	8.00	2.75	4.36	bio ME with F
TZ3	792	8.63	8.99	7.15	9.00	8.65	8.27	8.93	9.00	1.38	7.69	8.00	7.98	8.00	8.00	8.00	bio M, few V, slow expansion
TZ4	683	1.48	8.89	7.70	6.75	3.79	2.08	5.03	5.39	1.06	7.16	7.92	4.20	7.70	2.70	4.42	bio ME with SF, slow expansion
TZ5	423	5.44	8.73	7.15	5.19	6.42	1.08	7.52	2.20	1.63	3.59	7.92	7.98	2.76	4.94	7.52	bio M with WVE, slow expansion
TZ6	1134	2.44	8.60	1.16	7.00	2.99	3.70	5.49	5.66	1.16	2.71	7.68	4.18	7.59	3.32	5.23	bio M with VESF, fast expansion
TZ7	1339	2.20	8.87	1.25	7.66	2.85	3.85	8.00	6.37	1.03	7.91	7.98	8.00	7.55	2.21	5.15	bio ME with F, few W, fast expansion
TZ8	583	5.16	8.30	5.46	4.77	5.79	1.08	4.49	2.38	1.27	4.64	7.83	2.79	4.85	3.36	7.13	combo MS with EVW
TZ9	428	7.90	4.73	5.39	5.12	6.70	1.07	3.50	2.61	3.62	2.96	1.58	3.87	5.78	7.76	7.98	mech G with VSW, few bio

building order (1=built first, 9=never built)

unit frequency (1=always, 8=never)

Fig. 4. Extracted Terran strategies: versus Terran (top), versus Protoss (middle), versus Zerg (bottom) with average building orders (left), average unit frequencies (middle) and verbose descriptions (right); building order: 1=built as first, 9=never built; unit frequency: 1=always created, 8=never created

strategies TP0 and TP8). Also, the resulting sizes of clusters seem to be pretty varied, too, each containing a decent sample of at least 100 occurrences in match-ups. Strategy distributions in each match-up are shown in Fig. 5. For each match-up, there appears to exist one favorite strategy with a large margin before other strategies, e.g., TZ1 for TvZ.

Win rates of strategies from Fig. 4 are compared against each other in TvT in Fig. 6, against Protoss PvT strategies in Fig. 7 and against Zerg ZvT strategies in Fig. 8. The numbers of match-ups containing the exact pair of strategies is shown on the right sides of Figs. 6-8. For example, if in a TvP match-up the Protoss opponent is following strategy PT8, the best course of action for the Terran player is to choose strategy TP2, because it has the highest win rate against that enemy strategy (82.31 %). The number of match-ups involving these exact strategies was 147. The second best option would be to choose TP9 with a 77.62 % win rate. The diagonal of the TvT table is always 50 % because both players chose the same strategy and only one won. Note that some strategy match-ups never occurred (e.g., TZ3 versus ZT0). Win rates are not available in those cases. This learned data will be helpful for the agent strategic decision making during games. See III-F for details.

We analyze strategies further in Table VII. According to the results, in both TvT and TvP match-ups, the most played Terran strategy is not the best one (using weighted averages across all games where the strategy was involved). In the TvZ match-up, the most played Terran strategy is also the most successful one.

F. Agent function

All 30 extracted Terran strategies (see Fig. 4) are transformed into build orders and described in 30 *json* files, one for each strategy. The agent, KasoBot, is able to emulate all 30 strategies by reading the contents of these *json* files. At

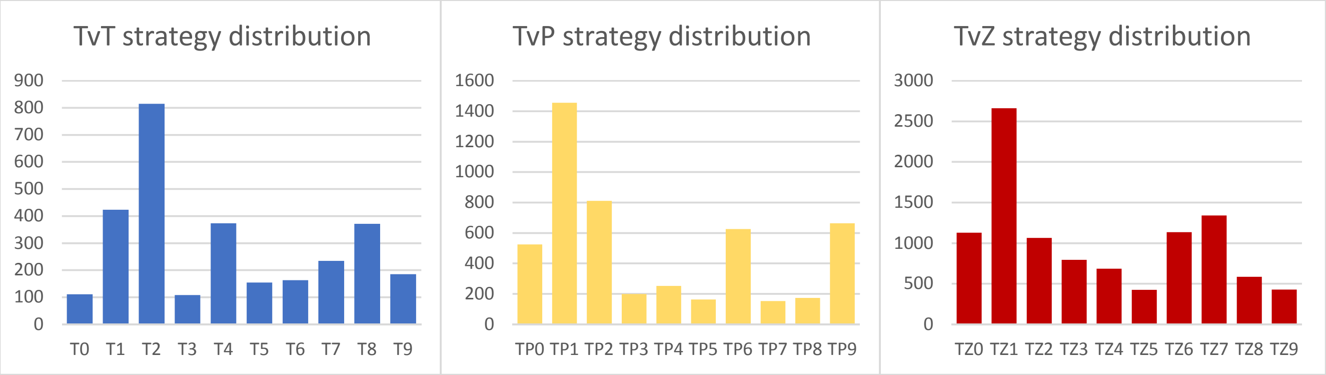


Fig. 5. Extracted strategy distributions: number of games the strategy has occurred in

Terran player B strategy

Terran player A strategy	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
T0	50.00	53.49	28.57	75.00	0.00	42.86	55.56	0.00	66.67	56.25
T1	46.51	50.00	42.41	100.00	46.34	75.00	68.42	43.24	50.00	80.00
T2	71.43	57.59	50.00	100.00	48.31	47.62	38.46	51.61	48.74	53.57
T3	25.00	0.00	0.00	50.00	N/A	100.00	50.00	N/A	0.00	0.00
T4	100.00	53.66	51.69	N/A	50.00	71.43	87.50	47.06	58.33	100.00
T5	57.14	25.00	52.38	0.00	28.57	50.00	36.36	75.00	N/A	61.54
T6	44.44	31.58	61.54	50.00	12.50	63.64	50.00	53.57	53.33	83.33
T7	100.00	56.76	48.39	N/A	52.94	25.00	46.43	50.00	57.50	77.78
T8	33.33	50.00	51.26	100.00	41.67	N/A	46.67	42.50	50.00	100.00
T9	43.75	20.00	46.43	100.00	0.00	38.46	16.67	22.22	0.00	50.00

Terran player B strategy

Terran player A strategy	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
T0	20	43	7	4	1	7	9	1	3	16
T1	43	38	158	5	41	20	19	37	42	20
T2	7	158	240	1	118	42	39	62	119	28
T3	4	5	1	88	0	1	2	0	1	6
T4	1	41	118	0	106	7	8	17	72	3
T5	7	20	42	1	7	36	11	4	0	26
T6	9	19	39	2	8	11	26	28	15	6
T7	1	37	62	0	17	4	28	36	40	9
T8	3	42	119	1	72	0	15	40	78	1
T9	16	20	28	6	3	26	6	9	1	70

Fig. 6. TvT: strategy win rates (left), strategy match-ups (right)

Protoss player strategy

Terran player strategy	PT0	PT1	PT2	PT3	PT4	PT5	PT6	PT7	PT8	PT9
TP0	30.77	59.34	30.36	50.00	29.73	14.29	33.33	0.00	47.20	19.05
TP1	63.38	61.54	56.84	60.00	53.72	80.00	53.57	N/A	70.77	50.35
TP2	56.72	58.82	46.25	87.50	67.57	55.56	37.63	N/A	82.31	41.57
TP3	33.33	11.43	0.00	7.14	11.11	23.08	0.00	100.00	19.44	0.00
TP4	0.00	43.94	25.00	34.33	25.00	57.14	N/A	52.38	33.82	0.00
TP5	N/A	29.63	N/A	23.53	33.33	40.00	50.00	34.33	20.00	N/A
TP6	54.55	47.83	58.00	60.71	60.00	50.00	47.83	0.00	67.48	38.18
TP7	72.73	50.00	66.67	27.27	61.11	50.00	32.43	0.00	64.52	50.00
TP8	33.33	52.94	33.33	54.17	46.15	55.56	50.00	100.00	54.00	0.00
TP9	60.19	80.00	46.67	78.57	58.78	54.29	25.53	N/A	77.62	36.73

Protoss player strategy

Terran player strategy	PT0	PT1	PT2	PT3	PT4	PT5	PT6	PT7	PT8	PT9
TP0	26	91	56	46	37	28	3	2	214	21
TP1	284	13	285	5	121	5	392	0	65	284
TP2	134	17	240	8	74	9	93	0	147	89
TP3	3	70	5	28	27	26	1	1	36	2
TP4	3	66	4	67	12	7	0	21	68	4
TP5	0	27	0	51	6	5	2	67	5	0
TP6	77	46	150	28	75	24	46	1	123	55
TP7	22	10	12	11	18	6	37	3	31	2
TP8	9	34	12	24	26	9	2	2	50	4
TP9	108	30	105	14	131	35	47	0	143	49

Fig. 7. TvP: strategy win rates (left), strategy match-ups (right)

Zerg player strategy

Terran player strategy	ZT0	ZT1	ZT2	ZT3	ZT4	ZT5	ZT6	ZT7	ZT8	ZT9
TZ0	72.22	67.50	50.00	65.43	48.14	50.00	31.71	37.04	51.04	65.00
TZ1	63.65	79.17	100.00	67.35	70.27	55.84	75.00	28.06	71.43	100.00
TZ2	14.29	20.00	74.07	25.00	37.74	7.81	29.72	0.00	63.04	64.14
TZ3	N/A	0.00	47.15	16.67	N/A	0.00	10.53	N/A	0.00	28.04
TZ4	47.45	61.11	82.35	50.00	46.88	34.02	46.03	13.33	82.76	79.31
TZ5	46.15	54.43	81.48	45.76	31.58	28.57	25.00	0.00	62.92	78.26
TZ6	60.81	44.44	100.00	60.00	50.00	65.00	40.00	21.71	58.33	50.00
TZ7	64.71	75.65	75.00	32.09	62.50	41.70	61.82	44.05	75.00	81.08
TZ8	59.76	56.25	100.00	62.35	55.36	50.63	50.00	25.93	57.97	60.00
TZ9	52.63	38.24	100.00	48.28	48.95	60.00	26.92	25.00	58.00	100.00

Zerg player strategy

Terran player strategy	ZT0	ZT1	ZT2	ZT3	ZT4	ZT5	ZT6	ZT7	ZT8	ZT9
TZ0	36	40	8	81	538	116	164	27	96	20
TZ1	1029	72	4	441	74	539	124	335	35	7
TZ2	7	10	81	92	53	64	471	2	46	237
TZ3	0	2	649	6	0	1	19	0	8	107
TZ4	137	36	17	150	32	97	126	30	29	29
TZ5	39	79	27	59	38	21	24	1	89	46
TZ6	518	9	1	105	12	100	25	350	12	2
TZ7	136	115	4	134	136	259	406	84	28	37
TZ8	164	16	1	162	56	79	4	27	69	5
TZ9	19	68	1	29	190	35	26	8	50	2

Fig. 8. TvZ: strategy win rates (left), strategy match-ups (right)

TABLE VII
STRATEGY ANALYSIS

Match-up	Most used	Average win rate	Highest avg. win rate
TvT	T2	50.86 %	T4 (54.16 %)
TvP	TP1	56.46 %	TP9 (58.31 %)
TvZ	TZ1	59.59 %	TZ1

the start of a game, it randomly selects one of the 3 best performing strategies to follow against the opponent, using data from Figs 6-8. Opponent's race is known prior to the game. It proceeds to create buildings according to the extracted average build orders with the goal to reach the corresponding target unit composition.

The agent sets its highest priority to complete the required buildings from the selected build order. Unit production has lower priority and is only launched when the required buildings are ready. Once the build order is completed, the agent starts to produce units according to the selected unit composition. If existing production facilities are busy and there are spare resources available, the agent adds more production facilities to speed up the unit production. During a game, it periodically checks the availability of all buildings from the build order and tries to reconstruct them if any were destroyed.

Once enough military units are produced, the agent starts to form individual unit squads with different tasks, including scouting (small squads), defending important positions (mainly expansions), or attacking multiple revealed opponent positions (mainly structures) simultaneously. If the opponent army is confronted, the units follow simple combat behavior (attack closest enemy), with some exceptions (e.g., Vulture, which employs hit-and-run tactics).

During a game, the agent is also able to switch between similar strategies and adjust its army composition slightly. For example, if using strategy TP2 and the enemy starts to create air units, the agent will add Goliaths (strong anti-air units) to its composition, effectively switching to a strategy very similar to TP1.

If a game progresses to the late stage (15 minutes), the agent lowers the priority for the current strategy and sets the highest priority to a special late game strategy, which was manually constructed specifically for late game scenarios. The agent will modify its army composition by constructing late game buildings and units not included in extracted strategies, e.g., Battlecruisers (overall strong air unit) or Valkyries (strong air-to-air support unit). It will also focus more on the weapon and armor upgrades, making units more powerful and tough. This late game strategy is a result of manual inspection of several longer games, where the Terran players' strategies seemed to converge towards one particular late game strong army composition.

Although the extracted strategies and unit compositions serve as a source for strategic decisions, the game of SC:BW encompasses many other tasks that are required to beat the opponent. These tasks include: producing enough worker units to keep the economy afloat, supporting unit production; producing defensive structures; choosing important areas of the map to scout, defend and attack; producing enough maximum unit supply increasing buildings to keep the army at the maximum possible size and strength; creating special defensive building formations to prevent the opponent access to certain areas. All these tasks are performed by the agent simultaneously with the strategy component described above.

TABLE VIII
STRATEGIES IN A RULE-BASED AGENT

Strategy	Brief description
LateGame	priority: Mutalisks, secondary: Hydralisks, Zerglings standard strategy used when it is late game
Mutalisk	priority: Mutalisks, secondary: Zerglings switch to LateGame if failed
Hydralisk	priority: Hydralisks, secondary: Zerglings switch to LateGame if failed
ZerglingRush	priority: Zerglings switch to Hydralisk or Mutalisk if failed

After the manual inspection of multiple games, we conclude that the agent is able to select a good starting strategy and switch between similar strategies, as mentioned above. However, it is lacking in combat scenarios against superior opponents, e.g., unit behavior in combat is very simple compared to other advanced agents. As mentioned earlier, we did not focus on the micromanagement tasks as much. Moreover, the agent is not very good at scouting in early game and instead relies heavily on the extracted statistics. These aspects could be improved as future work.

IV. RULE-BASED TRAINING

We manually analyzed 112 machine versus machine games between the top 16 contestants of the final tournament in the SSCAIT 2018/19 edition. Based on the results, we prepared the strategic reasoning module for the Zerg agent, nicknamed NuiBot and built from scratch, by defining a number of simple rules and defining a number of own and opponent strategies.

The agent has a total of 12 different strategies and is able to switch between them during a game if needed. It also has a total of 13 different opponent models and is able to assign a different model to an opponent during a game if updated information is available. The agent tries to actively scout the opponent during early and mid game and update the opponent model as frequently as possible. Some of the agent's strategies are listed in Table VIII and some of the opponent models are listed in Table IX.

Apart from the strategic decisions controlled by the defined rules, NuiBot also performs a number of additional tasks, similar to KasoBot. These tasks involve scouting using workers in early game, simple army movements, attacking enemy positions and so on. However, it can not create unit squads and attacks with large groups. Unit behavior in combat is also very simple, similar to KasoBot.

Additionally, the agent continually accumulates the information about each individual opponent it has met previously in a game. In particular, it stores statistical data about models

TABLE X
AGENT RESULTS IN SSCAIT TOURNAMENT AS OF JULY, 1ST 2020; ACCUMULATED OVER 4 MONTHS

Agent	Description	ELO rating	SSCAIT rank	Overall win rate	Last 50 games
KasoBot (Terran)	Trained on STARDATA	2004	D	27 %	48 %
NuiBot (Zerg)	rule-based	1915	E	29 %	32 %

TABLE IX
OPPONENT MODELS IN A RULE-BASED AGENT

Opponent model	Brief description
fastExpand	opponent has expansion early
CannonRush	opponent is Protoss has Forge, but not Gateway early
ZerglingRush	opponent is Zerg has Spawning Pool, but only 1 Hatchery early
Dark Templar	opponent is Protoss has Citadel of Adun early
hardDefense	opponent has high number of defensive structures
massFlights	opponent has more than 3 air units
Normal	default model used until switched to another model

assigned to opponents during games. It updates these statistics with each played game. This helps to set a correct strategy next time when it faces the same opponent, i.e., it makes assigning a model to an opponent easier during subsequent games.

V. EXPERIMENTS

Both agents, KasoBot and NuiBot, were placed into SSCAIT Ladder at the start of March 2020. In this competition format, the opponents are picked randomly from a roster of all other active contestants. The authors can upload new versions of agents anytime. Over the course of 4 months, the results were accumulated and are summarized in Table X and are valid as of July, 1st 2020 (https://sscaitournament.com/index.php?action=scores).

KasoBot was able to maintain a relatively stable average ELO rating of 2004 (current contestant ratings range from 1224 up to 2917) with a decent 27 % overall win rate and a good 48 % current win rate (from last 50 games) with a slightly below average 'D' SSCAIT competitive rank (current contestant rankings range from 'B' to 'E'). ELO rating is a common metric used in many games, e.g., chess. SSCAIT rank is an internal SSCAIT system used to qualitatively compare agents. In comparison, NuiBot achieved a lower ELO rating of 1915, 29 % overall win rate and 32 % current win rate. The higher overall win rate of the rule-based agent is caused by the known phenomenon where rule-based agents tend to have good win rates when freshly entering a competition, but are slowly surpassed over time by more advanced agents, as recently documented for example in the AIIDE 2017 competition [4]. The overall win rate of KasoBot is expected to raise above NuiBot's if its current win rate stays at the present level. We conclude that the STARDATA trained agent surpassed the simple rule-based agent in all important statistics of the competition.

VI. CONCLUSION

We evaluate the strategy diversity requirement of the recently published large human versus human SC:BW replay dataset called STARDATA. We show that a competitive SC:BW agent built from scratch, with its strategic decision making module trained solely on the unlabeled replay data from STARDATA, can be a strong contestant among other agents in a competitive environment. The dataset offers a good variety of player strategies and the agent was able to learn broad amount of domain knowledge from the replays alone. Therefore, we conclude that the diversity requirement of STARDATA is met. It is encouraged to use this dataset for further work in the areas and tasks outlined by the authors.

REFERENCES

[1] M. Buro, "Real-time strategy games: A new ai research challenge," *International Joint Conferences on Artificial Intelligence, IJCAI 2003,* pp. 1534-1535.

[2] Z. Lin, J. Gehring, V. Khalidov and G. Synnaeve, "STARDATA: A StarCraft AI Research Dataset," *13th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2017,* pp. 50–56, arXiv:1708.02139.

[3] S. Ontañon, G. Synnaeve, A. Uriarte, F. Richoux, D. Churchill and M. Preuss, "A Survey of Real-Time Strategy Game AI Research and Competition in StarCraft," *IEEE Transactions on Computational Intelligence and AI in games, IEEE Computational Intelligence Society, 2013,* 5(4), pp. 1–19, doi: 10.1109/TCIAIG.2013.2286295.

[4] Mi. Čertický, D. Churchill, K.-J. Kim, Ma. Čertický and R. Kelly, "StarCraft AI Competitions, Bots and Tournament Manager Software," *IEEE Transaction on Games, 2018,* 11(3), pp. 227–237, doi: 10.1109/TG.2018.2883499.

[5] O. Vinyals, I. Babuschkin et al., "Grandmaster level in StarCraft II using multi-agent reinforcement learning," *Nature, 2019,* 575, pp. 350–354, doi: 10.1038/s41586-019-1724-z.

[6] B. G. Weber and M. Mateas, "A data mining approach to strategy prediction," *IEEE Symposium on Computational Intelligence and Games, 2009,* pp. 140-147, doi: 10.1109/CIG.2009.5286483.

[7] H. C. Cho, K. J. Kim and S. B. Cho, "Replay-based strategy prediction and build order adaptation for StarCraft AI bots," *IEEE Conference on Computational Intelligence in Games (CIG), 2013,* pp. 1-7, doi: 10.1109/CIG.2013.6633666.

[8] G. Synnaeve and P. Bessière, "A Dataset for StarCraft AI & an Example of Armies Clustering," *Artificial Intelligence in Adversarial Real-Time Games, 2012,* arXiv:1211.4552.

[9] W. Gong, X. Zhang, B. Deng and X. Xu, "Palmprint Recognition Based on Convolutional Neural Network-Alexnet," *Federated Conference on Computer Science and Information Systems, FedCSIS 2019,* 18, ACSIS, pp. 313–316, doi: 10.15439/2019F248.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 401–412 DOI: 10.15439/2020F3 ISSN 2300-5963 ACSIS, Vol. 21

15 Years Later: A Historic Look Back at *"Quake 3: Ray Traced"*

Daniel Pohl
Intel Corporation,
Konrad-Zuse-Bogen 4,
Krailling, Germany
daniel.pohl@intel.com

Selvakumar Panneer
Intel Corporation,
2111 NE 25th Ave,
Hillsboro, OR, USA
selvakumar.panneer@intel.com

Deepak S. Vembar
Intel Corporation,
2111 NE 25th Ave,
Hillsboro, OR, USA
deepak.s.vembar@intel.com

Carl S. Marshall
Intel Corporation,
2111 NE 25th Ave,
Hillsboro, OR, USA
carl.s.marshall@intel.com

Abstract*—Real-time ray tracing has been a goal and a challenge in the graphics field for many decades. With recent advances in the hardware and software domains, this is becoming a reality today. In this work, we describe how we got to this point by taking a look back at one of the first fully ray traced games: *"Quake 3: Ray Traced"*. We provide insight into the development steps of the project with unreleased internal details and images. From a historical perspective, we look at the challenges pioneering in this area in the year 2004 and highlight the learnings in implementing the system, many of which are relevant today. We start by going from a blank screen to the full ray traced gaming experience with dynamic animations, lighting, rendered special effects and a simplistic implementation of the gameplay with basic AI enemies. We describe the challenges encountered with aliasing and the methods used to alleviate it. Lastly, we describe for the first time the unofficial continuation of the project, code named *"Quake 3: Team Arena Ray Traced"*, and provide an overview of the changes over the past 15 years that made it possible to generate fully ray-traced interactive gaming experiences with mass market hardware and an open software stack.

***Index Terms*—ray tracing, computer**

I. Introduction

Real-time ray tracing has been a dream for computer graphics programmers for many decades. By physically simulating how light interacts with surfaces, ray tracing can produce outputs that can closely resemble how light illuminates surfaces in the real world. The key challenges that were faced in bringing ray tracing to real-time was the need for enormous data parallel computing power and memory bandwidth combined with advances on the algorithmic side. While GPUs were primarily designed to accelerate rasterization, many explored re-purposing the thousands of general-purpose execution units in GPUs to perform ray tracing in real-time. The introduction of highly parallel programming languages like CUDA, OpenCL and Direct Compute enabled developers to explore a sub-set of real-time ray tracing techniques on the GPU such as indirect lighting, approximated global illumination and environment effects. In 2018, Microsoft officially announced the graphics API *DirectX Raytracing (DXR)* [1] which unified GPU vendors to support a dedicated GPU-accelerated ray tracing pipeline. This led to games taking advantage of GPU-accelerated ray tracing to perform real-time global illumination, environment effects such as reflections, refractions and shadows at a quality level which is much closer to realism than mimicking these effects using the rasterization pipeline.

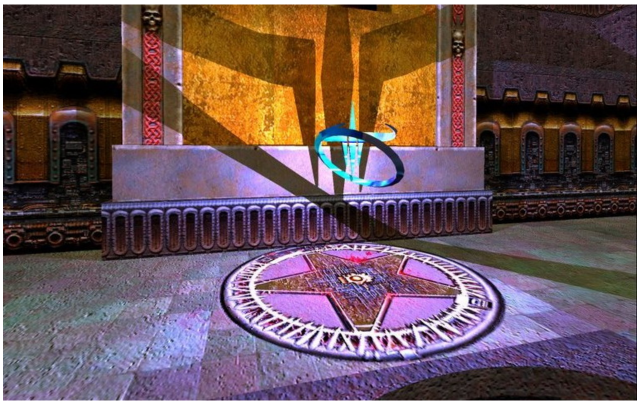


Figure 1. A blue light source is placed inside the quad damage item. Pixel-accurate real-time shadows are ray traced.

We want to use this moment to take a look back at the origins of ray tracing in games. We do this by going through a project which one of the authors of this paper developed during a bachelor's thesis at the University of Erlangen together with the Saarland University in 2004. At this time, we used the *Quake 3* game content from *id Software* in a novel game engine written from scratch together with a newly available real-time ray tracing library. The project was named *"Quake 3: Ray Traced"*, or Q3RT in short. A rendered image from it is shown in Figure 1. We share the challenges, learnings and benefits with many previously unreleased details, images and benchmarks.

Besides the historical documentation, we provide insights into what researchers and developers might encounter today in a similar matter with DirectX Raytracing as we did already one and a half decades ago. Furthermore, we disclose details about an unofficial and so far undocumented continuation of the project code named *"Quake 3: Team Arena Ray Traced"*.

In the following sections, we will start with the related work up until 2004 which was required to have the Q3RT project started. In chapter III, we give a short overview of the used OpenRT ray tracing library. This is followed by a description of our single PC hardware and a clustered network setup that were used during development. Chapter V outlines the motivation for starting this project. We continue with the first steps that describe the initial software setup and how rendered images were displayed. In chapter VII, we add static geometry from the game level into the engine. Following, different types of light sources with hard and soft shadows are discussed. In chapter IX, we add dynamics like texture animations, decals and player models to the ray traced game. In the section afterwards, we describe special effects like reflections, refractions, camera portals, ground fog and colored shadows and how they were enabled with ray tracing. We move on to chapter XI and describe the encountered issues with aliasing and which methods we applied against it. Next, we give details about the, so far, undocumented project continuation *"Quake 3: Team Arena Ray Traced"* with a ray traced water implementation and a test setup with one million reflecting spheres. Chapter XIII describes the achieved performance with the setup from 2004. After this, we share our learnings from the project. Before we finish with the conclusion, we provide a quick summary of what happened after the project until today and give a short outlook on potential future rendering architectures.

II. Related Work before Q3RT

The question of who invented ray tracing has been analyzed in an article by Hofmann [2] in 1990: going back to even before computers existed, famous artists like Leonardo da Vinci and Albrecht Dürer used between the years of 1480 and 1528 true perspective projections in their drawing. While this was done with paint instead of pixels in a frame buffer, these paintings could be seen as the first renderings with ray tracing.

Moving over to the electronic form of rendering, Appel [3] used in 1968 rays in the domain of computer graphics. In 1979, Whitted [4] used ray tracing for colorful renderings including simulations of rendered glass. Creating a single image with a resolution of 640 × 480 pixels and a total color depth of nine bits took between 45 and 120 minutes at that time.

In 1984, Cook et al. [5] described distributed ray tracing, allowing effects like blurred reflections, translucency, depth of field and motion blur. Glassner [6] optimized ray tracing for animations by using modified bounding volume hierarchies (BVH). Starting in 1995 with the military work from Muuss and Lorenzo [7], networked computers were used together to speed up ray tracing calculations. Parker et al. [8] scaled their isosurface rendering system across multiple shared-memory processors. Wald et al. [9] optimized ray tracing in 2001 for the usage of CPU-based SIMD vector extensions, offering another dimension of parallelization. A combination of scalable ray tracing across processors, SIMD and networked computers with flexibility in the rendering pipeline for different primitives and animated content was wrapped around the OpenRT ray tracing API [10], [11].

In 2004, the OpenRT rendering library was made available to us for the *"Quake 3: Ray Traced"* (Q3RT) project. During that time, the student Tim Dahmen was also exploring the usage of ray tracing in games with a newly created game named *"Oasen"* [12] in which an outdoor environment could be explored using a virtual magical flying carpet.

Parts of the work on the *"Quake 3: Ray Traced"* project have been written up in the bachelor's thesis of the student [13] in 2004, followed by a short summary in Schmittler et. al [12]. For our paper, we go in deeper with more details and with a retrospective look on that work with today's knowledge.

III. OpenRT

In 2002, OpenRT ray tracing library development was started at the Saarland University. At this time, OpenGL [14] programming was very commonly taught at universities and was the only available cross-platform low-level graphics API for the most common desktop and server computer systems. To let developers quickly adapt, one of the goals of OpenRT was to make the API as similar to OpenGL as possible.

OpenGL, at this time available in the version 1.5, had the option of using an immediate rendering mode. A triangle could be drawn as simple as in this code snippet:

```
glBegin(GL_TRIANGLES);
  glVertex3f(0, 0, 0);
  glVertex3f(0, 1, 0);
  glVertex3f(1, 1, 0);
glEnd();

glSwapBuffers();
```

Listing 1. Drawing a triangle in OpenGL 1.5

Many ray tracers, including OpenRT, were using acceleration structures like kd-trees or bounding volume hierarchies to store geometry. Therefore, the immediate rendering mode was not an available design option. Instead, a system comparable to OpenGL's display list was used where an object is described once and can later be instantiated.

```
int triangleObjID = rtGenObjects(1);
rtNewObject(triangleObjID, RT_COMPILE);
rtBegin(RT_TRIANGLES);
  rtVertex3f(0, 0, 0);
  rtVertex3f(0, 1, 0);
  rtVertex3f(1, 1, 0);
rtEnd();
rtEndObject();

int instID = rtInstantiateObject(triangleObjID);
rtSwapBuffers();
```

Listing 2. Drawing a triangle in OpenRT

For programmable shading, the OpenRTS ray tracing shading language was provided. It used a C++ interface with the possibility of creating new rays during shading, e.g. for tests regarding shadows or for tracing reflections.

IV. Development System

The main development system used at that time was a PC with a single-core Pentium 4 (code named "Northwood"), clocked at 2.66 GHz with 768 MB memory. Due to the relatively low performance of that system, many parts of the interactive development were done in a rendering resolution of either 64×64 or 128×128 pixels as shown in in Figure 2. By just pressing a key on the numpad, it was possible to decrease or increase that resolution on the fly. For analysis of correctness in higher resolutions, an offline-calculated rendering in higher resolution with supersampling was created through a key shortcut. To test animations, videos were rendered over night in higher resolution. While the work was targeting future interactive ray tracing in games, the development on a single-core machine was sometimes rather limited.

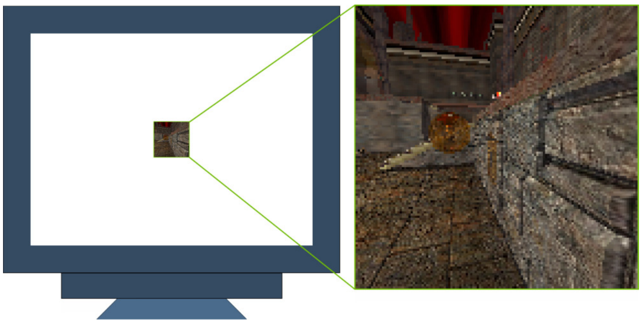


Figure 2. To give an understanding about the size of the interactive rendering resolution of 128×128 pixels, we show on the left such an image on a CRT monitor with a screen resolution of 1024×768 pixels. On the right, that image is enlarged and shows the pixelated content.

The true interactivity at higher resolutions came when using the PC cluster network at the Saarland University: 20 nodes with a dual processor system equipped with AMD MP 1800+ CPUs, clocked at 1.533 GHz, interconnected with 100 Mbit Ethernet. Due to the distance between the Q3RT development location and the cluster, this system was only used twice.

V. Quake 3: Ray Traced

The idea for researching the applicability of ray tracing for games emerged during a guest lecture from Professor Slusallek from the Saarland University. He presented recent advances in bringing ray tracing from an offline algorithm to real-time. The vision of getting this rendering fidelity combined with one of the most popular first-person shooters at that time led to the previously mentioned bachelor’s thesis to research real-time ray tracing for games.

In 2004, Quake 3 was still one of the best-looking computer games. It had an advanced shading system, offered vertex animation for player models, curved surfaces, decals, volumetric fog, portals and many more features. It came with level editors and some of the internal file formats were unofficially documented on the Internet. This made it an exciting choice to investigate its applicability to real-time ray tracing.

The full game offered 30 levels. For the relatively short time frame of nine month for the bachelor’s thesis, it made sense to limit the complexity by focusing on one of the levels exclusively. The level "q3dm7", short for "Quake 3 Deathmatch level number 7", was used. It is one of the larger levels with different interesting areas. Some experiments were still done with other levels which are shown in the Appendix.

VI. First steps

Because the source code of the original Quake 3 game was not available in 2004 yet, the Q3RT project was created from scratch in an empty Visual Studio 6.0 project. Libraries were used to quickly make progress: most importantly OpenRT for ray tracing as described before. For creating the display window, key inputs and mouse interactions libSDL was used.

The ray tracing frame buffer itself was an array in local PC memory, consisting of 32-bit RGBA data per pixel. The alpha channel was not used, but provided CPU-friendly data alignment. To display the frame buffer on the screen, for every rendered frame an OpenGL texture was created (or an existing one was modified with *glTexSubImage2D*). In 2004, many GPUs still had the limitations that textures had to be in a resolution of $2^n \times 2^m$. While with workarounds different aspect ratios could have been used as well, it was the easiest way to use resolutions like 128×128 in a 1:1 aspect ratio for this work. Once the texture has been updated, it was displayed on an OpenGL quad on the screen.

We learned that HUD, cross hair, and overlays in general were of higher quality and faster rendered on top of the image through the traditional OpenGL pipeline. They were never present in the ray traced world description where as 3D objects they might have interfered with ray intersections.

VII. Static Content

The first geometry to test if rendering works was just a single triangle. Once this was ray traced and showed up correctly through the OpenGL texture, the next step was to include a player camera model to simulate the WASD-movement known from first-person shooters. After completion, loading of the static Quake 3 geometry was addressed. The level files are stored in a .bsp file format which was unofficially documented on the Internet. The file extension .bsp hints at the usage of a binary space partitioning (BSP) tree [15] to optimize rendering on the GPU. The tree nodes contain information on the splitting plane within this volume, the bounding box of this volume and indices to the children. In the tree leaf nodes, information about the potentially visible sets [16] (PVS) were stored. The original rasterized rendering algorithm would determine in which leave node it currently is and then know which other nodes need to be rendered to not miss any geometry. The generation of the BSP tree with the PVS was done in a compiling step during level creation. To optimize performance further, professional level designers needed to place manually brushes into the scene in

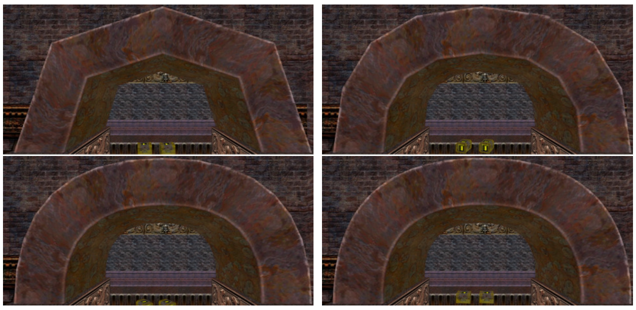


Figure 3. Top left shows the beziér patch at a resolution of 32 triangles. Top right uses 128, while bottom left has 512 and bottom right 7200 triangles.

the level editor with a hint for the PVS system to not continue to render beyond this volume.

Here is already an interesting difference between the original version for rasterization and engines using ray tracing. For Q3RT, we were not using any of the PVS data. It would even have been wrong to try to use it. For global effects like a reflection that bounces around, the PVS would not provide the full geometry that might be required to trace this ray.

Instead, we put the full level geometry into a single object for OpenRT. It internally built its acceleration structure for ray tracing on it. No further hints were required as the culling of other geometry happens implicitly on a per-ray level through the acceleration structure.

The number of triangles for the level q3dm7 is 20,000 which 15 years later appears very small and it surprises how much detail artists were getting out of this given the constraints. Besides the geometry we extracted so far, there is one more area which can increase the number of triangles. Quake 3 was one of the first games that supported rendering of beziér patches. Through changing parameters in the original game in the internal console, a different number of triangles were used for creating smoother curved surfaces. There are a total of 189 beziér patches in the level q3dm7.

For rendering curved surfaces like beziér patches, ray tracing has usually a great advantage [17]. A hit volume can be defined specifically for these patches. When tracing a ray inside such a volume, the mathematical equation can be used to calculate the exact hit point. This means that the visualization is always as smooth as possible and no single triangle-based surface becomes visible when looking very closely at these objects. However, this functionality was not yet supported by OpenRT in 2004. Therefore, we fell back to manually calculating the patches as triangle meshes during level loading. We tried various detail levels and measured the impact on the performance by adding additional geometry for the beziér patches on top of the 20K triangles of the level. We discuss the performance results in detail in section XIII. Renderings of the different beziér patch resolutions can be seen in Figure 3.

If a ray did not hit any target, the color was set to black. In DXR, this is now known as a miss shader. In game levels there is often a surrounding of a large skybox with cubical texture mapping on it. While the player moves through the scene, the sky is so far away that it appears fixed. In our case, if a ray hits outside the level, we used a sky sphere shader from OpenRT. Given the direction of the ray that does not hit any geometry, we calculate a texture offset into a spherical texture. The result is comparable to a skybox and shown in the red, cloudy area in Figure 4.

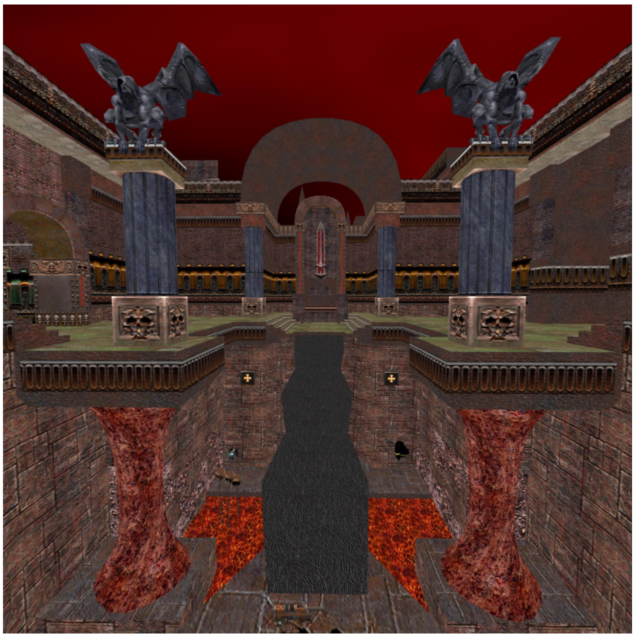


Figure 4. Level q3dm7 with all textures and the red, cloudy sky sphere. At this initial stage, there was only ambient lighting used.

VIII. Lighting

Enabling dynamic lighting through ray tracing and having accurate shadows was one of our key goals. To achieve this, OpenRT and Quake 3: Ray Traced supported different types of light sources. The most relevant ones with their adjustable parameters were:

- *point lights:* position, RGB intensity, attenuation
- *directional lights:* direction, RGB intensity, attenuation
- *spot lights:* position, direction, RGB intensity, attenuation, falloff angle

The original Quake 3 game uses hundreds of *point light* sources in the level q3dm7. However, during compile time of the map, the illumination of these is baked into static light maps [18]. While this looks reasonably well, it does not allow for fully flexible dynamic illumination. Therefore, in Q3RT, we did not use the light map data, but enabled some of the point lights manually in the scene.

The *directional light* fits very well for simulating a distant light source on top of the level like the sun.

Quake 3 in its original form does not use *spot lights*. However, we added a flash light option for the player. From the estimated hand position, a spot light was used facing forward into the viewing direction of the player. At this time, such

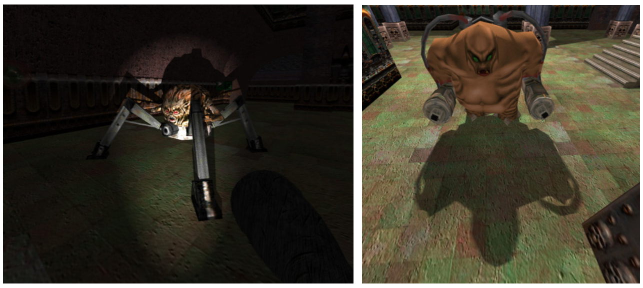


Figure 5. Left: flash light including casted hard shadows. Right: soft shadows.

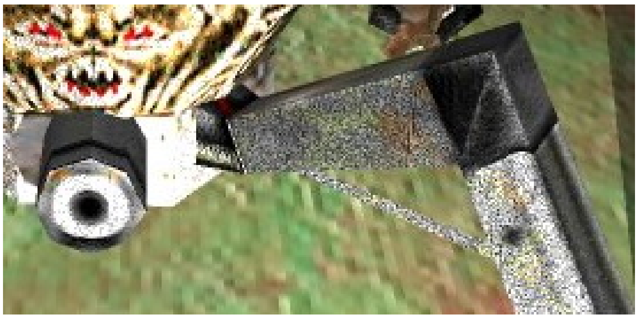


Figure 6. Epsilon issues during casting a shadow ray. The ray self-intersects with the surface to be shaded and produces a flickering moiré pattern.

an effect of dynamically lighting the scene with a flash light including real-time shadows casted from it was not available to gamers yet. An example rendering is given in Figure 5 left. Such an effect was becoming available to gamers shortly after this work finished: in August 2004, the game Doom 3 was released. In it, the shadow volume [19] algorithm was used to enable such a flash light effect including real-time shadows.

"Where there is light, there must be shadow" is a quote from Haruki Murakami [20]. One very interesting aspect of ray tracing is the shadow generation. When we shade a surface, we can shoot a shadow ray to a light source and check if the ray reaches there or if it is blocked. In the first case, we illuminate the surface with the light properties, in the second we do not change the lighting at his point, which leads to a shadow. The implementations used at this time were not optimized well for supporting a multitude of lights. Even if one light was very far away and clearly out of reach to illuminate a surface, the shadow ray was still cast to the distant light source. The previously mentioned project *"Oasen"* solved this more elegantly by defining a bounding volume for the light source on which it can impact other geometry. By doing a check first against this bounding volume, hundreds of local light sources were efficiently used.

Shooting a single shadow ray to a light source leads to a hard shadow with a pixel-exact border at the surface. However, in the real world, most scenarios do not have such a hard shadow. Still, in 2004, this was a desirable result and equals the hard shadows that Doom 3 did in the same year with the shadow volume algorithm. One experiment in the Q3RT project involved shooting six shadow rays with an offset around the center of the light instead of a single one for a point light source. While this had certainly a much higher impact on performance, the results looked visually more pleasing compared to hard shadows. Soft shadows are shown in Figure 5 right.

As easy as it was to get shadows working in ray tracing compared to implementing various shadow mapping algorithms [21]–[23] with their specific optimizations or using shadow volumes, there is also a drawback. When the shadow ray is cast from the surface to be shaded, it needs to be offset by a certain epsilon value in its ray direction to avoid hitting the same surface again due to floating point inaccuracies. As the q3dm7 level spans from very large geometry to smaller models with their own internal acceleration structures, it did happen that the chosen epsilon value worked in one case but not another. We show this in Figure 6. While it seems this issue has in practice often been tweaked and afterwards ignored as a real problem, there is now in 2019 a publication on avoiding self-intersection during ray tracing from Wächter and Binder [24]. Another elegant way to avoid these issues would have been to store in the shadow ray the surface ID of the previous hit and exclude it during intersection tests later.

IX. Dynamic Content

In 2004, the thoughts on real-time ray tracing were that it is not suited for dynamics. While the term "dynamics" goes into different dimensions, as we will show in this chapter, the reference was made towards dynamically updating geometry as this would require a costly rebuild of the acceleration structures used for ray tracing.

The way Q3RT was setup as described up until here is having static geometry and some light sources, most of them fixed except the flash light. At this stage, the impressions of the renderings are a lifeless scenario like being the only person in a virtual environment with no changes at all over time. To counter this and make a lifelike impression as in the original Quake 3 game, various stages are involved in adding dynamics.

Quake 3 was one of the first games with its own shading language [25]. This allowed various effects on otherwise flat textures. For example, with just a few lines of shading code texture animations were created. Often, the texture coordinates were modified over time before sampling. This could be combined with blending multiple textures together to create the perception of a dynamically changing environment. For the level q3dm7, we ported these effects over into the shading language OpenRTS.

Next, we added support for decals. These are used in the game to dynamically add sprites or animation effects into the scene. One example is when shooting the virtual machine gun and hit decals show up at the environment. Those stay for a few seconds and then disappear, to not overload the rendering over time. The implementation is often in the form of a textured quad with transparencies around the effect area.

One interesting property of OpenRT is that once geometric objects are instantiated, they will remain automatically in all future frames unless they are manually removed. As shown in the Appendix in Figure 18, one can easily "paint" the level by adding decals and never remove them. Of course, deletion of old objects was added afterwards into the rendering engine. From the ray tracing side, if a ray hits the quad with some transparent areas in it, it will shoot another ray behind that quad into the same direction. This can again lead to the previously mentioned epsilon issues regarding self-intersection.

The next dynamic objects are player models. In Quake 3, the animation for these are stored as pre-calculated keyframes. During rendering in the original game, the closest key frame according to the in-game time index is determined. The next closest is taken as well and the geometry is linear interpolated between those. In Q3RT, we tried this as well, but the frequent rebuilding of the acceleration structure of the player model was too costly and lowered the frame rate by a factor of 7 to 10. Therefore, during loading of Q3RT, we created a separate object for all possible keyframes of the player model. These were between 200 and 300 poses with around 1500 triangles. The acceleration structures were built during loading. In-game, we determined the closest key frame and instantiated it. In the next frame, we deleted the old instance and created the updated one. Figure 19 in the Appendix shows what happens if that deletion step is not used.

Using this method for player models meant that our animations were not as smooth as in the original game if both would be compared side-by-side at high frame rates. Newer approaches in the years afterwards provided much faster BVH buildings and therefore the option to just rebuild the interpolated player model at anytime. Other research was going towards mapping skeletal animation technologies directly into the acceleration structures [26]. Furthermore, refitting of BVHs became a viable method instead of a full rebuild [27], [28].

While the original Quake 3 game has a complex AI [29] system for moving the player models around, we took a simpler approach for Q3RT. Pre-defined way points were used for player models to move around in the levels. When a player model came too close to an AI-driven agent, the AI agent started firing into the direction of our own player. A basic game play logic was added with player health and damage given by various weapons and their projectiles.

X. Special Effects

With the combination of static and dynamic geometry, animation support and proper lighting, we explored several new special effects that were not in the original game and would be very hard to achieve without using ray tracing.

Reflections are very easy to use in a renderer with ray tracing. When a ray hits a reflecting surface, a new ray will be cast into the reflected direction, depending on the incoming angle and the normal of the hit surface. Besides showing the reflected color on the object like on a perfect mirror, there can also be a texture on the reflecting object which gets mixed together with the reflection. One of these examples is shown in Figure 7 left with the ammunition box for which we added reflecting properties, but also kept the original yellow-colored texture.

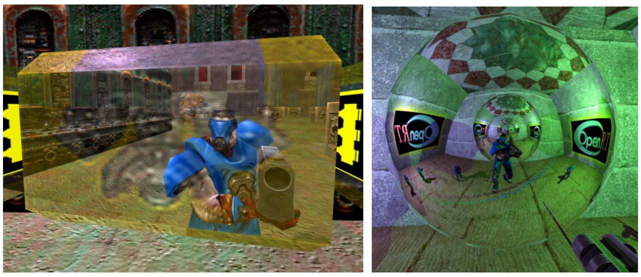


Figure 7. Left: ammo box with reflections. Right: multiple-bounce reflections.

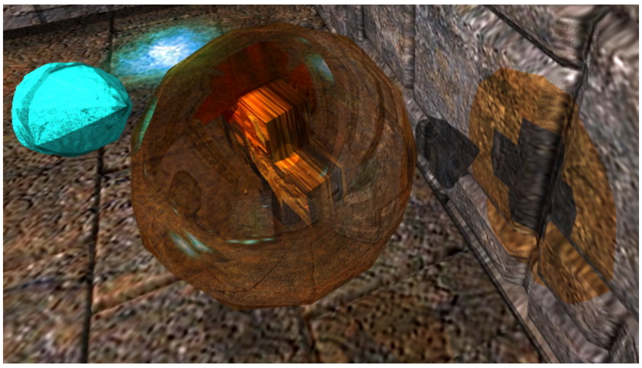


Figure 8. A glass shader on an orange sphere with reflections of the environment and refractions. Colored shadows on the wall on the right are cast through the glass using ray tracing.

As ray tracing works recursively through the shaders, there is no extra effort to be made by the developer to enable effects like a mirror inside a mirror including the multiple-bounced reflections that occur. While the concern might arise that these increase rendering time too much, it shall be noted that the impact is smaller than what people might expect. Because the reflection in the reflection is already decreasing in size compared to the original object, there are only very few rays that do a higher amount of multiple reflection bounces. Only these will require more performance, while the other ones are reasonably fast done with tracing. Other effects like lighting and shadows naturally work in the reflections as well without any additional development effort. An example of reflections in reflections can be seen in Figure 7 right.

Refractions and reflections together are found in glass. Ray tracers are known for the ability of rendering glass highly realistic. Besides the reflection ray on objects, a second ray for refraction gets traced as well, depending on the incoming angle and the refraction index of the surface or volume. An example is shown in Figure 8.

Camera portals are very easy to achieve with ray tracing. Once a ray hits the surface of the portal, another ray is set up. The new ray will have a positional offset and optionally a change in direction. It is traced from there and its shaded color value is used for the surface of the portal. This is shown in Figure 9 left. Even advancing to recursive effects showing portals in portals is possible with ray tracing at no extra development effort. Sample code for portals in OpenRTS:

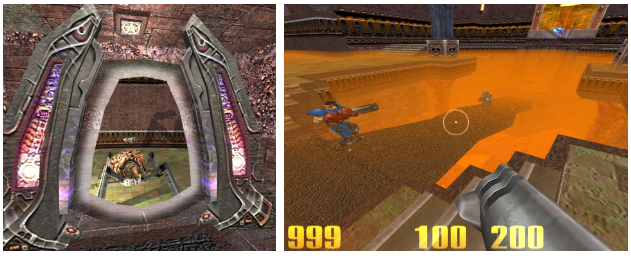


Figure 9. Left: camera portal. Right: Ground fog through ray tracing.

```
Vec3D newOrigin = ray.hitPosition + portalOffset;

// shooting a new ray with the added offset
color = traceNewRay(newOrigin, ray.direction);
```

Listing 3. Creating the portal effect with the OpenRTS shading language

Ground fog is used in the original Quake 3 game. With ray tracing, there is also an easy method of enabling this. The ground fog is modeled in a non-visible volume. Once a ray hits this volume, it will prepare another ray. That second ray continues with a small offset into the same direction. After its shading color has been determined, the length of this ray will be used. It is put into an exponential function which determines a blending value between a fog color, like orange in Figure 9 right, and the original shaded color. This way, the further the ray traveled, the more fog will be applied.

Colored shadows cast by partially transparent objects can be done in ray tracing as well. In the regular, single-colored shadow casting with ray tracing, we test with a shadow ray if any object is blocking the path from the surface to be illuminated towards the light source. In case of decals with quads that have partially transparent pixels, we must already execute shading code on it to gather the information of transparency or opaqueness. In a scenario like shown in Figure 8, we give the shadow ray that tests the glass surface a color offset if it can reach to the light source through the medium. Even though the direct light from the surface to shade to the light source is blocked, the indirect light going through the glass will now contribute the new shadow color.

It shall be noted that for many of the described effects that require shooting a secondary ray like an additional reflection or refraction ray, the previously mentioned issues of self-intersection can happen. Two examples are shown in Figure 10.

XI. Aliasing and Image Quality

As typical for rendered content, aliasing can happen - this is the same if a ray tracer or a rasterizer is used. In the original Quake 3, trilinear texture filtering is used. Full scene anti-aliasing (FSAA) could be enforced through the graphics driver.

The version of OpenRT that was provided to us at the given time was limited to bilinear texture filtering. As a result, textured objects farther away were flickering much more compared to the original version. As a workaround, we implement shader-based trilinear filtering. First, the mipmaps [30] were created and made available as different textures into the shader.

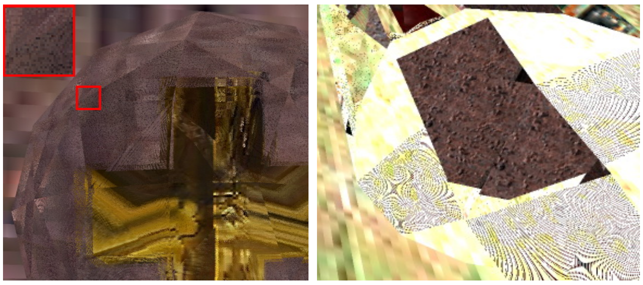


Figure 10. Self-intersection issues with a glass shader on the left and on reflections on the right.

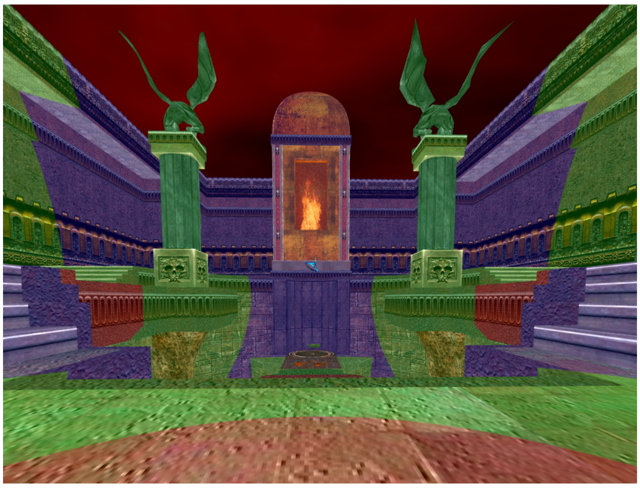


Figure 11. Visualization of mipmap levels based on distance.

Second, based on the distance of the original camera ray to the primary hit point, we determined the mipmap level. This is visualized with a color-coding in Figure 11. Last, with interpolation between the two closest mipmap levels, we achieved trilinear filtering. While this improved overall image quality, it shall be noted that this should usually be handled by the underlying ray tracing system instead of cluttering up the shading code. Furthermore, this way of using mipmaps is only a rough approximation. In the implemented form, it does not consider how large the surface is on which the texture is and what the texture and rendering resolution is. The correct way would be to use ray differentials [31] for this.

Full scene anti-aliasing was a commonly offered option in 2004 for rasterized games. It samples geometry in higher resolution, but shading is applied in the original rendering resolution. This is of course a trade-off between performance and image quality. For Q3RT, we experimented with supersampling [32]: multiple rays are shot instead of a single one for the virtual rendering camera. All rays will intersect the corresponding geometry and be shaded individually. The resulting color is then divided to create an average color between these samples. Multiple methods of how to sample the rays within a pixel are possible, e.g. using a regular grid, a rotated grid, random selections and various other stochastic methods. In Q3RT, we

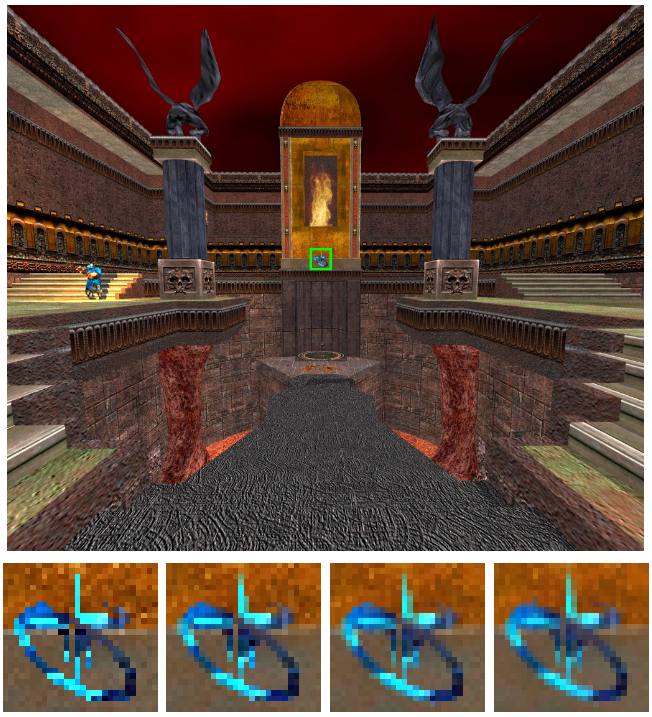


Figure 12. The top image shows the scene with 8× supersampling. The images on the bottom show a close-up of the marked green area from the top image. From left to right, these images use 1, 2, 4 and 8 samples per pixel.

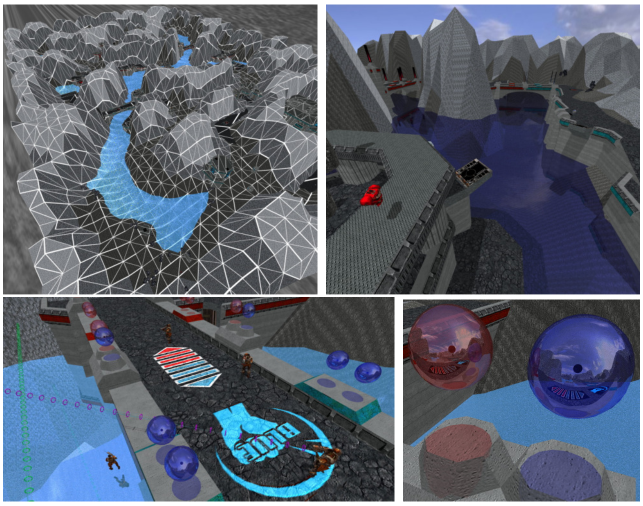


Figure 13. Team Arena level "mpterra2". Top left visualizes the triangle mesh. The other images show an initial implementation of a water shader, reflections in reflections and colored shadows from translucent objects.

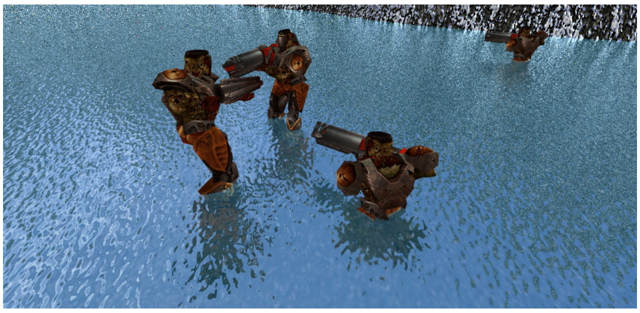


Figure 14. Ray traced water shader with a normal map for a ripple effect.

used a randomized sampling pattern. The resulting renderings with different settings for 1, 2, 4 and 8 samples per pixel can be seen in Figure 12.

XII. Quake 3: Team Arena Ray Traced

Quake 3: Team Arena was an official expansion pack to the original Quake 3 games. From the graphics side, the interesting change was that much larger outdoor levels were supported in this release. After the official work on Q3RT ended with the handover of the student's bachelor's thesis, the student continued behind closed doors on the research on ray tracing on games. While a few screenshots of this continued work have been released, there was never an official mentioning on the details of this work.

The continuation did look at the next available content from id Software, which was the Team Arena pack for Quake 3. In the continued work, one of the large outdoor levels named "mpterra2" has been chosen for ray tracing. A look from high above the level with visualization for the triangle edges and other in-game views are shown in Figure 13. The size of the level spans multiple kilometers in each dimension, which was not very common for first-person shooters at this time.

Ray traced water was a new special effects added to the ray traced version of Quake 3: Team Arena. The code was derived from the glass shader. The similarities are that both need an additional reflection and an additional refraction ray. The refraction index for the water shader was changed to 1.33 instead of 1.5 for glass. Just applying the shader on a flat surface in the world did not look very convincing. We added a normal map to the surface which simulated small ripples in the water. The normal map was animated over time, so the perception of ripples moving from wind was given. A small artistic fine tuning was done which increased the intensity of the blue color channel a bit more. An example with player models in the water is shown in Figure 14.

Having *a million reflecting spheres* in the level was the last test. The goal was to have a scenario in which ray tracing is the only feasible technology for rendering. At a certain distance to each other, about one million spheres where added into the level. As expected with ray tracing, reflections from other reflecting objects work naturally and there was no extra development required to have this working. Given the added complexity, interactivity was heavily reduced and only reasonably possible at a very low rendering resolution of 16×16 pixels. Nevertheless, we provide two offline-rendered images in higher resolution with supersampling in Figure 15.

XIII. Performance

The ray tracing performance is dependent on different factors: acceleration structure build time, the actual tracing of rays and

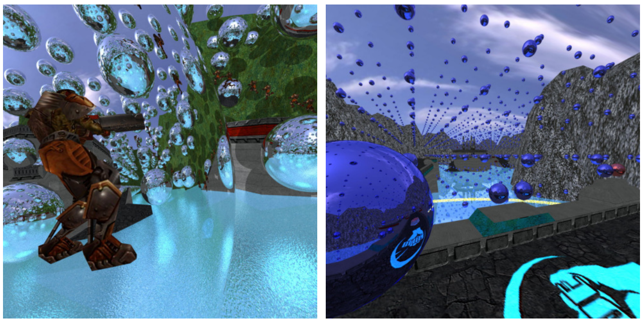


Figure 15. One million reflecting spheres in the Team Arena map

shading calculations including texturing.

The Quake 3: Ray Traced project was optimized to avoid the building of complex acceleration structures during runtime. The static geometry is only initialized once. The dynamic player models use pre-calculated acceleration structures for every animation step. The only change from frame to frame is instancing and deletion of dynamic objects like players and decals. This was implemented in OpenRT so efficiently, that no performance penalty was measurable from this.

For the tracing of rays, it was observed that, as an approximation, the number of computed rays had a linear impact on performance. However, not all rays perform equally. The primary rays shot from the camera shader showed a sub-linear impact due to cache coherency. As a rule of thumb, when increasing the rendering resolution by a factor of 4, the performance impact was a reduction factor of about 3.6 to 3.8. Using supersampling and increasing the samples per pixel from 1 to 4 was also around that area with a performance reduction of about 3.4 to 3.7. The achieved frame rates on the single-core Pentium 4 system are shown in Table I.

Resolution	**spp 1**	**spp 2**	**spp 4**	**spp 8**
128 × 128	41.5	22.9	12.2	6.3
256 × 256	11.9	6.1	3.3	1.7
512 × 512	3.1	1.6	<1	<1

Table I
Performance using a different number of samples per pixel (spp). Values in frames per second on a single-core Intel Pentium 4.

In the network cluster with 20 nodes, each equipped with a dual-core CPU, the frame rate was at 20 frames per second for a resolution of 512 × 512 pixels at 4× supersampling.

As outlined in section VII, we investigated the impact of the different static, pre-calculated beziér geometry. This is again on the single-core PC. The frame rate was averaged over a walk through the level. We show the results in Table II.

As we can interpret from these measurements, there is of course an impact from increasing the overall geometric complexity. However, specifically the step going from a total of 117K to 1380K triangles had the impact of lowering the frame rate only by a relatively low 13%. The acceleration structures that are used for each tracing of a ray pay off very well: as the beziér patches are not fully visible all the time during the walk through, only the rays that cast into the highly detailed beziér patches cause a higher performance cost. The per ray geometry culling with such structures has a logarithmic performance impact depending on the number of triangles [33]. While rasterized games can use these structures as well, they are usually not down to a per-pixel culling level, but still require some larger chunks which might turn out to not be fully visible.

	32Δ/p 20KΔ+ 6KΔ	128Δ/p 20KΔ+ 24KΔ	512Δ/p 20KΔ+ 97KΔ	7200Δ/p 20KΔ+ 1361KΔ
128x128	53.1	46.4	43.6	38.1

Table II
The main static geometry of the level Q3DM7 has 20K triangles. Depending on the detail level of the pre-calculated beziér patches, we show the performance in frames per second. For example, having 32 triangles per patch, we have the main geometry of 20K triangles added with 6K triangles for patches, resulting in an average rendering frame rate of 53.1.

XIV. Learnings

Now that we described all aspects of the Q3RT project, we want to discuss our learnings and impressions from it. One of the realizations during the project was that just because ray tracing is used, it does not magically improve the overall image quality by default as one might expect when previously viewing ray traced images from offline-rendered content only. The problems of aliasing still happen the same as with a rasterizer. In fact, rendering only the primary rays from the virtual ray tracing camera results in the same image as using rasterization. However, once specific effects like reflections, refractions and shadows are added, it became clear where the strengths of ray tracing are. Those effects are done with only a small amount of development effort and provide great, physically-based results. Furthermore, they are always per-pixel efficient. If only very few pixels of a reflecting object are visible, only those add an extra cost during rendering. This is in contrast with approaches like shadow mapping or reflection mapping, where the map needs to be created at a certain resolution even if only one pixel shows up in the final image using it. The combination of multiple effects, like reflections in reflections with the correct translucent colored shadowing worked flawless. No ordering of which effect to calculate first needs to be provided by the developer. It was also impressive to see the scaling of the ray traced rendering when more compute nodes were added. Ray tracing has been described before as an *embarrassingly parallel* algorithm [34]. Calculations of tracing one ray are completely independent of the other rays. Therefore, adding more computing nodes, more CPUs and other hardware units provides a great performance improvement.

The two most difficult aspects during the development of Q3RT were the low rendering resolution on a single-core PC. However, with the workarounds of quickly creating higher resolution screenshots offline and training of the human perception to this lower level of detail, this became less of an issue after a while. The other aspect that required tedious tuning of epsilon values were the self-intersection issues as described in the chapters before.

Overall, for the student working on the Q3RT project, it was clear afterwards that ray tracing will play an important role in the future of interactive games. The development of hardware was already going to multi-core CPU architectures and highly parallel GPUs, which would help in scaling ray tracing up to real-time for consumers eventually. The open question was therefore not if it will happen, but when it would happen and be available to consumers.

XV. Fast forward to 2019 and Outlook

Today, 15 years later from our original real-time ray tracing project, there has been a tremendous amount of advancement in algorithms, software APIs and hardware for enabling real-time ray tracing. Through the years, we have seen many research projects on bringing ray tracing to modern game content: Quake 4 [35], Enemy Territory: Quake Wars [36] and the Wolfenstein [37] (2009) game were changed to use ray tracing. Bikker [38] looked into advanced effects like ambient occlusion with ray tracing and provided performance optimizations. McGuire and Mara [39] used screen-space ray tracing in 2014 as an approximation for rasterized content.

Developers targeting the professional rendering market were able to apply ray tracing into their engines by using highly optimized libraries like Intel Embree [40] or Nvidia OptiX [41].

With the release of DirectX Raytracing (DXR) in 2018, real-time ray tracing became available to the consumer and gaming world. In conjunction with that release, samples starting with ray tracing a simple triangle up to building a small game engine using DXR have been released [42]. We are starting to see many popular gaming titles using this technology in a hybrid mixture between ray tracing and rasterization. For example, *Battlefield V*, *Metro Exodus*, *Shadow of the Tomb Raider* and *Wolfenstein: Youngblood* all have some features that can utilize ray tracing in real-time. A next step to expect would be the development of games using additional global illumination features, similar to the *Quake 2 Vulkan Path Tracing* [43] project. Further along, we might be heading towards fully ray traced games with full real-time global illumination. However, there is still a need for more advances in algorithms and hardware before this can be achieved.

Recently, we have seen how machine learning being applied to interactive ray-tracing can denoise an image [44], [45] at similar quality to a highly sampled ray traced image with far fewer ray samples per pixel. Today's graphics architectures have compute cores for general purpose, rasterization, ray tracing, and AI. A combination using those can provide additional benefits for rendering like a rasterized image with ray traced special effects, where the lighting gets denoised through AI and the final image gets upsampled with AI to a higher display resolution. For future work, we could see more areas where AI can impact the rendering pipeline to help accelerate ray tracing like guiding importance sampling, applying super-resolution across frames, and enhancing acceleration structures.

XVI. Conclusion

We have shown how in the year 2004 we created a full ray traced game in the research project *Quake 3: Ray Traced*. We described the first footsteps of applying ray tracing to gaming. We demonstrated how we handled multiple light types, shadows, dynamics, and special effects. A look at aliasing and ray length-based mipmapping with supersampling was provided. We shared our learnings from the Q3RT project. We gave an overview to what happened after it until today with an outlook on future rendering architectures. Although we were limited by processing power and, in turn, screen resolution and features at the time, this provided validation that as algorithms and hardware advanced that real-time ray tracing was possible. Today, it is even included in several AAA game titles.

XVII. Acknowledgment

Many thanks to Jörg Schmittler for supervising the *Quake 3: Ray Traced* project together with Prof. Marc Stamminger and Prof. Philipp Slusallek. Additional thanks to Tim Dahmen and the whole OpenRT team for providing the ray tracing library.

References

[1] C. Wyman and A. Marrs, “Introduction to DirectX Raytracing”, in *Ray Tracing Gems*, Springer, 2019, pp. 21–47.

[2] G. R. Hofmann, “Who invented ray tracing?”, *The Visual Computer*, vol. 6, no. 3, pp. 120–124, 1990.

[3] A. Appel, “Some techniques for shading machine renderings of solids”, in *Proceedings of the April 30–May 2, 1968, spring joint computer conference*, ACM, 1968, pp. 37–45.

[4] T. Whitted, “An improved illumination model for shaded display”, in *ACM SIGGRAPH Computer Graphics*, ACM, vol. 13, 1979, p. 14.

[5] R. L. Cook, T. Porter, and L. Carpenter, “Distributed ray tracing”, in *ACM SIGGRAPH computer graphics*, ACM, vol. 18, 1984, pp. 137–145.

[6] A. S. Glassner, “Spacetime ray tracing for animation”, *IEEE Computer Graphics and Applications*, vol. 8, no. 2, pp. 60–70, 1988.

[7] M. J. Muuss and M. Lorenzo, “High-resolution interactive multispectral missile sensor simulation for ATR and DIS”, in *Proceedings of BRL-CAD Symposium’95*, vol. 2, 1995.

[8] S. Parker, P. Shirley, Y. Livnat, C. Hansen, and P.-P. Sloan, “Interactive ray tracing for isosurface rendering”, in *Proceedings Visualization’98 (Cat. No. 98CB36276)*, IEEE, 1998, pp. 233–238.

[9] I. Wald, P. Slusallek, C. Benthin, and M. Wagner, “Interactive rendering with coherent ray tracing”, in *Computer graphics forum*, Wiley Online Library, vol. 20, 2001, pp. 153–165.

[10] I. Wald and C. Benthin, “OpenRT-A flexible and scalable rendering engine for interactive 3D graphics”, 2002.

[11] A. Dietrich, I. Wald, C. Benthin, and P. Slusallek, "The OpenRT application programming interface–towards a common API for interactive ray tracing", in *Proceedings of the 2003 OpenSG Symposium*, Citeseer, 2003, pp. 23–31.

[12] J. Schmittler, D. Pohl, T. Dahmen, C. Vogelgesang, and P. Slusallek, "Realtime ray tracing for current and future games", in *ACM SIGGRAPH 2005 Courses*, 2005, p. 23.

[13] D. Pohl, "Applying Ray Tracing to the Quake 3 Computer Game", *University of Erlangen*, 2004.

[14] M. Woo, J. Neider, T. Davis, and D. Shreiner, *OpenGL programming guide: the official guide to learning OpenGL, version 1.2*. Addison-Wesley Longman Publishing Co., Inc., 1999.

[15] H. Fuchs, Z. M. Kedem, and B. F. Naylor, "On visible surface generation by a priori tree structures", in *ACM Siggraph Computer Graphics*, vol. 14, 1980, pp. 124–133.

[16] D. P. Luebke and C. Georges, "Portals and mirrors: Simple, fast evaluation of potentially visible sets.", *SI3D*, vol. 95, p. 105, 1995.

[17] C. Benthin, I. Wald, and P. Slusallek, "Interactive ray tracing of free-form surfaces", in *Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*, ACM, 2004, pp. 99–106.

[18] M. Abrash, "Quake's lighting model: Surface caching", *Graphic Programming Black Book*, 2000.

[19] J. Carmack, *John Carmack on shadow volumes*, http://fabiensanglard.net/doom3_documentation/CarmackOnShadowVolumes.txt, 2000.

[20] H. Murakami, *1Q84*. Random House, 2012.

[21] C. Everitt, A. Rege, and C. Cebenoyan, "Hardware shadow mapping", *White paper, nVIDIA*, vol. 2, 2001.

[22] R. Fernando, S. Fernandez, K. Bala, and D. P. Greenberg, "Adaptive shadow maps", in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM, 2001, pp. 387–390.

[23] M. Stamminger and G. Drettakis, "Perspective shadow maps", in *ACM transactions on graphics (TOG)*, vol. 21, 2002, pp. 557–562.

[24] C. Wächter and N. Binder, "A fast and robust method for avoiding self-intersection", in *Ray Tracing Gems*, Springer, 2019, pp. 77–85.

[25] P. Jaquays and B. Hook, "Quake 3: Arena shader manual, revision 10", in *Game Developer's Conference Hardcore Technical Seminar Notes*, 1999.

[26] J. Günther, H. Friedrich, H.-P. Seidel, and P. Slusallek, "Interactive ray tracing of skinned animations", *The Visual Computer*, vol. 22, no. 9-11, pp. 785–792, 2006.

[27] C. Lauterbach, M. Garland, S. Sengupta, D. Luebke, and D. Manocha, "Fast BVH construction on GPUs", in *Computer Graphics Forum*, Wiley Online Library, vol. 28, 2009, pp. 375–384.

[28] D. Kopta, T. Ize, J. Spjut, E. Brunvand, A. Davis, and A. Kensler, "Fast, effective BVH updates for animated scenes", in *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, ACM, 2012, pp. 197–204.

[29] J. Van Waveren, "The Quake III Arena Bot", *University of Technology Delft*, 2001.

[30] P. S. Heckbert *et al.*, "Texture mapping polygons in perspective", Citeseer, Tech. Rep., 1983.

[31] H. Igehy, "Tracing ray differentials", in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 1999, pp. 179–186.

[32] F. C. Crow, "A comparison of antialiasing techniques", *IEEE Computer Graphics and Applications*, no. 1, pp. 40–48, 1981.

[33] J. Schmittler, S. Woop, D. Wagner, W. J. Paul, and P. Slusallek, "Realtime ray tracing of dynamic scenes on an FPGA chip", in *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, 2004, pp. 95–106.

[34] B. Freisleben, D. Hartmann, and T. Kielmann, "Parallel raytracing: A case study on partitioning and scheduling on workstation clusters", in *Proceedings of the thirtieth hawaii international conference on system sciences*, IEEE, vol. 1, 1997, pp. 596–605.

[35] *Quake 4: Ray Traced*, http://www.q4rt.de.

[36] *Quake Wars: Ray Traced*, http://www.qwrt.de.

[37] *Wolfenstein: Ray Traced*, http://www.wolfrt.de.

[38] J. Bikker, "Real-time ray tracing through the eyes of a game developer", in *2007 IEEE Symposium on Interactive Ray Tracing*, IEEE, 2007, pp. 1–10.

[39] M. McGuire and M. Mara, "Efficient GPU screen-space ray tracing", *Journal of Computer Graphics Techniques (JCGT)*, vol. 3, no. 4, pp. 73–85, 2014.

[40] *Intel embree*, http://www.embree.org.

[41] *Nvidia OptiX*, http://developer.nvidia.com/optix.

[42] *DirectX Raytracing samples*, http://github.com/microsoft/DirectX-Graphics-Samples/tree/master/Samples/Desktop/D3D12Raytracing.

[43] *Quake 2 Vulkan Path Tracer*, http://brechpunkt.de/q2vkpt.

[44] C. R. A. Chaitanya, A. S. Kaplanyan, C. Schied, M. Salvi, A. Lefohn, D. Nowrouzezahrai, and T. Aila, "Interactive reconstruction of monte carlo image sequences using a recurrent denoising autoencoder", *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 98, 2017.

[45] *Intel Open Image Denoise*, http://openimagedenoise.github.io.

APPENDIX

As an appendix, we provide more images that were created during the development. Not always everything worked on the first try or was fully implemented when taking these images. Furthermore, we show unreleased images from other levels of Quake 3, Quake 2 and the classic Wolfenstein game that were used for experimentation with ray tracing during the project.

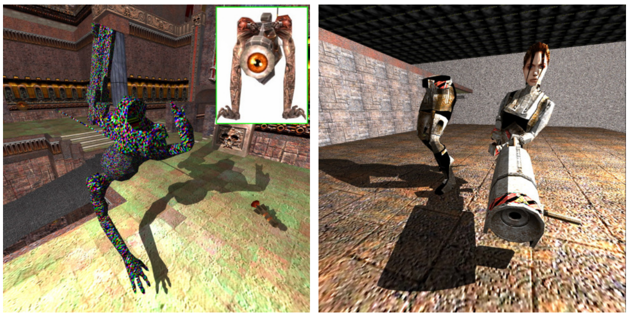


Figure 16. Left: invalid texture access for the Quake 3 model "Orbb". The texture visualizes random parts of the memory. Right: too large offset between the lower and upper body model.

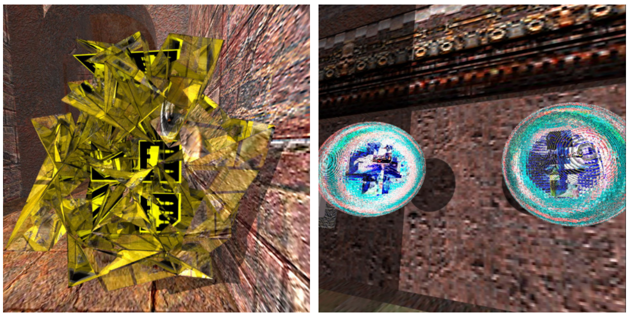


Figure 17. Left: something went wrong during the mesh setup of the ammunition box. Right: during rendering the glass of the health sphere, rays got stuck inside due to self-intersection problems.

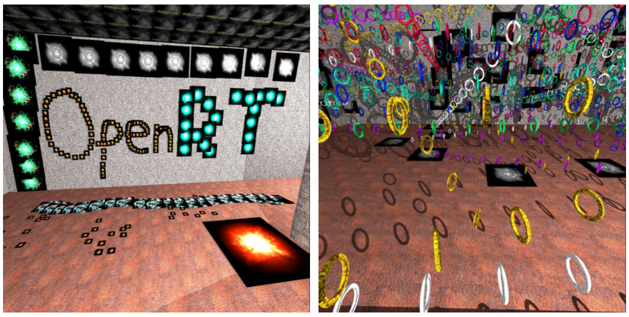


Figure 18. Left: shooting decals on the wall that last forever. Right: shooting rings of the virtual rail gun weapon without deletion.

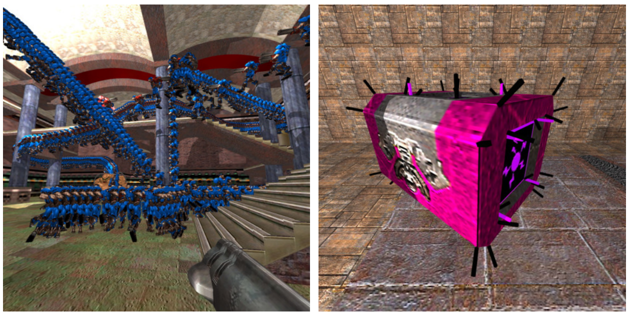


Figure 19. Left: moving character animations if they are not manually deleted. Right: visualizing surface normals through added geometry on models.

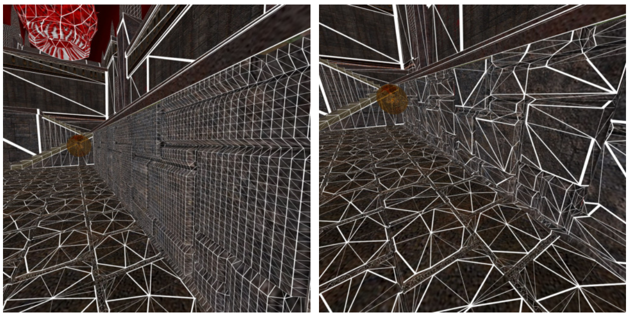


Figure 20. Both images show experiments replacing the original flat geometry with highly detailed meshes. The performance difference was only small.

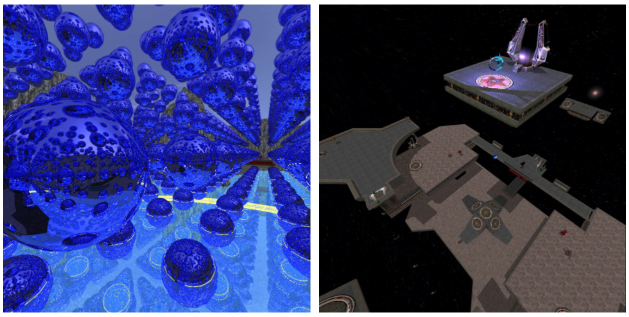


Figure 21. Left: the white pixels show areas where not enough ray tracing recursion depth was set for multiple bounced reflections. Right: the Quake 3 map q3dm17, ray traced.

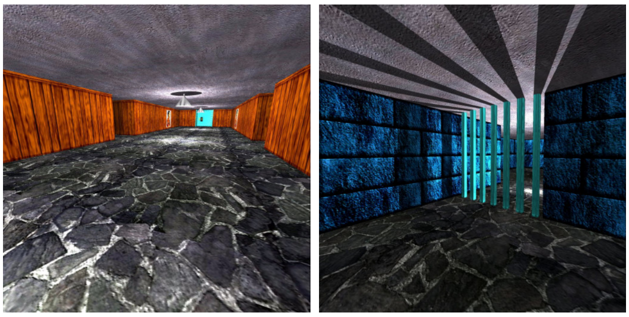


Figure 22. Both images show renderings of an enhanced level of Wolfenstein 3D (1992) with ray tracing.

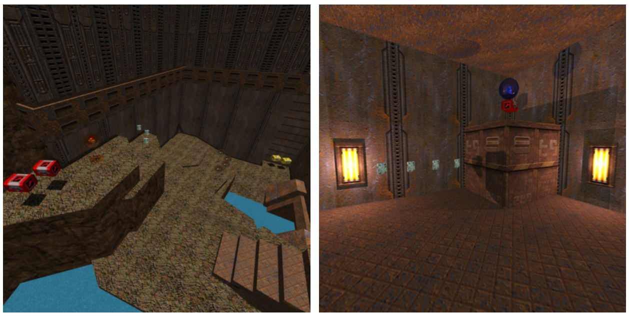


Figure 23. Both images show a modified version of the Quake 2 level q2dm1 using ray tracing.

DOI: 10.15439/2020F208
ISSN 2300-5963 ACSIS, Vol. 21

Simulator of a Supercomputer Job Management System as a Scientific Service

Gennadiy Savin, Boris Shabanov, Dmitriy Lyakhovets, Anton Baranov, Pavel Telegin
Joint Supercomputer Center of the Russian Academy of Sciences
Leninsky prospect, 32a, Moscow, 119334, Russian Federation
Email: [savin, shabanov]@jscc.ru, anetto@inbox.ru, antbar@mail.ru, ptelegin@jscc.ru

***Abstract*—Job management system (JMS) is an important part of any supercomputer. JMS creates a schedule for launching jobs of different users. Actual job management systems are complex software systems with a number of settings. These settings have a significant impact on various JMS metrics, such as supercomputer resources utilization, mean waiting time of a job in queue, and others. Various JMS simulators are widely used to study the influence of JMS settings or modifications, new scheduling algorithms, jobs input stream parameters or available computing resources for JMS efficiency metrics. The article presents the comparative analysis results of the actual JMS simulators (Alea, ScSF, Batsim, AccaSim, Slurm simulator) and their application areas. The authors consider new ways to use the JMS simulator as a scientific service for researchers. With such a service, the researchers are able to study various hypotheses about JMS efficiency, algorithms or parameters. This gives the folowing: (1) research is performed on the service side around the clock, (2) the simulator accuracy or adequacy is provided by the service, (3) the research results reproducibility is ensured, and the simulator-as-a-service becomes a single entry point for the researchers.**

I. Introduction

JOB MANAGEMENT system (JMS) is an essential software for multi-user high performance computing [1]. JMS handles a queue of user jobs, determines job launch order, allocates computing nodes for launched jobs, controls job termination and checks that nodes are freed after job termination. A number of metrics, such as resources utilization, mean waiting time of a job in queue, and others, measure JMS quality.

Modern JMS have been evolving for decades, and now JMS are complex software systems with a lot of adjustable parameters. The example of the parameters is a scheduling algorithm and its options, users and groups priority, job size limitation. The parameters optimization could be challenging because parameters influence on JMS quality can be far from obvious.

A job launch order in most of the JMS is based on job time limits specified by users. JMS terminates the job when it reaches job limit. Most of JMSs provide worst-case job launch time to a user (if no nodes are broken) when every job in queue ends at its limit. The research [2] shows that most of the jobs end far before their limits.

Every job launch time forecasting is a challenge. The forecast could provide to a user more precise launch time estimation. Such forecast is especially important for geographically distributed supercomputer systems. Note the integration of geographically distributed supercomputer resources is a steady high-performance computing trend [3]. The main goal of this integration is creation of a digital platform to meet the scientific, educational and industrial needs for high-performance computing. The digital platform can include several JMSs. Each JMS could have its own input job stream, so the digital platform integrates multiple input job streams. In general, any job from any input stream can be assigned to any JMS of the platform. As a result, the job management complexity increases significantly, so as the prediction complexity of the job launch time and location. The launch location is a supercomputer in the distributed digital platform, where the job will be executed. To schedule jobs in a distributed system efficiently, it is necessary to predict the release time of the required computing resources in each of the JMS accurately [4]. Job launch time forecast allows to determine the JMS where a job could be executed faster. The forecast could be used to schedule a global job queue for the distributed supercomputer system [5]. Job from the global queue could be executed on less busy JMS which reduces resource imbalance. This can be achieved by modelling of the management system in order to predict the launch time and location for each job. We assume to model up to hundreds of thousands jobs per year executed on tens of thousands nodes from thousands of different users.

Of particular interest are different aspects of JMS modelling to solve two tasks mentioned above. The first task is JMS parameters optimization. The second task is job launch time forecasting. Most popular JMS model implementation is a simulator, thus we would use the words «model» and «simulator» as synonyms.

II. Related Work

Nowadays a number of JMS is available, for instance, SLURM [6], PBS [7], LSF [8]. SUPPZ [9] is an example of domestic JMS, which has been used in Joint Supercomputer Center of the Russian Academy of Sciences (JSCC RAS) for more than 20 years. Of particular note is JMS for relatively small clusters named OAR [10]. In the paper [11] authors investigate some techniques to provide malleable behavior on

The work was carried out within the RFBR grant (project No. 18-29-03236), and the framework the JSCC RAS government assignment

MPI applications and the impact of this support upon the OAR resource manager.

Many JMS models can be found in the literature. There are both models of existing JMS (like SLURM, SUPPZ, OAR) and models of general JMS. Let us consider recent papers for JMS modelling. Existing JMS modelling tools can be divided into 3 classes: modelling languages, software platforms and simulators.

Modelling languages fully support the modelling process — the model time control and the object interaction in the system. This allows the researcher to focus on the description of the JMS model essential properties and characteristics. In this case, the researcher must independently reproduce the entire JMS operation logic — build a supercomputer model with the given characteristics and the jobs processing order, create a job scheduler with a given scheduling algorithm, describe the input job stream, develop program modules for conducting the experiment and collecting the necessary results. Specialized languages such as AnyLogic [12], ExtendSIM [12], GPSS World [13], Simulink [12] can be used to build a JMS model.

AnyLogic is a general-purpose modelling language developing since 2000. Model development is performed in a graphical interface, the Java programming language is supported to finalize the components. ExtendSIM has been in development since 1987. ExtendSim has a user-friendly interface and does not require special knowledge and programming skills. It is enough to draw a block diagram of the modeled process and enter the initial data using the necessary block settings. GPSS World is one of the earliest modelling languages created in 1961. A GPSS program is a sequence of statements displaying events. Simulink is a modelling language developing since 1984 that provides tight integration with MATLAB.

Software platforms for JMS modelling allow reducing the time to implement the model due to the parts of the modeled systems and components for displaying various data (for example, statistical) implemented in the platform. The software platform provides typical entities, such as «computing module», «job», «job scheduler» with a wide range of different characteristics. The developer builds a model from ready-made large blocks and configures them for solving the problem. Software platforms such as SimGrid [14], GridSim [15] are widespread. SimGrid is a software platform for developing distributed application simulators, developing since 1999. GridSim developing since 2002 is widely used by various researchers to model grid systems and JMS.

JMS simulators provide the researher with a ready model that needs to be configured. Model configuration may require code development. Examples of JMS simulators are MONARC [16], Alea [17], OptorSim [18], WorkflowSim [19]. MONARC has been developed since 2000 and is designed to analyze large-sized systems. A key aspect of this simulator is wide opportunities for monitoring system components [20]. Alea is based on GridSim and has been in development since 2007. The Alea main purpose is the scheduling algorithms study, and a number of scheduling algorithms is already implemented in the simulator. WorkflowSim has been in development since 2012. The WorkflowSim main purpose is the job stream processing optimization [20]. OptorSim has been under development since 2003. In OptorSim, it is possible to configure the network topology between computing nodes with their throughput and the job data volume.

We are going to take a detailed look at Batsim [21], created for OAR modelling. In the paper authors criticize current situation when other modelling researches are hard to reproduce. Authors mention that often models are not used after results publication. One of the problems in JMS modelling is complexity of experiment reproduction due to model unavailability, lack of input or output data (in case when average metrics are published, but not all of the outputs), no access to the experimental environment [21].The solution presented by the authors is their own model Batsim which can be used by other researchers. Beside that, the authors suggest to publish full experiment plan, including all parameters, inputs and outputs. Batsim validation consists of Gantt chart of workload, plot of the difference between the real and the simulated execution time (so as for submission and turnaround time) of all jobs in workload. The plot comparison is done visually. Authors note the difference between the real system and the model.

Paper [22] states that available modelling tools do not cover the full cycle of modelling, generation, simulation, and analysis. The authors present their own model — scheduler simulation framework (ScSF), model of SLURM. There is no way to use the fixed job input for ScSF because input generator is a mandatory part of ScSF. Paper notes the complexity of long simulations since their few weeks experiment sometimes failed due to power cuts, hypervisor or VM failures and system updates. There is no model validation found in the paper.

Paper [23] represents a new version of SLURM simulator. For model validation there are plots of job start time difference between simulated and real SLURM runs. The authors calculated average and standart deviation of all job start time differences. Plots for real system and its model are compared visually. There are data for natural modelling on a small cluster (10 computing nodes) in the paper.

There is a new Alea version presented in paper [24]. Alea is the model of general JMS. The paper provides technical details of the model and data about simulation speed. There is no model validation found in the paper.

Paper [25] presents an AccaSim model of general JMS, which is compared to Batsim and Alea. Comparison with ScSF is not done due to high system requirements and complex configuration. Also ScSF could not use fixed job input, only generation of a new one was possible during the experiment. There is no model validation found in the paper, but there is validation of input generator.

The above analysis of papers about modelling revealed two scientific problems for further development.

The first problem is model validation, more precisely lack of common ways to measure model adequacy and accuracy. Other researchers skip validation, or compare some plots visually or calculate average metrics. There are no analytic measures for

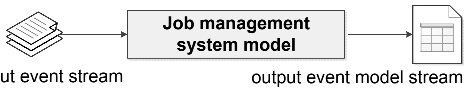


Fig. 1. JMS model research

model accuracy. The authors are investigating this problem, first results are presented at [26].

III. JMS Model as a Scientific Service

Modern way of JMS research could be explained as follows. A researcher creates a JMS model [26], for example, a JMS simulator. The researcher experiments with some input event stream and model with fixed parameters and saves output model stream (see Fig. 1). Output model stream could be saved in some kind of a plain text file or in a database. The database is not a bottleneck in this case. Authors use PostgreSQL. MySQL is used in SLURM simulator.

Then the researcher modifies input event stream (e.g., changing job density) or JMS model (e.g., changing scheduling algorithm or its options), repeats the experiment and gets new output model stream. By analyzing output stream before and after the changes the researcher can determine if modifications were good or bad (for example, old algorithm provided 95% average utilization and the new one provided 96%). Reability of the results is ensured by series of experiments, and results are presented as average of some metric and its standard deviation. Improvement is significant if it is bigger than the deviation.

The above way of research faces the following problems:

1) Researchers have to develop their own JMS model or to master the existing ones — its installing, configuring, ways to form an input event stream, getting and analyzing the result [21].
2) Researchers have to validate JMS model on their own.
3) It is hard to reproduce experiments of other researches. Used models could not be found publicly, or research could not provide all of the input or output streams (for example, only average metrics could be published).

All of the problems can be solved if a JMS model operates alongside with the real JMS. Organization providing the JMS for users could also provide a JMS model as a scientific service for researchers. In that case JMS model development, its installation, configuring and validation is done once by the organization, and researchers use the model over and over again.

JMS model as a scientific service extends the field of JMS research. Often the research relies on some old data (for a period in the past) or a generated input event stream (generated to statistically the same as real stream for some period). Simulator-as-a-service allows to experiment with most recent streams, close to the real-time. Such a simulator could be used to forecast job launch times in real-time. Besides, such a service could constantly calculate model adequacy.

Building the simulator-as-a-service is a challenging job. Let us consider various options of service organization.

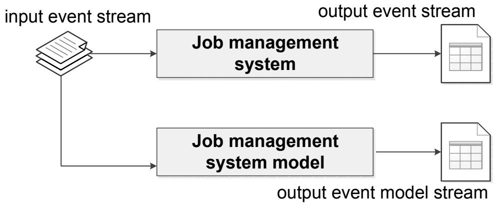


Fig. 2. Using JMS model in real-time

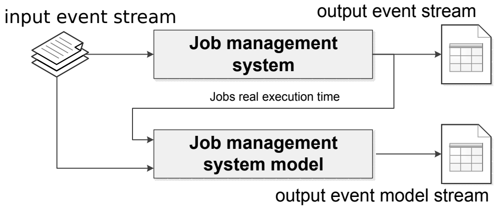


Fig. 3. Using JMS model in real-time with feedback

For real-time research we should duplicate input event stream to both real JMS and its model (see Fig. 2). In this case a technical problem reveals — there is no real job execution time in input event stream, which is often less than the limit specified by user.

Thus, JMS model does not have an important value in input stream — real execution time. Let us consider ways to get it. The first way is to add a feedback from JMS to JMS model (see Fig. 3). After the job ends, JMS notifies the model about real execution time.

The second way is to forecast real execution time. In this case for every submitted task real execution time is forecasted basing on statistics (see Fig. 4).

Result of combination of the two ways is JMS model with feedback, and forecast (see Fig. 5). For every submitted job, real execution time is forecasted basing on statistics. When the job ends, real JMS notifies the model and replaces the forecast with the actual execution time which allows to constantly correct forecasting and improve its precision.

Simulator-as-a-service reduces the complexity for researchers. They can concentrate on scientific aspects of experiments. Additional advantage is reproducibility improvements.

Using a JMS model in real-time enables the continuous comparison of real and model output streams which allows to modify a JMS model to improve its accuracy. Using a JMS model in real-time with forecasting and feedback allows to

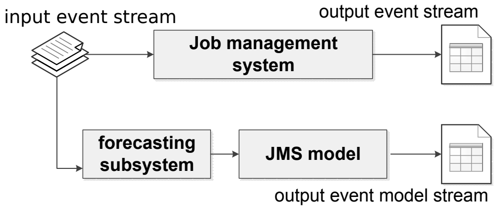


Fig. 4. Using JMS model in real-time with forecasting

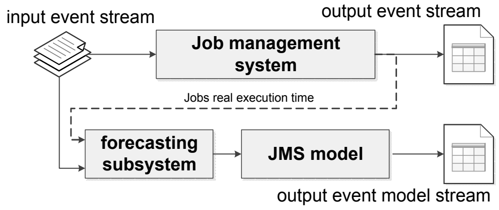


Fig. 5. Using JMS model in real-time with forecasting and feedback

improve forecasting subsystem which later could be used for scheduling.

IV. Conclusion

The authors' JMS models analysis revealed the complexity of other researchers' models reproduction. Lack of used software or its installation or configuration complexity, incomplete input or output event streams, lack of common ways to validate a model — all these makes every researcher to repeat a huge amount of work in order to search, configure, execute and validate a model.

We suggest a new approach for experimental study of JMS based on once developed and then publicly served JMS model used by researchers. The approach does not restrict researchers in using their own models or creating new simulators as a service. Simulator-as-a-service could attract more researchers making JMS model easy to use for any kind of research. The main purpose of the simulator-as-a-service is to research the influence of new scheduling algorithms or scheduling parameters on JMS quality metrics.

We outlined different ways to build a JMS model simulator-as-a-service: to work in real-time, in real-time with feedback, in real-time with forecasting, in real-time with forecasting and feedback.

References

[1] A. Reuther et al. "Scalable system scheduling for HPC and big data," *J. of Parallel and Distributed Computing,* vol. 111, 2018, pp. 76–92. https://dx.doi.org/10.1016/j.jpdc.2017.06.009

[2] A. V. Baranov, E. A. Kiselev, D. S. Lyakhovets, "The quasi scheduler for utilization of multiprocessing computing system's idle resources under control of the Management System of the Parallel Jobs," *Bul. of the South Ural State University. Series Comp. Math. and Software Engineering,* issue 3(4), 2014, pp. 75–84 (in Russian). https://dx.doi.org/10.14529/cmse140405

[3] B. Shabanov, A. Ovsiannikov, A. Baranov, S. Leshchev, B. Dolgov, and D. Derbyshev, "The distributed network of the supercomputer centers for collaborative research," *in Program systems: Theory and applications,* 8:4(35), 2017, pp. 245–262 (In Russian). https://dx.doi.org/10.25209/2079-3316-2017-8-4-245-262

[4] N. N. Kuzyurin, D. A. Grushin, and S. A. Fomin, "Two-dimensional packing problems and optimization in distributed computing systems," *in Proc.of the Institute for System Programming of the RAS,* vol. 26, no 1, 2014, pp. 483–502 (in Russian).

[5] A. I. Tikhomirov, "The English Auction Method for Scheduling Jobs in a Distributed Network of Supercomputer Centers," *Lobachevskii J. of Math.,* vol. 40, issue 5, 2019, pp. 606–613. https://dx.doi.org/10.1134/s1995080219050214

[6] A. B. Yoo, M. A. Jette, and M. Grondona, "SLURM: Simple Linux Utility for Resource Management," *Lecture Notes in Comp. Science,* vol. 2862, 2003, pp. 44–60. https://dx.doi.org/10.1007/10968987_3

[7] R. L. Henderson, "Job scheduling under the Portable Batch System," *Lecture Notes in Comp. Science,* vol. 949, 1995, pp. 279–294. https://dx.doi.org/10.1007/3-540-60153-8_34

[8] IBM Spectrum LSF overview, https://www.ibm.com/support/knowledgecenter/en/SSWRJV_10.1.0/lsf_foundations/chap_lsf_overview_foundations.html

[9] G. I. Savin, B. M. Shabanov, P. N. Telegin, and A. V. Baranov, "Joint Supercomputer Center of the Russian Academy of Sciences: Present and Future," *Lobachevskii J. of Mathematics,* vol. 40, issue 11, 2019, pp. 1853–1862. https://dx.doi.org/10.1134/S1995080219110271

[10] N. Capit et al., "A batch scheduler with high level components," *in IEEE Int. Symp. on Cluster Comp. and the Grid,* Cardiff, Wales, UK, vol. 2, 2005, pp. 776–783. https://dx.doi.org/10.1109/CCGRID.2005.1558641

[11] M. C. Cera et al., "Supporting Malleability in Parallel Architectures with Dynamic CPUSETs Mapping and Dynamic MPI," *in Distributed Computing and Networking,* 2010, pp. 242–257. https://dx.doi.org/10.1007/978-3-642-11322-2_26

[12] I. M. Yakimov, M. V. Trusfus, V. V. Mokshin, and A. P. Kirpichnikov, "AnyLogic, ExtendSim and Simulink Overview Comparison of Structural and Simulation modelling Systems," *in Proc. 3rd Russian-Pacific Conf. on Computer Technology and Applications (RPC), Vladivostok,* 2018, pp. 1–5. https://dx.doi.org/10.1109/RPC.2018.8482152

[13] S. W. Cox, "GPSS World: A brief preview," *in 1991 Winter Simulation Conference Proceedings, Phoenix, AZ, USA,* 1991, pp. 59–61. https://dx.doi.org/10.1109/WSC.1991.185591

[14] A. Legrand, M. Quinson, H. Casanova, and K. Fujiwara, "The SIMGRID Project Simulation and Deployment of Distributed Applications," *in 15th IEEE Int. Conf. on High Performance Distributed Computing, Paris,* 2006, pp. 385–386. https://dx.doi.org/10.1109/HPDC.2006.1652196

[15] S. R. Chelladurai, "Gridsim: a flexible simulator for grid integration study," 2017. https://dx.doi.org/10.24124/2017/1375

[16] I. C. Legrand and H. B. Newman, "The MONARC toolset for simulating large network-distributed processing systems," *in Winter Simulation Conf. Proc. (Cat. No.00CH37165), Orlando, FL, USA,* vol.2, 2000, pp. 1794–1801. https://dx.doi.org/10.1109/WSC.2000.899171

[17] D. Klusacek, H. Rudova, "Alea 2: job scheduling simulator," *in SIMUTOOLS ICST,* 2010. https://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8722

[18] W. H. Bell, D. G. Cameron, F. P. Millar, L. Capozza, K. Stockinger, and F. Zini, "Optorsim: A Grid Simulator for Studying Dynamic Data Replication Strategies," *The Int. J. of High Performance Computing Applications,* 17(4), 2003, pp. 403–416. https://dx.doi.org/10.1177/10943420030174005

[19] W. Chen and E. Deelman, "WorkflowSim: A toolkit for simulating scientific workflows in distributed environments," *in 2012 IEEE 8th Int. Conf. on E-Science, Chicago, IL,* 2012, pp. 1–8. https://dx.doi.org/10.1109/eScience.2012.6404430

[20] J. Taheri, A. Zomaya, S. Khan, "Grid Simulation Tools for Job Scheduling and Data File Replication," *in Scalable Computing and Communications: Theory and Practice,* New Jersey: Wiley, 2013, pp. 777–797.

[21] P. F. Dutot, M. Mercier, M. Poquet, O. Richard, "Batsim: a realistic language-independent resources and jobs management systems simulator," *in Job Scheduling Strategies for Parallel Processing,* 2015, pp. 178–197. https://dx.doi.org/10.1007/978-3-319-61756-5_10

[22] G. P. Rodrigo, E. Elmroth, P. Ostberg, L. Ramakrishnan, "ScSF: A Scheduling Simulation Framework," *Lecture Notes in Comp. Science,* vol. 10773, 2017. https://dx.doi.org/10.1007/978-3-319-77398-8_9

[23] N. A. Simakov et al., "A Slurm Simulator: Implementation and Parametric Analysis," *Lecture Notes in Comp. Science,* vol. 10724, 2017. https://dx.doi.org/10.1007/978-3-319-72971-8_10

[24] D. Klusacek, M. Soysal, F. Suter, "Alea — Complex Job Scheduling Simulator," *Lecture Notes in Comp. Science,* vol. 12044, 2019. https://dx.doi.org/10.1007/978-3-030-43222-5_19

[25] C. Galleguillos, Z. Kiziltan, A. Netti et al., "AccaSim: a customizable workload management simulator for job dispatching research in HPC systems," *in Cluster Comput.* vol. 23, 2020, pp. 107–122. https://dx.doi.org/10.1007/s10586-019-02905-5

[26] A. Baranov, P. Telegin, B. Shabanov, D. Lyakhovets, "Measure of Adequacy for the Supercomputer Job Management System Model," *in Proc. of the 2019 Fed. Conf. on Computer Science and Information Systems, ACSIS,* vol. 18, 2019, pp. 423–426. https://dx.doi.org/10.15439/2019F186

DOI: 10.15439/2020F157
ISSN 2300-5963 ACSIS, Vol. 21

Dynamic Clustering Personalization for Recommending Long Tail Items

Diogo Vinícius de Sousa Silva
Federal University of Bahia - UFBA
Computer Science Departament,
Salvador - BA, Brazil
Federal Institute of Maranhão - IFMA
Bacabal - MA, Brazil
Email: diogovss@ufba.br

Frederico Araújo Durão
Federal University of Bahia - UFBA
Computer Science Departament,
Salvador - BA, Brazil
Email: fdurao@ufba.br

***Abstract*—Recommendation strategies are used in several contexts in order to bring potential users closer to products with a strong probability of interest. When recomendations focus on niche items, they are called recommendations in the long tail. In these cases, they also look for less popular items and try to find your target custumer, niche market. This paper proposes a long tail recommendation approach that prioritizes relevance, diversity and popularity of recommended items. For that, a hybrid approach based on two techniques are used. The first is clustering with dynamic parameters that adapt from according to the dataset used and the second is a type of Markov chains for to calculate the distance of interest of a user to an item of relevance for this user. The results show that the techniques used have a better relevance indexes at the same time more diverse and less popular recommendations.**

I. Introduction

RECOMMENDATION System is a computational model that aims to approximate the objects to consumers with real interests in acquiring them. To find these associations, a specific research area arose over the 1990s to focus on a study on personalized recommendation techniques [1]. These technicians use several types of recommendation algorithms to process the user profile informations, items metadata, among other information, to conduct a specific user to a service, product, or any other item that is interesting.

In the 20th century, the entertainment industry was based heavily on hits products, that is, those that achieved great success facing the public consumer. However, at the end of the 20th century and the beginning of the century 21th, with the help of the Internet, the industry starts to consider the niche culture. With the experience of the newly created e-commerce sites (with electronic commerce), it was realized that the revenue from a large number of products with low sales index can be equal to or even greater than the revenue of successful products, as these last are made available in a smaller amount of options [2]. If before, the focus on niche products represented high costs with the Internet, the costs of making products visible to customers were smaller and smaller. Before, the focus was on placing products on the shelves more consumed and save with the logistics necessary for the control of stock. With the Internet it became possible to offer products and services each more and more specific, without having to demand costs with physical infrastructure of a showcase or shelves. Products are exposed through images, texts, video files, representing just a few more bits in the infrastructure computationally with low cost. With that, what was previously expensive to be exposed and offered to the public, became accessible, increasing the amount of visible products for potential consumers.

With the increasing spread of the Internet, and through computers, smartphones, TVs, tablets, access has been growing and niches are appearing more. Niche culture is what brings us to the term long tail. So there are greater amount of products available to the consumer and with niches more and more specific. Today, with the Internet and a greater amount of content available, there is a tendency for users to choose more specific items and make the tail longer and longer. Considering the importance niche groups, which have very specific interests, the problem of recommendations on the long tail opens possibilities for studying techniques that improve the performance of such recommendations. They are not relevant to recommendation, but also in other aspects such as popularity, diversity and hybridization. Due to the fact that long tail items are low popular, the accuracy of these recommendations tends to be less than the items most consumed by most users. Since the long tail items are less popular, they have a lower amount of ratings from other users and consequently provide less inputs for the calculation more accurate recommendation.

The combination of different techniques is also an opportunity to decrease the recommendation problem in the long tail. When done correct, can help to improve the long tail recommendations in several aspects, not limited to just one or two metrics as a measure of evaluation. For this reason, the search for new strategies and also the combination of different techniques already in existence is an important topic to be studied. The main objective of this article is to propose a recommendation model that guide better users to niche items, located on the long tail. Thus, it leads them to a greater diversity and relevance products at the same time. For this, clustering techniques and represents matrices will be explored.

Thus, an hybrid strategy will be presented. Considering hybrid strategies, a common technique for recommends that the long tail negatively affect the accuracy already obtained by another technical. This fact will also be addressed in this work so that the approach hybrid approach can add good relevance to the recommendations, without affecting business actively recommend the diversity of recommended items.

It is important to highlight that this work does not intend to analyze the performance related to the execution time, nor of latency. The object of study is limited to improving item recommendations by exploring the long tail. Thus, metrics were chosen to specifically assess the relevance of recommendations from long tail items.

The remaining sections of this paper are divided as follows: in Section II we present a overview in relation to the state of the art in research on recommends long tail. Section III shows the related works. Section IV the model proposed in this work is shown. Section V explains the evaluation model used, as well as the results of this evaluation. Section VI we discuss the conclusions and future work.

II. Background

When studying niche culture, [2] was the first to coin the term long tail. The use of this term was influenced by a paper published on a mailing list by [3] in which the author describes social behavior from the rise of the Internet. The term long tail is the name of a part of a supplier's entire set of items. This part is formed by items that have a small sales margin, however they have a large amount, representing well over 50% of all the stock. By representing this entire volume of items in a graph, such as shown in Figure 1, it can be observed that the right-most side extends until reach zero level. Observing this characteristic of the graphic the author realized that the curve resembled the long tail distribution curve, as denominated in Statistics. It is also common to use the terms tail heavy and fat tail to reference this type of curve in the area of Statistics. In marketing the term long tail started do be used by [2] as a reference to the niche market.

According to [2], very popular products are generally quite commercial by several other companies and, so competition for sales are great. As there is a great demand for these products, the price tends to be as low as possible due to competition. With that, the profit rate of these products is quite low. With items with low demand, it is possible to define a higher profit margin, because users interested in purchasing them will be more willing to pay a higher price for the low availability of the product. Another benefit of exploring long tail products is the one-stop effect shopping convenience, in which the user has the facility to find several types of products in just one store. A store that offers tail products and popular products offer extra convenience for your customer, since this find is everything you need in one place.

On the other hand, the term short head has also come to be used to reference the left-most end of the graphic, it is possible to observe the Figure 1. In this part of the graphic a small number of items are represented, but they are only great success, making the graphic head narrower and, due to the great higher sales quantity. Thus, it was agreed to use the term long tail to represent niche products with great diversity and low popularity. The term short head started to make the products that are highly successful (mass market) and less successful degree of diversity, since they represent a very small number. The greater number of items, the greater the tail. It is on this tail that culture is established niche. Figure 1 shows a graphic with two axis. The abscissa axis represent the items present in the stock of a particular service or product provider. The ordinate axis represent the demand (purchase or download) of each.

With the new characteristics of this niche market, the means of publication content has started to have a greater demand for filtering content. Due to the growing amount of content available, it was increasingly more necessary a mechanism that would help people to find their preferred contents. Two features would need to be implemented: i) make everything available and; ii) help the users to find what they want/need. This gap started to be filled with the Recommendation Systems with approaches specific for the niche market, i.e. recommendations on the long tail.

III. Related Work

The phenomenon of the long tail has been studied with the main objective of leveraging sales of products less by recommending these in conjunction with more popular items. [2] studies how the long tail phenomenon can influence the future of Business. The author brings a more focused analysis to marketing and puts in evidence the importance of Recommendation Systems as a long tail exploration. Thus, the author is able to show how the phenomenon long tail can be seen as a way to increase the profits of a company. [4] propose new algorithms based on chains of Markov for recommending long tail items. Graphs are used to represent the relationships between users and items and, based on those relationships, calculate the distances of interests between these entities. [5] and [6] apply the techniques of adaptive clustering and spplitting of clusters, respectively, to improve performance and decrease the error rate. The last work continue the previous one. In them, the authors evolve a clustering model by making it adaptive to the cutoff point in the dataset for long tail items and short head items. In addition, the size and quantity of the clusters are calculated according to the dataset to be applied the proposed approach. [7] use probabilistic collaborative filtering to generate recommendations for items long tail. Through probability calculations based on relevance models of items the system recommends less popular products and achieves a higher degree relevance in the recommendations. [8] evaluate the coldstart problem and propose a method to generate recommendations for new users based on a user model called Contextual Conditional Preferences. In their experiments, the authors evaluate accuracy measures as well as serendipity, novelty and diversity and obtain recommendations in cold start situations as good as in non cold start situations. [9] discuss the use of collective clustering for recommendation

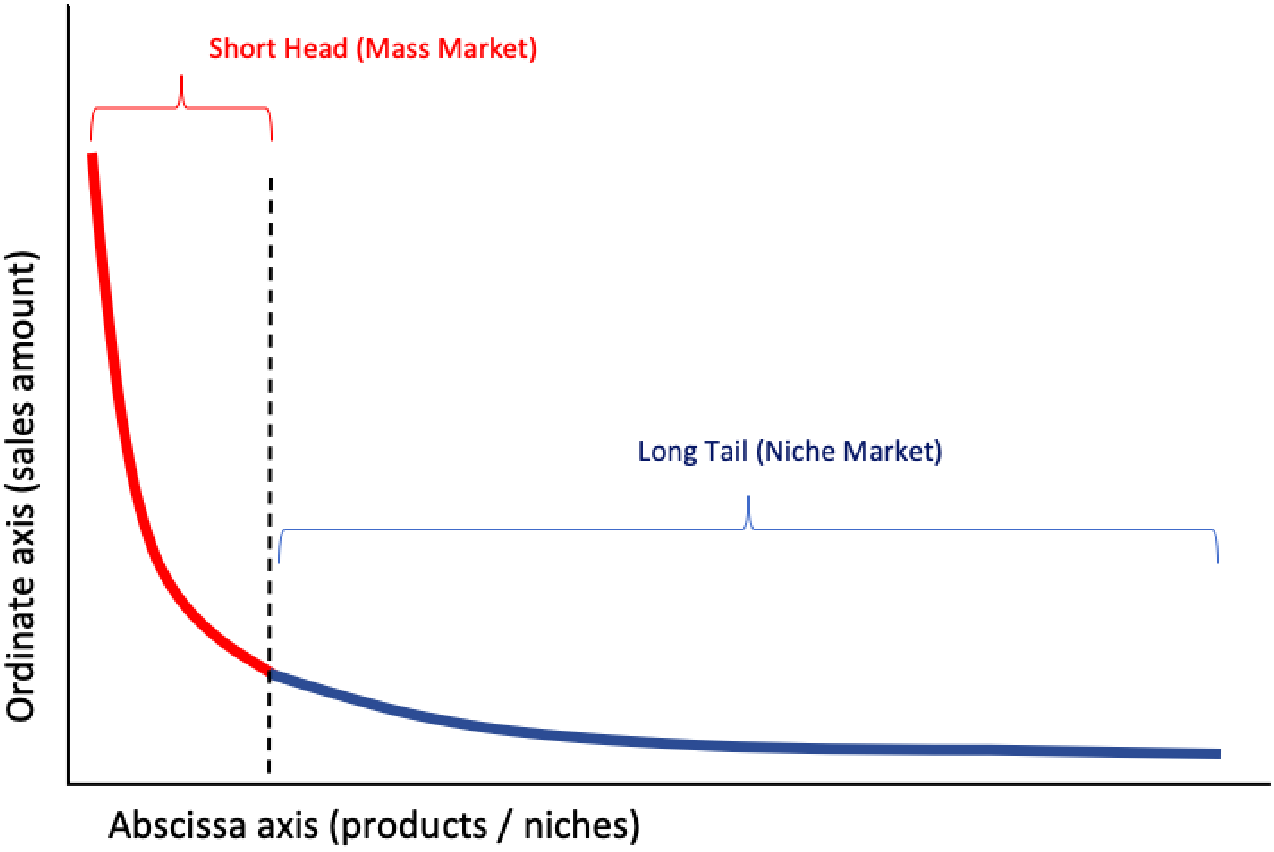


Fig. 1. Representation of the long tail of a hypothetical retailer's inventory relating all products (abscissa axis) to the quantity of sales for each (ordinate axis).

in the context of e-commerce. For this, the authors use data mining combined with RFM techniques. In the work of [10] the authors propose a model hybrid based on the Hitting Time algorithm of the work of [4] and also on the clustering performed in the work of [5]. However, the proposed clustering parameterizes some variables such as number of clusters and the coefficient used to adjust the result obtained in the first technique with Markov chain. With this hybrid approach, the authors perform experiments and presents good results. However, the clustering performed in this work use fixed parameters that may not be the most suitable when used in different datasets.

In most of the works presented above, the technique used is Collaborative Filtering, the most popular for generating recommendations [11]. However, this technique faces a challenge in relation to long tail items. How are items that have a very small amount of classifications by part of the users, the tendency is that these items are little recommended to users. Thus, a relevant item that is in the long tail, has a low chance of being recommended. A simple solution would be identify long tail items, as this task is not costly, and force them to enter the recommendations generated by the Collaborative Filtering. However, this strategy does not guarantee the relevance of recommendations. Recommending long tail items doesn't just mean identifying the items that are present in the long tail. In addition, it is necessary to find relevant items and conduct them to the correct users. At the same time, such recommendations need to be diversified to ensure that each user is actually receiving personalized suggestions. It is also interesting that the recommendation are not popular items, i.e. items that are of interest of few users, and as a consequence generating less obvious recommendations.

IV. Model Description - Personalized-Hitting Time Clustered (P-HTCL)

This section describes the proposed model called P-HTCL (Personalized-Hitting Time Clustered). The model is based on Hitting Time Clustered (HTCL) algorithm [10], which applies clustering and Markov chains to calculate distances of interest between users and items.

A. *Hitting Time Clustered*

For the initial calculation of the distance between users and items, [10] consider the work of [4], in which the dataset with the users' ratings for the items are represented using a graph *user x item*. The edges of the graph are represented by the ratings given by users to the items in the dataset. Representing the graph through an adjacency matrix, the algorithm uses a Markov chain called random walker to traverse through the graph calculating the distance between users and items in the dataset.

Subsequently, [10] apply a clustering technique to generate recommendations more assertive for long tail items. This clustering is applied in the dataset on items classified as long tail. The long tail of the dataset is composed of a set of items that represent the niche market, that is, items with low sales rates, as shown in blue in the graph in Figure 1. In order to classify items in the dataset, the Pareto [12] rule is taken into account. That is, 20% of the most searched items represent 80% of total sales. Thus, the remaining 80% of the items are the least popular and consequently represent the long tail. For the items present in the long tail, the average score of each one is taken into account, i.e. the average of the ratings provided by all users for that item. Using 4 different clusters the items are grouped according to the average score. In our context, a cluster is a set of items grouped according to the average score calculated for each item. In this way, a cluster gathers a subset of items that are present in the long tail. For each of these clusters, a coefficient called the Adjustment Factor (AF) is applied. This factor will conduct the recommendations in order to make them more assertive for long tail items. Thus, the distance between items and users, which was calculated by walking the graph by the random walker, is adjusted according to the following criteria:

- Average Score < 2 — Item allocated to the cluster A and AF = +20%
- Average Score < 3 — Item allocated to the cluster B and AF = +10%
- Average Score < 4 — Item allocated to the cluster C and AF = -10%
- Average Score $<= 5$ — Item allocated to the cluster D and AF = -20%

Under these criteria, there will be a new ordering of the most relevant recommendations according to the new values obtained from the distance between items and users of the dataset.

B. The P-HTCL Algorithm (Personalized Hitting Time Clustered)

The model proposed in the present work performs a calculation to dynamically define the optimal values to be used as AF. The algorithm chooses a set of optimal values based on some dynamic tests. In a sequence, the set of AF values that achieves the best results in the recommendations for long tail items is chosen. Thus, depending on the dataset and the ratings made by users, this same model can use different values for the AF of the clusters. That is, there will be a customization of the AF for each domain, hence the name of the approach: Personalized Hitting Time Clustered (P-HTCL).

In order to find out the optimal values for the AF, the algorithm assess various combinations and compares them to each other. As the intention is to explore recommendations for long tail items, the comparisons is based on the diversity of the recommendations generated with the combination of the AF. For testing the combinations, two sequences are taken into account for the AF values of the clusters. Table I shows the sequence with the values that the algorithm uses to find the optimal AF value for each of the 4 clusters. The first is an exponential sequence S with base 2, i.e. S =[2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6, 2^7....]. Thus, the sequence of values for the AF to be tested in cluster A is S = [1, 2, 4, 8, 16, 32, 64, 128 ...]. The second sequence is applied to cluster B. In this case, the sequence is the set of natural numbers N = [0, 1, 2, 3, 4, 5, 6, 7, 8 ...]. For the values of clusters C and D the same sequences are used as clusters B and A, respectively. However, in these clusters the percentages are negative. Thus, the sequence for cluster C, for example, is N = [0, -1, -2, -3, -4, -5, -6, -7, -8 ...].

Using the sequences of values shown in Table I, a diversity measurement is performed for each test. Each measurement is compared with the results obtained by the previous tests. It is natural that the first tests achieve values closer to the Hitting Time algorithm. This is due to low influence of the AF, since the AF values are still small. Figure 2 shows that at some point there is a peak of diversity. In this illustration, the peak occurs in test number 4. The following values tend to decrease. The peak found then represents the optimal value that the algorithm will use to generate the recommendations.

C. Personalized Hitting Time Clustered Pseudo-Code

The Algorithm 1 represents the implementation of the P-HTCL. To generate the recommendations a graph with users, items and ratings is required. From there, the algorithm will return the top@N set of items recommended for a specific user. Initially, the dataset is prepared to split long tail items (or niche items) from the rest (also called hit items or short head items). Then, 4 clusters are created to receive the long tail items according to the average score of each one (as shown in lines 2 to 6 of Algorithm 1).

After creating the clusters, it is necessary to define the AF value that will be used in the cluster. In line 7, the variables that will control the tested values are initialized. The tests are performed according to the sequence shown in Table I. The tests are described in lines 8 to 14. In this part of the algorithm, diversity measurements are made with each of the sets of AF values.

The obtained diversity results are compared with previous tests until a pattern is found that indicates a peak of diversity for a given AF value In line 17, the distances from the items to the user are calculated using the Hitting Time algorithm. In lines 18 and 19, the AF value obtained is applied. Consequently, in line 21 there is a need for a new sorting to return the most relevant results according to the Personalized - Hitting Time Clustered approach.

TABLE I
SEQUENCE OF VALUES ANALYZED BY THE ALGORITHM P-HTCL IN SEARCH FOR THE OPTIMAL SET OF THE AF.

Cluster	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9	Test 10	Test ...
A	1	2	4	8	16	32	64	128	256	512	...
B	0	1	2	3	4	5	6	7	8	9	...
C	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	...
D	-1	-2	-4	-8	-16	-32	-64	-128	-256	-512	...

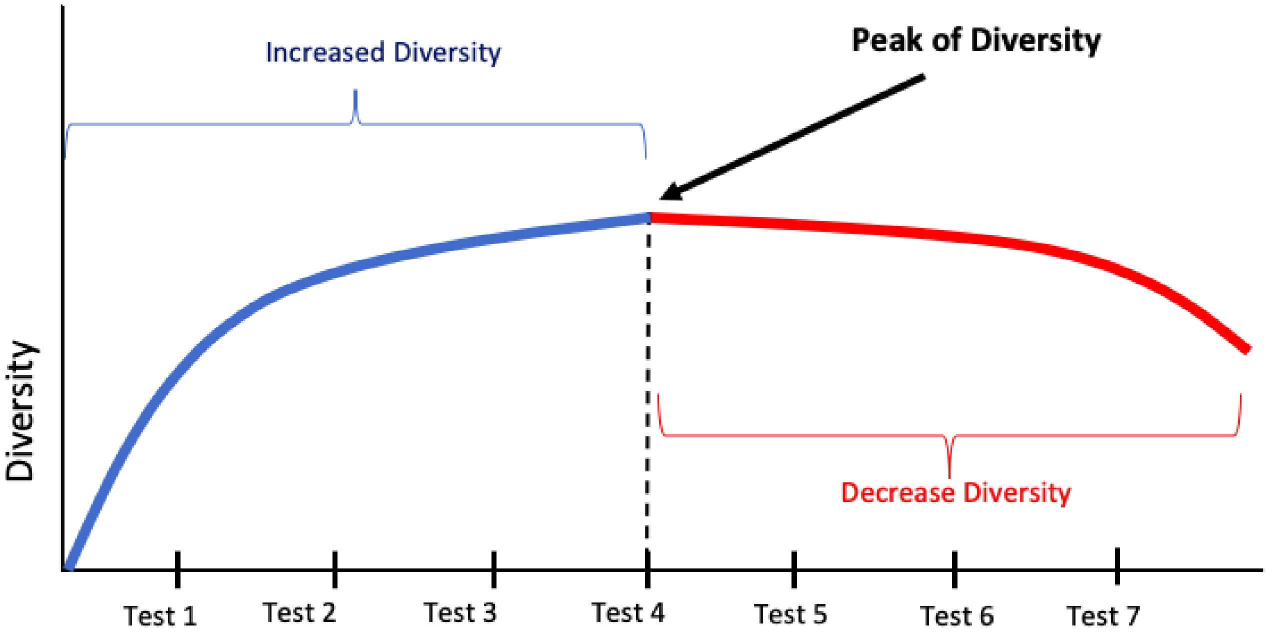


Fig. 2. Graphic that relates the level of diversity of the recommendations with the tests carried out with a set of different values for the AF.

Up to line 14 the algorithm performs an initialization so that the recommendations are computed. This initilization needs to be performed only once. After that, the system will generate recommendations as many times as necessary. Over time, the dataset data will naturally change and a new initialization will be necessary. In this new initialization, it is possible for the AF to change again, since it is calculated dynamically and depends on each dataset. From line 15 our approach will use the initialized data and then the recommendations can be generated. Thus, starting from line 15 this model is possible to run in parallel enironment.

V. Evaluation

This section presents the experimental evaluation of the approach, as well as the comparative results of the proposed model with other baselines.

A. Methodology

We performed offline experiment to evaluate the proposed approach. The dataset is generally divided into two subsets, one for training the recommendation algorithm and the second for testing and evaluating the recommendations generated. The algorithm generate the recommendations based on the set of users who have their evaluation histories already known. The correct ones are computed and then the results of the metrics are obtained. To give more statistical confidence to the experiment, the tests are perform 100 times each. From these data, it is possible to find out if there is a significant difference between the averages found.

To assess of the results of the P-HTCL, we compare it with other two state-of-the-art baselines: i) Hitting Time (HT) - algorithm presented in the work of [4] and ii) Hitting Time Clustered (HTCL) - algorithm presented in the work of [10]. These two algorithms are explained in Section III and Section IV, respectivelly.

The dataset used to measure metrics and analyze the results of all baselines is Movelens [13]. The dataset is a movie domain, with 1,682 titles and 943 users. The total rating is 100,000 with a density of 6.3%, that is, a sparse matrix in which most users have not yet rated most films.

As to the evaluation metrics, according to [14] recall is one of the main metrics used to evaluate Recommendation Systems, and this is used in the evaluation of this work. When we focus on long tail recommendations, it is interesting that the evaluation metrics can reduce the bias generated when recommending short head items. So, to know the performance of a technique for recommending items that have low demand

Algorithm 1: Recommendation using Personalized-Hitting Time Clustered (P-HTCL)

```
Data: Graph with items, users and ratings;
A user and your user-item graph, G(V,E) with
adjacency matrix A;
The amount top@N itens recommendations;
Result:
Split item dataset between long tail and short head
  using Pareto's Rule;
Create 4 clusters for each rating range;
while not at end of itemsLongTailList do
    Compute item score;
    Allocate to their respective cluster;
end
Initialize variable diversity, peak of diversity, sequence;
while diversity $>$ peak of diversity do
    Compute diversity with new adjustment factor;
    if diversity $>$ peak of diversity then
        peak of diversity $<=$ diversity;
    end
    Select next adjustent factor from sequence;
end
Select a list itemList with all items unrated by the user;
while not at end of itemsList do
    Compute hitting time from item to user;
    Apply the respective cluster adjustment factor;
    Add to map user<personalized hitting time
      clustered, item>;
end
Sort the map by personalized hitting time clustered
  asc;
Return the top@N results;
```

by most users. Thus, the diversity and popularity metrics help evaluate recommendation approaches for long tail items [4], and therefore will also be used in this evaluation. The following metrics were considered:

1) Recall: - This metric represents a measure of relevance of the recommendations. In Equation 1 we have the formal representation of the metric. The calculation is based on the ratio between two numbers. The first number represents the relevant items and at the same time recommended. The second one represents the number of items only recommended. In olher hands, the recall focuses on the number of relevant items that have been recommended correctly, considering all relevant items that may be recommended.

$$Recall = \frac{relevants \cap recommended}{relevants} \tag{1}$$

where the number of relevant items is represented by $relevants$ and the number of recommended items is represented by $recommended.$

2) Diversity: - This metric calculates for a given set of users the top@N items to be recommended. To calculate diversity, we check how many repeated items appear once and then calculate the proportion with the total, as shown in Equation 2.

$$Diversity = \frac{\mid \bigcup I_u \in I \mid}{\mid U \mid \cdot top@N} \tag{2}$$

where I_u is the set of unique items recommended for all users. The I element is the dataset set, U represents the set of users and $top@N$ is the recommended number of items for each user, that is, N represents the the number of items that the algorithm will return as recommendation to the user. For example, $top@5$ means that the algorithm will return 5 ordered recommendations from the most relevant to the least.

3) Popularity: - This metric calculates the frequency of a particular item. It is based on the proportion of the number of ratings compared to the other ratings in the dataset [4]. The calculation is based on the average popularity of the items present in the ranking of each user. For each user the average is calculated to find the final value, according to Equation 3.

$$Popularity = \frac{R_u}{\sum \mid R_d \mid} \tag{3}$$

where

$$R_u = \frac{\sum \mid R_r \mid}{\mid U \mid \cdot top@N} \tag{4}$$

where R_d represents the rating set for the entire dataset, U represents the set of users selected for the popularity calculation and $top@N$ is the number of items recommended for each user belonging to the U set. In Equation 4, R_u normalized rates considering the number of users and items recommended for each one of them.

B. Results

To analyze adherence to normality the experiments used the test AD (Anderson Darling). After ensuring that the means have a normal distribution, we use a parametric test called the Test-T. For the comparison of the averages is considered the p-value $<$ 0.05. All results of the P-HTCL approach are compared with the other baselines. In all comparisons, the result obtained for the p-value is less than 0.001. Thus, all the results of the metrics presented below are statistically different.

1) Recall Results: - Figure 3 illustrates the evolution of P-HTCL tests compared to HT and HTCL. In all top@N the approach proposed in this work achieves better results against the other two approaches. Regarding recall, the best performance is achieved in the top@25, when the P-HTCL exceeds HTCL by 148%.

Table II presents the results of the recall metric in each top@N and in each baseline. As highlighted in bold, note that in the top@25 and top@30 the recall of the P-HTCL approach is twice as good as HTCL approach.

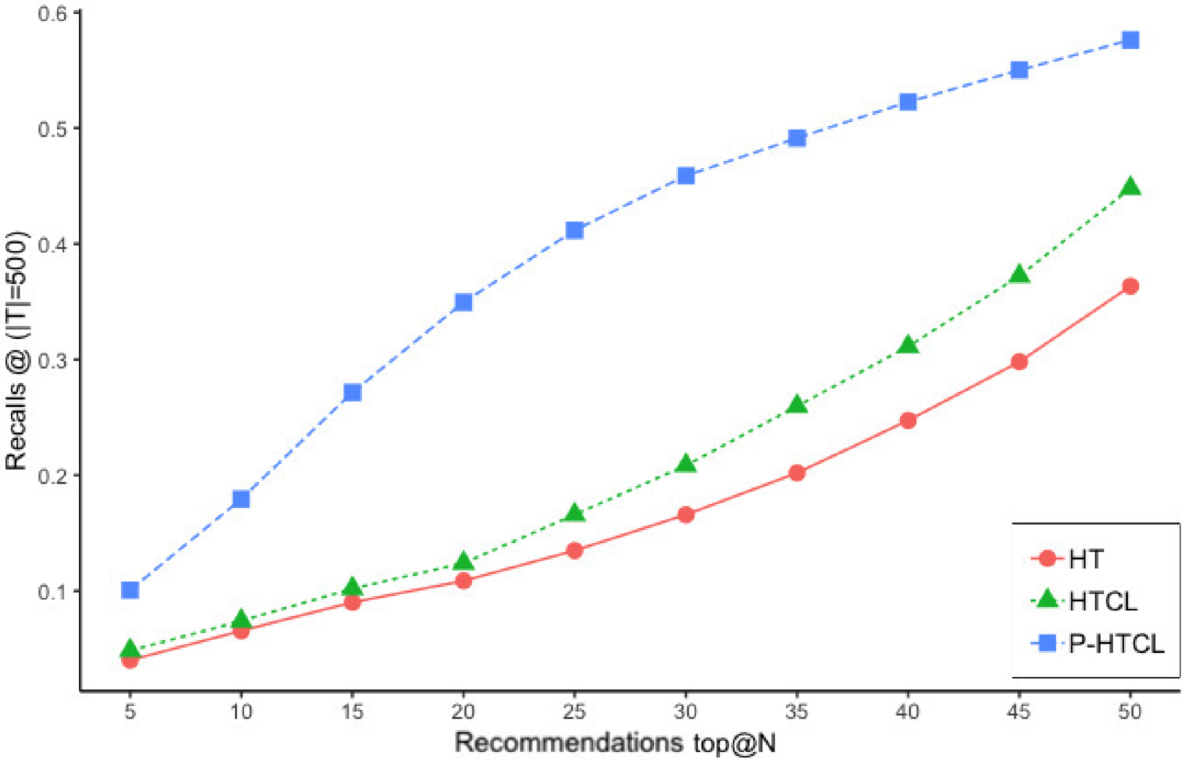


Fig. 3. Recall of the top@N items in 500 test cases.

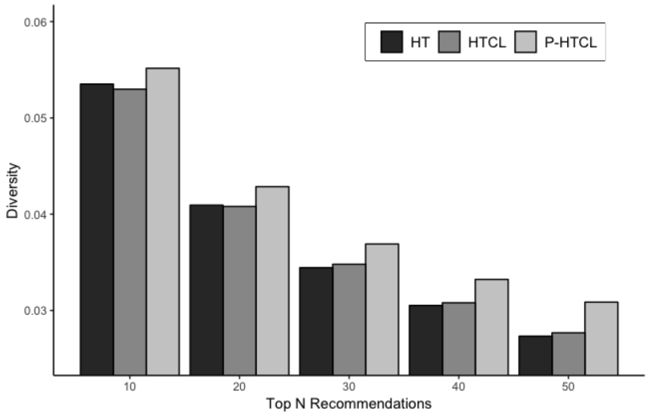


Fig. 4. Results of the diversity metric on Movielens 100k using 200 random users.

2) Diversity Results: - Figure 4 presents the evolution of the three approaches from top@10 recommnedations to top@50. It is possible to observe that from the top@10 the P-HTCL approach always is the best. From the top@20 the difference increases even more, thus showing that it is also the best approach in terms of diversity.

Table III shows the results of the diversity metric of each approach and in all top@N recommendations. Note that the top@50 is the moment when the P-HTCL stands out among the other baselines, in which the diversity of 0.0309 is 11.5% higher than the HTCL, as highlighted in bold in the Table III.

TABLE II
RESULTS OF THE RECALL METRIC FROM TOP@5 TO TOP@50 EXECUTED IN ALL BASELINES: HT (HITTING TIME), HTCL (HITTING TIME CLUSTERED) AND P-HTCL (PERSONALIZED-HITTING TIME CLUSTERED)

	HT	HTCL	P-HTCL
top@05	0,0401	0,0484	0,1007
top@10	0,0656	0,0740	0,1794
top@15	0,0901	0,1019	0,2716
top@20	0,1087	0,1241	0,3495
top@25	0,1350	**0,1658**	**0,4117**
top@30	0,1658	**0,2085**	**0,4588**
top@35	0,2020	0,2596	0,4913
top@40	0,2473	0,3113	0,5226
top@45	0,2981	0,3722	0,5501
top@50	0,3634	0,4479	0,5762

TABLE III
RESULTS OF THE DIVERSITY METRIC FROM EXECUTED IN ALL BASELINES: HT (HITTING TIME), HTCL (HITTING TIME CLUSTERED) AND P-HTCL (PERSONALIZED-HITTING TIME CLUSTERED)

	HT	HTCL	P-HTCL
top@10	0,0535	0,0530	0,0552
top@20	0,0409	0,0408	0,0429
top@30	0,0344	0,0348	0,0369
top@40	0,0305	0,0308	0,0332
top@50	0,0273	**0,0277**	**0,0309**

3) Popularity Results: - Figure 5 shows the evolution of the popularity metric of the items recommended in the top@N. Unlike diversity and recall metrics, the popularity is expected

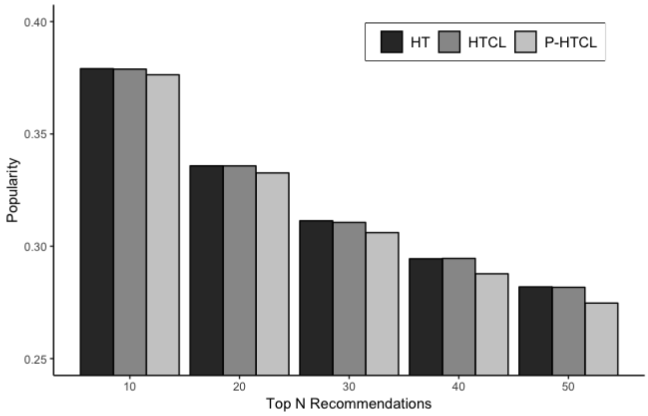


Fig. 5. Results of the popularity metric on Movielens 100k using 500 random users.

TABLE IV
RESULTS OF THE POPULARITY METRIC EXECUTED IN ALL BASELINES: HT (HITTING TIME), HTCL (HITTING TIME CLUSTERED) AND P-HTCL (PERSONALIZED-HITTING TIME CLUSTERED)

	HT	HTCL	P-HTCL
top@10	0,3791	0,3789	0,3763
top@20	0,3358	0,3358	0,3327
top@30	0,3114	0,3106	0,3061
top@40	0,2944	0,2946	0,2878
top@50	0,2819	**0,2817**	**0,2747**

to be as low as possible. In this way, the system recommends less popular items, i.e. more targeted to the niche market. Figure 5 presents that in all the top@N the P-HTCL approach achieves better results than the other baselines.

Table IV presents the average scores. Similarly to the diversity metric, we observe that the greater the top@N the greater the effectiveness of P-HTCL approach. Note that at top@50 we achieve the greatest distance from HTCL with an average of 0.2817, against P-HTCL, which achieves only 0.2747, as highlighted in Table IV.

As a result of the evaluation, improvements have been achieved compared to the HTCL and HT. Recall that the adjustment made was in the automatic configuration of the AF of the clusters used. In HTCL, the 4 clusters are AF: Cluster A = 20; Cluster B = 10; Cluster C = -10; Cluster D = -20. With the customized approach of P-HTCL, various AF are analyzed from measurements of the diversity metric until finding the combination that achieves the best results.

VI. CONCLUSION

This work proposes an hybrid recommendation model to increase diversity, recall and popularity in long tail recommendations. The proposal is an extension of the HTCL model shown in [10]. In this model, the authors propose a graph and cluster based recommendation algorithm. The adjustment factors aim at improving the recommendations on the long tail. These adjustment factors are fixed in the algorithm. In our approach, called P-HTCL, we use a dynamic model to define the value of adjustment factor in runtime. Two rules are used to generate and test variations of the adjustment factors. Then the algorithm tests each one. The first rule is an exponential sequence of order 2 for clusters A and D (the most external), and the second is a numeric sequence natural for clusters B and C (the most internal). In this way, the P-HTCL performs various executions until the results of the diversity metric improve. The objective is to find the maximum point, i.e. the highest peak diversity index.

The tests are performed using Movielens dataset. The results of popularity metric are satisfactory, as it was possible to reduce it by 2,55% percent compared to HTCL on top@50 recomendations. The lower the popularity index of an item, the better for the long tail recommendation, demonstrating that it is a niche item. The opposite happens in the recall and diversity metrics, i.e. the higher the results, the better for the recommendation. As a result, the diversity improved 11,6% over the HTCL method on top@50. The recall improved 148% over the best compared method on top@25. The increase in diversity means that the algorithm recommends a large number of different items. In this way, the algorithm ensures that it does not generate recommendations that are biased in conducting users to the same set of items. In addition, the decrease of the popularity gives us indications that the recommendations are more directly related to long tail items, i.e. the niche items.

The P-HTCL approach has the potential to be used in several contexts, since the P-HTCL does not use any data specific to a particular domain. An great example is in e-commerce. Considering the possibility of increasing sales by exposing more assertively forgotten items, e-commerce companies can benefit from using P-HTCL to recommend items long tail. Video or music streaming companies can also take advantage of our approach, since there is also the positive aspect of exploring the long tail without having to use domain specific data. In addition, it is necessary to carry out more experiments in different domains. The experiments used in our work considered only the Movielens dataset, in which the domain is related to the recommendation of movies. However, for different types of markets, our approach may be applied, such as book recommendation, clothing items, that is, any item that large retailers sell online. In this way, experiments with other datasets are also scheduled as a future work.

One of the limitations of this work is the fixed amount of the number of clusters used, only four so far. We intend to develop an algorithm that will look for the ideal value for the number of clusters to be used. In addition, is important to try other graph based algorithms that focus on improving item recommendations long tail. The work of [15] can be a starting point in this direction. The authors use a tripartite graph that provides more information in the graph structure *user x item*, improving the recommendations.

REFERENCES

[1] G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, June 2005.

[2] C. Anderson, *The Long Tail: Why the Future of Business Is Selling Less of More*. Hyperion, 2006.

[3] C. Shirky, "Power laws, weblogs, and inequality," Economics & Culture, Media & Community, Open Source, 2003, february 08, 2003. [Online]. Available: http://shirky.com/writings/powerlaw_weblog.html

[4] H. Yin, B. Cui, J. Li, J. Yao, and C. Chen, "Challenging the long tail recommendation," *Proc. VLDB Endow.*, vol. 5, no. 9, pp. 896–907, May 2012. [Online]. Available: http://dx.doi.org/10.14778/2311906.2311916

[5] Y.-J. Park and A. Tuzhilin, "The long tail of recommender systems and how to leverage it," in *Proceedings of the 2008 ACM Conference on Recommender Systems*, ser. RecSys '08. New York, NY, USA: ACM, 2008, pp. 11–18. [Online]. Available: http://doi.acm.org/10.1145/1454008.1454012

[6] Y. J. Park, "The adaptive clustering method for the long tail problem of recommender systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 8, pp. 1904–1915, Aug 2013.

[7] D. Valcarce, J. Parapar, and A. Barreiro, "Item-based relevance modelling of recommendations for getting rid of long tail products," *Know.-Based Syst.*, vol. 103, no. C, pp. 41–51, Jul. 2016. [Online]. Available: http://dx.doi.org/10.1016/j.knosys.2016.03.021

[8] A. Karpus, T. di Noia, and K. Goczyła, "Top k recommendations using contextual conditional preferences model," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 11. IEEE, 2017, pp. 19–28. [Online]. Available: http://dx.doi.org/10.15439/2017F258

[9] M. Pondel and J. Korczak, "Collective clustering of marketing data-recommendation system upsaily," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 15. IEEE, 2018, pp. 801–810. [Online]. Available: http://dx.doi.org/10.15439/2018F217

[10] D. V. Silva and F. A. Durao, "A hybrid approach to recommend long tail items," in *Anais Estendidos do XXIV Simpósio Brasileiro de Sistemas Multimidia e Web*. Porto Alegre, RS, Brasil: SBC, 2018, pp. 7–12. [Online]. Available: https://sol.sbc.org.br/index.php/webmedia_estendido/article/view/4049

[11] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender Systems Handbook*. Springer, 2011, pp. 1–35.

[12] K. Yamashita, S. McIntosh, Y. Kamei, A. E. Hassan, and N. Ubayashi, "Revisiting the applicability of the pareto principle to core development teams in open source software projects," in *Proceedings of the 14th International Workshop on Principles of Software Evolution*, ser. IWPSE 2015. New York, NY, USA: ACM, 2015, pp. 46–55. [Online]. Available: http://doi.acm.org/10.1145/2804360.2804366

[13] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 19:1–19:19, Dec. 2015. [Online]. Available: http://doi.acm.org/10.1145/2827872

[14] A. Gunawardana and G. Shani, "A survey of accuracy evaluation metrics of recommendation tasks," *J. Mach. Learn. Res.*, vol. 10, pp. 2935–2962, Dec. 2009. [Online]. Available: http://dl.acm.org/citation.cfm?id=1577069.1755883

[15] J. Johnson and Y.-K. Ng, "Enhancing long tail item recommendations using tripartite graphs and markov process," in *Proceedings of the International Conference on Web Intelligence*, ser. WI '17. New York, NY, USA: ACM, 2017, pp. 761–768. [Online]. Available: http://doi.acm.org/10.1145/3106426.3106439

Future Graduate Salaries Prediction Model Based On Recurrent Neural Network

Jakub Siłka, Michał Wieczorek and Marcin Woźniak
Faculty of Applied Mathematics
Silesian University of Technology
Kaszubska 23, 44-100 Gliwice, POLAND
kubasilka@gmail.com, michal_wieczorek@hotmail.com, marcin.wozniak@polsl.pl

***Abstract*—Prediction models are widely applied in several fields. In this study we present a discussion on using Recurrent Neural Network as predictor for salaries of future graduates. The model is based on feature analysis which leads to input values of the predictor. We have analyzed several compositions and ideas. As a result we have selected Recurrent Neural Network to be the most accurate. Presented results confirm this selection and show high precision.**

***Index Terms*—Recurrent Neural Networks, Prediction model.**

I. Introduction

PREDICTION models are used in various fields to help on future trends estimation. We can find many places where analysis of potential outcomes for the future is assumed as a key factor. One of such topics is salary prediction. We can find many discussions and proposed models which analyse financial factors influencing level of the financial outcome for various professions, countries, ages, etc. Our model is based on Recurrent Neural Network to precisely fit the prediction by the applied structure and proposed modified gradient descent training algorithm. Results, compared for various methods, show that our proposition has much advance.

In [1] was presented an economical approach with considering social aspects to analyze salary ranging from higher education level. The model considered many features which may result in future fluctuation of the salary. In [2] the aspects of salary level related to academic roots was considered for european universities. Authors compared several of them to select the best possible fields of studies in european society. Another aspects widely considered in research on salary fluctuation is race. As presented in [3] the origin of workers may have an active influence on the level of their salary. Similarly, differences in the salary may be visible if we compare woman and man, a study in this field for academic libraries workers was presented in [4]. There are also several aspects which should be analyzed for different countries. In [5] we can read about historical analysis over work positions in India. There are many methods which are used in such studies, ie. [6] proposed deep learning approach model with Convolutional Neural Networks. In all analysis we can find social aspects, economic features and mathematical models. The best option is to model such approach with the most flexible but at the same time precise approach.

In our study we have decided to use methods of machine learning. There are several models which found application to prediction systems. However recently many studies report Recurrent Neural Networks as architectures of advances over classical ones. A comprehensive study with several findings about optimal settings was presented in [7]. Recurrent Neural Networks (RNN) are very efficient in text analysis, voice comparison, technical systems, medicine, etc. In [8] a type of long-short term memory model was implemented for acoustic data prediction. In [9] such structures were used for text recognition in chinese language. Medical signals of epileptic symptoms were classified by RNN as reported in [10], while in [11] these architectures helped in modeling the quality of coal fuel for power stations. In any case, the structure which uses machine learning must be trained. The second part of our study over salary prediction is devoted to best training algorithm. We have examined some variations of gradient descent approaches in our research. Such training algorithms are reported in many research, therefore we decided to examine them in our topic. In [12] was presented a study for possible training by the use of gradient descent methods. In [13] the study concentrated on devoted versions for deep learning was presented. The general model for using such algorithm in training of neural networks was defined in [14], while association rules and their relation to the training efficiency was discussed in [15]. Another aspect of efficient training is convergence analysis to the optimum while preserving the step size. The model for efficient adaptive approach was discussed in [16].

Our approach is based in RNN, which is modeled to learn form the input data about possible results from abilities of graduates related to their education, origin, experience and other social aspects. Proposed model was trained by some selected algorithms sourced in the idea of using gradient descent of the network error function in the model of weight corrections. We have performed a comparative analysis to compare efficiency of the training methods and final classification. Results are compared in charts and discussed to draw conclusions. The novelty of our approach is sourced in flexibility of our model, which is able to predict the future result from multi type input set. Additionally the structure composed for our research is easily trained and results show very good measures for many applied algorithms. We have also modified training algorithm to better fit the input values, what is visible in increased results for such option.

II. Neural Network Architecture

1) Recurrent Neural Network: We have done tests on different Neural Network architectures but the one that had the best accuracy was a Recurrent Neural Network using LSTM layers. Normally RNN are used for timestep data but in our case it also strongly improved accuracy. In our model we have used two activation functions:

- hiperbolic tangent - for all hidden layers,
- softmax - for the output layer.

2) Recurrent layers: In our model we have used Long Short-Term Memory (LSTM) layers with a forget gate. Mathematical functions used in the model are

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \tag{1}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{2}$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{3}$$

$$\hat{c}_t = \sigma_h(W_c x_t + U_c h_{t-1} + b_c) \tag{4}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \hat{c}_t \tag{5}$$

$$h_t = o_t \circ \sigma_h(c_t) \tag{6}$$

where $c_0 = 0$ and $h_0 = 0$ for x_t - input vector to the LSTM unit, f_t - forget gate's activation vector, i_t - input/update gate's activation vector, o_t - output gate's activation vector, h_t - hidden state vector, $\tilde{c}_t$ - cell input activation vector, c_t - cell state vector, W,U,b - weight matrices and bias vector parameters, σ_g - sigmoid function, σ_c - hyperbolic tangent function, σ_h - hyperbolic tangent function.

3) System training: To improve our model and speed up the training process we have used an Adaptive Moment Estimation Algorithm - NAdam. This method is computationally efficient and does not require huge amount of memory so it's widely used in Machine Learning research. To improve accuracy even more and shorten training time we have used learning rate decay to firstly make bigger steps during training but after that in our approach the network is making small steps to polish the model accuracy.
Adam formula can be described as follows:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t, \tag{7}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, \tag{8}$$

where g is the current gradient value of error function for the training and β parameters are constant values called hyperparameters. Values m_t and t_m are used for calculation of the correlations marked as $\hat{m}_t$ and $\hat{v}_t$ according to these equations:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{9}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}. \tag{10}$$

Using above calculated correlations, the final formula for changing weights in our neural network can be defined as follows:

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t, \tag{11}$$

where ϵ is a constant small value and η is a learning rate (in this case we have used learning rate of 0.0003).
We need to apply NAG to Adam using these equations:

$$w_t = w_{t-1} - \eta \frac{\beta_1 m_{t-1}}{\beta_2 v_{t-1} + (1 - \beta_2) g_t^2 + \epsilon} \tag{12}$$

$$-\eta \frac{(1 - \beta_1) g_t}{\sqrt{\beta_2 n_{t-1} + (1 - \beta_2) g_t^2 + \epsilon}}$$

Now we modify Adam's update rule. The first term no longer depends on g_t so we need to change expressions for $\hat{m}_t$ and w_t:

$$\hat{m}_t = (1 - \beta_1) g_t + \beta_{1_{t+1}} m_t \tag{13}$$

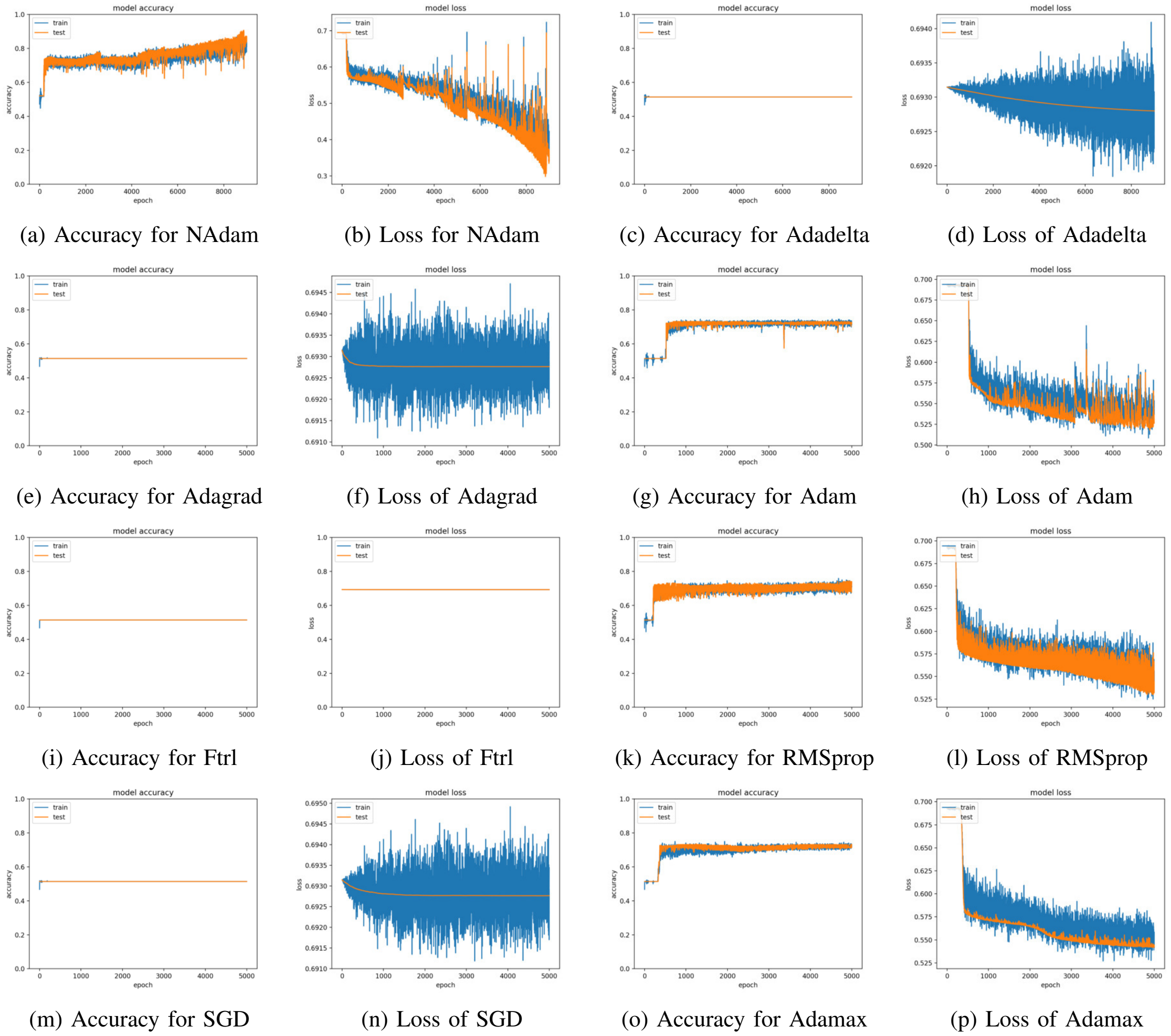
$$w_t = w_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\beta_{2_t}} + \epsilon} \tag{14}$$

III. Dataset

The dataset for our research experiments was downloaded from kaggle.com. Initially it contained data on the education history of individual persons as well as their annual salary on the day the database was created. Some attributes, however, have been removed for processing. We removed id due to irrelevance in entering the neural network. We removed also column stating whether someone is employed or not because in our opinion this will allow the network to be used in more flexible contexts. That's why what's left is:

Gender, Secondary Education percentage- 10th Grade, Board of Secondary Grade Education, Higher Secondary Education percentage- 12th Grade, Board of Higher Secondary Education, Specialization in Higher Secondary Education, Degree Percentage, Under Graduation(Degree type)- Field of degree education, Work Experience, Employability test percentage (conducted by college), Post Graduation(MBA)- Specialization, MBA percentage, salary.

All this information was provided by graduates of Jain University Bangalore, India. According to the 4icu.org website, this university is on the 73rd place in the national classification, despite the fact that those who did not find employment and who earn less than 250k per year were in the dataset. We have established that our network will recognize this threshold. However, not only the assessments affect future employment, it is possible, among others, for two people with the same grades but different personalities, and therefore one of them will not undergo the recruitment process or negotiate a significantly lower salary.



(a) Accuracy for NAdam (b) Loss for NAdam (c) Accuracy for Adadelta (d) Loss of Adadelta

(e) Accuracy for Adagrad (f) Loss of Adagrad (g) Accuracy for Adam (h) Loss of Adam

(i) Accuracy for Ftrl (j) Loss of Ftrl (k) Accuracy for RMSprop (l) Loss of RMSprop

(m) Accuracy for SGD (n) Loss of SGD (o) Accuracy for Adamax (p) Loss of Adamax

Fig. 1: Results of training using different optimization algorithms. In all charts we can see measures of accuracy and loss for the process of training.

IV. Searching for the best optimization algorithm

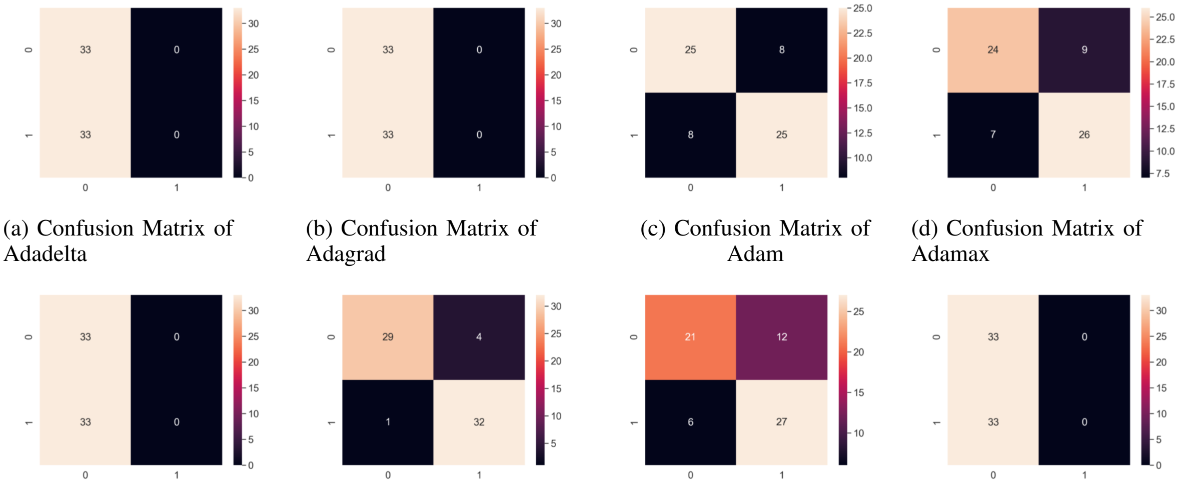
Because we wanted the best accuracy of the RNN model, we have tested different learning algorithms. Some of them were better, some worse and in few cases the network did not train at all. The comparison of accuracy and loss changes during training is presented in Fig. 1. After deep analysis these are the conclusions:

- SGD - Even with double number of iterations the network categorized all values as ”less then 250k”.
- RMSprop - The network reached around 72.73% final accuracy.
- Adam - Network was learning but even with increased iterations maximal reached accuracy was under 75.76% and 73.14% for training data.
- Adadelta - All values were categorized as ”less then 250k”
- Adagrad - All values were categorized as ”less then 250k”
- Adamax - Network was learning and reached about 75.76% for test data and around 73.61% for training data. Because of that it is the second best option, however it is still a lot behind NAdam which reached around 92.42%.
- NAdam - Using NAdam gave the best results with less iterations needed.
- Ftrl - All values were categorized as ”less then 250k”

Results of our initial tests on categorization are shown in Fig. 2. In each experiment we have applied classical division for training and test data 70:30.

V. Conclusions and Future Works

Our solution helps to understand the future and capabilities of graduates, it allows for this thanks to a well-chosen neural network training algorithm, and that’s why we obtained about 92.42% final accuracy in a given dataset. As a result we can find out above all whether we should work more to improve our results through education to have opportunities in the



(a) Confusion Matrix of Adadelta

(b) Confusion Matrix of Adagrad

(c) Confusion Matrix of Adam

(d) Confusion Matrix of Adamax

(e) Confusion Matrix of Ftrl

(f) Confusion Matrix of NAdam

(g) Confusion Matrix of RMSprop

(h) Confusion Matrix of SGD

Fig. 2: Resulting confusion matrices using different optimization algorithms.

future for decent working conditions. However, it should be remembered that many factors influence the future of a given person. Despite this, behavior during your study may answer the question of what personality a person has and thus predict the fortune. In the future, we aim to further develop the project in such way that it can be determined at a young age whether a person will succeed.

In this research we have evaluated eight different training models sourced in gradient descent optimization. The results have shown that the training model of the applied RNN architecture must be also fitted to the data. Some of the methods gave very bad results and some of the methods very good, even if all of them are very similar in assumptions. Therefore in the future research we will concentrate on defining relations between input data and training procedure to achieve the best final result of classification.

References

[1] L. L. Taylor, J. N. Lahey, M. I. Beck, and J. E. Froyd, "How to do a salary equity study: With an illustrative example from higher education," *Public Personnel Management*, vol. 49, no. 1, pp. 57–82, 2020.

[2] M. Kwiek, "Academic top earners. research productivity, prestige generation, and salary patterns in european universities," *Science and Public Policy*, vol. 45, no. 1, pp. 1–13, 2018.

[3] M. Hernandez, D. R. Avery, S. D. Volpone, and C. R. Kaiser, "Bargaining while black: The role of race in salary negotiations." *Journal of Applied Psychology*, vol. 104, no. 4, p. 581, 2019.

[4] E. Silva and Q. Galbraith, "Salary negotiation patterns between women and men in academic libraries," *College & research libraries*, vol. 79, no. 3, p. 324, 2018.

[5] A. Pawha and D. Kamthania, "Quantitative analysis of historical data for prediction of job salary in india-a case study," *Journal of Statistics and Management Systems*, vol. 22, no. 2, pp. 187–198, 2019.

[6] M. He, D. Shen, Y. Zhu, R. He, T. Wang, and Z. Zhang, "Career trajectory prediction based on cnn," in *2019 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*. IEEE, 2019, pp. 22–26.

[7] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," *arXiv preprint arXiv:1312.6026*, 2013.

[8] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," 2014.

[9] J. Wang and Z. Cao, "Chinese text sentiment analysis using lstm network based on l2 and nadam," in *2017 IEEE 17th International Conference on Communication Technology (ICCT)*. IEEE, 2017, pp. 1891–1895.

[10] A. Petrosian, D. Prokhorov, R. Homan, R. Dasheiff, and D. Wunsch II, "Recurrent neural network based prediction of epileptic seizures in intra-and extracranial eeg," *Neurocomputing*, vol. 30, no. 1-4, pp. 201–218, 2000.

[11] H. Yao, H. Vuthaluru, M. Tade, and D. Djukanovic, "Artificial neural network-based prediction of hydrogen content of coal in power station boilers," *Fuel*, vol. 84, no. 12-13, pp. 1535–1542, 2005.

[12] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[13] J. Zhang, "Gradient descent based optimization algorithms for deep learning models training," *arXiv preprint arXiv:1903.03614*, 2019.

[14] D. Soydaner, "A comparison of optimization algorithms for deep learning," *International Journal of Pattern Recognition and Artificial Intelligence*, p. 2052013, 2020.

[15] N. Rajesh and A. A. L. Selvakumar, "Association rules and deep learning for cryptographic algorithm in privacy preserving data mining," *Cluster Computing*, vol. 22, no. 1, pp. 119–131, 2019.

[16] A. Barakat and P. Bianchi, "Convergence analysis of a momentum algorithm with adaptive step size for non convex optimization," *arXiv preprint arXiv:1911.07596*, 2019.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 431–434 DOI: 10.15439/2020F72 ISSN 2300-5963 ACSIS, Vol. 21

Automatic beams detection used for LiFi car-2-car communication

Cosmin Stoica Spahiu,
Computer Science and Information Technology, University of Craiova, Romania
Email: cosmin.spahiu@edu.ucv.ro

Abagiu Marian
Computer Science and Information Technology, University of Craiova, Romania
Email: marian.abagiu@gmail.com

Liana Stanescu
Computer Science and Information Technology, University of Craiova, Romania
Email: liana.stanescu@edu.ucv.ro.

***Abstract*—The car-2-car communication feature is a must for the next generation of cars. Such a functionality will increase the road safety by enhancing the drivers awareness about road conditions and eventual obstacles. It can also be used to inform the cars coming from behind about risky situations in order to trigger early avoidance actions (e.g. breaking). The paper proposes an alternate solution to the WIFI technology which is the most studied for this purpose. The proposed solution is based on Light Fidelity (LiFi) and analyse the feasibility of such a system, from image processing point of view during automatic beams detection. The solution consists of two parts: the information sender, which is the car's LED beams, and the information receiver which is an onboard camera running a special software. This solution has the advantage of using already available car's hardware with small adaption, and only integrate the new software.**

I. Introduction

THE car-2-car communication is a functionality highly researched in the last years due to the multiple benefits that might bring, and several solutions have been proposed in this direction.

The direct communication between cars will increase road safety and reduce the traffic incidents. Based on this functionality, each car will receive enough information in advance to adapt the driving based on the surroundings, in a matter of milliseconds. The drivers can be informed about hazard situations, even before they are visible on the road.

This functionality is currently in the research phase and several solutions have been proposed. However, none of them was generally adopted by OEM producers in order to be effective on large scale. To reach this goal it is needed a standard agreed by all cars manufacturers.

The "car-2-car.org" consortium was founded, aiming to define such a solution generally applicable, where all main car manufacturers participate as active members [1]. The CAR 2 CAR Communication Consortium (C2C-CC) aims at assisting towards accident free traffic (vision zero) at the earliest possible date. The consortium does not offer on-the shelves solutions but only a standard and guidance in the implementation of a common solution that is robust and reliable enough to be used in real life traffic.

There were proposed several solutions in this direction. Among them, the WI-FI communication is widely spread, as it has been proved to be a mature enough technology and very reliable.

One of the first implementations in this direction is Volkswagen's Local Hazard Warning (LHW) [2] which is a system that uses short-range communication between cars. For example, when a car is equipped with the LHW feature it might issue a warning to other vehicles if it had broken down in the middle the highway or had been involved in a collision.

The technology used in this implementation is based on an automotive optimized variant of WLAN technology known as "ETSI ITS-G5". A car who is equipped with this technology will always be informed on such situations in advance, and it is not needed to rely anymore on GSM internet network [2].

Fig 1. Wi-Fi solution in car-2-car communication

The WiFi solution has some drawbacks which must be overcome before this standard will be used on large scale. One of the most important is that each vehicle creates a virtual space around it where information can be exchanged (the WiFi range). There is a high volume of information which needs to be manipulated and filtered by each vehicle. This electromagnetic „pollution" can highly increase in the crowded areas, and the possible interferences can decrease the communication performances.

The second main drawback is the security for safe data access, which is harder to be ensured in open wi-fi communication systems.

The current paper proposes an original solution which is based on the LiFi technology. This solution overcomes the drawbacks presented above and have several additional advantages: provides high transmission rates and it's performances are not decreasing in crowded areas [3].

The paper is structured as following: Section 2 presents the State-Of-the art in LiFi technology, Section 3 describes the proposed solution, Section 4 presents the experiments and Section 5 presents the Conclusion of the paper.

II. Related Work

The paper considers an original solution that can be used for car-2-car communication, based on LiFi technology. The solution can be integrated in the already available cars lightning systems. Using this type of system, the cars can communicate each-other by using their beams which are pulsing in such manner that can transmit information. The receiver for this transmission is an on-board camera which is able to detect beams from front-coming cars and read the sent information [3][11].

The feasibility study for this solution can be split in two parts:

- analyse the performances of the LiFi technology
- analyse the implementations of the existing technologies for cars beams management.

A. LiFi on the market

The LiFi (Light Fidelity) technology is based on the Visible Light Communications transmitting wireless internet data at very high speeds using only light beams. The LiFi technology was proposed initially by the German physicist Harald Haas during the Technology, Entertainment and Design Global Talk Conference (2011), after the invention of LED bulbs [5].

A LiFi network is very simple: it is needed only a light emitter on one end (i.e. an LED transmitter), and a photo sensitive detector (light sensor) on the other end. The data input to the LED transmitter is encoded into the light by varying the flickering rate (PWM) at which the LEDs turns “on” and “off” in order to generate different strings of bits of 1s and 0s.

The LED intensity is modulated so rapidly that human eye cannot notice, so the light of the LED appears constant to humans. In fact, the current LED lights available for room lightning, or cars beams already use a PWM pulse to maintain the light intensity constant ignoring the voltage drops.

These types of systems are already available on the market offering gigabit speed for data transmission, but their applicability is currently limited to internet communication in closed environments. Even if the speed is above WiFi capabilities, they are sensitive to external noise (light) and therefore it is hard to be used in open environments. This can be considered an advantage in certain situations, as the network’s security is more reliable and less liable on hacking.

There are several manufacturers on the market which offers complete solutions[6][8] and some universities have already adopted LiFi communications for their intranet [7].

B. Automatic High Beam Management Systems

The second question to which this paper tries to answer is if LiFi technology is feasible to be extended for car-2-car communication and what impact will have to the already existing cars architecture.

The current generation of high-end cars have solutions based on onboard cameras for automatically managing high-beams, called high-beams automatic assistants. The assistants recognize the oncoming vehicles during night, trace their direction and switches the headlights automatically from high beam to low beam when cars are approaching. This feature is very useful during night driving, making the activity much less stressful for the driver.

The intelligent headlight control is composed by an onboard camera for data acquisition and special image processing algorithms for measuring the ambient brightness and estimating the distance from vehicles placed in front and the oncoming traffic [9][10][11].

Fig 2. Sample multi-purpose camera used for High Beam assistance

The headlamps of the car are controlled by an electronic unit, based on the decisions taken after the captured images are processed. The road users who are in the beam range will be automatically excluded from the light distribution of the beam and put in a dark zone.

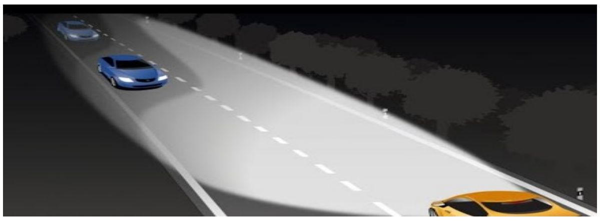


Fig 3. Light tunnel created to protect detected vehicles

Depending on the performances of the beamer, the high beam can be either turned off completely, or it can be dynamically configured to produce a “light tunnel” where the upcoming detected vehicle remains in the dark (e.g. LED matrix beamer)[10].

The solution considered in the paper is an adaption of the already existing technology, where the biggest change is in the software processing and the hardware changes are minimal (LED beams are already controlled via PWM, onboard camera for high-beams management is already available).

III. Solution Overview

The feasibility of using a camera for car-2-car communication is presented in [11]. In the paper it is analysed the camera minimum performances needed to record communication from up-front coming cars. The conclusion of the paper was that a minimum frequency of 100fps is needed.

If a lower frequency would be used, the camera will not be able to record the communication in due time and with an acceptable error rate.

Although the current technology allows a top framerate of more than 1000fps for a camera, the capability is limited by the images processing speed needed to detect beams and read data.

A. Beams Detection

The solution implemented in this paper is based on the Open CV. This is an open source computer vision and machine learning software library. It has more than 2500 basic and state-of-the-art computer vision and machine learning algorithms. It is cross platform with interfaces for C++, Java, Python and MATLAB, runs on Windows, Linux, Android and Mac OS. OpenCV is mostly used for real-time vision applications.

The image processing algorithm from Open CV library was extended with a Python wrapper that allows detection of beams lights.

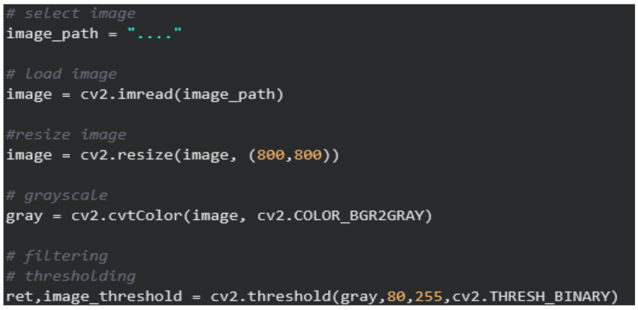
For detecting if the headlights are on or off, a simple approach is used. The continuous video data acquisition is split in sequential image frames which are independently processed, basically a continuous frame grabbing and processing. The algorithm performed for each image consists on several steps performed in loop: each image is transformed in a grayscale image; a thresholding function is applied afterwards to obtain a binary image. On this modified image is applied an edge detection or an area measurement. Based on the surface generated by the ON headlights the software can generate a distance using a scaling function applied on the continuous measurement of the image stream.

Once the beams are detected, further algorithms will be applied for monitoring and reading the data received from the beam light similar as in [3].

The advantage of this algorithm is that the processing time is reduced, and it can be applied during real-time acquisition process.

All the above functions, except the scaling of the image are by default implemented in Open CV. The needed adaption refers only to the threshold parameters which needs to be calibrated, depending on external environment conditions (e.g. fog light, day, night, snow, rain etc.)

Below, an example of the algorithm use can be observed.



```
# select image
image_path = "...."

# load image
image = cv2.imread(image_path)

#resize image
image = cv2.resize(image, (800,800))

# grayscale
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

# filtering
# thresholding
ret,image_threshold = cv2.threshold(gray,80,255,cv2.THRESH_BINARY)
```

Fig 4. Python wrapper used for Image processing

When no beams are detected in the image, then the resulted image is completely black, those no size for the region.

B. Experiments

The scope of the experiments performed was to determine the accuracy of the algorithm for detecting beam lights in images, and to measure the processing time needed for this detection. This will directly influence the maximum frequency of the acquisition camera accepted by the system.

During testing it was used an images database which present cars having the beams on in different environment conditions: day, night, fog.

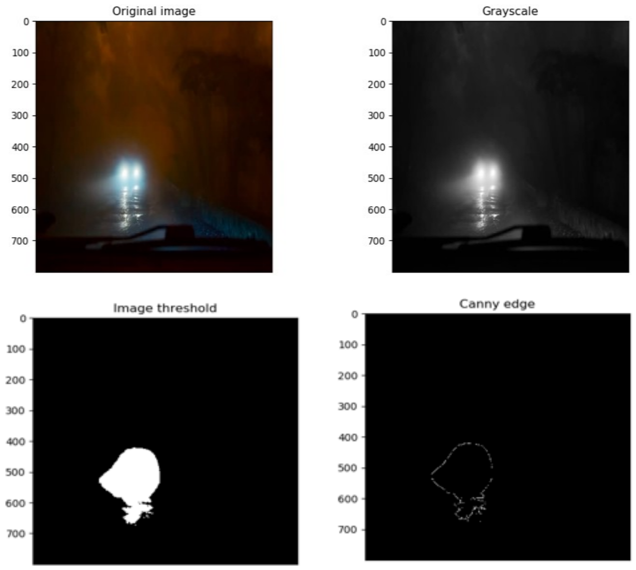


Fig 5. Image processing example

For the sets of images considered, the algorithm successfully detected the beams in images with a precision of more than 90% during night, and approx. 50% during daytime.

Fig 6 presents the results obtained during region processing for the beams detection in different environments.

It was observed that the algorithm used obtains the best results during night, with low traffic conditions.

In order to increase the performances of the system, a multi-layer application should be considered, based on machine learning algorithms.

- the first layer only detects the general weather condition and, based on this, it is chosen which algorithm to be executed next for image processing
- the second layer performs beams detection using the algorithm decided during the execution of step 1.

The second step of the experiment is to measure the processing time needed for one snapshot to detect the beams.

A full image processing operation is measured to be executed in average in 30 ms on a PC, considering a 800x800 pixels image.

Depending on the image resolution and environment data, the image processing times varies from 20 ms for a resolution of 600x600 up to more than 60 ms for a high traffic image.

No noticeable influence was observed by varying the image resolution.

Table 1 presents the average processing time measured for different images taken in different weather conditions and with different resolutions.

TABLE 1: Processing time measured

Time (ms)	20	29	62	32	45
Resolution	600x600	800x800	800x800	800x800	1200x1200
Environment	Night, multiple cars	night, single car	Night multiple cars	Fog, single car	night, single car

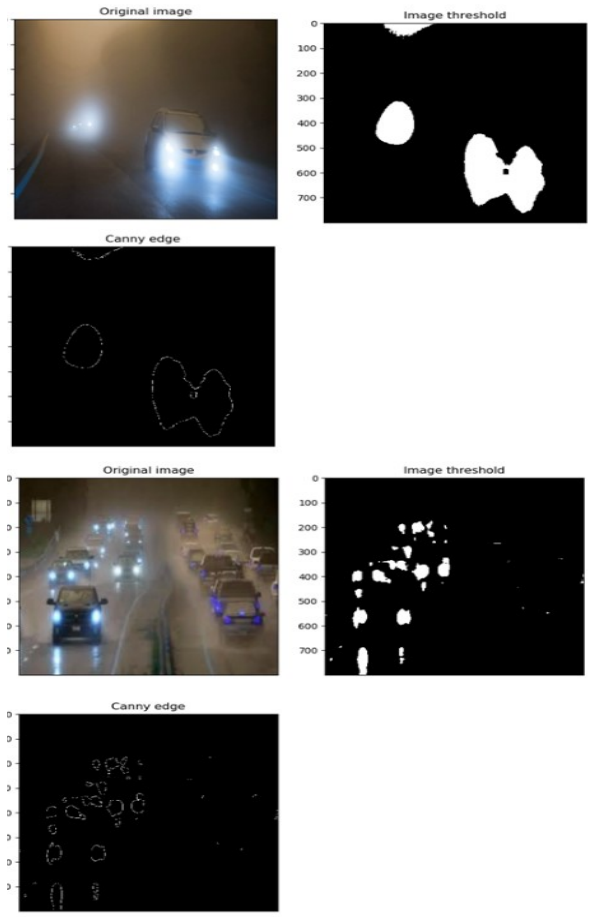


Fig 6. Image processing example

C. *System Improvements*

As it can be observed in the previous chapter, the average processing speed for the implemented algorithm is 40 ms for a 800x800 image resolution. That means the beams detection capability of the system will be of only 25 snapshots /sec.

Considering the results obtained in [11] it is needed a minimum framerate of 100fps in order that this technology to be feasible to be applied in real world.

Although at the first sight the two results are contradicting each other, this is not totally true. The detection algorithm is executed in loop only until on-coming beams are detected. Once they are detected, a second algorithm is started to trace beams direction, to read the flickering of beams and transform this into data. That means there is no need that the detection algorithm to be executed on a similar speed as camera acquisition speed.

The algorithm that needs to be implemented has 4 steps:

1. decide the beams detection algorithm that will be executed based on weather conditions
2. execute the beams detection algorithm decided on step 1 in loop until on-coming beams are detected in the camera visible range. The scope is to detect the region of the image where the beams are located
3. once the region is detected, the third step of the algorithm is started to trace the beams direction while there are in the visible range of the camera
4. the last step of the algorithm will read the flickering of the up-front vehicle beams and convert this into data

IV. Conclusion

The paper presented an original solution that can be implemented for car-2-car communication based on LiFi technology and analyse the feasibility of such a system from images processing point of view.

The system is using the new generation of car illumination systems based on LED technology to encode and send the information to the other cars, and an onboard camera capable to detect beams from front-coming cars and record the sent data.

The on/off activity of the LED transmitter is totally invisible to human eye which do not perceive any flickering if a minimum frequency is used.

The demo application presented in this paper is based on the Open CV library for image processing and obtained good results in term of reliability and processing time.

However, in order to be adapted for an embedded system it will be needed an application based on C++ interface of Open CV with different improvements for the runtime and application-oriented optimization.

References

[1] Car 2 Car Communication Consortium https://www.car-2-car.org/index.php?id=5

[2] https://www.car-2-car.org/fileadmin/press/pdf/volkswagen-local-hazard-warning.pdf

[3] C. Stoica Spahiu, L. Stanescu and M. Brezovan, "Improving driver warnings accuracy using low-cost sensors," 2018 19th International Carpathian Control Conference (ICCC), Szilvasvarad, 2018, pp. 377-382, doi: 10.1109/CarpathianCC.2018.8399659.

[4] S. Eichler, C. Schroth, J. Eberspäche, "Car-to-Car Communication" Institute of Communication Networks, Technische Universität Vehicle-to-Vehicle Communications:Readiness of V2V Technology for Application.US Department of transportation,National Traffic Safety administration 2014, DOT HS 812 014

[5] H. Haas, "Opportunities and Challenges of Future LiFi," 2019 IEEE Photonics Conference (IPC), San Antonio, TX, USA, 2019, pp. 1-2.

[6] LiFi (Light Fidelity) & its Applications, FN Division, TEC (2014) pureLiFi. [Online]. http://purelifi.com/

[7] LiFI applicability: https://purelifi.com/case-study/lifi-in-a-classroom/

[8] Oldedcomm LIFI systems:
https://www.oledcomm.net/category/products/lifinet-modems-accessories/

[9] M. Alsumady, S. Alboon, "Intelligent Automatic High Beam Light Controller". Old City Publishing, Inc. Published by license under the OCP Science imprint, a member of the Old City Publishing Group., pp 1-8, 2013

[10] High Beam assistance:
https://www.hella.com/techworld/au/Technical/Automotive-lighting/High-beam-assist-583/

[11] C.Stoica Spahiu, L.Stanescu, M.Brezovan, F.Petcusin, "LiFi Technology Feasibility Study for Car-2-Car Communication", 21st International Carpathian Control Conference, 2020

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 435–441 DOI: 10.15439/2020F128 ISSN 2300-5963 ACSIS, Vol. 21

Design of a R-ID in order to determine the position of the vehicle

Frederik Valocký, Peter Drahoš, Oto Haffner, Alena Kozáková
Institute of Automotive Mechatronics
Slovak University of Technology in Bratislava
Bratislava, Slovakia
frederik.valocky, peter.drahos, oto.haffner, alena.kozakova [@stuba.sk]

Miloš Orgoň
Institute of Multimedia and ICT
Slovak University of Technology in Bratislava
Bratislava, Slovakia
milos.orgon@stuba.sk

***Abstract*—In this article, we design a road identifier (R-ID). The R-ID must be easily and quickly recognizable by a camera mounted on the vehicle. The camera captures this R-ID and then calculates the position information from it. The article describes the decision-making procedure for the design of an R-ID using camera recognition of geometric shapes. Parameters such as the uniqueness of the R-ID also play a role in this recognition, so that it is not interchangeable with other traffic signs. Another parameter is the percentage needed for the R-ID to be correctly recognized by the camera to obtain the necessary data from the overlay image. The outcome of this article is therefore a road identifier (a pattern) placed on the road which will be captured by the camera mounted on the vehicle .**

***Index Terms*—machine vision, computer vision, autonomous car, data transfer, picture recognition**

I. Introduction

AUTONOMOUS vehicles are coming to the fore more and more. An important element of the navigation of such a vehicle is the determination of the exact position and thus its location. The most common satellite systems are used to locate the vehicle. However, these satellite systems lose accuracy when vehicles are moving, and of course their reliability is very low in tunnels or covered halls where vehicles are moving. Therefore, it is necessary to devise a method that would be sufficiently accurate and fast enough to ensure the location of the vehicle even in areas where the reliability and possibilities of satellite location is declining [1].

The proposal is to mount a camera in the front of the vehicle, the task of which would be to capture the R-ID road identifier located on the road, and from it to determine the position of the vehicle and the rotation of the vehicle at what angle the vehicle came to this R-ID. These data will be used to enable the vehicle to perform other tasks for autonomous driving with high accuracy.

Recognition of objects and shapes with the camera is much more extensive. The carmakers deal with this problem, where they use the camera to capture traffic signs, which they then display on the display in the vehicle. German motor vehicle manufacturer - Opel equips its selected models with the Opel Eye system This advanced system has four main functions: traffic sign alert (TSA), lane departure warning (LDW), distance indicator (FDI) and collision warning function front (FCA). Traffic Sign Alert (TSA) works with a new camera that provides a higher detection rate and better features [2]. The camera uses high-frequency image exposure technology and an image processor with significantly higher performance, which allows the system to perform multiple operations simultaneously. In addition to displaying speed limits, the new system also recognizes and displays signs that are related to the maximum speed limit (snow, rain, etc.). The Opel Eye also monitors lane separation lines for lane departure warning (LDW), which is activated immediately if the vehicle leaves the lane without using the turn signals. An audible signal and a mark on the display warns the driver of the possibility of exiting the lane without being signalled by the turn signals. In Fig.1 shows an Opel Eye camera and Fig.2 shows an LDW system [3], [4].

Fig. 1. Innovative Opel Eye camera [3]

Fig. 2. LDW system in Opel Insignia [4]

In conditions of reduced visibility and at night, a system called ”night vision” is very useful. Night vision uses infrared light, which the human eye does not perceive and thus does

The paper was partially supported by the Slovak Research and Development Agency, grant No. APVV-17-0190 and the Slovak Cultural Educational Grant Agency, grant No.038STU-4/2018.

not blind oncoming vehicles. In addition to the usual lights, the road is also illuminated by two infrared headlights and, when the dipped beam headlights are switched on, it expands the driver’s field of view by another 150 m. The night visibility system will help identify pedestrians, cyclists, parked vehicles and other obstacles on the road much earlier. An infrared camera is built into the windshield, which records the image of the road ahead and transmits it to the display. The imaging system of a vehicle equipped with ”night vision” is shown in Figure 3 [5].

Fig. 3. Display system for a vehicle equipped with ”night vision” in a Mercedes-Benz S 550 [6]

II. Importance of R-ID in Transport

We decided to design the road identifier R ID in order to specify the position of the vehicle in locations where it is not possible to use other sensors than the position sensor etc.. The main task of this identifier is to guide autonomous vehicles in closed production halls where the construction of the building weakens the signals of satellite location systems. The road identifier will be placed on the ground and a camera placed on the vehicle will capture the road identifier. Following the acquisition of the identifier, it calculates the distance between the identifier and the vehicle. It also calculates the angle at which the vehicle approached the identifier and, if it did not meet the requirements of the planned route, would direct the vehicle in the correct direction. Data from the camera will be transmitted to the central unit of the vehicle where they will be processed. The processing of these data from the camera means fusion with other sensors to ensure redundancy and these other sensors are the already mentioned satellite positioning system, vehicle odometry, camera ensuring vehicle safety, vehicle safety means early braking if an obstacle enters the vehicle and this camera subsequently sends a signal that an object has entered the route, which may cause a collision, in which case the vehicle will stop them or perform another necessary action to prevent their collision. Thus, the signal from the COGNEX is7802 camera will be fused with all these sensors .

The accuracy of the road identifier on the road will be ensured by a static satellite signalling sensor, by which the surveyor measures the constant position of the identifier. Since the static satellite locator can determine an accuracy of a few millimetres, each of these patterns will have the exact position determined by such a sensor. Thus, if the vehicle captures the road identifier, it looks at the previous identifier of its position and looks at information from other less accurate sensors, such as the Satellite Location Sensor. If this information fits within the tolerance, the pattern on the ground will calculate the displacement position and thus refine the navigation of the vehicle in an enclosed space [7].

A. *Parameters of R-ID*

The main task of the tests was to find out which parameters are important precession identifiers. We focused mainly on parameters such as the uniqueness of the identifier, which means that it must not be interchangeable. Identifier recognition and identifier testing was performed using a COGNEX is 7802 camera.

This camera has many pre-programmed functions, such as calibration of the distorted image if the image is at an angle or also pattern search, this function is called PatMax. In our experiments, we mainly used the PatMax function. At the beginning we learned the camera what shape or shape to look in the camera image. And then we moved this shape in different directions and different positions to find out to what extent it can ensure image recognition even when rotated in a vertical position and also what size the identifier can change. to be recognizable below 100 ms.

III. Data Transfer between Camera and Computer

A camera from COGNEX was chosen for distance measurement, as it has high-quality and widely usable software with a fast support and a friendly In-Sight Explorer configuration environment installed on a computer in which the camera is also controlled. The type of camera used is a monochrome IS7802. This camera has a resolution of 1600 x 1200 pixels and the frame rate is 53. As camera lens was used autofocus module with 8mm lens (ISAF-7000-8mm) and illumination with white LED ring light cover to protect lens. The camera communicates via an Ethernet connection and supports protocols such as Ethernet / IP, FTP, PROFINET, OPC, Modbus / TCP, TCP / IP, SLMP and RS-232. As the COGNEX is7802 camera we use also supports the FTP protocol, we decided to use an FTP server created on a computer with the Windows 10 operating system to transfer data [8].

A. *Create an FTP server*

We decided to create an FTP server on a computer that is on the same network as the camera. Windows 10 supports user and server creation on a single computer. First of all, it is necessary to create a bookmark that will represent the storage space for data. Users will have access to this tab so they can write data to it. The next step is to create an account that will have rights to access the bookmark. This account represents a logged in user with a username and password to access the FTP server. On the path ”Control Panel All items of the Control Panel Administrative Tools” we will find a tool called ”Internet Information Services (IIS) Manager” and in this tool we will create an FTP page as can be seen in figure 4.

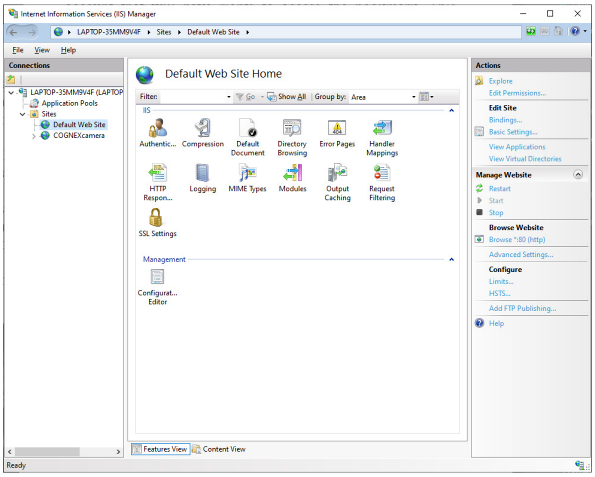
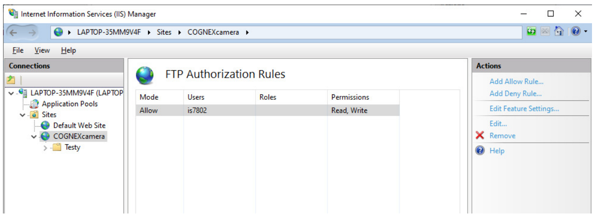


Fig. 4. Creating an FTP Server

Next, in this tool it is necessary to set the path to the shared directory and specify it as a bookmark of the FTP server. In this tool, it is necessary to add your new user and add all the rights (read and write) as can be seen in figure 5.



Mode	Users	Roles	Permissions
Allow	is7802		Read, Write

Fig. 5. sers with access to the FTP Server

B. Configure FTP transfer on the camera

The camera is configured in In-Sight Explorer, which is the software that controls the camera. In this software, all the parameters of the camera are set, and it also determines which image to capture and what parameters to store and process further. In insight explorer, the user has two options for displaying the environment. The first is ”EasyBuilder” and the second is ”Spreadsheet”. EasyBuilder is an intuitive and very friendly graphical display in which the user can create the basic recognition that the camera offers without any knowledge. However, for more complex tasks such as composing multiple functions and then using mathematical operations from the outputs of these functions, it is far better to use the Spreadsheet view. Spreadsheet view offers more features and tools than the EasyBuilder view menu. FTP transfer is feasible in both display options [9].

C. Konfiguration in EasyBuilder view

When configured in the middle, the setup is relatively simple. In the panel that offers application steps, select the ”Communication” option and in the lower left corner you will see the types of possible connections, including FTP. The window with application steps can be seen in figure 6.

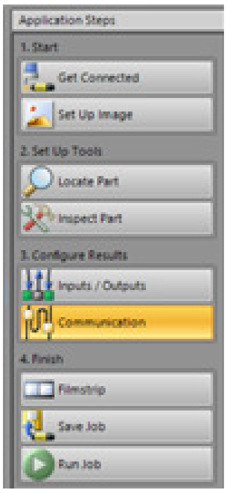
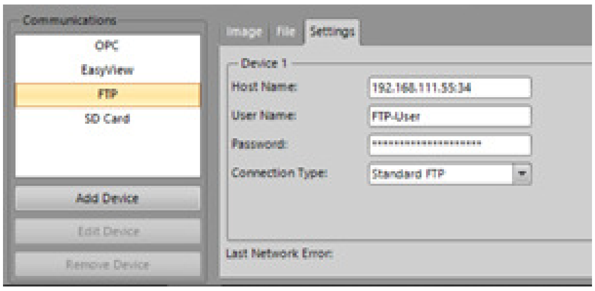


Fig. 6. Application steps

Then assign Host Name in the communication settings, which is the IP address and port of the FTP server in the format “IPaddress:Port”. It is necessary to set a different FTP server port than the camera itself. Next, fill in the User Name and Password, these are the data of the user you created. You can see the connection settings in Figure 7.



Host Name: 192.168.111.55:34

User Name: FTP-User

Password: ********************

Connection Type: Standard FTP

Fig. 7. Connection settings in EasyBuilder

D. Konfiguration in Spreadsheet view

The spreadsheet view represents a cellular environment similar to Microsoft Office Excel. It is possible to insert functions or function outputs into these cells for further possible data processing. Functions ensuring FTP data transfer can be found in the Input / Output¿ Network category. Figure 8 shows the position of the palette with functions in right side and the cell editor in left side.

When configured, the WriteFTP or WriteImageFTP function has the write format:

$$WriteFTP(Event, HostName, UserName, Password, \\ FileName, DataFormat, String, Append) \tag{1}$$

On figure 9 you can see example how are written data chosen by us only for one type of testing process from camera to PC.

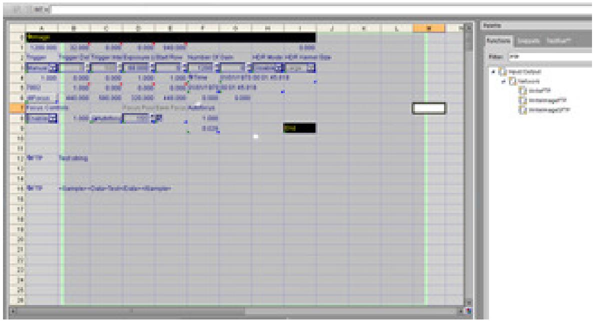
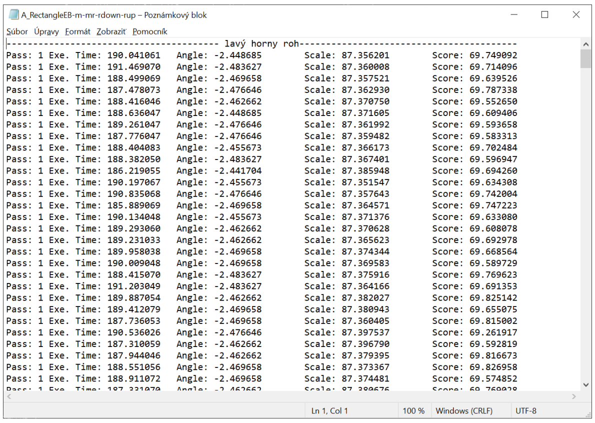


Fig. 8. Spreadsheet view



A_RectangleEB-m-mr-rdown-rup – Poznámkový blok
Súbor Úpravy Formát Zobraziť Pomocník

```
--------------------------------------- lavý horny roh-----------------------------------------
Pass: 1 Exe. Time: 190.041061   Angle: -2.448685        Scale: 87.356201        Score: 69.749092
Pass: 1 Exe. Time: 191.469070   Angle: -2.483627        Scale: 87.360008        Score: 69.714096
Pass: 1 Exe. Time: 188.499069   Angle: -2.469658        Scale: 87.357521        Score: 69.639526
Pass: 1 Exe. Time: 187.478073   Angle: -2.476646        Scale: 87.362930        Score: 69.787338
Pass: 1 Exe. Time: 188.416046   Angle: -2.462662        Scale: 87.370750        Score: 69.552650
Pass: 1 Exe. Time: 188.636047   Angle: -2.448685        Scale: 87.371605        Score: 69.609406
Pass: 1 Exe. Time: 189.261047   Angle: -2.476646        Scale: 87.361992        Score: 69.593658
Pass: 1 Exe. Time: 187.776047   Angle: -2.476646        Scale: 87.359482        Score: 69.583313
Pass: 1 Exe. Time: 188.404083   Angle: -2.455673        Scale: 87.366173        Score: 69.702484
Pass: 1 Exe. Time: 188.382050   Angle: -2.483627        Scale: 87.367401        Score: 69.596947
Pass: 1 Exe. Time: 186.219055   Angle: -2.441704        Scale: 87.385948        Score: 69.694260
Pass: 1 Exe. Time: 190.197067   Angle: -2.455673        Scale: 87.351547        Score: 69.634308
Pass: 1 Exe. Time: 190.835068   Angle: -2.476646        Scale: 87.357643        Score: 69.742004
Pass: 1 Exe. Time: 185.889069   Angle: -2.469658        Scale: 87.364571        Score: 69.747223
Pass: 1 Exe. Time: 190.134048   Angle: -2.455673        Scale: 87.371376        Score: 69.633080
Pass: 1 Exe. Time: 189.293060   Angle: -2.462662        Scale: 87.370628        Score: 69.608078
Pass: 1 Exe. Time: 189.231033   Angle: -2.462662        Scale: 87.365623        Score: 69.692978
Pass: 1 Exe. Time: 189.958038   Angle: -2.469658        Scale: 87.374344        Score: 69.668564
Pass: 1 Exe. Time: 190.009048   Angle: -2.469658        Scale: 87.369583        Score: 69.589729
Pass: 1 Exe. Time: 188.415070   Angle: -2.483627        Scale: 87.375916        Score: 69.769623
Pass: 1 Exe. Time: 191.203049   Angle: -2.483627        Scale: 87.364166        Score: 69.691353
Pass: 1 Exe. Time: 189.887054   Angle: -2.462662        Scale: 87.382027        Score: 69.825142
Pass: 1 Exe. Time: 189.412079   Angle: -2.469658        Scale: 87.380943        Score: 69.655075
Pass: 1 Exe. Time: 187.736053   Angle: -2.469658        Scale: 87.360405        Score: 69.815002
Pass: 1 Exe. Time: 190.536026   Angle: -2.476646        Scale: 87.397537        Score: 69.261917
Pass: 1 Exe. Time: 187.310059   Angle: -2.462662        Scale: 87.396790        Score: 69.592819
Pass: 1 Exe. Time: 187.944046   Angle: -2.462662        Scale: 87.379395        Score: 69.816673
Pass: 1 Exe. Time: 188.551056   Angle: -2.469658        Scale: 87.373367        Score: 69.826958
Pass: 1 Exe. Time: 188.911072   Angle: -2.469658        Scale: 87.374481        Score: 69.574852
Pass: 1 Exe. Time: 187.331070   Angle: -2.462662        Scale: 87.380676        Score: 69.769028
```

Fig. 9. Written Data

IV. Comparison of Recognition speeds of basic Geometric Shapes

First, it was necessary to find out which geometric shape is most easily recognizable. During the testing, we tested 3 geometric shapes, namely a rectangle, a circle and a triangle. The test was carried out. Thus, these geometric shapes were each printed with a size of 15 cm and a side thickness of 1 cm.

These faces were attached to a solid surface with which it moved in front of the camera at a distance of 2.5 m. The purpose of these tests was to find out which department recognizes the shortest and also to find the advantages and disadvantages of use for each department. We decided to perform 3 experiments for these units. The first attempt was focused on recognizing features in different positions in front of the camera. These positions were distributed throughout the camera's field of view. The second attempt was focused on the percentage overlap of the pattern when image can still be recognized. The 3rd experiment was designed to determine how the recognition time will affect the rotation of the pattern in front of the camera.

These experiments are based on parameters that can be set on the camera we use. These parameters include Image Distortion, for example, if the lens uses the Fisheye Effect. The first test was also focused on this. Another parameter is the angle below which the image must recognize, and therefore for each shape it is necessary to set the angle according to the sides of the geometric shape.

A. *Pattern Position in Camera View*

In front of the camera, we measured the positions where we will place the objects. There were 9 of these possible object fields. You can see them in figure 10 as smaller rectangles with position indicators. Camera view is represented by bigger full rectangle border. This camera view is picture which can be captured by camera.

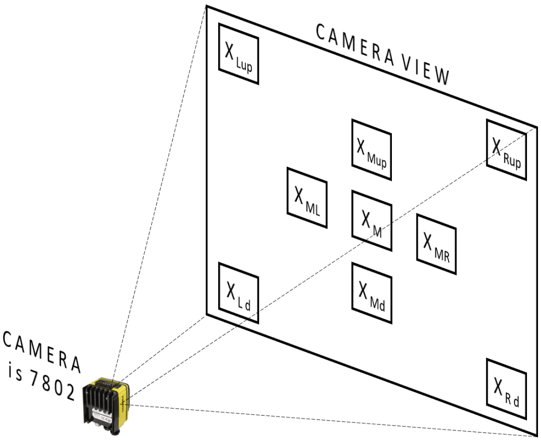


Fig. 10. Camera View

In InSight Explorer, we created a script to train and learn to further recognize a geometric shape if it appears in front of the camera and if it recognizes time information, how long it took to discover other information, such as resizing an image or how many percent it matches the trained one. picture. In this test, the decisive time for which the image was discovered and also how much it was necessary to increase the image alignment parameter to be recognized in the corners of lenses where the image is distorted.

The calibrated image or thus the trained image was at the exact center and the parameters were further adjusted so that this image was recognizable to other parts of the screen. This way we have achieved the accuracy that at any place where the camera captures the image and this image will be recognized. In this test, 100 separate measurements were performed in each position on 100 different images together, so there were 900 measurements for which we calculated the average recognition time and stopped the Graph at which the position took longer and at which the time took shorter.

In figures 11 and 12, you can see graphs that plot the recognition time and the score at which the image was recognized. The score above represents the percentage of how much the pattern resembles the learned pattern. The x-axis represents the time course of 100 consecutive images.

B. *Percentage overlap of the pattern*

In the second test, we wanted to find out what percentage of geometric shapes can be overlapped and still be recognized

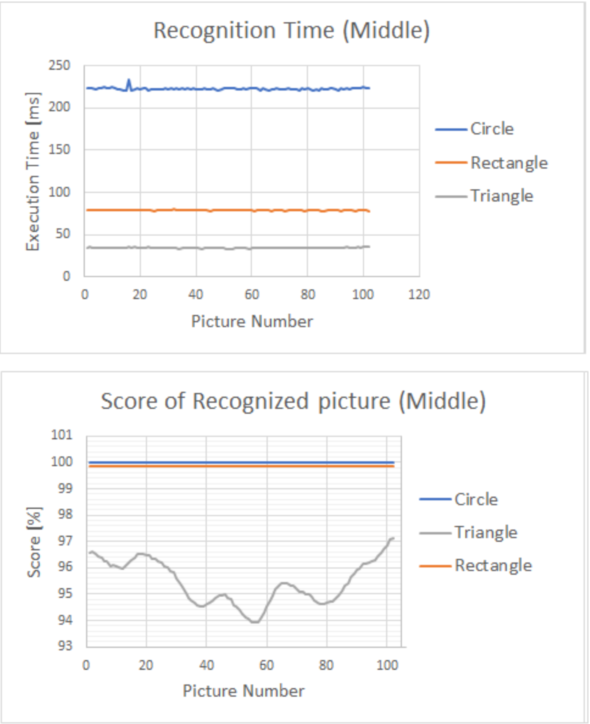


Fig. 11. Graphs of Recognition in the middle of camera view

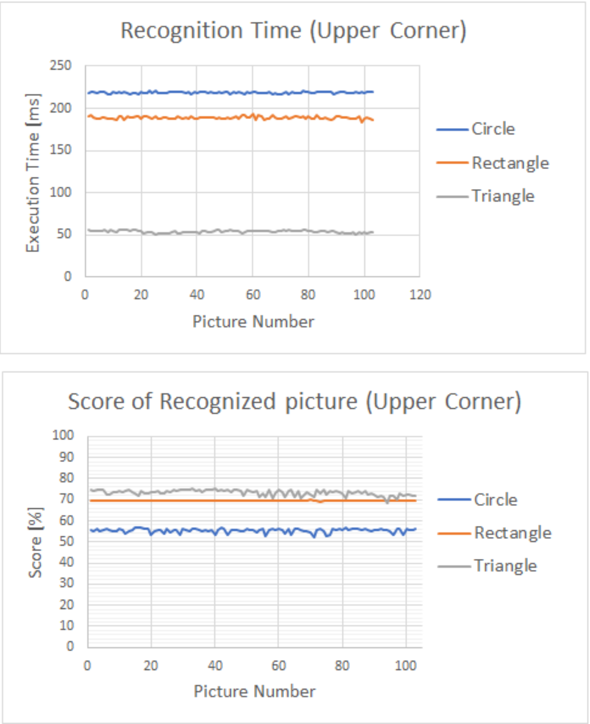


Fig. 12. Graphs of Recognition in the Upper Corners of camera view

by the camera. This test was intended to assume that these signs would be placed on the road. And so there is a risk that they may be partially damaged or covered or soiled. In this test, the pattern was calibrated exactly to the centre of the camera. Other parameters such as rotation or percentage enlargement or reduction of the image were set to zero so that it was purely detectable without any other parameters. This percentage is represented by a variable called a score. You can see in the graphs in figure 13 the minimum value for the coverage the system was still able to recognize the pattern.

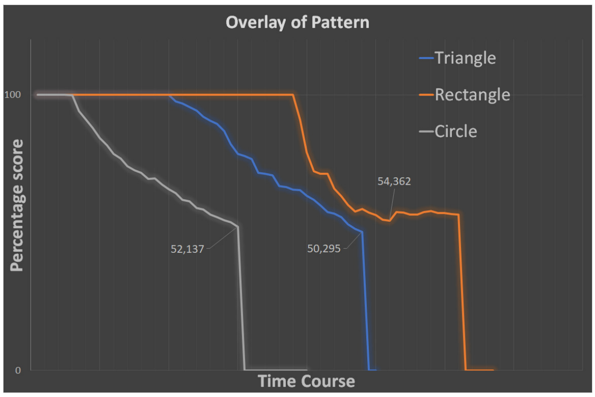


Fig. 13. Graph of Overlay of Pattern

This test was performed so that the pattern with the pattern gradually overlapped from each Cardinal directions and their combination, that is, there were 8 tests for each pattern, which were averaged, and the last data in the variable score was the lowest value at which the camera can recognize the pattern.

C. Rotation in Front of the Camera

Another test was to determine the time needed to recognize the pattern if it rotates with it. This test was to determine how the angle and rotation of the pattern would affect the recognition time. This text was converted so that for each image you set the angle needed for 100% image recognition. For a circle it was 0 °, for a rectangle it was 90 ° and for a triangle this setting was 120 °. These settings are set in InSight Explorer, which provides camera control. In this test, two hundred images were used for each pattern. Each of these images was different and the output from the camera recognition was to find out how the camera system perceives the change of angle for further image processing. You can see the results of these tests in the following graph in figure 14.

As can be seen in Figure 14, the camera is absolutely unable to detect the rotation of the pattern. In this test, the graph should rotate to form a sinusoid at the square and triangle and at the circle will be straight. However, the camera cannot work with the same side, for example if we imagine a square with sides a1, a2, a3 and a4 with a1 being down and a3 up in a horizontal position and we rotate the pattern 90 or 180 degrees etc. so for the camera it still represents from the pattern is rotated by 0 degrees. The same goes for the triangle just that there is an angle of 120 degrees.

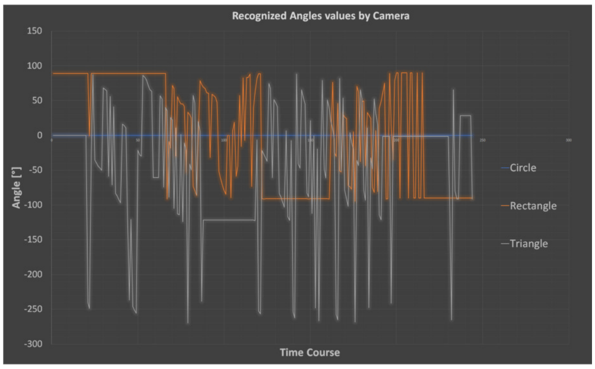


Fig. 14. Recognized Angles values by Camera

V. Design of R-ID

After the tests, we decided to use the triangle as the most ideal shape for recognition by our camera. However, it still needs to be modified so that it can provide information about the rotation of the vehicle. It should also be borne in mind that the shape will be painted on the road and therefore the substrate must consist of one piece. Therefore, we adjusted the shape to the form as can be seen in figure 15.

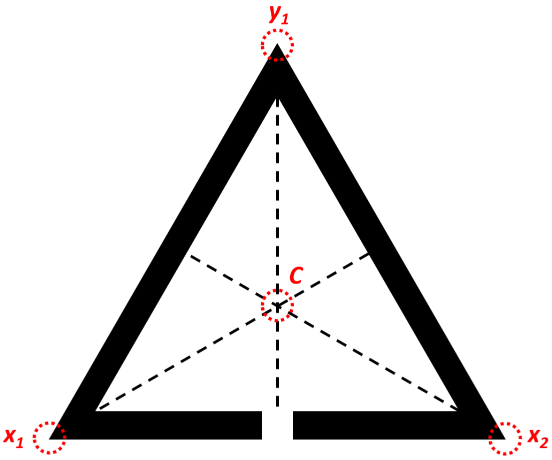


Fig. 15. Simple R-ID

With these changes we will provide a reference point C which will be accurately measured by surveyors. It will also provide us with points y_1 and x_1 and x_2, which will be subject to further measurements and measurement accuracy in order to determine the distance between the camera mounted on the vehicle and the road identifier.

VI. Future Work

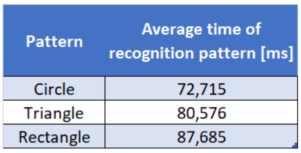
Since in this research we found that the triangular shape is the most ideal for use as a road identifier. It is necessary to test this the upper corner of the triangle and in the middle of the opposite side of this triangle.

The centre of the triangle is the intersection of such 3 sections of lines that begin at the vertex and end at the centre of the opposite side of the vertex. At this intersection will be the calibrated position using a static satellite module which will be accurately measured by the surveyor and its accuracy should be a maximum of a few tens of millimetres. With the help of thousands of tests, this static GPS module can determine a very precise position, which cannot be determined with high accuracy when the vehicle is moving in enclosed covered areas. Thus, this pattern will represent an exact imaginary milestone that will determine the exact position of the vehicle and also the direction in which the vehicle came to the image.

Other plans include the fusion of several sensors and systems and technologies, such as a camera with a new network that will ensure safety in front of the vehicle by having trained algorithms and neural networks to recognize objects of persons or other obstacles that unexpectedly appear in front of the vehicle. The other data that will be merged will come from the Vehicle Odometry and this data will talk about the speed of the expected time to the next identifier of the rotation of the vehicle's wheels and the tracking of the trajectory through which the vehicle has passed. The vehicle will also contain a GPS module, which does not assume high accuracy, but will be a security element that will specify the camera system whose purpose is to recognize the pattern on the ground. It is also possible to use systems such as LIDAR when mounted on servomotors and ultrasonic sensors around the perimeter of the vehicle from each side.

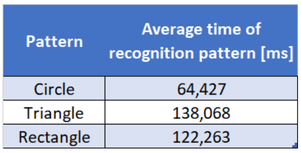
VII. Conclusion

The article describes the decision-making procedure for the design of an R-ID using camera recognition of geometric shapes. Important parameters are e.g. the uniqueness of the R-ID and the percentage needed for the R-ID to be correctly recognized by the camera and to obtain the necessary data from the overlay image. However, the most important parameters to be found in the shortest possible time are the most accurate position of the vehicle and the direction from which it comes to this position; they are still to be determined.



Pattern	Average time of recognition pattern [ms]
Circle	72,715
Triangle	80,576
Rectangle	87,685

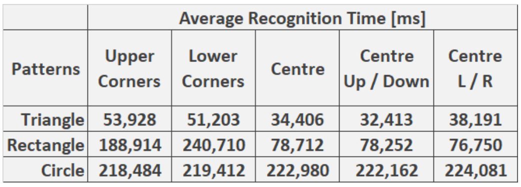
Fig. 16. Execution Times of Overlay test



Pattern	Average time of recognition pattern [ms]
Circle	64,427
Triangle	138,068
Rectangle	122,263

Fig. 17. Execution Times of Rotation test

As can be seen from the tables (figures 16,17 and 18), the largest differences in recognition were measured in tests when there were maximum recognition requirements, i.e. it



	Average Recognition Time [ms]				
Patterns	Upper Corners	Lower Corners	Centre	Centre Up / Down	Centre L / R
Triangle	53,928	51,203	34,406	32,413	38,191
Rectangle	188,914	240,710	78,712	78,252	76,750
Circle	218,484	219,412	222,980	222,162	224,081

Fig. 18. Execution Times of Position test

was necessary to recognize a pattern in any part of the scanned image. When rotating the image, the triangle ended with the time in the last place, but in real operation it is not possible for the vehicle to approach the image painted on the road at an angle greater than 120 degrees, so this result should not be considered too important. In further tests, a lower range of possible angles will be assumed. When testing the overlap of the pattern, the times were relatively the same, but it should be noted that when recognizing overlapping patterns, the best percentage of the right triangle was 50.295 (meaning that 49.705 % of the image can be obscured from the pattern), 52.573 which is an overlap of 47.827 % and with a rectangle score of 54.362 is an overlap of 45.638 %. Therefore, we decided to use a triangle as R-ID.

Camera limitations consist mainly in a poor resolution. On the resolution depends the calibration and the conversion of pixels to millimeters required when measuring the position. During testing, the camera was not mounted on the vehicle but was located only on a tripod that represented a model of a stationary vehicle. This means that it was mounted at the same height and under the same inclination as will be mounted on the vehicle, and the tested patterns were placed in similar distances as they will be painted on the road. After transferring the information from the camera to the server, the computer selects the information from the server and further processes it. The computer's task will be to calculate the distance and rotation from the data (x and y values) received from the camera. This data is then compared with the data from other sensors according to the time stamp. When the computer knows the exact position of the vehicle, it sends a direction correction so that the vehicle is directed to the next mark. The aim of road identifiers is not navigate the autonomous vehicle but to specify its position in places where other sensors are less reliable, and in the end to assist in the navigation of the autonomous vehicle.

ACKNOWLEDGMENT

The paper was partially supported by the Slovak Research and Development Agency, grant No. APVV-17-0190 and the Slovak Cultural Educational Grant Agency, grant No.038STU-4/2018.

REFERENCES

[1] F. Valocký, P. Drahoš, and O. Haffner, "Measure distance between camera and object using camera sensor," in *30th Conference Cybernetics & Informatics*. FEI STU BA, 2020.

[2] B. Buková, R. Madleňák, and K. R., "Elektronické podnikanie v doprave a logistike." Iura Edition, 2009.

[3] Opel, Groupe PSA, "Innovative Opel Eye camera," 2014. [Online]. Available: https://http://www.opel.pl

[4] T. Čorejová, A. Križanová, and N. A., "Tendencies in the professional education for transportation sector in slovakia." Iura Edition, 2009.

[5] J. Mikulski and P. Czech, "Vibration signals to diagnose damage of head gasket in internal combustion engine of a car. in: Telematics." TST 2014, 2014.

[6] VolvoCars, "NightVision," 2014. [Online]. Available: http://www.volvocars.pl/september2014

[7] R. Fujdiak, P. Masek, P. Mlynek, J. Misurec, and A. Muthanna, "Advanced optimization method for improving the urban traffic management," in *Proceedings of the 18th Conference of Open Innovations Association FRUCT*. FRUCT Oy, 2016, pp. 48–53.

[8] COGNEX, "Cognex Support," 2019. [Online]. Available: [6]https://support.cognex.com/docs/is_580/web/EN/ise/Content/Reference/WriteFTP.htm?tocpath=Communications%20Reference%7CFTP%20Communications%7C_____2

[9] F. Valocký, P. Drahoš, O. Haffner, and M. Orgoň, "Transfer data from camera required for autonomous driving via ftp," in *22nd Conference of Doctoral Students Faculty o fElectrical Engineering and Information Technology*. FEI STU BA, 2020.

Interpolation merge as augmentation technique in the problem of ship classification

Dawid Połap* and Marta Włodarczyk-Sielicka†
*Marine Technology Ltd.
ul. Roszczynialskiego 4/6, 81-521 Gdynia, Poland
†Maritime University of Szczecin
Waly Chrobrego 1-2, 70-500 Szczecin, Poland
Email: *d.polap@marinetechnology.pl, †m.wlodarczyk@am.szczecin.pl

***Abstract*—Quite a common problem during training the classifier is a small number of samples in the training database, which can significantly affect the obtained results. To increase them, data augmentation can be used, which generates new samples based on existing ones, most often using simple transformations. In this paper, we propose a new approach to generate such samples using image processing techniques and discrete interpolation method. The described technique creates a new image sample using at least two others in the same class. To verify the proposed approach, we performed tests using different architectures of convolution neural networks for the ship classification problem.**

I. Introduction

IMAGE classification is considered one of the leading problems in today's AI where many inconvenient situations may occur. One such element is too much variety of objects belonging to the same class. Besides, each of these objects can be placed on a different background, in a different light the same as from a certain angle. Another one is too small several samples for selected classes to train the classifier at a given level of accuracy. Unfortunately, there is also the situation of overtraining the model which means correct classification at the training database level, but the large probability of incorrect classification for new samples. These are only selected problems encountered in the classification task.

To reduce the likelihood of any of them occurring, a proper detection [1] or data augmentation can be used. It is a process of increasing the number of samples in the database using existing ones. This kind of generating process is mainly based on rotation, changing brightness or zooming. All of them are basic image processing techniques but provides more samples. It was proved that accuracy can be higher using this type of approach what can be seen in [2], where the authors analyzed the impact of augmented data on the effectiveness of convolutional neural network (CNN). Again in [3], the idea of generating data during the process of driving and using them to generate newer is described. A big problem in a small amount of data is medical data, where the diseases can be of different form and shape. In [4], the idea of using augmented cytological images of cells was presented. Similar ideas of using generating sample techniques with CNN were used in the target recognition problem in [5], [6]. In [7], the idea of using CNN for semantic segmentation was presented. In this solution, to increase the accuracy level of CNN, augmentation was added to achieve wanted results. Moreover, in [8], authors used the occlusion technique, according to which random rectangles composed of pixels in the HSL color model are randomly obscured. This type of action also changing image, so as a result a new sample is generated. Another approach of augmentation solution was presented in [9], wherein the architecture of CNN, the authors added a layer that is responsible for this process. Similar idea was described in [10],

In this paper, we propose a novel technique of data augmentation for the classification process. Our approach is based on interpolated data from two images for merging into the third one as a hybrid one. This approach was tested in the problem of ship classification.

II. Interpolation merge technique

Having two images from the same class I_1 and I_2, there is a way to merge them into a new image. At first, each of these images must be processed to calculate the merging ability. For this purpose, some image processing technique is used for simplicity and adaptation to interpolation.

Each image is subjected to convolutional operation wchich is a modification of image using some filter g. A filter is described as a matrix of size $k \times k$, and this oparation can be described as following

$$\begin{aligned} I'[i,j] &= I[i,j] * g \\ &= \sum_{t=-\lfloor k/2 \rfloor}^{\lfloor k/2 \rfloor} \sum_{r=-\lfloor k/2 \rfloor}^{\lfloor k/2 \rfloor} I[i+t, j+r] \cdot g[t,r]. \end{aligned} \tag{1}$$

The coefficient of matrix g depends on the filter. In our proposition, each image is converted as

$$I'[i,j] = \left((I[i,j] * g_1)^5 * g_2 \right) * g_3, \tag{2}$$

where the filters g_i define Gaussian blur, emboss and edge detection as follows

$$g_1 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \tag{3}$$

$$g_2 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}, \tag{4}$$

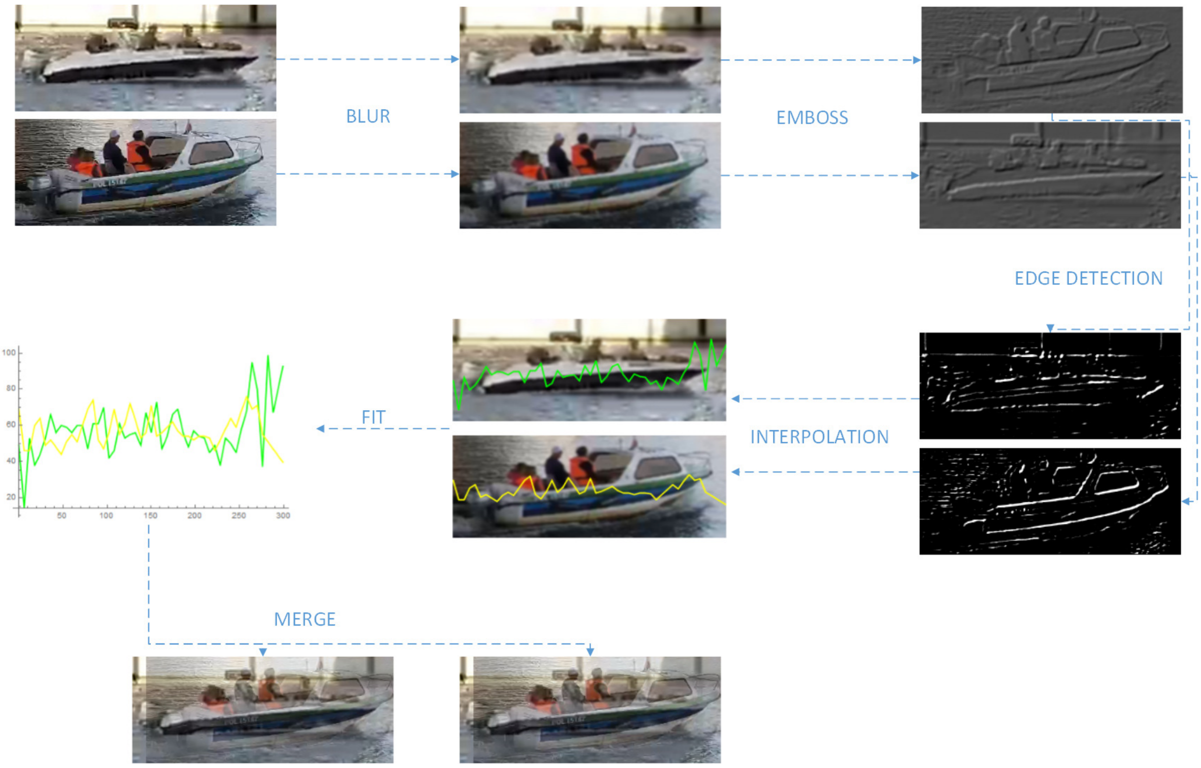


Figure 1: Visualization of the proposed augmentation technique.

$$g_3 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}. \tag{5}$$

The above filters are needed to obtain only some pixels that represent the shape of a ship. In Eq. 2, there is five times blur due to the removal of as much noise as possible and smaller elements, and above all water. The remaining not black pixels in obtained image I' must be corrected before using them in the interpolation technique which requires uniquely defined points. To achieve this uniqueness to the OX axis, for each column of pixel in I', the average value of all values non-black pixels will be calculated according to

$$\left(\frac{1}{p}\sum_{i=1}^{p} x_i, y\right), \tag{6}$$

where p is the number of all non-black pixels in x column.

Obtained points can be defined as a set $\Theta = \{(x_1, y_1), (x_2, y_2), \ldots, (x_p, x_{p-1})\}$ where each point can be presented as $y = f(x)$. These points are interpolated by some function $F(x) = \sum_{o=0}^{m} a_o \phi_o$ (ϕ_o are a polynomial system). As interpolation, we understood finding all coefficient a_m of $F(x)$ in such a way that error is the smallest one, what can be defined as

$$||f - F|| = \sum_{i=1}^{p} w(x_i)[f(x_i) - F(x_i)]^2. \tag{7}$$

As a result, we obtain an interpolated functions F_{I_1} and F_{I_2}. Both functions are compared with each other to minimize the area between curves as

$$\int_0^{width} |F_{I_1}(x) - F_{I_2}(x)|\, dx. \tag{8}$$

When the error value is the smallest, both images are superimposed to fit the function offset. The described operation is performed concerning the width, although it can be done in the same way in height – then the fit will be more accurate. Image joining consists of changing the alpha channel α on the shifted image, which will cause it to shine through. An example of augmentation is illustrated in Fig. 1.

Of course, it is possible to use basic augmentation techniques like zooming, rotation, etc, which will give more images with a more accurate fit.

III. Experiments

Described technique was implemented in Wolfram Mathematica 11 and C# language. For testing purposes, two classes representing cargo (2120 images) and military (1167 images) vessels from the Deep Learning Hackathon publicly available collection organized by Analytics Vidhya were used, and one class was created as part of the SHREC project [11] consisting of 688 images of yachts. This database was used for analyzing proposed technique for classical architecture of convolutional neural network like AlexNet [12], VGG16 [13] and Inception [14].

	CNN	10 iterations	15 iterations
without augmentation	AlexNet	0,62	0,74
	VGG16	0,64	0,75
	Inception	0,63	0,77
$\alpha = 0.2$	AlexNet	0,67	0,78
	VGG16	0,68	0,8
	Inception	0,59	0,82
$\alpha = 0.4$	AlexNet	0,64	0,85
	VGG16	0,62	0,86
	Inception	0,67	0,87
$\alpha = 0.6$	AlexNet	0,65	0,84
	VGG16	0,67	0,85
	Inception	0,71	0,87
$\alpha = 0.8$	AlexNet	0,59	0,8
	VGG16	0,61	0,76
	Inception	0,6	0,83

Table I: Comparison of the effectiveness of various CNN architectures in relation to the proposed augmentation and the number of training iterations (without freezing method).

Statistical coefficient	Value
accuracy	0,865454545
sensitivity	0,962760131
specificity	0,390374332
precision	0,885196375
negative predictive value	0,682242991
miss rate	0,037239869
fall-out	0,609625668
false discovery rate	0,114803625
false omission rate	0,317757009
F1 score	0,922350472

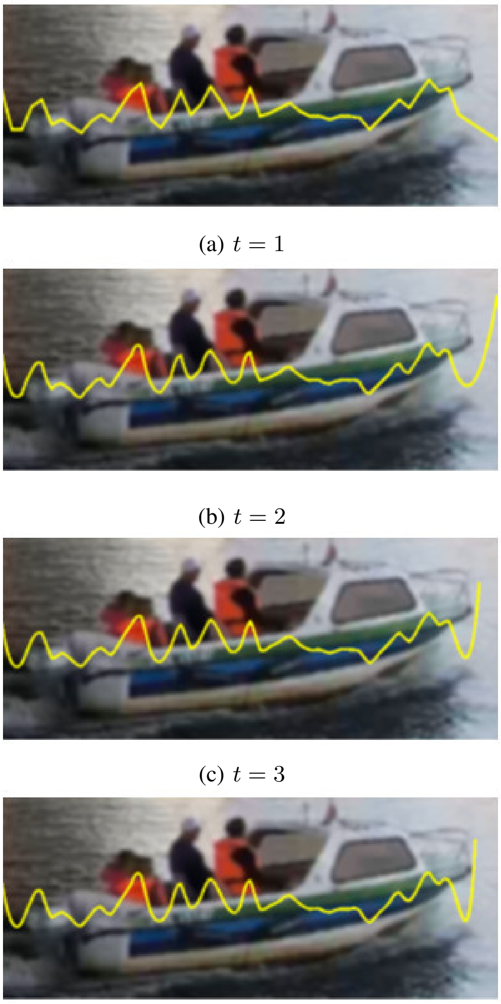
Table II: Statistical values for Inception architecture trained on the basis of data augmentation.

For each of class, 75% of samples were used for training purpose (with a ratio 80:20 – training:validation samples), the remaining 25% was used for verification purpose to analyze accuracy. Then, for each of class, 20% of images was chosen for augmentation, which allowed for a 10% increase in each class.

The images used for augmentation were selected randomly and created according to the described technique. As part of the analysis, the following degrees t of polynomials were used – 2, 3 and 4. Sample interpolation function with t-th degree are presented in Fig. 2. We notice, that for degrees higher than 2 and small samples, the function can reach extremum above or even below image which is not advisable to use – due to the later fit. When a grade 2 polynomial is used, the graph is more rounded, so matching between two graphs can be simpler but formally there are more calculations to find this polynomial.

Samples were created for second degree of interpolated polynomial. All samples has transparency coefficient $\alpha \in \{0.2, , 0.4, 0.6, 0.8\}$. Each of CNN was trained with this database for 10 and 15 iterations and then checked accuracy using the verification database (with additional 107 images of other types of ship).

The obtained results are presented in Tab. I. Based on these results, using proposed augmentation increases accuracy by an average of 7%. It is easy to see that the increase in the number of iterations resulted in an increase in accuracy for each architecture without as well as with augmentation. The best results were obtained for the transparency parameter value of 0.4 and 0.6. The reason for these results is the fact that at these levels both ships are visible with a minimum clearance relative to the other and are stacked on top of each other (by width). Also, for Inception architecture and transparency values of 0.4 as well as 0.6, the highest accuracy was reached at 0.87. Other statistical coefficients were calculated for this configuration and are presented in Tab. II and in Fig. 3. Despite the high accuracy and precision, the architecture showed that the sensitivity of the model is very high, as it is 0.96, and is understood as the probability that the classification will be correct if the entering image was correct. Also, the specificity was almost 0.4, which is not the best result, as it is the probability of correct classification for a false sample. Besides, the false discovery rate has reached a probability level of 0.11, which means that only 11% of the positive results will not be correct.



(a) $t = 1$

(b) $t = 2$

(c) $t = 3$

(d) $t = 4$

Figure 2: Samples with specific degree t of interpolation function.

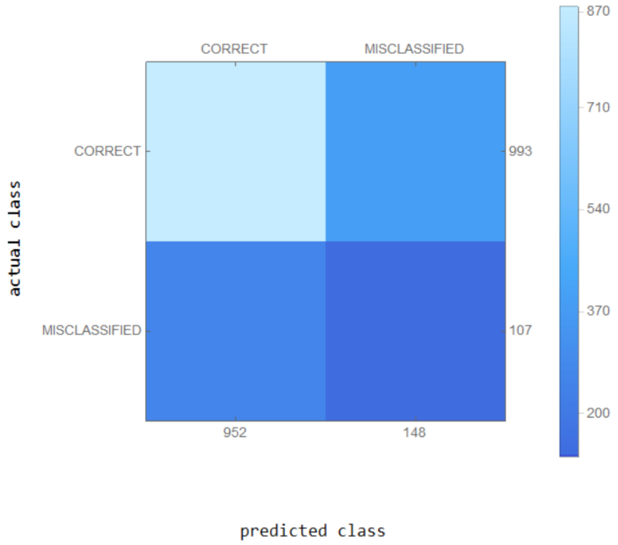


Figure 3: Confusion matrix for the tested Inception architecture.

The results indicate the correct impact of the newly generated data on the quality of the classification when performing tests on three classic architectures using learning transfer. Besides, the calculated statistical coefficients confirm that the convolution network model is properly trained and has high efficiency in detecting correct samples as well as a high probability of classification into a given class.

IV. Conclusion

Data augmentation is important in the training process when there is no sufficient amount of samples. In this paper, we propose an alternative approach for creating image data based on two other samples using merging images and locating them based on the interpolation technique. We showed that this approach can be useful for training a convolutional neural networks. Based on the obtained results, our approach reached an average accuracy value 7% better than the classic approach without using an extended database.

In the future, we plan to analyze other techniques of mixing two images which might prove to be a better way to generating new image samples.

Acknowledgments

This work was supported by the National Centre for Research and Development (NCBR) of Poland under grant no. LIDER/17/0098/L-8/16/NCBR/2017.

References

[1] T. Hyla and N. Wawrzyniak, “Ships detection on inland waters using video surveillance system,” in *IFIP International Conference on Computer Information Systems and Industrial Management*. Springer, 2019, pp. 39–49.

[2] M. A. Kutlugün, Y. Sirin, and M. Karakaya, “The effects of augmented training dataset on performance of convolutional neural networks in face recognition system,” in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019, pp. 929–932.

[3] W. Zhang, P. M. Chu, K. Huang, and K. Cho, “Driving data generation using affinity propagation, data augmentation, and convolutional neural network in communication system,” *International Journal of Communication Systems*, p. e3982, 2019.

[4] A. Teramoto, A. Yamada, Y. Kiriyama, T. Tsukamoto, K. Yan, L. Zhang, K. Imaizumi, K. Saito, and H. Fujita, “Automated classification of benign and malignant cells from lung cytological images using deep convolutional neural network,” *Informatics in Medicine Unlocked*, vol. 16, p. 100205, 2019.

[5] N. J. Tustison, B. B. Avants, Z. Lin, X. Feng, N. Cullen, J. F. Mata, L. Flors, J. C. Gee, T. A. Altes, J. P. Mugler III *et al.*, “Convolutional neural networks with template-based data augmentation for functional lung image quantification,” *Academic radiology*, vol. 26, no. 3, pp. 412–423, 2019.

[6] F. Gao, T. Huang, J. Sun, J. Wang, A. Hussain, and E. Yang, “A new algorithm for sar image target recognition based on an improved deep convolutional neural network,” *Cognitive Computation*, vol. 11, no. 6, pp. 809–824, 2019.

[7] K. Cho *et al.*, “Retrieval-augmented convolutional neural networks against adversarial examples,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 563–11 571.

[8] J. M. Haut, M. E. Paoletti, J. Plaza, A. Plaza, and J. Li, “Hyperspectral image classification using random occlusion data augmentation,” *IEEE Geoscience and Remote Sensing Letters*, 2019.

[9] G. Chen, C. Li, W. Wei, W. Jing, M. Woźniak, T. Blažauskas, and R. Damaševičius, “Fully convolutional neural network with augmented atrous spatial pyramid pool and fully connected fusion path for high resolution remote sensing image segmentation,” *Applied Sciences*, vol. 9, no. 9, p. 1816, 2019.

[10] L. Mou, Y. Hua, and X. X. Zhu, “A relation-augmented fully convolutional network for semantic segmentation in aerial scenes,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 12 416–12 425.

[11] N. Wawrzyniak and A. Stateczny, “Automatic watercraft recognition and identification on water areas covered by video monitoring as extension for sea and river traffic supervision systems,” *Polish Maritime Research*, vol. 25, no. s1, pp. 5–13, 2018.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” pp. 1097–1105, 2012.

[13] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[14] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

13th Workshop on Computer Aspects of Numerical Algorithms

NUMERICAL algorithms are widely used by scientists engaged in various areas. There is a special need of highly efficient and easy-to-use scalable tools for solving large scale problems. The workshop is devoted to numerical algorithms with the particular attention to the latest scientific trends in this area and to problems related to implementation of libraries of efficient numerical algorithms. The goal of the workshop is meeting of researchers from various institutes and exchanging of their experience, and integrations of scientific centers.

TOPICS

- Parallel numerical algorithms
- Novel data formats for dense and sparse matrices
- Libraries for numerical computations
- Numerical algorithms testing and benchmarking
- Analysis of rounding errors of numerical algorithms
- Languages, tools and environments for programming numerical algorithms
- Numerical algorithms on coprocesors (GPU, Intel Xeon Phi, etc.)
- Paradigms of programming numerical algorithms
- Contemporary computer architectures
- Heterogeneous numerical algorithms
- Applications of numerical algorithms in science and technology

TECHNICAL SESSION CHAIRS

- **Bylina, Beata,** Maria Curie-Sklodowska University, Poland
- **Bylina, Jaroslaw,** Maria Curie-Sklodowska University, Poland
- **Stpiczyński, Przemysław,** Maria Curie-Sklodowska University, Poland

PROGRAM COMMITTEE

- **Amodio, Pierluigi,** Università di Bari, Italy
- **Anastassi, Zacharias,** De Montfort University, United Kingdom
- **Banaś, Krzysztof,** AGH University of Science and Technology, Poland
- **Bielecki, Wlodzimierz,** West Pomeranian University of Technology in Szczecin
- **Brugnano, Luigi,** Universita' di Firenze, Italy
- **Fialko, Sergiy,** Tadeusz Kościuszko Cracow University of Technology, Poland
- **Fourneau, Jean-Michel**
- **Georgiev, Krassimir,** IICT - BAS, Bulgaria
- **Gepner, Paweł**
- **Knottenbelt, William,** Imperial College London, United Kingdom
- **Kozielski, Stanislaw,** Silesian University of Technology, Poland
- **Kucaba-Pietal, Anna,** Politechnika Rzeszowska, Poland
- **Lastovetsky, Alexey,** University College Dublin, Ireland
- **Lirkov, Ivan,** Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Bulgaria
- **Luszczek, Piotr,** University of Tennessee, United States
- **Marchiori, Massimo**
- **Marowka, Ami,** Bar-Ilan University, Israel
- **Mele, Valeria**
- **Petcu, Dana,** West University of Timisoara, Romania
- **Shishkina, Olga,** Max Planck Institute for Dynamics and Self-Organization, Germany
- **Trivedi, Kishor S.,** Duke University, United States
- **Tudruj, Marek,** Inst. of Comp. Science Polish Academy of Sciences/Polish-Japanese Institute of Information Technology, Poland
- **Vardanega, Tullio,** University of Padova, Italy
- **Vazhenin, Alexander,** University of Aizu, Japan

DOI: 10.15439/2020F85 ISSN 2300-5963 ACSIS, Vol. 21

Efficient Computation of RNA Partition Functions Using McCaskill's Algorithm

Chunchun Zhao
Amadeus IT Group
Email: zhaochunchun@gmail.com

Sartaj Sahni
Department of Computer and Information
Science and Engineering
University of Florida
Email: sahni@cise.ufl.edu

***Abstract*—We develop efficient single- and multi-core algorithms to compute partition functions for RNA sequences. Our algorithms, which are based on McCaskill's algorithm, are benchmarked against state-of-the-art fast algorithms obtained using the parallelizing source-to-source compilers PLUTO and TRACO. On our Intel I9 computational platform, our best single core algorithm takes up to 81.2% less time than the single core algorithm resulting from PLUTO, which is faster than that obtained from TRACO. Our best multi-core algorithm takes up to 84.7% less time than the multi-core algorithm obtained using TRACO when run with 20 threads (our I9 has 10 cores and supports hyperthreading); the TRACO multi-core algorithm is faster than the PLUTO one.**

I. Introduction

SEVERAL algorithms have been proposed to determine the minimum energy secondary structure of an RNA molecule. Smith and Waterman [1] and Nussinov et al. [2] have proposed a dynamic programming algorithm to maximize the number of complementary base pairs. These algorithms are oversimplified and do not generate good RNA secondary structure predictions. Zuker et al. [3] first defined five different loops that are basic units of the RNA secondary structure and proposed an energy model for each loop. Using this energy model, they developed a dynamic programming algorithm to determine the total minimum free energy structure for a given RNA sequence. The original approach of Zuker et al. [3] has been enhanced in [4], [5] by Zuker to handle more complex cases. However, Zuker's approach only provides a globally optimal secondary structure at equilibrium where each loop has been evaluated by the energy model without errors. To calculate the variance of the secondary structure, Zuker [6] proposed a further extension to calculate all suboptimal secondary structures of an RNA sequence. McCaskill [7] has proposed a totally different method to compute the full equilibrium partition function, which is the sum of the contributions of the suboptimal structures. McCaskill's algorithm computes the probabilities of each individual base pair of (i, j) in the RNA sequence. These probabilities are used, for example by the software LocaRNA [8] and PMComp/PMMulti [9] for simultaneous folding and alignment and in algorithms to predict RNA structure with a maximum expected accuracy [10], [11].

Many parallel algorithms for RNA secondary structures have also been proposed. (see, for e.g., [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23]). Fekete et al. [22] first parallelized the McCaskill algorithm for a computer cluster. More recently, Palkowski and Bielecki [23] used the parallelizing source-to-source compilers PLUTO [24], [25] and TRACO [26] to automatically generate cache efficient and multi-core codes for the McCaskill algorithm.

In this paper, we begin by rewriting McCaskill's dynamic programming equations using a single matrix rather than two as used in the original equations. Then, using the row and box computation methods proposed by us in [27] and [28] to develop cache efficient algorithms for Nussinov's and Zuker's methods, respectively, for RNA folding, we develop new algorithms to compute the partition functions of McCaskill using a single array. The performance of our algorithms is compared experimentally with that of McCaskill's original algorithm and optimized versions obtained automatically by the optimizing compilers PLUTO and TRACO. Code for the original, PLUTO, and TRACO algorithms was obtained from [23]. Our experiments indicate that we are able to reduce run time by as much as 81.2% relative to the fastest previously known single core code and by as much as 84.7% relative to the fastest multi-core code using 20 threads on a 10 core Intel I9 CPU that supports hyperthreading.

The rest of the paper is organized in the following way. McCaskill's original dynamic programming equations and corresponding algorithm are presented in Section II. In Section III, we give our rewrite of McCaskill's equations and corresponding algorithms. Experimental results are presented in Section IV. Finally, Section V presents concluding remarks.

II. McCaskill's Equations and Algorithms

Let $A[1 : n] = a_1 a_2 \cdots a_n$ be an RNA sequence. McCaskill [7] develops an $O(n^4)$ and an $O(n^3)$ algorithm to compute the partition functions of A. The $O(n^3)$ algorithm uses a simplified (constant) energy function and it is this algorithm that is the focus of [23] and this paper. Let $Q(i, j)$ be the partition function of the subsequence $A[i : j]$ and let $Q^{bp}(i, j)$ be the partition function of the subsequence $A[i : j]$ when $A[i]$ and $A[j]$ form a base pair ($Q^{bp}(i, j)$ is 0 when $A[i]$ and $A[j]$ do not form a base pair). McCaskill's simplified dynamic programming equations are:

$$Q_{i,i-1} = 1,\ 1 \leq i \leq n \tag{1}$$

Algorithm 1 McCaskill Original

```
for i=N-1; i>=0; i−− do
    for j=i+1; j<N; j++ do
        Q[i][j] = Q[i][j-1];
        for k=i; k<j-l; k++ do
            $Q^{bp}$[k][j] = Q[k+1][j-1] * $exp(-E_{bp}/RT)$ * $pair$(k,j);
            Q[i][j] += Q[i][k-1] *$Q^{bp}$[k][j];
        end for
    end for
end for
```

Algorithm 2 OneArray

```
1: for d=0; d<=N; d++ do // d: index of diagonal
2:     for i=0; i<=N; i++ do // i: index of row
3:         j = d+i; // j: index of column
4:         Q[i][j] = Q[i][j-1];
5:         for k=i; k<j-l; k++ do
6:             Q[i][j] += Q[i][k-1] * Q[k+1][j-1] * $exp(-E_{bp}/RT)$ * $pair$(k,j);
7:         end for
8:     end for
9: end for
```

$$Q_{i,j} = Q_{i,j-1} + \sum_{i \le k < j-l} Q_{i,k-1} * Q^{bp}_{k,j} \tag{2}$$

$$Q^{bp}_{i,j} = Q_{i+1,j-1} * exp(-E_{bp}/RT) * pair(i,j) \tag{3}$$

where $pair(i,j)$ is 1 if $A[i]$ and $A[j]$ are complementary base pairs such as AU, GC and GU, and 0 otherwise; E_{bp} is the fixed energy, contributed by a base pair; R is a gas constant; T is the temperature; and l is the minimum loop length.

Using these equations, the partition functions Q and Q^{bp} may be computed using the algorithm $Original$ (Algorithm 1) [23], which computes the Qs and Q^{bp}s by rows from bottom to top and within a row from left to right.

Palkowski and Bielecki [23] have experimented with optimized single core and multi core versions of Algorithm $Original$ obtained using the source-to-source parallelizing compilers PLUTO [24], [25] and TRACO [26]. We use the terms $Orig$, $Pluto$, and $Traco$ to refer to their single core single thread codes for the original algorithm and its optimization using PLUTO and TRACO, respectively.

III. Our Algorithms

A. Base Algorithm For Q Using One Triangular Array

McCaskill's dynamic programming equations for Q are easily rewritten as below by eliminating Q^{bp} from Equation 2 using Equation 3. The rewritten equations are:

$$Q_{i,i-1} = 1, \ 1 \le i \le n \tag{4}$$

$$\begin{aligned} Q_{i,j} = Q_{i,j-1} + \sum_{i \le k < j-l} & Q_{i,k-1} * Q_{k+1,j-1} \\ & *exp(-E_{bp}/RT) * pair(k,j) \end{aligned} \tag{5}$$

$Q^{bp}(i,j)$, if needed, may be computed from $Q(i+1,j-1)$ using Equation 3 once all the Qs have been computed.

Using the rewritten equations, Q may be computed using a single array as shown in Algorithm 2 ($OneArray$). This algorithm computes Q by diagonals and within a diagonal from top to bottom. It's run time is $O(n^3)$ and it uses $n(n+1)/2$ space when an upper triangular two-dimensional array [29] is used for Q. Q^{bp} may be computed from Q in $O(n^2)$ time using Equation 3.

Notice that Qs that are on the same diagonal may be computed simultaneously. So, algorithm $OneArray$ is easily parallelized. Let $OneArrayP$ be the parallel version of $OneArray$ obtained by dividing each diagonal into t tiles of approximately same size, where t is the number of parallel threads.

B. ByRow (ByRow Algorithm)

In the loop of lines 5 and 6 of Algorithm $OneArray$ (Algorithm 2), the memory accesses for $Q[k+1][j-1]$ are cache inefficient as these correspond to a column access while the memory accesses for $Q[i][k-1]$, which correspond to a row access, are cache efficient. As noted by us in [27], [28] cache efficiency is enhanced by computing the Qs by rows bottom to top rather than by diagonals. Within a row, the computation is done left to right. Algorithm 3, $ByRow$, does exactly this. Though the computation order is the same in Algorithm $ByRow$ as in Algorithm $Original$ (Algorithm 1), there is a significant difference between the two algorithms besides the fact that Algorithm $ByRow$ does not use Q_{bp} explicitly. Once an element $Q[i][j]$ has been calculated in $ByRow$, all elements to its right which are on the same row are updated. This does not happen in $Original$. Relative to Algorithm 2, Algorithm 3 eliminates the inefficient column access for Q.

Algorithm 3 ByRow

```
for i=N-1; i>=0; i−− do // i: index of row
    for k=i; k<N-l; k++ do
        Q[i][k] = Q[i][k-1];
        for j=k+l+1; j<N; j++ do
            Q[i][j] += Q[i][k-1] * Q[k+1][j-1] * $exp(-E_{bp}/RT)$ * $pair$(k,j);
        end for
    end for
end for
```

The innermost loop of Algorithm $ByRow$ (i.e., the j loop) is easily parallelized. The resulting parallel algorithm is called $ByRowP$.

C. ByBox (ByBox Algorithm)

The most cache efficient method developed by us for dynamic programming computes the elements in the upper

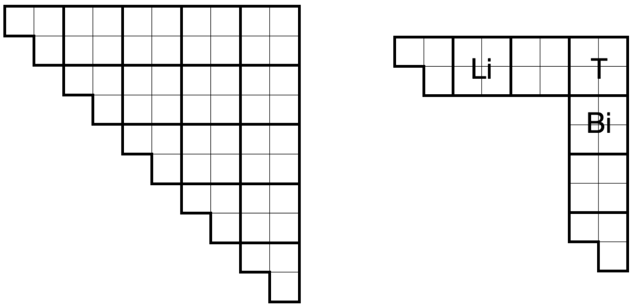


Fig. 1. Partitioning the upper triangle of Q for computation by boxes

triangle of Q by boxes rather than by rows [27], [28]. In this method, the upper triangle is divided into strips with p rows each. Each strip is divided into pxp square boxes except the leftmost partition in each strip, which is a pxp triangular box (Figure 1). p is chosen to be a multiple of the cache-line width w. Q is computed by strips bottom to top. Within a strip, the computation is done by boxes left to right. The corresponding algorithm is $ByBox$ (Algorithm 4).

Algorithm 4 ByBox

```
for r=N-p; r>=0; r=r-p do
    Calculate triangular box $(r, r+p, r, r+p)$ using ByRow
    for c=r+p; c<N; c=c+p do
        Let T be the square box $(r, r+p, c, c+p)$ that is to be computed
        Let $L_0$, $L_1$, $\cdots$, $L_{k-1}$ be the boxes to the left of $T$. ($L_0$ is the first triangular box)
        Let $B_1$, $B_2$, $\cdots$, $B_k$ be the boxes below $T$. ($B_k$ is the last triangle box)
        for t=1; t<k; t++ do
            Update $T$ using the pair $(L_i, B_i)$
        end for
        Update $T$ using the pair $(L_0, T)$ and $(B_k, T)$.
    end for
end for
```

In our parallel version $ByBoxP$, we first compute the triangular blocks. Since these blocks are independent of one another, they may be computed in parallel using one processor per block. Next, the square blocks are computed one at a time bottom to top and within a strip left to right. When computing block T (Figure 1), multiple block pairs (L_i, B_i) can be handled simultaneously.

IV. Experimental Results

A. Experimental Platforms and Test Data

We implemented the single- and multi-core versions of the algorithms $OneArray$, $ByRow$, and $ByBox$ of Section III in C; openMP was used in the parallel codes. The performance of our codes was compared to that of the codes $Orig$, $Pluto$, and $Traco$ obtained from [23]. The tile sizes for $Pluto$ and $Traco$ used by us were the defaults (16 x 16 x 16) and (1 x 128 x 16) set by Palkowski et al. in these codes, respectively. For $ByBox$ and $ByBoxP$, the box size p was set to 32. All codes were compiled using the gcc compiler with the -O3 option and run on the platform: Intel I9-7900X Ten-Core processor 3.30GHz with 14MB LLC cache (Hyper threading supported).

For test data, we used RNA sequences obtained from the National Center for Biotechnology Information (NCBI) database [30].

B. Performance on I9

Table I gives the run times, in seconds, for the single core McCaskill algorithms on our I9 platform and shows the speedup obtained by $ByBox$ relative to each of the single core single thread algorithms for each of our test sequences. $Pluto$ is the fastest of the codes in [23] when run using a single thread and both $ByRow$ and $ByBox$ are consistently faster than $Pluto$. $ByBox$ is the fastest of our codes. The run time reduction obtained by $ByBox$ relative to $Pluto$ ranges from 41.24% to 81.22% (column labeled B vs P). $ByBox$ achieves a speedup of up to approximately 5.3 relative to $Pluto$.

Table II gives the run times, in seconds, for the multi core McCaskill algorithms on our I9 platform. Times are given using both 10 threads and 20 threads. $Traco$ was consistently faster with 20 threads than with 10 threads (recall that the I9 supports hyperthreading) and $ByBoxP$ was faster with 20 threads for sequence sizes more than 4000. $ByBoxP$ was the fastest multicore algorithm on the I9 for all sequence sizes and $Traco$ came in second. When 20 threads are used, $ByBoxP$ takes between 57.36% and 84.76% less time than does $Traco$ on our RNA sequences. Speedups of up to approximately 7.1 were obtained relative to $Traco$.

V. Conclusion

Using a rewrite of McCaskill's dynamic programming equation, we have obtained a one array implementation, $OneArray$, of McCaskill's algorithm to find the partition functions of an RNA sequence. Two cache efficient versions of $OneArray$ along with parallel multi-core versions of all three of these algorithms have been developed. On our Intel I9 computational platform, our best single core algorithm, $ByBox$, takes up to 81.2% less time than the best single core algorithm in [23] and our best multi-core algorithm, $ByBoxP$, takes up to 84.7% less time than the best multi-core algorithm in [23].

Acknowledgment

This work was supported, in part, by the National Science Foundation under award 1447711.

References

[1] M. S. Waterman and T. F. Smith, "RNA secondary structure: A complete mathematical analysis," *Mathematical Biosciences*, vol. 42, no. 3-4, pp. 257–266, 1978.

[2] R. Nussinov, G. Pieczenik, J. R. Griggs, and D. J. Kleitman, "Algorithms for loop matchings," *SIAM Journal on Applied mathematics*, vol. 35, no. 1, pp. 68–82, 1978.

TABLE I
RUN TIME FOR SINGLE CORE MCCASKILL ALGORITHMS, IN SECONDS, ON I9

Seq	Size	Orig	Pluto	Traco	OneArray	ByRow	ByBox	B vs P
AH001815.2	1,000	0.65	0.52	0.93	0.380	0.433	0.304	41.24%
XR_002696555.2	2,048	11.24	7.21	14.37	4.343	2.750	2.324	67.77%
NR_076111.2	3,048	44.37	29.51	50.48	24.152	10.476	7.773	73.66%
XR_002696557.1	4,026	128.87	82.39	162.11	65.872	20.427	18.281	77.81%
AJ966883.1	5,039	275.73	188.32	337.89	148.208	45.133	35.368	81.22%

TABLE II
RUN TIME FOR MULTI CORE MCCASKILL ALGORITHMS, IN SECONDS, ON I9

		10 threads					20 threads				
Seq	Size	Traco	OneArrayP	ByRowP	ByBoxP	B vs T	Traco	OneArrayP	ByRowP	ByBoxP	B vs T
AH001815.2	1,000	0.15	0.05	1.03	0.08	47.73%	0.15	0.04	1.48	0.06	57.36%
XR_002696555.2	2,048	1.72	0.42	4.87	0.35	79.39%	1.53	0.46	7.34	0.38	75.13%
NR_076111.2	3,048	6.72	2.51	11.04	1.13	83.21%	5.57	2.20	15.50	1.24	77.77%
XR_002696557.1	4,026	18.37	7.16	21.02	2.60	85.83%	15.54	6.35	30.60	2.51	83.85%
AJ966883.1	5,039	37.47	16.51	31.09	4.87	87.00%	31.27	14.02	44.98	4.77	84.76%

[3] M. Zuker and P. Stiegler, "Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information," *Nucleic acids research*, vol. 9, no. 1, pp. 133–148, 1981.

[4] M. Zuker and D. Sankoff, "RNA secondary structures and their prediction," *Bulletin of mathematical biology*, vol. 46, no. 4, pp. 591–621, 1984.

[5] M. Zuker, "Computer prediction of RNA structure," in *Methods in enzymology*. Elsevier, 1989, vol. 180, pp. 262–288.

[6] ——, "On finding all suboptimal foldings of an RNA molecule," *Science*, vol. 244, no. 4900, pp. 48–52, 1989.

[7] J. S. McCaskill, "The equilibrium partition function and base pair binding probabilities for RNA secondary structure," *Biopolymers: Original Research on Biomolecules*, vol. 29, no. 6-7, pp. 1105–1119, 1990.

[8] S. Will, K. Reiche, I. L. Hofacker, P. F. Stadler, and R. Backofen, "Inferring noncoding RNA families and classes by means of genome-scale structure-based clustering," *PLoS computational biology*, vol. 3, no. 4, 2007.

[9] I. L. Hofacker, S. H. Bernhart, and P. F. Stadler, "Alignment of RNA base pairing probability matrices," *Bioinformatics*, vol. 20, no. 14, pp. 2222–2227, 2004.

[10] Z. J. Lu, J. W. Gloor, and D. H. Mathews, "Improved RNA secondary structure prediction by maximizing expected pair accuracy," *Rna*, vol. 15, no. 10, pp. 1805–1813, 2009.

[11] M. Palkowski and W. Bielecki, "Parallel tiled cache and energy efficient codes for o (n4) RNA folding algorithms," *Journal of Parallel and Distributed Computing*, vol. 137, pp. 252–258, 2020.

[12] J. Li, S. Ranka, and S. Sahni, "Multicore and GPU algorithms for Nussinov RNA folding," *BMC bioinformatics*, vol. 15, no. 8, p. S1, 2014.

[13] A. Mathuriya, D. A. Bader, C. E. Heitsch, and S. C. Harvey, "GTfold: a scalable multicore code for RNA secondary structure prediction," in *Proceedings of the 2009 ACM symposium on Applied Computing*, 2009, pp. 981–988.

[14] M. S. Swenson, J. Anderson, A. Ash, P. Gaurav, Z. Sükösd, D. A. Bader, S. C. Harvey, and C. E. Heitsch, "GTfold: Enabling parallel RNA secondary structure prediction on multi-core desktops," *BMC research notes*, vol. 5, no. 1, p. 341, 2012.

[15] G. Tan, N. Sun, and G. R. Gao, "A parallel dynamic programming algorithm on a multi-core architecture," in *Proceedings of the nineteenth annual ACM symposium on Parallel algorithms and architectures*, 2007, pp. 135–144.

[16] T. Estrada, A. Licon, and M. Taufer, "CompPknots: a framework for parallel prediction and comparison of RNA secondary structures with pseudoknots," in *International Symposium on Parallel and Distributed Processing and Applications*. Springer, 2006, pp. 677–686.

[17] F. Xia, Y. Dou, X. Zhou, X. Yang, J. Xu, and Y. Zhang, "Fine-grained parallel RNAalifold algorithm for RNA secondary structure prediction on FPGA," *BMC bioinformatics*, vol. 10, no. S1, p. S37, 2009.

[18] A. Jacob, J. Buhler, and R. D. Chamberlain, "Accelerating Nussinov RNA secondary structure prediction with systolic arrays on FPGAs," in *2008 International Conference on Application-Specific Systems, Architectures and Processors*. IEEE, 2008, pp. 191–196.

[19] Y. Dou, F. Xia, and J. Jiang, "Fine-grained parallel application specific computing for RNA secondary structure prediction using SCFGS on FPGA," in *Proceedings of the 2009 international conference on Compilers, architecture, and synthesis for embedded systems*, 2009, pp. 107–116.

[20] G. Rizk, D. Lavenier, and S. Rajopadhye, "GPU accelerated RNA folding algorithm," in *GPU Computing Gems Emerald Edition*. Elsevier, 2011, pp. 199–210.

[21] D.-J. Chang, C. Kimmer, and M. Ouyang, "Accelerating the Nussinov RNA folding algorithm with CUDA/GPU," in *The 10th IEEE International Symposium on Signal Processing and Information Technology*. IEEE, 2010, pp. 120–125.

[22] M. Fekete, I. L. Hofacker, and P. F. Stadler, "Prediction of RNA base pairing probabilities on massively parallel computers," *Journal of Computational Biology*, vol. 7, no. 1-2, pp. 171–182, 2000.

[23] M. Palkowski and W. Bielecki, "Parallel cache-efficient code for computing the mccaskill partition functions," in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019, pp. 207–210.

[24] U. Bondhugula, A. Hartono, J. Ramanujam, and P. Sadayappan, "A practical and fully automatic polyhedral program optimization system," in *ACM SIGPLAN PLDI*, vol. 10, no. 1375581.1375595, 2008.

[25] U. Bondhugula, M. Baskaran, S. Krishnamoorthy, J. Ramanujam, A. Rountev, and P. Sadayappan, "Automatic transformations for communication-minimized parallelization and locality optimization in the polyhedral model," in *International Conference on Compiler Construction*. Springer, 2008, pp. 132–146.

[26] M. Palkowski and W. Bielecki, "TRACO: source-to-source parallelizing compiler," *Computing and Informatics*, vol. 35, no. 6, pp. 1277–1306, 2017.

[27] C. Zhao and S. Sahni, "Cache and energy efficient algorithms for Nussinov's RNA folding," *BMC bioinformatics*, vol. 18, no. 15, p. 518, 2017.

[28] ——, "Efficient RNA folding using Zuker's method," in *2017 IEEE 7th International Conference on Computational Advances in Bio and Medical Sciences (ICCABS)*. IEEE, 2017, pp. 1–6.

[29] S. Sahni, "Data structures, algorithms, and applications in c++, second edition." Silicon Press, 2005.

[30] "Ncbi database," http://www.ncbi.nlm.nih.gov/gquery.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 453–460 DOI: 10.15439/2020F198 ISSN 2300-5963 ACSIS, Vol. 21

Analysis of asymptotic time complexity of an assumption-free alternative to the log-rank test

Lubomír Štěpánek
Department of Statistics and Probability
Faculty of Informatics and Statistics
University of Economics
nám. W. Churchilla 4, 130 67 Prague, Czech Republic
lubomir.stepanek@vse.cz
&
Institute of Biophysics and Informatics
First Faculty of Medicine
Charles University
Salmovská 1, Prague, Czech Republic
lubomir.stepanek@lf1.cuni.cz

Filip Habarta
Department of Statistics and Probability
Faculty of Informatics and Statistics
University of Economics
nám. W. Churchilla 4, 130 67 Prague, Czech Republic
filip.habarta@vse.cz

Ivana Malá
Department of Statistics and Probability
Faculty of Informatics and Statistics
University of Economics
nám. W. Churchilla 4, 130 67 Prague, Czech Republic
malai@vse.cz

Luboš Marek
Department of Statistics and Probability
Faculty of Informatics and Statistics
University of Economics
nám. W. Churchilla 4, 130 67 Prague, Czech Republic
marek@vse.cz

***Abstract*—Comparison of two time-event survival curves representing two groups of individuals' evolution in time is relatively usual in applied biostatistics. Although the log-rank test is the suggested tool how to face the above-mentioned problem, there is a rich statistical toolbox used to overcome some of the properties of the log-rank test. However, all of these methods are limited by relatively rigorous statistical assumptions.**

In this study, we introduce a new robust method for comparing two time-event survival curves. We briefly discuss selected issues of the robustness of the log-rank test and analyse a bit more some of the properties and mostly asymptotic time complexity of the proposed method. The new method models individual time-event survival curves in a discrete combinatorial way as orthogonal monotonic paths, which enables direct estimation of the p-value as it was originally defined. We also gently investigate how the surface of an area, bounded by two survival curves plotted onto a plane chart, is related to the test's p-value. Finally, using simulated time-event data, we check the robustness of the introduced method in comparison with the log-rank test.

Based on the theoretical analysis and simulations, the introduced method seems to be a promising and valid alternative to the log-rank test, particularly in case on how to compare two time-event curves regardless of any statistical assumptions.

I. Introduction

IN SURVIVAL analysis, the response variable is usually two-dimensional, since it takes into account both the time of the event of our interest and whether the event (or the censoring) even occurred. More than intuitively, such a target variable suggests being plotted in a two-dimensional plot. As usual, while a number of subjects who do not evinced the event of interest to all subjects is plotted on a vertical axis at a given time point, the time points where the event occurred are aligned with the horizontal axis, see also Fig. 1 That is the way how Kaplan-Meier estimators are commonly illustrated [1]. Therefore, the survival curve as a response variable could be represented as a monotonic orthogonal path, i. e., a polygonal path of a finite number of horizontal and vertical segments, in the Cartesian two-dimensional chart. Since such a variable deals both with the events of interest and their times, it is ordinarily called the time-event (survival) curve.

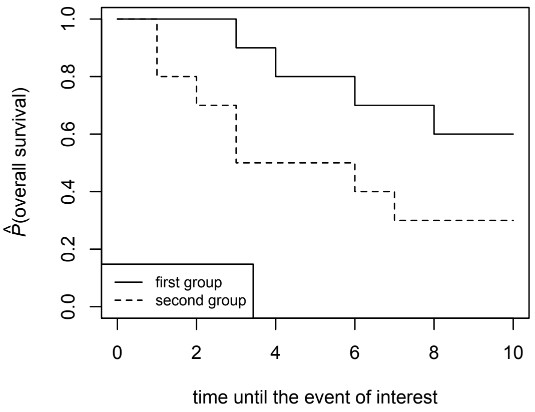


Fig. 1. Two time-event survival curves in a survival plot.

Whenever two or more time-event survival curves, describing evolution of events in time within two groups of individuals, are to be compared, several well-established methods could be used. A classical log-rank test could solve the problem, when there are only two groups supposed to be compared [2]. Assuming some special settings, particularly when the time-event survival curves are constructed using data that are not censored, i. e. the data fully describe all events of interest occurred in the groups, then a simple Wilcoxon rank-sum test might be applied. If more than two groups are to be compared, the problem could be battled using either a score-rank test, or even a Cox proportional hazards model [3]. All the approaches mentioned above may be performed in various software, including R language and environment [4], such that a pure R package `stats` or a package `survival` [5] could be employed to do the job. Nevertheless, each one of the described methods has its limitations, and its application is determined by meeting relatively rigorous statistical assumptions.

Application of the log-rank test that compares two non-crossing time-event survival curves (similar as plotted in Fig. 1) is limited mostly by assuming the fact that censoring (i) should not induce anyhow the observed events and (ii) is equally likely to occur in both the groups. What is more, the counts of events of interest should be large enough to satisfy asymptotic properties of χ^2 distribution and fulfill the central limit theorem to let the log-rank test statistic follow an asymptotically normal distribution. That means an incidence of the events of interest in each group across all the time points should be neither too small nor too large.

To overcome the limitations of the classical log-rank test, several diverse modifications of the log-rank test were published to either increase efficiency of the test, or its robustness against violation of the statistical assumptions, or both. Whereas Kong (1997) in [6] adjusted the log-rank test efficiency by improving the hazard functions, i. e. functions of rates of events based on fixed proportions of the events in the past, Song et al. (2008) dived deeper into covariate matrix decomposition, by which they derived formulas estimating minimal sample sizes that enables a valid usage of the log-rank test [7]. Several authors such as Peto and Peto (1972) [8], Yang and Prentice (2010) [9], and Li (2018) [10], suggested the usage of weights of individual observations, usually lower weights for later events when there the numbers of observations tend to be not so high; by this they improve the validity of the log-rank test outputs.

There are also articles handling with exact discrete calculations when compare two survival curves which is more similar to our proposed approach. Thomas (1975) simplified the computations by fixing total numbers in the compared groups [11]. The algorithm was improved a bit computationally by Mehta et al. (1985) [12]. Finally, Heinze et al. (2003), similarly asymptotic approaches above, incorporated a weighting scheme into the calculations to increase significance of earlier observations [13].

Studies that go deeper into asymptotic complexities of the statistical inference test, particularly the exact ones that exhaustively compute over a polynomial universe, are generally missing. Some significant pieces of related knowledge focused on complexity of classic but robust and computationally-hard inference tests are discussed by Mosler (2002) [14], Smolinski et al. in (2008) [15], and Kulikov et al. (2014) [16].

Vast majority of the papers listed above work with a hazard function, which is event of interest rate in a given time point conditional on overall survival rate until the time point or they assume constant total numbers of subjects in all the compared groups. Unlike them, in this proceeding, besides a brief discussion on limitations of the log-rank test, we model the time-event survival curves using a discrete combinatorial approach, considering the survival curves to be orthogonal monotonic paths on a plane of two-dimensional plot (as shown in Fig. 1 and Fig. 2), and taking into account their mutual "Manhattan" grid distances. That indicates how easily the p-value of this modified log-rank test could be calculated using its original statistical definition as a conditional probability of observing data of given properties. Then we analyse asymptotic time complexity of algorithmic approaches behind the proposed method. We also briefly discuss the possible relationship between the two-dimensional surface bounded by two non-crossing survival curves in the plot and the test's p-value. Finally, using simulations of artificial survival curves, the first type errors as rates of detection the cases, when similar curves are supposed to be different, are estimated for both the log-rank test and our proposed alternative, mutually compared and discussed within the frame of the robustness of the methods.

II. Principles, assumptions and limitations of the log-rank test

Firstly, we gently introduce principles of the log-rank test, by which we can better understand its assumptions and limitations.

A. Principles of the log-rank test

Let's assume two groups of individuals (marked by indices 1, and 2, respectively) and $k \in \mathbb{N}$ distinct event times. At each event time, we can construct a 2×2 contingency table and compare the event rates between the two groups. Let the $(t_1, t_2, \ldots, t_k)^T$ be an ordered tuple of the event time points, then for the j-th event time t_j, such that $j \in \{1, 2, 3, \ldots, k\}$, we can construct the (contingency) table Tab. I. At j-th event time, there are $d_{1,j}$ and $d_{2,j}$ individuals who experienced the events in the group 1 and 2, respectively, and $r_{1,j}$ and $r_{2,j}$ subjects at risk (who have not yet had the event or been censored) in the groups 1 and 2, respectively, see Tab. I.

TABLE I
Numbers of the events of interest in both groups at time t_j.

	event of interest at the event time t_j		
group	yes	no	total
1	$d_{1,j}$	$r_{1,j} - d_{1,j}$	$r_{1,j}$
2	$d_{2,j}$	$r_{2,j} - d_{2,j}$	$r_{2,j}$
total	d_j	$r_j - d_j$	r_j

The log-rank test checks the null hypothesis H_0 that both groups have identical hazard functions, i. e. that rates of the events of interest in time conditional on fixed rates in the past are the same. Under the null hypothesis H_0, the observed numbers of the events could be considered as random variables $D_{1,j}$ and $D_{2,j}$ following a hypergeometric distribution with parameters $(r_j, r_{i,j}, d_j)$ for both $i \in \{1,2\}$. Thus, the expected value of the variable $D_{i,j}$ is $\mathbb{E}(D_{i,j}) = r_{i,j}\frac{d_j}{r_j}$ and variance is $\mathrm{var}(D_{i,j}) = \frac{r_{1,j}r_{2,j}d_j}{r_j^2}\left(\frac{r_j - d_j}{r_j - 1}\right)$ for both $i \in \{1,2\}$. For all $j \in \{1,2,3,\dots,k\}$ we can compare the observed numbers of events of interest, $d_{i,j}$, to their expected values $\mathbb{E}(D_{i,j}) = r_{i,j}\frac{d_j}{r_j}$, under H_0. So, the test statistic for both $i \in \{1,2\}$ is finally

$$\chi^2_{\text{log-rank}} = \frac{\left(\sum_{j=1}^{k} d_{i,j} - \mathbb{E}(D_{i,j})\right)^2}{\sum_{j=1}^{k} \mathrm{var}(D_{i,j})} = \\ = \frac{\left(\sum_{j=1}^{k} d_{i,j} - r_{i,j}\frac{d_j}{r_j}\right)^2}{\sum_{j=1}^{k} \frac{r_{1,j}r_{2,j}d_j}{r_j^2}\left(\frac{r_j - d_j}{r_j - 1}\right)}, \tag{1}$$

which follows under H_0 a χ^2 distribution with 1 degree of freedom, $\chi^2_{\text{log-rank}} \sim \chi^2(1)$. For feasible large r_j, at least $r_j \geq 30$, a square root of $\chi^2_{\text{log-rank}}$ follows a standard normal distribution, $\sqrt{\chi^2_{\text{log-rank}}} \sim \mathcal{N}(0, 1^2)$.

B. Some of the assumptions and limitations of the log-rank test

Firstly, censoring is assumed not to affect anyhow the occurrence of event of interest, and the proportion of censored data are supposed being of nearly equal size in both the groups, as well. Otherwise, the test statistic $\chi^2_{\text{log-rank}}$ calculated using (1) either for $i = 1$, or for $i = 2$, respectively, could be biased and therefore mutually different. That may affect the interpretability, i. e. the robustness of the log-rank test applied on such data.

Then, since the test statistic $\chi^2_{\text{log-rank}}$ follows a χ^2 distribution, the initial total number of individuals r_0 and the number of all event times k should be large enough. Analogously but inversely, whenever the numbers of individuals d_j experiencing the event of interest are generally large (relatively to r_j), than both the numerator and denominator of the fraction in the formula (1) is relatively small, too, and, consequently, one could expect that the $\chi^2_{\text{log-rank}}$ statistic (or the derived $\sqrt{\chi^2_{\text{log-rank}}}$ statistic) does not fulfil its assumed asymptotic properties, and its estimate could be thus biased. That might influence both the robustness and the power of the log-rank test when applied to data of such limitations.

By researching the denominator of the equation (1) a bit deeper, we can realize the test statistic $\chi^2_{\text{log-rank}}$ is the highest when the denominator $\sum_{j=1}^{k} \mathrm{var}(D_{i,j})$ is as low as possible given the values $d_{i,j}$ and $r_{i,j}$ for all $i \in \{1,2\}$ and $j \in \{1,2,3,\dots,k\}$. It is worth mentioning this holds just when the proportions $\frac{r_{1,j}}{r_j} = \frac{r_{1,j}}{r_{1,j}+r_{2,j}}$ and $\frac{r_{2,j}}{r_j} = \frac{r_{2,j}}{r_{1,j}+r_{2,j}}$ are both constant (and mutually different enough) across all the time points $(t_1, t_2, \dots, t_k)^T$, and then the log-rank test is the most powerful; i. e. in other words, its ability to reject the null hypothesis H_0, claiming the survival curves are equivalent, when they are in fact different, is maximal possible. That used to be the most usual issue that may decrease the power of the log-rank test. The mentioned proportions are typically not constant when the time event curves change a lot their mutual distance across the time points or when they even cross themselves one or more times. Consequently, the power of the log-rank test may be decreased by any deviations from the constant values of the proportions $\frac{r_{1,j}}{r_j}$, and $\frac{r_{2,j}}{r_j}$, respectively.

III. Introduction of an assumption-free alternative to the log-rank test

Within this section, we introduce an assumption-free alternative to the log-rank test. The alternative algorithm for two time-event curves comparison is based on a discrete combinatorial calculation of possible states (i. e. all possible time-event curves) that would be theoretically obtained and that are at least as extreme as the original two survival curves. This approach corresponds to an original definition of a p-value as a probability of obtaining data at least as extreme as the data currently observed, assuming that the null hypothesis is true (i. e. the observed survival curves are not statistically different).

All the possible states could be considered as monotonic orthogonal paths in the two-dimensional chart of two original survival curves, excluding (for simplicity) the crossing curves. By calculating (or estimating) the numbers of all the paths at least at extreme as the plotted two curves, i. e. all the paths such that one is above the first observed one and the other is below the second observed one, we get a point estimate of the p-value as a proportion of all pairs of orthogonal paths contradicting the same way or even more to the observed survival curves. Or in other words, as a proportion of all pairs of orthogonal paths that are at least as distant one from the other than the original two time-event curves.

A. Principle of the proposed assumption-free alternative to the log-rank test

Again, let two groups of individuals (marked by indices 1, and 2, respectively) to be compared and $k \in \mathbb{N}$ distinct event times when events of interest could occur. Let the $(t_1, t_2, \dots, t_k)^T$ be an ordered tuple of the event times. At each event time, we can compute the number of individuals who experienced the event at the j-th event time t_j for both groups, similarly to the construction of contingency tables, as shown in table Tab. I. By repeating this approach k times, consequently, once we get the proportions of subjects at risk, $\frac{r_{1,j}}{r_j}$, and $\frac{r_{2,j}}{r_j}$, respectively, for each event time t_j, we could plot the time-event survival curves based on the proportions of individual in risk $\frac{r_{1,j}}{r_j}$, and $\frac{r_{2,j}}{r_j}$ similarly to Fig. 1.

For simplicity, the survival curves are assumed not to cross themselves. More technically spoken, it for each j-th event time t_j holds

$$\frac{r_{1,j}}{r_j} \geq \frac{r_{2,j}}{r_j}, \tag{2}$$

as illustrated in Fig. 1. By adding a grid into the Fig. 1, we get Fig. 2, which is a bit closer to an idea of calculating (or estimating) a number of monotonic orthogonal paths starting at the proportion of subjects at risk $\frac{r_{i,0}}{r_0} = 1$ and ending — after k event times — at the proportion of subjects at risk $\geq \frac{r_{i,k}}{r_k}$ (one of such possible paths is the blue line for $i = 1$ in Fig. 2) $\leq \frac{r_{i,k}}{r_k}$ (similarly to the red line for $i = 2$ in Fig. 2).

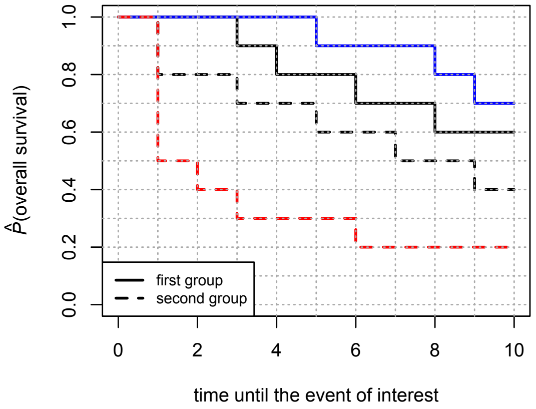


Fig. 2. Two original time-event survival curves in a survival plot (black lines) and an example of a pair of monotonic orthogonal paths such that one is above (blue solid line, $i = 1$) the upper original survival curve and the second one is below (red dashed line, $i = 2$) the lower original survival curve.

Let $N_{(1,k,u,v)}$ stands for the number of all orthogonal paths (respecting the grid, i. e. all segments of such a path are parallel to horizontal or vertical lines of the grid and its edges are aligned to grid points) starting at the proportion 1 (left upper corner of the Fig. 2) and ending after k event times at the proportion of subjects at risk $\frac{u}{v}$ (a point with coordinates $[k, \frac{u}{v}]$ in Fig. 2). Eventually, let $N^{+}_{(1,k,u,v)}$ be a number of all orthogonal paths starting at the proportion 1, going above the 1–st survival curve or tangentially meeting it (without crossing it) and ending at the proportion of subjects at risk $\geq \frac{u}{v}$. Analogously, let $N^{-}_{(1,k,u,v)}$ be a number of all orthogonal paths starting at the proportion 1, going below the 2–nd survival curve or tangentially meeting it (without crossing it) and ending at the proportion of subjects at risk $\leq \frac{u}{v}$ after k event times. The numbers $N^{+}_{(1,k,u,v)}$ and $N^{-}_{(1,k,u,v)}$ could be computed perhaps exhaustively in a combinatorial way (this is an open problem) or could definitely be estimated by numerical simulations.

Let us define a null hypothesis H_0 that claims the original (observed) survival curves are not significantly different. On of the tricky part on the proposed method is that, since we do not need any more initial assumptions for this testing, we also do not require modelling a null distribution. The p-value, as mentioned above, is the probability of obtaining data (expected survival curves described as monotonic orthogonal paths in the survival plot) at least as extreme as the data currently observed (the two original survival curves), assuming that the null hypothesis H_0 is correct. Following the definition of the p-value and marking it as p, we get

$$p = p\text{-value}$$
$$p = P(\text{getting data at least as extreme as the observed}|H_0)$$
$$p = P\left(\frac{N^{+}_{1,k,r_{1,k},r_k} \cdot N^{-}_{2,k,r_{2,k},r_k}}{\left(\sum_{j=0}^{r_k} N_{k,j,r_k}\right)^2 - N_{\text{cc}},}\right) \tag{3}$$

where N_{cc} is a number of pairs of survival curves crossing each other. Again, the number N_{cc} can be calculated probably either using a a discrete combinatorial analysis, or be numerically simulated (which is far easier).

In comparison with the term in the denominator of the equation (3), the number of pairs of survival curves depicted by the numerator can not include any crossing curves. Since we assume all curves ending in the proportion $\frac{r_{1,k}}{r_k}$ or greater, and all curves ending in the proportion $\frac{r_{2,k}}{r_k}$ or lower, considering that $\frac{r_{1,j}}{r_j} \geq \frac{r_{2,j}}{r_j}$ for each $j \in \{1, 2, 3, \ldots, k\}$ as stated in (2), thus, since $\frac{r_{1,k}}{r_k} \geq \frac{r_{2,j}}{r_j}$ for all time points, there are no pairs of crossing curves taken into account in the numerator of (3). The curves could tangentially meet themselves (in case of $=$) or run one above the other (in case of $>$), but could not cross each other.

B. A brief analysis of surface bounded by two non-crossing survival curves and the test's p-value

Surfaces above the first, upper survival curve (let us mark it as S_1^{+}) and below the second, bottom curve (let us mark it as S_2^{-}) in Fig. 2 suggest investigating on how are the surfaces related to the p-value of the test.

By following the first impression, when S stands for a surface of the whole canvas of the chart in Fig. 2, it seems that p-value is proportional to the term $\frac{S_1^{+}+S_2^{-}}{S}$. However, the relationship between the p-value and the surfaces is more complex and not so straightforward. The numbers of all orthogonal paths in some dedicated surface, let us assume e. g. $N^{+}_{(1,k,u,v)}$, is not proportional to the size of the surface. As a sketch of a proof by contradiction, let us suppose we are to calculate the number of $N^{-}_{(1,k,0,v)}$ curves below a horizontal curve crossing the point $[k, v]$. Then, simply using combinatorial rules, we realize that $N^{-}_{(1,k,0,v)} = \binom{k+v}{k}$. However, if we now want to calculate the number of $N^{-}_{(1,k,0,2v)}$ curves below a horizontal curve crossing the point $[k, 2v]$, we get that $N^{-}_{(1,k,0,2v)} = \binom{k+2v}{k}$. Whereas the proportion of the surfaces below the two lines crossing the points $[k, 2v]$ and $[k, v]$ is equal to 2, the proportion of the numbers of the paths is in general much greater than 2, since generally $\frac{\binom{k+2v}{k}}{\binom{k+v}{k}} \gg 2$. Thus, $\frac{N^{+}_{1,k,r_{1,k},r_k}}{N^{-}_{2,k,r_{2,k},r_k}} \neq \frac{S_1^{+}}{S_2^{-}}$ in general and the p-value is not (!) proportional to the term $\frac{S_1^{+}+S_2^{-}}{S}$.

C. Approaches on calculation the p-value of the proposed alternative to the log-rank test

The terms such as $N^{+}_{1,k,r_{1,k},r_k}$, $N^{-}_{2,k,r_{2,k},r_k}$, $\sum_{j=0}^{r_k} N_{k,j,r_k}$, and N_{cc}, respectively, in equation (3) could by estimated either numerically by re-sampling, or calculated exhaustively. A fully analytical approach is under current research.

Numerical estimation. All the terms such as $N^{+}_{1,k,r_{1,k},r_k}$, $N^{-}_{2,k,r_{2,k},r_k}$, $\sum_{j=0}^{r_k} N_{k,j,r_k}$, and N_{cc}, respectively, in equation (3) could be numerically estimated by *re-sampling* approach. Let us assume we got two non-crossing survival curves similarly to the plot in Fig. 1, so that we know the values k, $r_{1,j}$, $r_{2,j}$, $d_{1,j}$, and $d_{2,j}$ for all $j \in \{1, 2, 3, \ldots, k\}$. Let us suppose we generate n pairs of survival curves. Then, let $N(\forall+, \forall-)(n)$ be the number of all pairs such that one of the curves is completely above the first original curve and the other is completely below the second original curve (and, thus, they do not cross each other), in all n generated pairs. Let $N(\text{non-crossing})(n)$ be the number of all pairs such that the curves of the pair do not cross each other, in all n generated pairs. Then we can simply derive that

$$N^{+}_{1,k,r_{1,k},r_k} \cdot N^{-}_{2,k,r_{2,k},r_k} = \lim_{n\to\infty} N(\forall+, \forall-)(n)$$

$$\left(\sum_{j=0}^{r_k} N_{k,j,r_k}\right)^2 - N_{\text{cc}} = N(\text{non-crossing})(n),$$

and, consequently, by replacing in equation (3)

$$\hat{p} = \lim_{n\to\infty} \frac{N(\forall+, \forall-)(n)}{N(\text{non-crossing})(n)}.$$

By this *re-sampling* approach, we can obtain for reasonably large $n \in \mathbb{N}$ an unbiased estimate of p-value in equation (3) of the proposed alternative to the log-rank test. The algorithm is also described in Algorithm 1

Algorithm 1: Re-sampling approach on how to obtain for reasonably large $n \in \mathbb{N}$ an unbiased estimate of p-value in equation (3) of the proposed alternative to the log-rank test.

```
Data: two non-crossing survival curves
Result: an unbiased estimate of $p$-value in equation (3)
        of the proposed alternative to the log-rank test
$k, r_{1,j}, r_{2,j}, d_{1,j}, d_{2,j}$   // parameters of the original two survival curves ;
$n$   // number of repetitions;
$N(\forall+, \forall-)(0) = 0$   // number of all pairs such that one of the curves is completely above the first original curve and the other is completely below the second original curve;
$N(\text{non-crossing})(0) = 0$   // number of all pairs such that the curves of the pair do not cross each other;
for $j = 1 : n$ do
    generate a pair of two survival curves;
    if the curves of the pair do not cross each other then
        $N(\text{non-crossing})(j) = N(\text{non-crossing})(j) + 1$;
        if one of the curves is completely above the first original curve and the other is completely below the second original curve then
            $N(\forall+, \forall-)(j) = N(\forall+, \forall-)(j) + 1$;
end
calculate an estimate of $p$-value as
    $\hat{p} = \frac{N(\forall+, \forall-)(n)}{N(\text{non-crossing})(n)}$ ;
```

Exhaustive approach. Let us again assume we got two non-crossing survival curves similarly to the plot in Fig. 1, so that we know the values k, $r_{1,j}$, $r_{2,j}$, $d_{1,j}$, and $d_{2,j}$ for all $j \in \{1, 2, 3, \ldots, k\}$. The exhaustive, greedy approach is based on grid search for all possible pairs of survival curves such that one of the curves is completely above the first original curve and the other is completely below the second original curve (and, thus, they do not cross each other). In case the exhaustive approach is finished successfully, one could obtain more confident estimate of p-value in equation (3) of the proposed alternative to the log-rank test than in case of the numerical re-sampling.

Since the exhaustive approach is greedy, so that one could expect a large asymptotic time complexity, we enumerated worst-case scenarios estimates of all terms such as $N^{+}_{1,k,r_{1,k},r_k}$, $N^{-}_{2,k,r_{2,k},r_k}$, $\sum_{j=0}^{r_k} N_{k,j,r_k}$, and N_{cc}, respectively, in equation (3).

Since $N^{+}_{(1,k,r_{1,k},r_k)}$ (or $N^{-}_{(2,k,r_{2,k},r_k)}$) is a number of all orthogonal paths starting at the proportion 1, going above (or below) the 1–st (or the 2-nd) survival curve or tangentially meeting it (without crossing it) and ending at the proportion of subjects at risk $\geq \frac{u}{v}$, number of such paths could not be larger than a number of all monotonic orthogonal paths in a rectangle of size $k \times d_{1,k}$ (or $k \times d_{2,k}$). Then,

$$N^{+}_{1,k,r_{1,k},r_k} \leq \sum_{j=0}^{d_{1,k}} \binom{k+j}{k} = \binom{d_{1,k}+k+1}{k+1}$$

$$N^{-}_{2,k,r_{2,k},r_k} \leq \sum_{j=0}^{r_{2,k}} \binom{k+j}{k} - \sum_{j=0}^{d_{2,k}} \binom{k+j}{k} = \\ = \binom{r_{2,k}+k+1}{k+1} - \binom{d_{2,k}+k+1}{k+1}.$$

Number of all monotonic orthogonal paths in the grid, $\sum_{j=0}^{r_k} N_{k,j,r_k}$, is by assuming (for simplicity) $r_{1,k} = r_{2,k}$ similarly

$$\sum_{j=0}^{r_k} N_{k,j,r_k} \leq \sum_{j=0}^{r_{2,k}} \binom{k+j}{k} = \binom{r_{2,k}+k+1}{k+1}.$$

Since parts of crossing curves in a pair could be rearranged such that the crossing segments could be "re-coloured" eventually, i. e. switched so that the curves only tangentially meet each other keeping them monotonic, we can assume that $N_{\text{cc}} \ll \sum_{j=0}^{r_k} N_{k,j,r_k}$.

Putting all the derivations together, we can estimate an upper estimate $\Theta(\bullet)$ of all the monotonic orthogonal paths' grid searching by the formula

$$\begin{aligned}
\Theta(\bullet) = \quad & \Theta\left(N^{+}_{1,k,r_{1,k},r_k}\right) + \Theta\left(N^{-}_{2,k,r_{2,k},r_k}\right) + \\
& + \Theta\left(\sum_{j=0}^{r_k} N_{k,j,r_k}\right) + \Theta\left(N_{\text{cc}}\right) = \\
= \quad & \Theta\left(\binom{d_{1,k}+k+1}{k+1}\right) + \\
& + \Theta\left(\binom{r_{2,k}+k+1}{k+1} - \binom{d_{2,k}+k+1}{k+1}\right) + \\
& + \Theta\left(\binom{r_{2,k}+k+1}{k+1}\right) + \\
& + \Theta(0) = \\
= \quad & \Theta\left((k+d_{1,k}/2)^{d_{1,k}}\right) + \\
& + \Theta\left((k+r_{2,k}/2)^{r_{2,k}} - (k+d_{2,k}/2)^{d_{2,k}}\right) + \\
& + \Theta\left((k+r_{2,k}/2)^{r_{2,k}}\right).
\end{aligned}$$

Since we assume $r_k = r_{1,k} = r_{2,k}$, then there is $r_{1,k} \geq d_{1,k}$ and the final worst-case scenario's asymptotic time complexity of p-value exhaustive calculation using the proposed method alternative to the log-rank test is equal to $\Theta(\bullet) = \Theta\left((k + r_k/2)^{r_k}\right)$.

In comparison to the novel method's time complexity, the log-rank test's $\chi^2_{\text{log-rank}}$ statistic based on equation (1) is significantly simpler, considering its asymptotic time complexity. Both by inspecting the numerator and denominator of the fraction in (1), we can see the calculation is based only on two summations of k terms, so the asymptotic time complexity of the log-rank test's $\chi^2_{\text{log-rank}}$ statistic as about $\Theta(k)$, where k is number of time points where the event of interest may occur.

Analytical approach. At the current moment, a fully analytical approach on how to calculate the term $N^{+}_{1,k,r_{1,k},r_k}$, $N^{-}_{2,k,r_{2,k},r_k}$, $\sum_{j=0}^{r_k} N_{k,j,r_k}$, and N_{cc}, respectively, in equation (3) is an open problem and requires authors' ongoing research.

IV. Simulation study

We compared the log-rank test and the assumption-free method proposed above by simulating many pairs of random non-crossing curves, assuming the curves in the pairs are not significantly different. Then, we calculated the first type errors rates, i. e. rates of the situations, when the inference test (either the log-rank test, or the new method) claims that two statistically similar survival curves are (falsely) detected as different. Finally, we assume that more robust the method is, the lower value of the first type error it should return.

The simulation study was performed using R programming language and environment [4]. There is more on numerical applications of R programming language to various fields in [17]–[21].

When generating the pairs of survival curves, we applied the following negatively exponential survival function,

$$s(t) = \sigma\left(e^{-\frac{10+\varepsilon}{10000}t}\right)$$

where ε is a random noise term that follows a standard normal distribution, i. e. $\varepsilon \sim \mathcal{N}(0, 1^2)$, and $\sigma(\bullet)$ is a function rounding its argument to the nearest integer, e. g. $\sigma(4.3) = 4$, $\sigma(4.5) = 5$ or $\sigma(5.8) = 6$.

There were $n = 1000$ pairs of significantly non-different survival curves generated in total and within each pair, the curves were compared using both the log rank test, and the above-proposed method. By summing up numbers of cases where p-value was lower than or equal to 0.05 regardless of the method, we got the point estimates of the first type error frequencies as illustrated in Tab. II.

TABLE II
Point estimates of the first type error rates for the log-rank test and the proposed method, based on the simulation described above.

	method	
	the log-rank test	the proposed method
# of simulated cases in total	1000	1000
# of cases p-value ≤ 0.05	54	15
first type error rate estimate	0.054	0.015

Whereas the log-rank test output a point estimate of the first type error rate about 0.054, the method introduced above returned a point estimate of the first type error rate about 0.015, therefore lower than the one for the log-rank test. Thus, the proposed method seems to be more robust than the log-rank test, based on the simulation described above. The first type error settings follows the common value of the alpha level equal to 0.050, as usual in applied sciences.

V. Conclusion remarks

By calculation of monotonic orthogonal paths in the grid of survival plot, we can get a ratio of the number of all pairs of the paths that are more distant one to each other, which opposes the null hypothesis, and the number of all non-crossing pairs of possible paths. This is a suggested point estimate of the p-value of the proposed alternative to the classical log-rank test.

Based on the simulation, the introduced method proved to be of higher robustness than the log-rank test. So, the assumption-free version of the log-rank test seems to be a valid alternative for the comparison of two time-event curves. However, while the numerical estimation of the p-value seems to be relatively simple-to-follow, exhaustive (greedy) calculation of exact values is of a high asymptotic time complexity and analytical derivations of the p-value formula requires following research.

Besides, the method and the computational aspects could also be a topic for a new R package development.

VI. Acknowledgement

This research was supported by the grant no. F/45/2020 provided by Internal grant agency of University of Economics, Prague.

References

[1] E. L. Kaplan and Paul Meier. "Nonparametric Estimation from Incomplete Observations". In: *Journal of the American Statistical Association* 53.282 (June 1958), pp. 457–481. DOI: 10.1080/01621459.1958.10501452. URL: https://doi.org/10.1080/01621459.1958.10501452.

[2] Nathan Mantel. "Evaluation of survival data and two new rank order statistics arising in its consideration". In: *Cancer chemotherapy reports* 3.50 (1966), pp. 163–170.

[3] Huimin Li, Dong Han, Yawen Hou, et al. "Statistical Inference Methods for Two Crossing Survival Curves: A Comparison of Methods". In: *PLOS ONE* 10.1 (Jan. 2015). Ed. by Zhongxue Chen, e0116774. DOI: 10.1371/journal.pone.0116774. URL: https://doi.org/10.1371/journal.pone.0116774.

[4] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2017. URL: https://www.R-project.org/.

[5] Therneau T. *survival: A Package for Survival Analysis in R*. Vienna, Austria, R package version 3.1-12. URL: https://CRAN.R-project.org/package=survival/.

[6] F. Kong. "Robust covariate-adjusted logrank tests". In: *Biometrika* 84.4 (Dec. 1997), pp. 847–862. DOI: 10.1093/biomet/84.4.847. URL: https://doi.org/10.1093/biomet/84.4.847.

[7] Rui Song, Michael R. Kosorok, and Jianwen Cai. "Robust Covariate-Adjusted Log-Rank Statistics and Corresponding Sample Size Formula for Recurrent Events Data". In: *Biometrics* 64.3 (Dec. 2007), pp. 741–750. DOI: 10.1111/j.1541-0420.2007.00948.x. URL: https://doi.org/10.1111/j.1541-0420.2007.00948.x.

[8] Richard Peto and Julian Peto. "Asymptotically Efficient Rank Invariant Test Procedures". In: *Journal of the Royal Statistical Society. Series A (General)* 135.2 (1972), p. 185. DOI: 10.2307/2344317. URL: https://doi.org/10.2307/2344317.

[9] Song Yang and Ross Prentice. "Improved Logrank-Type Tests for Survival Data Using Adaptive Weights". In: *Biometrics* 66.1 (Apr. 2009), pp. 30–38. DOI: 10.1111/j.1541-0420.2009.01243.x. URL: https://doi.org/10.1111/j.1541-0420.2009.01243.x.

[10] Chenxi Li. "Doubly robust weighted log-rank tests and Renyi-type tests under non-random treatment assignment and dependent censoring". In: *Statistical Methods in Medical Research* 28.9 (July 2018), pp. 2649–2664. DOI: 10.1177/0962280218785926. URL: https://doi.org/10.1177/0962280218785926.

[11] Donald G. Thomas. "Exact and asymptotic methods for the combination of 2 × 2 tables". In: *Computers and Biomedical Research* 8.5 (Oct. 1975), pp. 423–446. DOI: 10.1016/0010-4809(75)90048-8. URL: https://doi.org/10.1016/0010-4809(75)90048-8.

[12] Cyrus R. Mehta, Nitin R. Patel, and Robert Gray. "Computing an Exact Confidence Interval for the Common Odds Ratio in Several 2 × 2 Contingency Tables". In: *Journal of the American Statistical Association* 80.392 (Dec. 1985), p. 969. DOI: 10.2307/2288562. URL: https://doi.org/10.2307/2288562.

[13] Georg Heinze, Michael Gnant, and Michael Schemper. "Exact Log-Rank Tests for Unequal Follow-Up". In: *Biometrics* 59.4 (Dec. 2003), pp. 1151–1157. DOI: 10.1111/j.0006-341x.2003.00132.x. URL: https://doi.org/10.1111/j.0006-341x.2003.00132.x.

[14] Karl Mosler. *Multivariate dispersion, central regions, and depth : the lift zonoid approach*. New York: Springer, 2002. ISBN: 0387954120.

[15] Tomasz Smolinski. *Computational intelligence in biomedicine and bioinformatics : current trends and applications*. Berlin: Springer, 2008. ISBN: 978-3-540-70776-9.

[16] Alexander Kulikov. *Combinatorial pattern matching : 25th annual symposium, CPM 2014 Moscow, Russia, June 16-18, 2014, proceedings*. Cham: Springer, 2014. ISBN: 978-3-319-07565-5.

[17] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. "Evaluation of facial attractiveness for purposes of plastic surgery using machine-learning methods and image analysis". In: *2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom)*. IEEE, Sept. 2018. DOI: 10.1109/healthcom.2018.8531195. URL: https://doi.org/10.1109/healthcom.2018.8531195.

[18] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. "Machine-learning at the service of plastic surgery: a case study evaluating facial attractiveness and emotions using R language". In: *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems*. IEEE, Sept. 2019. DOI: 10.15439/2019f264. URL: https://doi.org/10.15439/2019f264.

[19] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. "Evaluation of Facial Attractiveness after Undergoing Rhinoplasty Using Tree-based and Regression Methods". In: *2019 E-Health and Bioengineering Conference (EHB)*. IEEE, Nov. 2019. DOI: 10.1109/ehb47216.2019.8969932. URL: https://doi.org/10.1109/ehb47216.2019.8969932.

[20] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. "Machine-Learning and R in Plastic Surgery – Evaluation of Facial Attractiveness and Classification of Facial Emotions". In: *Advances in Intelligent Systems and Computing*. Springer International Publishing, Sept. 2019, pp. 243–252. DOI: 10.1007/978-3-030-30604-

5_22. URL: https://doi.org/10.1007/978-3-030-30604-5_22.

[21] Patricia Martinková, Lubomír Štěpánek, Adéla Drabinová, et al. "Semi-real-time analyses of item characteristics for medical school admission tests". In: *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*. Ed. by M. Ganzha, L. Maciaszek, and M. Paprzycki. Vol. 11. Annals of Computer Science and Information Systems. IEEE, 2017, pp. 189–194. DOI: 10.15439/2017F380. URL: http://dx.doi.org/10.15439/2017F380.

11^{th} Workshop on Scalable Computing

THE world of large-scale computing continuously evolves. The most recent addition to the mix comes from numerous data streams that materialize from exploding number of cheap sensors installed "everywhere", on the one hand, and ability to capture and study events with systematically increasing granularity, on the other. To address the needs for scaling computational and storage infrastructures, concepts like: edge, fog and dew computing emerged.

Novel issues in involved in "pushing computing away from the center" did not replace open questions that existed in the context of grid and cloud computing. Rather, they added new dimensions of complexity and resulted in the need of addressing scalability across more and more complex ecosystems consisting of individual sensors and micro-computers (e.g. Raspberry PI based systems) as well as supercomputers available within the Cloud (e.g. Cray computers facilitated within the MS Azure Cloud).

Addressing research questions that arise in individual "parts" as well as across the ecosystem viewed from a holistic perspective, with scalability as the main focus is the goal of the Workshop on Scalable Computing. In this context, the following topics are of special interest (however, this list is not exhaustive).

TOPICS

- General issues in scalable computing
  - Algorithms and programming models for large-scale applications, simulations and systems
  - Large-scale symbolic, numeric, data-intensive, graph-oriented, distributed computations
  - Fault-tolerant and consensus techniques for large-scale computing
  - Resilient large-scale computing
  - Data models for large-scale applications, simulations and systems
  - Large-scale distributed databases
  - Load-balancing / intelligent resource management in large-scale applications, simulations and systems
  - Performance analysis, evaluation, optimization and prediction
  - Scientific workflow scheduling
  - Data visualization
  - On-demand computing
  - Virtualization supporting computations
  - Volunteer computing
  - Scaling applications from small-scale to exa-scale (and back)
  - Big data real-time computing / analytics
  - Economic, business and ROI models for large-scale applications
- Emerging technologies for scalable computing
  - Cloud / Fog / Dew computing architectures, models, algorithms and applications
  - High performance computing in Cloud / Fog / Dew
  - Green computing in Cloud / Fog / Dew
  - Performance, capacity management and monitoring of Cloud / Fog / Dew configuration
  - Cloud / Fog / Dew application scalability and availability
  - Big Data cloud services
  - Architectures for large-scale computations (GPUs, accelerators, quantum systems, federated systems, etc.)
  - Self* and autonomous computational / storage systems

TECHNICAL SESSION CHAIRS

- **Ganzha, Maria,** Warsaw University of Technology, Poland
- **Gusev, Marjan,** University Sts Cyril and Methodius, Macedonia
- **Paprzycki, Marcin,** Systems Research Institute Polish Academy of Sciences, Poland
- **Petcu, Dana,** West University of Timisoara, Romania
- **Ristov, Sashko,** University of Innsbruck, Austria

PROGRAM COMMITTEE

- **Barbosa, Jorge,** University of Porto, Portugal
- **Camacho, David,** Universidad Autonoma de Madrid, Spain
- **Carretero, Jesus,** Universidad Autonoma de Madrid, Spain
- **D'Ambra, Pasqua,** IAC-CNR, Italy
- **Durillo, Juan,** Leibniz Supercomputer of the Bavarian Academy of Sciences and Humanities, Germany
- **Garcia Valdez, Mario,** National Technological Institute of Mexico, Mexico
- **Gordon, Minor,** Software development consultant, United States
- **Gravvanis, George,** Democritus University of Thrace, Greece
- **Grosu, Daniel,** Wayne State University, United States
- **Holmes, Violeta,** The University of Huddersfield, United Kingdom
- **Kalinov, Alexey,** Cadence Design Systems, Russia

- **Kecskemeti, Gabor,** Liverpool John Moores University, United Kingdom
- **Kitowski, Jacek,** AGH University of Science and Technology, Department of Computer Science, Poland
- **Knepper, Richard,** Indiana University, United States
- **Lang, Tran Van,** Vietnam Academy of Science and Technology, Vietnam
- **Lastovetsky, Alexey,** University College Dublin, Ireland
- **Margaritis, Konstantinos G.,** University of Macedonia, Greece
- **Nosovic, Novica,** Faculty of Electrical Engineering, University of Sarajevo, Bosnia and Herzegovina
- **Pawłowski, Wiesław,** University of Gdańsk, Poland
- **Prodan, Radu,** University of Klagenfurt, Austria
- **Schikuta, Erich,** University of Vienna, Austria
- **Schreiner, Wolfgang,** Johannes Kepler University Linz, Austria
- **Shen, Hong,** University of Adelaide, Australia
- **Telegin, Pavel,** JSCC RAS, Russia
- **Tudruj, Marek,** Inst. of Comp. Science Polish Academy of Sciences/Polish-Japanese Institute of Information Technology, Poland
- **Vazhenin, Alexander,** University of Aizu, Japan
- **Wei, Wei,** School of Computer science and engineering, Xi'an University of Technology, China
- **Wyrzykowski, Roman,** Czestochowa University of Technology, Poland
- **Zavoral, Filip,** Charles University, Czech Republic

DOI: 10.15439/2020F76 ISSN 2300-5963 ACSIS, Vol. 21

How well a multi-model database performs against its single-model variants: Benchmarking OrientDB with Neo4j and MongoDB

Martin Macak[1,2], Matus Stovcik[1,2], Barbora Buhnova[1,2], and Michal Merjavy[2]
[1]*Institute of Computer Science, Masaryk University*
[2]*Faculty of Informatics, Masaryk University*
Brno, Czech Republic
{macak, mstovcik, buhnova, merjavy}@mail.muni.cz

***Abstract*—Digitalization is currently the key factor for progress, with a rising need for storing, collecting, and processing large amounts of data. In this context, NoSQL databases have become a popular storage solution, each specialized on a specific type of data. Next to that, the multi-model approach is designed to combine benefits from different types of databases, supporting several models for data. Despite its versatility, a multi-model database might not always be the best option, due to the risk of worse performance comparing to the single-model variants. It is hence crucial for software engineers to have access to benchmarks comparing the performance of multi-model and single-model variants. Moreover, in the current Big Data era, it is important to have cluster infrastructure considered within the benchmarks.**

In this paper, we aim to examine how the multi-model approach performs compared to its single-model variants. To this end, we compare the OrientDB multi-model database with the Neo4j graph database and the MongoDB document store. We do so in the cluster setup, to enhance state of the art in database benchmarks, which is not yet giving much insight into cluster-operating database performance.

I. Introduction

THE AGE we live in is governed by information. Our society daily produces excessive amounts of raw data. Big Data tools were created as the help to this issue, each being a better fit for a different data set or scenario [1]. Existing surveys describe these tools either generally or are focused on the visualization, processing, or storage options. Focusing on the Big Data storage tools, it is difficult to navigate between them and choose the most effective tool for the given problem [2]. Therefore, for an efficient solution, it is crucial to make the right choice of storage technology, reflecting data variety and other characteristics [3].

Organizations are thus left to either compromise the functionality and use one (i.e., single-model) database strategy or combine more single-model solutions. Implementing and maintaining more solutions means more load on the database engineers and often higher costs [4]. Multi-model databases were proposed as an answer to the need for new, more general, data storing solution [5], offering a possibility to work with one database with multiple types of data.

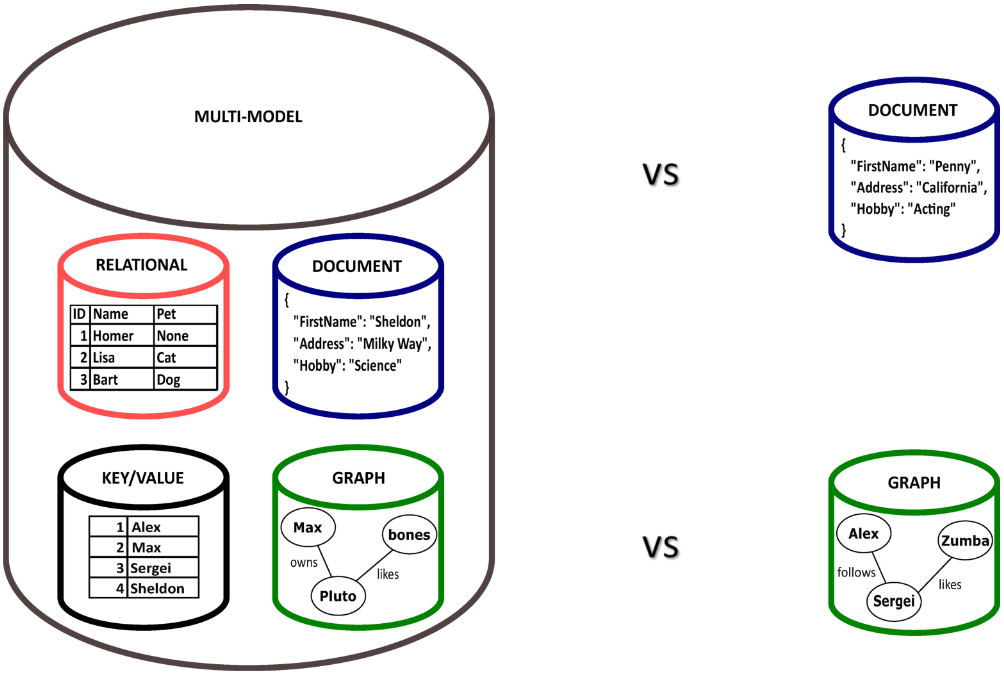
The ability to effectively store and process data is further emphasized in the context of Big Data, which calls for operating the database solutions in a cluster setup. Even though there are papers comparing databases in a cluster-based scenario [6], the area of comparing multi-model databases with their single-model variants using cluster setup is so far unexplored.

In this paper, we aim to address this gap and contribute to state of the art in the decision about storage technology, specifically choosing among multi-model database and its single-model variants when there is a need to operate them in a cluster setup. In the context of Big Data, NoSQL databases, and specifically the document stores and graph databases, belong among the most popular storage technologies. In our research, we have, therefore, decided to compare these two storage strategies with the multi-model variant (as can be seen in Figure 1). Specifically, we compare the OrientDB multi-model database with the Neo4j graph database and the MongoDB document store. We chose OrientDB, as it is currently one of the most popular and advanced multi-model database [7], [8], whereas MongoDB and Neo4j are suitable representatives of document [9] and graph [10] databases. As for the comparison metric, we use the execution time of queries as it is a standard metric for comparison also in other (non-cluster) benchmarks [11], [12], [13].

To make the benchmarks relevant to the possible real-world Big Data scenarios, we work with a big data set that asks for a storage solution in a cluster of computers. Hence we import this data to distributed versions of the databases, i.e., OrientDB, MongoDB, and Neo4j, exploring how the queries behave in a cluster setup.

The structure of the paper is as follows. Following the related work overview in Section II, we explain our choice of the compared database technologies in Section III. Section IV presents the used cluster configuration, chosen datasets, and designed queries for our experiments. Then, Section V contains the results of the experiments. We provide a summary of these results, including key observations, in Section VI. In Section VII, the threats to validity are discussed, and Section VIII contains the conclusion of this paper.

IEEE Catalog Number: CFP2085N-ART ©2020, PTI



ID	Name	Pet
1	Homer	None
2	Lisa	Cat
3	Bart	Dog

```
{
  "FirstName": "Sheldon",
  "Address": "Milky Way",
  "Hobby": "Science"
}
```

1	Alex
2	Max
3	Sergei
4	Sheldon

```
{
  "FirstName": "Penny",
  "Address": "California",
  "Hobby": "Acting"
}
```

Fig. 1: Compared data structures

II. Related work

Relevant related work considering a multi-model database can be identified in two directions, based on the metric of the comparison. The first direction is based on the efficiency of compared databases (Section II-A), i.e., the comparison of their performance. The second direction is based on the effectiveness of compared databases (Section II-B), i.e., the comparison of their characteristics.

A. Comparisons based on performance

There are many comparisons of multi-model databases with different representatives of its single-model variants. The significant number of these comparisons used OrientDB and compared it with Neo4j and MongoDB [12], [14]. However, none of them used a cluster setup in their comparison.

The work of Ataky et al. [11] presented a paper about performance testing of OrientDB and Neo4j. They focused on the difference between query performance on the index and non-indexed data. They conclude that Neo4j is more suitable for non-indexed data, while MongoDB performed better on data with indexes.

Messaoudi et al. [12] also used Neo4j, MongoDB, and OrientDB in their comparisons. Their research was focused on biomedical data. Their benchmarks covered different depth levels but also some CRUD operations. They found out that OrientDB better handles deeper levels of traversal. A similar study by Messaoudi et al. [14] uses, as in the previous case, Neo4j, MongoDB, and OrientDB. They evaluate databases' ability to manage proteomics data. They observe that MongoDB has better performance for importing protein information in both large and small datasets, but not always in the case when protein fields had a great number of fields. They also found instances in which Neo4j outperformed OrientDB.

Jayathilake et al. [13] have enlarged the scope of focus considering database tools by Cassandra and MemBase. The primary interest was on handling a highly heterogeneous tree. They did a ranking of database tools. Among those tools, MongoDB, Neo4j, and OrientDB stood their place within a higher ranking.

The work performed by Oliveira and del Val Cura [15] focused on benchmarking NoSQL multi-model databases with a polyglot persistence approach. They designed benchmarks with three databases OrientDB, ArrangoDB, and the combination of Neo4j and MongoDB. In contrary to our work, they did not test a multi-model to single-model variant but to a combination of two databases, document and graph database. These tests showed that the combination of Neo4j and MongoDB in some cases performed as good as OrientDB, depending on the size of the dataset, and that ArrangoDB has better performance on more document-based queries with smaller depth levels.

B. Comparisons based on effectiveness

Several studies are extensively comparing the characteristics, features, and benefits of the multi-model databases against their single-model variants.

Fernandes and Bernardino [16] provided characteristics of NoSQL databases, mainly comparing graph database and multi-model database with graph aspect. They discussed and explained the features and benefits of representation of graph data in different databases, including AllegroGraph, ArangoDB, InfiniteGraph, Neo4j, and OrientDB. The recommendation was offered that Neo4j and ArrangoDB are the correct way of representing a graph database in case of a single-model database.

In the paper by Bathla et al. [17], the authors categorized different databases. Among them, we can find Neo4j, MongoDB, and OrientDB. They drew guidelines for choosing the right database tools for users.

The survey by Mazumdar et al. [18] provided a better understanding of choosing the right database solution based non-functional requirements.

III. Compared databases

For our benchmark, we have chosen one representative of a multi-model database, OrientDB, and two representatives of single-model variants: the Neo4j graph database and the MongoDB document store. When selecting the representatives, we considered the popularity of databases based on the DB-Engines website[1]. This ranking of popularity is based on various factors, like the frequency of Google search, relevance in social networks, and a number of job offers. We were looking only for non-commercial open-source databases with the ability to run in a cluster setup.

A. MongoDB

MongoDB[2] version 4.0.12 was chosen as a representative of document store databases. It is classified as a NoSQL database. Data are managed in structure-free storage, with the capability of every collection to be different. Therefore, it stores data in a more flexible way than SQL databases. Its flexibility is one of the reasons for being chosen in our work. MongoDB works with a JSON-like document schema. MongoDB uses collections instead of tables, and it also implements references for faster querying. In Listing 1 is an example query that returns all females in a collection of people.

```
db.people.find( { sex: { "female" } } )
```

Listing 1: MongoDB query example

B. Neo4j

Neo4j[3] version 3.5.12 is an open-source native graph database that is also ACID compliant, highly available, and scalable. Neo4j is one of the leading software solutions in graph databases with active support and development. Its storage was explicitly designed for the management and storage of graphs [19]. Neo4j appears to be the right choice as a representative of graph databases; it has performed well in many comparisons [20], [21], [22]. Neo4j stores data in graphs format using nodes and relationships. Relationships are used for connecting nodes and traversing through data, which is less costly than using a SQL-like approach, i.e., joins. Nodes and relationships are labeled by name, have properties, and are grouped to sets. We can improve the performance of graph traversals by dividing parts of the graphs and using indexes. Data in Neo4j can be accessed through two different query languages.

Neo4j implements cluster setup via core and read-only nodes. This setup does not use the traditional concept of master and slave hierarchy; a leader is voted every period to maintain freshness and availability. Only the leader has the ability to use write operations; followers are used only for read operations. In Listing 2 is a query that returns a number of friends of Jennifer using Neo4j.

```
match (a:Person {name: 'Jennifer'})
      -[:Friend]->
      (b:Person)
return count(b) as count;
```

Listing 2: Neo4j query example

C. OrientDB

OrientDB[4] version 3.0.23 is a multi-model open source NoSQL database management system that supports document, graph, key-value, and object data model. It was released in 2010, is implemented in Java, and is being developed by OrientDB Ltd. It supports distributed architecture with replication and is transactional.

OrientDB uses Paginated Local Storage for storing data [23]. It is disk-based storage that uses a page model to work with data and consists of several components that use disk data trough disk cache. Paginated local storage is a two-level disk cache that works together with a write-ahead log. Files are split into pages, and this allows operations to be atomic at a page level. Two-level disk cache allows OrientDB to cache often accessed pages, separate pages that are not accessed frequently, minimize the amount of disk head seeks during data writes. It also enables the mitigation of pauses that are needed to write data to the disk by flushing all changed or newly added pages to the disk in a background thread. Disk cache consists of two parts read cache and write cache. Read cache is based on the 2Q cache algorithm and write cache is based on WOW cache algorithm. One of the possibilities or manipulating database data is using Java, SQL with extension for graphs and Gremlin. OrientDB supports schema-less, schema-full, or schema-mixed data. OrientDB uses the Hazelcast Open Source project for automatic discovery of nodes, storing cluster configuration, and synchronization between nodes. Distributed architecture can be used in different ways to achieve better performance, scalability, and robustness. OrientDB also provides a web interface that can be used for viewing graphs and data manipulation [16].

OrientDB can use the SQL-like approach for querying. The following query in Listing 3 returns all people with sex equal to 'Female'.

```
SELECT FROM People WHERE sex LIKE 'Female'
```

Listing 3: OrientDB SQL-like query example

[1] https://db-engines.com/en/ranking

[2] https://www.mongodb.com/

[3] https://neo4j.com/

[4] https://www.orientdb.org/

The example of a query that uses an approach of a graph database and returns a number of friends of Jennifer is in Listing 4.

```
SELECT
 both('HasFriend').size() AS FriendsNumber
FROM `Person`
WHERE Name='Jennifer'
```

Listing 4: OrientDB graph query example

IV. Design of experiments

This section presents the used cluster configuration and datasets that were used for testing, together with all queries that were designed. Each query contains a brief description of the results it produces.

Our primary interest was to mirror real-world use cases. Therefore, we designed our queries accordingly in the respective data models. Graph databases stand out with abilities to connect data with relationships and to query data using relationship traversal. Therefore, we wanted to underline these characteristics using queries with a various depth of relationship traversing. Document databases store their data in a structure that offers a suitable environment for filter querying with single or multiple conditions. Hence, it is a natural decision to mirror this into out benchmark queries.

For graph queries, we mostly focused on traversal between nodes because we expect relationships to be more relevant than information stored in individual nodes. In document queries, we concentrated mainly on filtering the data with and without indexes.

A. Setup

Each database is configured in a cluster of three nodes. Nodes are located on the OpenStack cloud platform. All three nodes are running on Ubuntu 18.04.2 LTS. One node runs on dual-core CPU with 2GHz for each core and uses 4GB of RAM. Remaining two nodes run on quad-core CPU witch 2GHz each and use 8GB of ram.

1) Graph dataset: To compare the graph database, we use the 22.3 GB dataset of Twitter followers [24]. Files in this dataset were processed into a format suitable for importing it into the database. We are using CSV format for import as both Neo4j and OrientDB support it. In Neo4j, we use the Neo4j-admin import tool with a file that contains names of all files about to be imported and the name of the database.

For loading data in OrientDB, we utilized the ETL tool that requires a JSON file to define Extractor, Transformer, and Loader. An extractor is responsible for extracting data from a source file and defining other options for extraction, such as separator, columns, and date format. The transformer defines to which class it imports the data and edges that are related to this class. The loader contains the name of the database, type of the database, indexes for classes.

2) Document dataset: As dataset for a document database, we use records of taxi rides in New York City from 2013 [5]. This 18.3 GB dataset is already in the CSV format; hence, it does not need any further alteration. We use this dataset for both MongoDB and OrientDB. To MongoDB, it is imported by the mongoimport tool, which required a path to file, which we want to import and the database name. In OrientDB, we use the same ETL tool as in the case with the graph database.

B. Graph queries

In these queries, we are using relationships for traversal between nodes. Queries are labeled from GQ1 to GQ8. We use a setup described in IV-A, with the exception of queries GQ2 and GQ7, where we used 4GB of RAM on all nodes. The description of the queries is as follows:

- GQ1 counts connected nodes that have less than 1000 followers until depth two,
- GQ2 identical to GQ1 (4GB of RAM on all nodes)
- GQ3 identical to GQ1 depth three,
- GQ4 identical to GQ1 depth four,
- GQ5 identical to GQ1 depth five,
- GQ6 finds the shortest path between two nodes where the desired path is three edges long,
- GQ7 identical to GQ6 (4GB of RAM on all nodes),
- GQ8 finds the shortest path where the path between nodes does not exist.

C. Document queries

In a document database, we focus on queries that deal with filtering data for results that satisfy their requirements, grouping data by some common denominators. Queries are labeled from DQ1 to DQ10. We use a setup described in IV-A, with the exception of queries DQ2 and DQ8, where we used 4GB of RAM on all nodes. The description of the queries is:

- DQ1 counts how many documents fulfill one condition[6],
- DQ2 identical to DQ1 (4GB of RAM on all nodes),
- DQ3 counts how many documents fulfill two conditions,
- DQ4 counts how many documents fulfill three conditions,
- DQ5 counts how many documents fulfill four conditions,
- DQ6 sum of total tip amount on different types of payments,
- DQ7 counts how many documents fulfill one condition on an indexed property where a number of these documents is more than ten million,
- DQ8 counts how many documents fulfill one condition on an indexed property where a number of these documents is more than ten million (4GB of RAM on all nodes),
- DQ9 counts how many documents fulfill one condition on an indexed property where a number of these documents is less than one hundred thousand,
- DQ10 combined index for two properties.

[5] https://chriswhong.com/open-data/foil_nyc_taxi/

[6] conditions are understood as WHERE statements

V. Results of the experiments

This section presents all measurements of queries and interpretation of the results for each comparison between OrientDB and its single-model variant, namely Neo4j and MongoDB. We provide further discussion and interpretation of the results.

The comparison between Neo4j and OrientDB is using a built-in timer within. For comparing MongoDB and OrientDB, we are using Unix command time. Each query was executed five times. The final time is calculated as an average from all runs of the query.

A. Graph queries measurements

TABLE I: Graphs queries

Query	Neo4j	OrientDB
GQ1	**24.586s**	1m16s
GQ2	**30.638s**	4m41s
GQ3	**3m43s**	10m52s
GQ4	30m38s	**15m37s**
GQ5	251m19s	**28m41s**
GQ6	2m7s	**20.436s**
GQ7	3m32s	**1m22s**
GQ8	1m14s	**8.247s**

Neo4j outperformed OrientDB in the first three queries, as shown in Table I. However, in queries GQ4 and GQ5 where the depth was four and five, respectively, OrientDB came out on top with a significant lead in both cases. In our last three queries where the objective was to find the shortest path between selected nodes, OrientDB was also significantly faster than Neo4j.

In queries GQ2 and GQ7, one can see that lowering the size or RAM negatively affects both OrientDB and Neo4j.

B. Document queries measurements

TABLE II: Document queries

Query	MongoDB	OrientDB
DQ1	**9m27s**	38m2s
DQ2	**17m6s**	51m32s
DQ3	**10m15s**	32m13s
DQ4	**12m47s**	32m3s
DQ5	**11m4s**	34m42s
DQ6	**18m42s**	42m57s
DQ7	**2.571s**	50.324s
DQ8	**2.652s**	77.893s
DQ9	**0.673s**	1.078s
DQ10	**0.562s**	1.459s

Queries we tested are focused on filtering a different number of properties and grouping data. In MongoDB filtering, a different number of properties did add some additional time to query executions. As we can see from Table II, in queries DQ2 and DQ3, where we were filtering two and three properties respectively, OrientDB performed better than in DQ1. We suspect that lower times in query DQ3 and DQ4 was due to the lower number of results. In DQ5 execution time for OrientDB increased, we assume it is because the last added property was a string.

In the last four queries, there were indexes created for properties that we used to filter the data. In OrientDB, it was SB-Tree Index; for MongoDB, it was Single Field. Both indexes are based on the B-tree algorithm. Both indexes were chosen based on recommendations found in respective documentations. In query DQ7, we can see that MongoDB outperformed OrientDB, but in the query DQ9, that uses the same index where the number of results was significantly smaller. The difference in these queries in real-time was negligible. In the last query, we used a combined index that uses more than one property for the index, and OrientDB performed a little bit better than MongoDB in this case. Since not in all cases, it is beneficial to create an index on all the data because it requires additional disk space and in most cases, slows inserts it is essential to know how we want to use a database.

In the case of non-indexed data, MongoDB outperforms OrientDB, and in the case of indexed data, OrientDB does not perform very differently to its variant. In query DQ7, where we used only 4GB of RAM instead of 8GB, we also found that having more RAM improved the performance of OrientDB on indexed data significantly, and for MongoDB, it did not have a significant impact in this case. The amount of RAM on non-indexed data improved performance in both cases, as shown in DQ1 and DQ2. This improvement was very similar in all queries for non-indexed data.

VI. Summary of results

This section provides a summary of our results. We provide evaluation and interpretation of the results as well as an explanation for our outcomes. We also include a list of key observations that emerged from our experiments.

A. Results of comparison with a graph database

For graph queries, the objective was to find the shortest path between nodes with varying depths as we wanted to emphasize the power of traversals. Our results in Figure 2 show that it is advisable to use Neo4j up to a depth of three. With higher levels of depths, OrientDB outperforms Neo4j; the difference is most significant in the case of GQ5. However, when the objective in graph data is to traverse different nodes up to the depth of three, Neo4j appears to be a more suitable choice.

B. Results of comparison with a document database

Moving to document queries, we wanted our queries to emphasize the difference between querying upon an indexed and non-indexed field.

Figure 3 shows that the execution times of OrientDB were many times larger than those of MongoDB. Our benchmarks display how good these databases handle non-indexed data, which means that MongoDB provides better management of document data than OrientDB.

Fig. 2: Average times of GQ queries (in seconds)

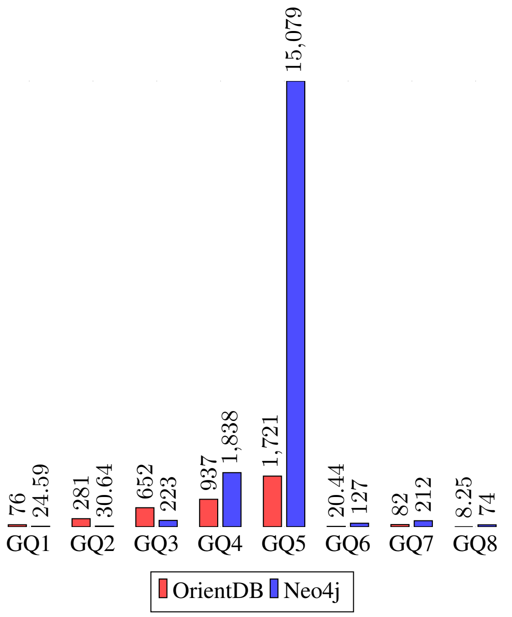
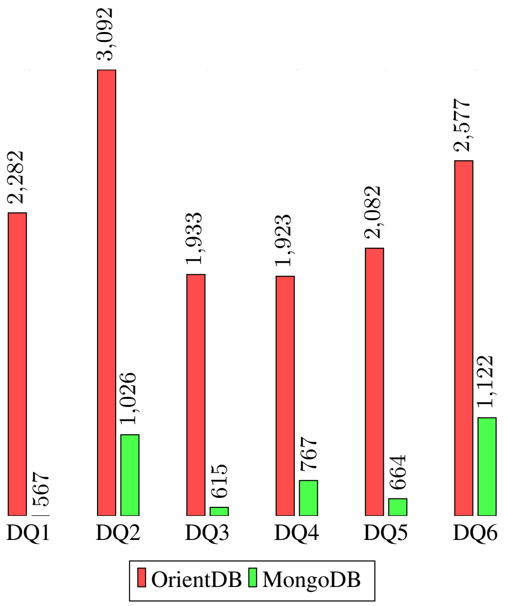
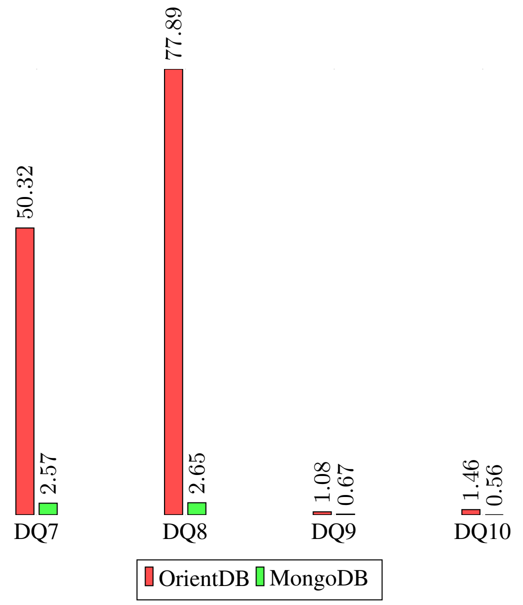


Fig. 3: Average times of DQ queries (in seconds)



Indexing fields resulted in a significant improvement of both MongoDB and OrientDB performances, as shown in Figure 4. In our test case, an increasing amount of RAM had a minor effect on MongoDB, but rather a significant impact on query latencies of OrientDB. For OrientDB, the number of results that our query produces is also very important, as we can see in queries DQ7 and DQ9. In both queries, we used the same data, but the difference between execution times of MongoDB and OrientDB was remarkably smaller. In the case of DQ9 and DQ10 queries, indexing data made the performance of OrientDB comparable to the MongoDB.

Fig. 4: Average times of DQ queries (in seconds)



C. Key observations

In this section, we summarize a list of key observations from our experiments. They also serve as recommendations for future work, which could explore these observations further.

- **KO1**: OrientDB is beneficial when the practitioner is unsure whether more data models will be needed in the future.
- **KO2**: In the case of using only document data, MongoDB is a more suitable choice than OrientDB.
- **KO3**: In the case of queries containing a significant level of depth, OrientDB is a better choice than Neo4j. On the other hand, Neo4j performed better for queries with a smaller level of depth.

From our experiments, it was visible that OrientDB is comparable to (or even better than) Neo4j for graph data and MongoDB for document data. We have to take into consideration that using two different database management systems would have an impact on overhead time. Therefore we assume OrientDB is more beneficial to use when more than one data model is needed. Furthermore, as stated in **KO1**, it is also beneficial to consider OrientDB when the practitioner is unsure whether the support of more data models will be needed in the future.

On the contrary, if the practitioner is aware that only the document model will be used, as mentioned in **KO2**, MongoDB is a more suitable choice because its performance was significantly better in both indexed and non-indexed data.

Moreover, if the practitioner knows that only graph data will be used, we can recommend the proper database based on the level of depth of queries, as stated in **KO3**. When the objective is to traverse different nodes up to the depth of three, Neo4j performed better. However, if the queries use a higher level of depth, OrientDB is a more suitable choice.

VII. Threats to validity

In this study, we needed to narrow our scope to keep focus. This section discusses the limitations we opted for, together with our reasoning behind it.

A. Construct validity threats

We are aware that the fact that in our comparisons, we used only one metric, the query response time, which might be limiting in the decision guidance. Measuring other metrics, like throughput, memory usage, or processor usage, shall be considered in future work to provide more detailed results, although it is out of the scope of this paper.

Another construct validity is the fact that the queries we used in our tests are not complete. There might be many more queries that could be run in a cluster setup to determine the suitability of multi-model database OrientDB over Neo4j and MongoDB. In the future, we would suggest adding queries with complicated aggregate functions. Using queries with even greater depth could result in the possibility of showing another threshold where Neo4j may outperform OrientDB. However, we believe that our tests sufficiently contribute to state of the art.

B. Internal validity threats

We are aware that the configuration of each benchmarked database can affect the results of experiments. We have tried several configurations of each one, and we believe that the chosen configurations are designed for the best efficiency. An exhaustive search of all configurations would be infeasible.

However, it is worth considering that experimenting with multiple configurations of the nodes in a cluster may provide different results. Also, the results may vary when the database cluster contains a different number of nodes.

C. External validity threats

In order to get generalized results, there is a need to perform more tests on different datasets. We have chosen a sufficiently large graph and document dataset. However, this selection might have an impact on the results. Despite this, we believe that our work provides a step towards this goal.

D. Conclusion validity threats

Datasets we used are available online. Therefore they can be downloaded for further investigation and replication. We provided the configuration of our cluster so the tests might be performed again on the same configuration. Using the same approach of testing, we should obtain the same results. However, as the development of chosen databases is fast, it does not make sense to perform these tests on the older versions.

VIII. Conclusion

In this paper, we compared the OrientDB multi-model database with the Neo4j graph database and MongoDB document database. We describe these databases, together with a brief description of used datasets. We compared the performance of these databases on several different queries that were focused on various properties. Our work was aimed at the cluster setup, precisely three nodes.

These queries are split into two different categories. The first category is focused on queries related to graph data, and the second is focused on document data. In each group, we try to aim at queries that are possible for real-world scenarios. For graph data, we focus on traversal between nodes. On the other hand, for document data, we focus on filtering.

Based on the experiments, we provide a set of key observations. We believe that they are proper candidates for future examination in this area.

Acknowledgment

The work was supported by the European Regional Development Fund Project *CERIT Scientific Cloud* (No. CZ.02.1.01/0.0/0.0/16_013/0001802). Access to the CERIT-SC computing and storage facilities provided by the CERIT-SC Center, under the "Projects of Large Research, Development, and Innovations Infrastructures" programme (CERIT Scientific Cloud LM2015085), is greatly appreciated.

References

[1] M. Macak, H. Bangui, B. Buhnova, A. J. Molnár, and C. I. Sidló, “Big data processing tools navigation diagram.” in *IoTBDS*, 2020, pp. 304–312.

[2] F. Gessert, W. Wingerath, S. Friedrich, and N. Ritter, “Nosql database systems: a survey and decision guidance,” *Computer Science-Research and Development*, vol. 32, no. 3-4, pp. 353–365, 2017.

[3] S. Kaisler, F. Armour, J. Espinosa, and W. Money, “Big data: Issues and challenges moving forward,” 01 2013. doi: 10.1109/HICSS.2013.645. ISBN 978-1-4673-5933-7 pp. 995–1004.

[4] P. J. Sadalage and M. Fowler, *NoSQL distilled: a brief guide to the emerging world of polyglot persistence*. Pearson Education, 2013.

[5] E. Raguseo, “Big data technologies: An empirical investigation on their adoption, benefits and risks for companies,” *International Journal of Information Management*, vol. 38, no. 1, pp. 187 – 195, 2018. doi: https://doi.org/10.1016/j.ijinfomgt.2017.07.008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0268401217300063

[6] M. Macak, M. Stovcik, and B. Buhnova, “The suitability of graph databases for big data analysis: A benchmark.” in *IoTBDS*, 2020, pp. 213–220.

[7] A. Messina, P. Storniolo, and A. Urso, “Keep it simple, fast and scalable: A multi-model nosql dbms as an (eb) xml-over-soap service,” in *2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, 2016, pp. 220–225.

[8] T. P. Hong and P. Do, “Combining apache spark orientdb to find the influence of a scientific paper in a citation network,” in *2018 10th International Conference on Knowledge and Systems Engineering (KSE)*, 2018, pp. 113–117.

[9] W. Schultz, T. Avitabile, and A. Cabral, “Tunable consistency in mongodb,” *Proc. VLDB Endow.*, vol. 12, no. 12, p. 2071–2081, Aug. 2019. doi: 10.14778/3352063.3352125. [Online]. Available: https://doi.org/10.14778/3352063.3352125

[10] T. T. Aung and T. T. S. Nyunt, “Community detection in scientific co-authorship networks using neo4j,” in *2020 IEEE Conference on Computer Applications(ICCA)*, 2020, pp. 1–6.

[11] S. Ataky T. M, L. Ferreira, M. Ribeiro, and M. Prado Santos, “Evaluation of graph databases performance through indexing techniques,” *International Journal of Artificial Intelligence & Applications (IJAIA)*, vol. 06, pp. 87–98, 09 2015. doi: 10.5121/ijaia.2015.6506

[12] C. Messaoudi, M. Amrou, R. Fissoune, and B. Hassan, “A performance study of nosql stores for biomedical data,” 11 2017.

[13] D. Jayathilake, C. Sooriaarachchi, T. Gunawardena, B. Kulasuriya, and T. Dayaratne, “A study into the capabilities of nosql databases in handling a highly heterogeneous tree,” in *2012 IEEE 6th International Conference on Information and Automation for Sustainability*, 2012, pp. 106–111.

[14] C. Messaoudi, R. Fissoune, and B. Hassan, “A performance evaluation of nosql databases to manage proteomics data,” *International Journal of Data Mining and Bioinformatics*, vol. 21, pp. 70–89, 09 2018. doi: 10.1504/IJDMB.2018.10016724

[15] F. R. Oliveira and L. del Val Cura, “Performance evaluation of nosql multi-model data stores in polyglot persistence applications,” in *Proceedings of the 20th International Database Engineering & Applications Symposium*, ser. IDEAS ’16. New York, NY, USA: Association for Computing Machinery, 2016. doi: 10.1145/2938503.2938518. ISBN 9781450341189 p. 230–235. [Online]. Available: https://doi.org/10.1145/2938503.2938518

[16] D. Fernandes and J. Bernardino, “Graph databases comparison: Allegrograph, arangodb, infinitegraph, neo4j, and orientdb,” in *Proceedings of the 7th International Conference on Data Science, Technology and Applications - Volume 1: DATA,*, INSTICC. SciTePress, 2018. doi: 10.5220/0006910203730380. ISBN 978-989-758-318-6 pp. 373–380.

[17] G. Bathla, R. Rani, and H. Aggarwal, “Comparative study of nosql databases for big data storage,” *International Journal of Engineering & Technology*, vol. 7, no. 2.6, pp. 83–87, 2018. doi: 10.14419/ijet.v7i2.6.10072. [Online]. Available: https://www.sciencepubco.com/index.php/ijet/article/view/10072

[18] S. Mazumdar, D. Seybold, K. Kritikos, and Y. Verginadis, “A survey on data storage and placement methodologies for cloud-big data ecosystem,” *Journal of Big Data*, vol. 6, no. 1, p. 15, Feb 2019. doi: 10.1186/s40537-019-0178-3. [Online]. Available: https://doi.org/10.1186/s40537-019-0178-3

[19] F. Holzschuher and R. Peinl, “Performance of graph query languages: comparison of cypher, gremlin and native access in neo4j,” in *Proceedings of the Joint EDBT/ICDT 2013 Workshops*. ACM, 2013, pp. 195–204.

[20] D. Dominguez-Sal, P. Urbón-Bayes, A. Giménez-Vañó, S. Gómez-Villamor, N. Martínez-Bazán, and J. L. Larriba-Pey, “Survey of graph database performance on the hpc scalable graph analysis benchmark,” in *Web-Age Information Management*, H. T. Shen, J. Pei, M. T. Özsu, L. Zou, J. Lu, T.-W. Ling, G. Yu, Y. Zhuang, and J. Shao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. ISBN 978-3-642-16720-1 pp. 37–48.

[21] S. Jouili and V. Vansteenberghe, “An empirical comparison of graph databases,” in *2013 International Conference on Social Computing*, Sep. 2013. doi: 10.1109/SocialCom.2013.106 pp. 708–715.

[22] M. Ciglan, A. Averbuch, and L. Hluchy, “Benchmarking traversal operations over graph databases,” in *2012 IEEE 28th International Conference on Data Engineering Workshops*, April 2012. doi: 10.1109/ICDEW.2012.47 pp. 186–189.

[23] A. S. Mondal, M. Sanyal, S. Chattopadhyay, and K. C. Mondal, “Comparative analysis of structured and un-structured databases,” in *Computational Intelligence, Communications, and Business Analytics*, J. K. Mandal, P. Dutta, and S. Mukhopadhyay, Eds. Singapore: Springer Singapore, 2017. ISBN 978-981-10-6430-2 pp. 226–241.

[24] R. A. Rossi and N. K. Ahmed, “The network data repository with interactive graph analytics and visualization,” in *AAAI*, 2015. [Online]. Available: http://networkrepository.com

Advances in Network Systems and Applications

THE rapid development of computer networks including wired and wireless networks observed today is very evolving, dynamic, and multidimensional. On the one hand, network technologies are used in virtually several areas that make human life easier and more comfortable. On the other hand, the rapid need for network deployment brings new challenges in network management and network design, which are reflected in hardware, software, services, and security-related problems. Every day, a new solution in the field of technology and applications of computer networks is released. The ANSA technical session is devoted to emphasizing up-to-date topics in networking systems and technologies by covering problems and challenges related to the intensive multidimensional network developments. This session covers not only the technological side but also the societal and social impacts of network developments. The session is inclusive and spans a wide spectrum of networking-related topics.

The ANSA technical session is a great place to exchange ideas, conduct discussions, introduce new ideas and integrate scientists, practitioners, and scientific communities working in networking research themes.

TOPICS

- Networks architecture
- Networks management
- Quality-of-Service enhancement
- Performance modeling and analysis
- Fault-tolerant challenges and solutions
- 5G developments and applications
- Traffic identification and classification
- Switching and routing technologies
- Protocols design and implementation
- Wireless sensor networks
- Future Internet architectures
- Networked operating systems
- Industrial networks deployment
- Software-defined networks
- Self-organizing and self-healing networks
- Mulimedia in Computer Networks
- Communication quality and reliability
- Emerging aspects of networking systems

TECHNICAL SESSION CHAIRS

- **Awad, Ali Ismail,** Lule&aring; University of Technology, Sweden
- **Essai, Mohamed Hassan,** AL-Azhar University, Egypt
- **Furtak, Janusz,** Military University of Technology
- **Hodoň, Michal,** University of Žilina, Slovakia

PROGRAM COMMITTEE

- **Ajlouni, Naim,** Istanbul Aydin University, Turkey
- **Brzoza-Woch, Robert,** AGH University of Science and Technology, Poland
- **Chumachenko, Igor,** Kharkiv National University of Municipal Economy named after Beketov, Ukraine
- **Davidsson, Paul,** Malmö University, Sweden
- **Dotsenko, Sergii,** Ukrainian State University of Railway Transport, Ukraine
- **Długosz, Rafał,** UTP University of Science and Technology, Bydgoszcz, Poland, Poland
- **Elmougy, Samir,** Mansoura University, Egypt
- **Faria, Lincoln,** Department of Computer Science, Fluminense Federal University, Brazil
- **Khairova, Nina,** National Technical University Kharkiv Polytechnic Institute, Ukraine
- **Kochláň, Michal,** University of Žilina, Slovakia
- **Lavrov, Eugeniy,** Sumy State University, Ukraine
- **Salem, Abdel-Badeeh M.,** Ain Shams University, Egypt
- **Smolarz, Andrzej,** Lublin University of Technology, Poland
- **Stamatescu, Grigore,** University "Politehnica" of Bucharest, Romania
- **Tymchuk, Sergiy,** Kharkiv National Technical University of Agriculture. Petro Vasilenko, Ukraine
- **Zieliński, Zbigniew,** Military University of Technology

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 473–477 DOI: 10.15439/2020F88 ISSN 2300-5963 ACSIS, Vol. 21

On the Community Discovery Methods for Complex Networks: A Case Study

Kirubel W. Afrassa, Genco Cosgun
Computer Engineering Dept.
Yildiz Technical University
Istanbul, Turkey
{kirubel.afrassa, genco.cosgun}@std.yildiz.edu.tr

Ulku F. Gursoy, Enes M. Yildiz
R&D Center
Intellica Business Intelligence Consultancy
Istanbul, Turkey
{ulku.gursoy, enes.yildiz}@intellica.net

Mehmet S. Aktas
Computer Engineering Dept.
Yildiz Technical University
Istanbul, Turkey
aktas@yildiz.edu.tr

***Abstract*—The inherent knowledge discovery problem regarding networks that represent complex real world phenomenon is a popular research topic. Specifically, in social network analysis (SNA), several community discovery techniques with various approaches have been put forward to distinguish closely related entities. Identifying the relevant techniques to utilize based on the context of the application is a key difficulty researchers face. In this study we propose a methodology for classifying these techniques, visualize a prototype, and analyze the performance and quality of selected approaches over a real world call detail record (CDR) data set.**

***Index Terms*—community discovery, community detection algorithms, visualization, CDR**

I. Introduction

ONE of the many applications of networks is community discovery. Intuitively, community ensues entities that are closer to each other within any arbitrary group, than outside it. Closeness maybe defined by common properties, similar roles or various measurements made on entity interaction. In a network, entities can be characterized by nodes and interactions among them can be embodied using edges. Despite the huge literature available on communities represented in a network, scholars do not have an agreement on what a network with communities corresponds to. However, the widely accepted definition is the *planted l-partition model* [1]. In this model p_{in} and p_{out} signify probability of each node being connected to nodes in its group and different groups respectively. If $p_{in} > p_{out}$ the network has communities present otherwise, the graph is random.

Community discovery within a network of such description can be viewed as maximizing the number of edges between any k groups within that community and minimizing the number of edges outside each of those groups. In terms of nodes, it can also be expressed as a generalization of a data mining problem that is analogous to unsupervised node clustering. But this definition doesn't account for nodes relational behavior. Different community discovery applications and algorithms use diverse and specialized interpretations for community detection. Consequently, there are many types of community discovery algorithms. They mainly constitute varying definitions based on node relationship and the definition of a community. Therefore, we propose a methodology for classifying different community discovery techniques in social network analysis (SNA) in order to narrow down the multitude of available methods. After that, we present a case study on a large scale call detail record (CDR) data set using the selected approaches. The selected approaches are implemented and evaluated based on a strategic ground truth definition for unlabeled data sets. Moreover, the performance and scalability of the selected algorithms are tested on both CDR and YouTube data sets. As part of the experimental study we also develop a visualization software that illustrates networks and discovered communities.

The rest of this paper is organized as follows. Section II reviews literature, Section III introduces the methodology used in choosing community discovery methods, on Section IV a prototype of discovered communities visualizer is presented, Section V defines different metrics in order to evaluate the performance of community discovery algorithms over larger scale social network and CDR data sets and Section VI summarizes the results obtained and possible future works.

II. Literature Review

Community detection techniques have been widely used for variety of purposes. Among these SNA is a common application. Social-based metaheuristic optimization algorithms have been used in order to identify overlapping communities [2] [3]. Moreover, recognized communities can pertain to improve performances of other operations. For instance, community discovery is used to boost low accuracy ranking algorithms in identifying top information spreaders [4].

In SNA and other applications, visualization tools have been made with their own individual variations. While hierarchical community structure and fence-sitting nodes visualization is performed on [5], there are also tools that use some nuance of Newman's modularity optimization algorithm for clustering, prior to visualization [6], [7]. CDR data has been impactful in the analysis of varying models and its application is growing exceedingly. Among these usages, urban sensing and planning [8], [9], traffic engineering [10], [11], predicting energy consumption [12], improved churn prediction using both CDR data and community detection [13] can be cited. Graph data

This study is funded by Scientific and Technological Research Council of Turkey (TÜBİTAK) project no. 3181048

analysis have been studied in different research fields such as provenance field [14]–[22]. Provenance graph is mainly used to understand the data lineage. Different from previous work in provenance research filed, in this study, we focus on analysing graph data to identify the sub networks.

III. Methodology

The variety of community discovery algorithms are more diversified by their abilities to support different types of networks. The main categories to consider would be their capability to support data sets that are directed or weighted and whether or not communities overlap.

Overlapping community discovery refers to a node's ability to be a member of multiple communities at the same time.

On the other hand, in non-overlapping community detection, no two or more communities share a common node. The computational complexity of non-overlapping community discovery algorithms is generally good because of the deducted operation of identifying nodes that are a member of multiple communities.

In real world SNA community discovery is unsupervised i.e. there is no ground truth data to justify the acquired results. And most community discovery algorithms just as other unsupervised learning processes, involve hyper parameters and initialization procedures that may lead to degenerate results or local minima / maxima. The additional diverse approaches explained thus far point out that careful selection of community discovery algorithms is necessary for increasing the quality of obtained results. Therefore, from the above broadly classified approaches, we believe researchers should aim to identify algorithms that are suitable for their data set and the respective outcome required.

Considering this application driven proposition, in this case study, we identify two models that are prominent in terms of their approach and relevancy to SNA.

These approaches are III-A Diffusion Model and III-B Motif-based Model. As it will be self-evident, we aim to take advantage of the different perspective these two approaches provide on community definition and detection.

A. Diffusion Model

One of the main approaches of community detection algorithms is Diffusion. Algorithms that take on diffusion model are not only used in community discovery, but also in viral marketing and churn analysis.

Group of nodes that are clustered through propagation of the same or similar properties summarizes community detection using this model [23], [24]. For example, in social network of a university, similarity among students may be defined by their common hobbies or lessons. These common information make nodes densely connected thus, creating a community. Similarity and shared information is also the basis of influence a node has in its community.

This approach makes the analysis of group dynamics apparent since the behavior of nodes is very closely related to the influence that drives it. For instance, if a highly influencer node leaves a network, eventually, it affects the existence of other nodes of the same community. And what is more, there is a high possibility that the community it belongs to scatters since the influencer node was crucial. Likewise, a node is attracted to a community that is more similar to itself.

B. Motif-based Model

In the above community detection approach, low-level connectivity can be seen as a theme. On other hand, motif-based approaches aim to address insights that can be gained by considering high-level connectivity. These methods achieve this through detecting dense subgraphs that appear in the network a lot more than those in a randomized network. The substructures are defined by a distinct pattern of interaction between nodes. The implication being that these sets of nodes within the hypergraph reflect a specific function or relationship.

Network motifs were first proposed by [25] and can be formally denoted as $M = \{V_{\mathbb{M}}, E_{\mathbb{M}}\}$ where $V_{\mathbb{M}}$ is a set of m nodes and $E_{\mathbb{M}}$ is a set of edges between $m-1$ (line motif) and $\frac{m(m-1)}{2}$ (clique motif) in the motif $\mathbb{M}$ [26]. But generally a network motif consists between 3-8 number of nodes [27]. This is because higher-order motifs are structurally complicated.

There are many variations implemented on this core approach that seek to enhance different defects or achieve a certain goal [26], [28], [29].

This perspective of over-watching the organization of a network for community detection must be incorporated with other techniques that advance towards addressing the negligence of lower-order connectivity and enhance the ability to find multi-layered motifs.

IV. Prototype

To demonstrate community discovery in a network we use Dash [33] Python framework for web applications that extends Cytoscape.js [34] and renders it. Fig. 1 shows a visualization of Label Propagation algorithm over a Twitch data set described in Section V-A. The shape of the networks represented in this figure can be adjusted using an interactive graphical user interface (GUI). Moreover, from the GUI, the user can choose among the two algorithms mentioned so far and other example data sets. After selection of these preferences, the network is rendered in the user's browser along with basic statistical, centrality and network defining properties. Due to space limitation these details are omitted in the figure. The network depicted on the left shows the original network and on the right we see the community substructure of nodes that belong to a selected community.

The cone shaped circular figure depicts hierarchy based on connectivity i.e., the most inner nodes found at the center have the most number of edges. Similarly connectivity decreases going out further in to the outer arcs. Fig. 2 zooms in on the most inner circle of the network and discovered community substructures. In fig. 2 (left) Node 166 (colored red to show membership) on the network can be identified as one of the

Fig. 1: Twitch data set visualization (Left). Discovered Community (Right).

most well connected nodes therefore, it is depicted at the very center of the network. And as anticipated, as shown in fig. 2 (right), the same node is at the center of it's community.

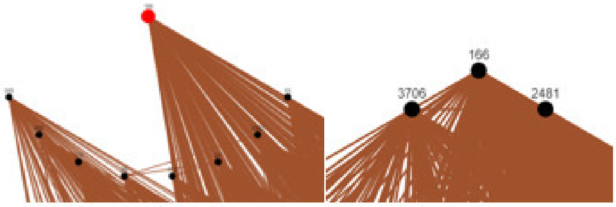


Fig. 2: Node 166 in the network (Left). Node 166 in the its Community (Right).

V. Evaluation

A. Data Set

In this paper we have used two test data sets. The first is a small Twitch [36] data set of 7126 nodes and 35324 edges for visualizing a prototype of a network and discovered communities. The second is a YouTube data set provided by [35]. It defines over 1.1 million nodes as YouTube users and almost 3 million edges represent user friendships. We use this data set for evaluating our ground truth method that is later used in the case study. Both data sets can also be found Stanford Network Analysis Project (SNAP) [32]. The CDR data set is a weighted edge list that consists of over 1.8 million number of nodes and almost 1.6 million number of edges.

B. Algorithm Selection

In order to implement the selected approaches in Section III over the unlabeled CDR data, we have picked two algorithms. Each of these algorithms are selected because they provide a unique advancements on top of the core approaches described and to take advantage of both lower-order and higher-order community detection. These algorithms are:

1) Label Propagation [30]: Each node in the network has a label attached that denotes its community membership. A node in the network joins a community based on the maximum number of neighbors that have a particular label. Therefore, the label propagates through the network quickly and at the end, each node will have been assigned a label. Consequently, nodes with the same label are clustered together as a member of one community. The time complexity of this algorithm is $O\ (m + n)$ where n and m are number of nodes and edges.

2) Edge enhancement approach for Motif-aware community detection (EdMot) [31]: This method fundamentally performs just as explained in Section III-B with measure improvements that other motif-based approaches neglect. That is, the fragmentation problem which is resulted from isolated nodes. This algorithm addresses this issue by deriving a clique from each partitioned module and rewiring the original network. This solution is shown to have increased the quality of higher-order community detection. The time complexity of this algorithm is $O\ (m^{1.5} + n \log n)$ where $O\ (m^{1.5})$ is the time required to find triangle motifs.

C. Data Input and Output

Data input and output format in the case study is made to be uniform for simplicity and consistency. A network that represents a data set is described as an edge list just as shown in Table I . After fitting the graph with the selected model, the output is a list of nodes that are indexed by a community id as shown in Table II.

TABLE I: Edge List Representation of a Social Network

From Node	To Node	Weight (if applicable)
1	6	290
2	7	79
3	545	388
4	210	12
...	...	...

TABLE II: Example Community Membership Output

Community id	Member nodes
1	1, 2, 4, 5...
2	3, 7, 8, 9...
3	6, 10, 11...
...	...

D. Ground Truth Definition

In our work, since we aim to discover communities from unlabeled CDR data set, we use intersectional communities as ground truth. Specifically, let C_L and C_M be a set of all communities discovered by *Label Propagation* and *EdMot* respectively, then $\forall C_l \in \mathbb{C_L}$ and $\forall C_m \in \mathbb{C_M}$, we compute the agreement ratio defined as:

$$Agreement(C_l, C_m) = \frac{\big|\,(C_l \cap C_m)\,\big|}{\big|\,(C_l \cup C_m)\,\big| - \big|\,(C_l \cap C_m)\,\big|} \times 100 \tag{1}$$

Subsequently, those common communities with an agreement threshold $\tau > 40\%$ are included in the set of intersectional ground truth communities C_G. Once ground truth is determined, output of each algorithm, C_L and C_M, is evaluated against C_G using a matrix as shown in Table III.

During evaluation (1) is applied by replacing C_L with C_G when evaluating C_M and vise versa.

Furthermore, (1) also solves the label assignment problem in discovered communities. That is, when evaluating community detection algorithms in such a way, there is high likelihood that *community id* assigned to the output of an algorithm having a high agreement ratio to a community in the ground truth labeled differently. For instance, sets of nodes labeled as C_2 from Table III, are highly similar to C_0 in the ground truth data. This is due to various approaches each algorithm uses and there is no computationally inexpensive way to control the label assignment problem.

TABLE III: Ground Truth Agreement Evaluation Matrix

		Algorithm Output		
		C_0 (3,4,7,8)	C_1 (2,3,4,5,7)	C_2 (1,2,4,5)
Ground Truth	C_0	14.28%	50%	100%
	C_1	50%	42.84%	12.5%
	C_2	16.66%	33.3%	40%

In this arbitrary example, in Table III, we assume nodes (1,2,4,5) $\in$ C_0, (2,3,7,8,9) $\in$ C_1 and (4,5,6) $\in$ C_2 as the ground truth C_G.

E. Evaluation Metrics

We have summarized our result matrix for different algorithm comparisons using three metrics. These are of importance in order to measure algorithms success rate.

1) The number of correct communities: For a community detection algorithm output, an agreement ratio of p along a ground truth agreement comparison matrix column k (i.e., single community), we set a threshold value t, such that if $p_k > t$ then the community detected is accepted as successful. The total number of correct communities η from an algorithm is considered as the first metric.

2) The rate of correct communities per number of ground-truth communities: For a number of ground truth communities N^G and correctly identified communities η, the rate of correct communities discovered is $\rho = \frac{\eta}{N^G}$. This allows for precision evaluation.

3) The mean agreement rate of correct communities: For communities discovered above the threshold value t $(C_1, C_2, ..., C_n)$ with respective agreement rate of $(A_1, A_2, ..., A_n)$, we evaluate the mean agreement rate $\mu = \frac{(A_1, A_2, ..., A_n)}{n}$. This metrics permits us to measure the mean correctness of discovered communities. Even though identified communities are above the set threshold value, evaluating the average agreement rate against the intersectional ground truth communities summarizes an algorithms success well.

Moving forward for better presentation of results, we will use the symbols described in Table IV to represent the above metrics.

TABLE IV: Symbol Representation of Metrics

Symbol	**Corresponding Metric**
N^G	Number of intersectional ground truth communities
ν	Number of discovered communities
η	Number of correct communities
t	Correctness threshold
ρ	Rate of correct communities per number of ground-truth communities
μ	Mean agreement rate of correct communities

F. Performance Evaluation

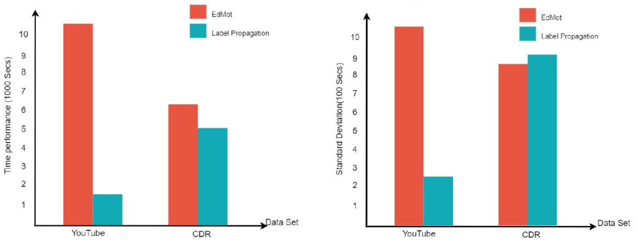
In order to run the following experiments, Amazon EC2 Linux version 5.3.0-1019 instance with 31GB of RAM was used.

TABLE V: Accuracy Performance

Metrics	**YouTube**		**CDR***	
	EdMot	Label P.	EdMot	Label P.
N^G	4114		16908	
ν	9451	62790	253589	50589
$\eta,\ t > 50\%$	2754	4100	11882	16908
ρ	66.94%	99.65%	70.27%	100%
μ	76.47%	99.58%	66.74%	99.98%

* communities with number of nodes < 4 were removed.

Fig. 3 shows the time performance and standard deviation over 30 runs.



(a) Time Performance.

(b) Standard Deviation.

Fig. 3: Time Performance Analysis for YouTube and CDR Data sets..

VI. Conclusion and Future Work

Community discovery in unlabeled real world data sets is subjected to a lot of uncertainty due to its inherit unsupervised nature. Although many promising advances are made continuously, there is room for improvement. In the scope of this study, in order to increase the quality of acquired results, we have shown that selective and context-driven methodology is necessary. And the results demonstrate intersectional ground truth can be used to strengthen the available community discovery algorithms by enforcing correctness through double approval scheme. For instance, from the 50,589 number of communities originally found by Label Propagation in the CDR data set, 16,908 of them were approved by EdMot with average intersectional ground truth agreement rate of 99,98%. This can be interpreted as it has successfully discovered 16,908 number of communities.

In the future, this work can be utilized in order to improve churn prediction models in the telecommunication sector by identifying influential entities.

Acknowledgment

We thank Intellica Business Intelligence Consultancy for providing us the CDR data set and for their continuous support in this case study. This study is supported by TUBITAK TEYDEB under the project ID 3181048.

References

[1] A. Condon and R. M. Karp, “Algorithms for graph partitioning on the planted partition model”, Random Struct Algor 18, pp.116–140, 2001.

[2] F. Altunbey and B. Alatas, “Overlapping Community Detection in Social Networks Using Parliamentary Optimization Algorithm”, International Journal of Computer Networks and Applications (IJCNA) 2, no. 1, pp. 12–19, 2015.

[3] N. Du, B. Wu, X. Pei, B. Wang, and L. Xu, “Community detection in large-scale social networks”, The 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis (WebKDD/SNA-KDD ’07). Association for Computing Machinery, New York, NY, USA, pp. 16-–25, 2007.

[4] M. S. Khan, A. W. A. Wahab, T. Herawan, G. Mujtaba, S. Danjuma and M. A. Al-Garadi, “Virtual Community Detection Through the Association between Prime Nodes in Online Social Networks and Its Application to Ranking Algorithms”, in IEEE Access, vol. 4, pp. 9614-9624, 2016.

[5] L. Runpeng, H. Jun and W. Xiaofan, “VCD: A network visualization tool based on community detection”, 2012 12th International Conference on Control, Automation and Systems, JeJu Island, pp. 1221–1226, 2012.

[6] M. Crampes, M. Plantie, “A unified community detection, visualization and analysis method”, Advances in complex systems, vol. 17, no. 01, pp. 1450001, 2014.

[7] J. David Cruz, C. Bothorel, and F. Poulet, “Community detection and visualization in social networks: Integrating structural and semantic information”, ACM Trans. Intell. Syst. Technol. 5, 1, Article 11 pp. 26, 2013.

[8] R. A. Becker, R. Caceres, K. Hanson, J. M. Loh, S. Urbanek, A. Varshavsky, and C. Volinsky, “A tale of one city: Using cellular network data for urban planning”, IEEE Pervasive Computing, 10(4), pp. 18-–26, 2011.

[9] F. Calabrese, L. Ferrari, and V. D. Blondel, “Urban sensing using mobile phone network data: A survey of research”, ACM Comput. Surv., 47(2):25:1–25:20, 2014.

[10] L. Alexander, S. Jiang, M. Murga, and M. C. Gonz´alez, “Origin–destination trips by purpose and time of day inferred from mobile phone data”, Transportation Research Part C: Emerging Technologies, 58 pp. 240-–250, 2015.

[11] O. J¨arv, R. Ahas, E. Saluveer, B. Derudder, and F. Witlox, “Mobile phones in a traffic flow: a geographical perspective to evening rush hour traffic analysis using call detail records”, PloS one, 7(11) pp. 1-–12, 2012.

[12] A. Bogomolov, B. Lepri, R. Larcher, F. Antonelli, F. Pianesi, and A. Pentland, “Energy consumption prediction using people dynamics derived from cellular network data”, EPJ Data Science, 5(1):13, 2016.

[13] K. Kim, C. Jun, J. Lee, “Improved churn prediction in telecommunication industry by analyzing a large network”, Expert Systems with Applications, vol. 41, Issue 15, pp. 6575–6584, 2014.

[14] Baeth, M.J. et al. (2019). Detecting misinformation in social networks using provenance data, CONCURR COMP-PRACT E, 31(3).

[15] Baeth M. J. et al. (2018) An approach to custom privacy policy violation detection problems using big social provenance data, CONCURR COMP-PRACT E, 30(21).

[16] Baeth, M.J. et al. (2017). Detecting misinformation in social networks using provenance data, SKG-17.

[17] Baeth, M.J. et al. (2015). On the Detection of Information Pollution and Violation of Copyrights in the Social Web, SOCA-15.

[18] Dundar, B. et al. (2016) A Big Data Processing Framework for Self Healing Internet of Things Applications, SKG-16.

[19] Aktas, M.S. et al. (2019), Provenance aware run-time verification of things for selfhealing Internet of Things applications, CONCURR COMP-PRACT E, DOI: 10.1002/cpe.4263.

[20] Aktaş M.S., (2018) Hybrid cloud computing monitoring software architecture, CONCURR COMP-PRACT E, 30(21).

[21] Riveni, M. et al. (2019). Application of provenance in social computing: A case study, CONCURR COMP-PRACT E, 31(3).

[22] Tas, Y. et al. (2016) An Approach to Standalone Provenance Systems for Big Provenance Data, SKG-16.

[23] Newman, Mark EJ. ”The structure and function of complex networks.” SIAM review 45.2, pp. 167–256, 2003.

[24] C. Michele, F. Giannotti, and D. Pedreschi. ”A classification for community discovery methods in complex networks.” Statistical Analysis and Data Mining: The ASA Data Science Journal 4.5, pp. 512-546, 2011.

[25] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, “Network motifs: simple building blocks of complex networks”, Science, vol. 298, no. 5594, pp. 824—827, October 2002.

[26] L. Huang, C. Wang, H. Chao, “Higher-Order Multi-Layer Community Detection”, The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19), pp. 9945–9946, 2019.

[27] S. Yu, J. Xu, C. Zhang, F. Xia, Z. Almakhadmeh and A. Tolba, “Motifs in Big Networks: Methods and Applications”, in IEEE Access, vol. 7, pp. 183322–183338, 2019, doi: 10.1109/ACCESS.2019.2960044.

[28] A. Benson, D. Gleich, and J. Leskovec, “Higher-order organization of complex networks”, Science 353, 6295, pp. 163-–166 2016.

[29] L. Huang, C. Wang, and H. Chao, “A Harmonic Motif Modularity Approach for Multi-layer Network Community Detection” IEEE International Conference on Data Mining, ICDM, Singapore, November 17-20, pp. 1043-–1048, 2018.

[30] U. N. Raghavan, R. Albert, and S. Kumara, “Near linear time algorithm to detect community structures in large-scale networks,”, Physical Review E, vol. 76, p. 036106, 2007.

[31] P. Li, L. Huang, C. Wang, and J. Lai, “EdMot: An Edge Enhancement Approach for Motif-aware Community Detection.”, The 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD ’19), Association for Computing Machinery, New York, NY, USA, pp. 479-–487, 2019, DOI:https://doi.org/10.1145/3292500.3330882

[32] Stanford Network Analysis Project snap,https://snap.stanford.edu/,Accessed: 2020-05-25

[33] Dash framework, https://dash.plotly.com/, Accessed: 2020-05-25

[34] Franz M, Lopes CT, Huck G, Dong Y, Sumer O, Bader GD, “Cytoscape.js: a graph theory library for visualisation and analysis”, Bioinformatics, 32, (2), pp.309–311, 2015

[35] Jaewon Yang, Jure Leskovec, “Defining and evaluating network communities based on ground-truth”, Knowl. Inf. Syst. 42, 1 (January 2015), 181–213. DOI:https://doi.org/10.1007/s10115-013-0693-z

[36] B. Rozemberczki, C. Allen and R.Sarkar, “Multi-scale Attributed Node Embedding”, 1909.13021, cs.LG, 2019.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 479–488
DOI: 10.15439/2020F40
ISSN 2300-5963 ACSIS, Vol. 21

How Coarse-grained Clock Impacts on Performance of NDN Rate-based Congestion Control with Explicit Rate Reporting

Toshihiko Kato, Takumi Enda, Ryo Yamamoto and Satoshi Ohzahata
Graduate School of Informatics and Engineering, University of Electro-Communications
1-5-1, Chofugaoka, Chofu, Tokyo 182-8585 Japan
Email: {kato, enda, ryo_yamamoto, ohzahata}@net.lab.uec.ac.jp

***Abstract*—Named Data Networking (NDN) is a widely adopted future Internet architecture well-suited to large scale content retrieval. The congestion control is one of the research topics actively studied, and the rate-based congestion control method is considered to be fitted to NDN. From the viewpoint of implementation, however, the rate-based method has an issue that it requires the fine-grained clock management, which is hard to implement in off-the-shelf computers. Among the rate-based congestion control methods, an approach in which intermediate nodes report a maximum rate explicitly for a flow is considered to work well. In this paper, we pick up the Multipath-aware ICN Rate-based Congestion Control as an example of explicit rate reporting scheme, and examine how coarse-grained clock gives impacts to its performance. This paper provides the performance evaluation when consumers and NDN routers use the system clock with long time interval. This paper also proposes a method for smoothening Interest sending under a coarse-grained clock and evaluates the performance of proposed method.**

I. Introduction

RESULTING from a drastic increase of content retrieval traffic over the Internet [1], there are many studies on the future Internet architecture called Information Centric Network (ICN). Among them, Named Data Networking (NDN) [2] is a platform widely adopted among the ICN researches. The fundamental concept adopted in NDN is the name of required content, not the address of hosts containing the content. NDN uses two types of packets in all communications: an *Interest packet* and a *Data packet*. A *consumer* that requests a specific content sends an Interest packet containing the content name. A *producer* that provides the corresponding content data returns a Data packet to the consumer. An Interest packet is forwarded using the name prefix it contains, and a Data packet traverse the reverse path of the corresponding Interest packet. *NDN routers* transferring a Data packet cache the packet itself for future redistribution.

The congestion control is one of the hot research topics in NDN [3]. Although it has been also a hot topic in TCP, the mechanisms in TCP congestion control are limited to the congestion window management at data senders [4] and the simple explicit congestion notification [5]. In contrast, various techniques are adopted to the NDN congestion control.

The receiver-driven window-based congestion control approach in NDN is similar to that in TCP. In this approach, the window for Interest packets is maintained in consumers, and Interest packets are sent back to back within the window size. In the traditional proposals, congestion is detected by timeout [6], [7] or the congestion notification [8], and the window size is managed heuristically through an Additive Increase and Multiplicative Decrease (AIMD) mechanism. There are some newly introduced methods, including one which adopts CUBIC TCP like window management [9], ones which use the active queue management scheme such as CoDel [10], watching out the delay of packets in sending queues for congestion detection [11], [12], and one adopts multiple window increase methods dependent on the usage of communication links [13].

In NDN, the rate-based congestion control approach is also studied actively. In this approach, a consumer and routers maintain a rate, in which Interest packets are transmitted regularly with a fixed interval. The rate is determined heuristically by use of congestion notification [14], [15] or by the explicit rate reporting [16]-[20]. Among these methods, the rate-based method with explicit rate reporting provides the best performance. Here, each router monitors the total of data packet traffic receiving over an individual link to an upstream router or a producer, and calculates the optimal Interest packet rate for each Interest-Data flow. Each router sends this optimal rate in a Data packet, and a consumer sends Interest packets according to the reported rate.

From the viewpoint of implementation, however, the rate-based congestion control approach has a problem. Since the transmission speed in recent data links becomes high, e.g., 1 Gbps for typical access links, the fine-grained clock management is required in the rate-based congestion control. For example, if the Data packet size is 10,000 bits and the link speed is 1 Gbps, the interval of Interest packet transmission is 10 μsec when Interest packets are transmitted in a line speed.

If the rate is 0.5 Gbps or 0.3 Gbps, the Interest transmission interval will be 20 µsec or 33.33 µsec, respectively. In order to handle these cases, it is supposed that higher precision clock with shorter tick, such as 1 µsec, will be required in order to manage the Interest packet sending timing.

On the other hand, it is considered that the fine-grained clock management is hard to implement in off-the-shelf computers. In the real world, TCP implementation uses 200 msec (5Hz) and 500 msec (2Hz) clocks for the delayed acknowledgement and retransmission, respectively [21]. So, implementing a rate-based mechanism with micro second order clock is extremely hard, especially in NDN nodes handling a large number of flows simultaneously.

We pointed out this issue and discussed how a coarse-grained clock influences the NDN rate-based congestion control in our previous paper [22]. We adopted the Stateful Forwarding [14] as a target system of the evaluation, and showed that the performance, specifically the Data packet throughput, is degraded largely when a coarse-grained clock is introduced.

However, the Stateful Forwarding is not the best example of the NDN rate-based congestion control methods. As stated above, the explicit rate reporting methods, especially the Multipath-aware ICN Rate-based Congestion Control (MIRCC), provide better performance. In this paper, we examine how the coarse-grained clock influences the performance of MIRCC. We gave some study on this topic in our recent paper [23] but the analysis in this paper is insufficient. Moreover, we propose a method to send Interest packets more smoothly even in the coarse-grained clock environment.

It should be noted that [24] focused on a similar issue on the processing overhead of fine-grained clock management for the rate-based congestion control, but it took a method that exploits a hop-by-hop window control, which does not require the clock management.

The rest of this paper is organized as follows. Section 2 gives the related work focusing on the overview of NDN congestion control, the overhead of fine-grained clock, and MIRCC. Section 3 describes the implementation of MIRCC over the ndnSIM simulator [25], which is a widely used network simulator for NDN, and the modification under the coarse-grained clock. Section 4 gives the performance evaluation results of the original MIRCC with coarse-grained clock. Section 5 proposes a mechanism for smoothening the Interest sending under coarse-grained clock and shows its performance evaluation. In the end, Section 6 concludes this paper.

II. Related Work

A. Related work on NDN congestion control

As described above, most of the congestion control methods in NDN are classified into the receiver-driven window-based and the rate-based methods. Tables 1 and 2 show examples of those methods.

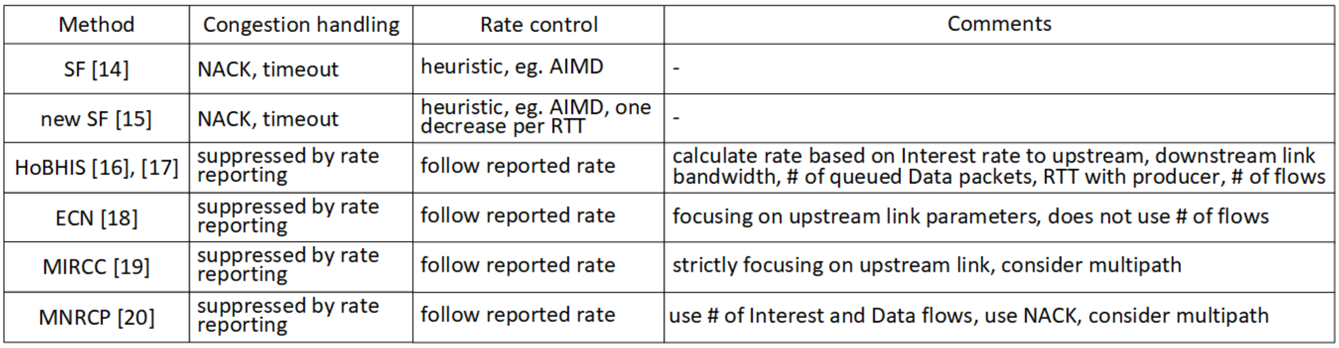
As examples of the traditional receiver-driven window-based methods, the Interest Control Protocol (ICP) [6] and the Content Centric TCP (CCTCP) [7] follow the traditional TCP window control, where a consumer sends Interest packets with the limitation of window size, and the window size is changed according to the AIMD mechanism triggered by Data packet reception and congestion detected by timeout. ICP uses one timer for one flow, just like the TCP round-trip time (RTT) estimation mechanism. CCTCP introduces a timer for an individual Interest packet by inserting a timestamp in Interest and Data packets. The Chunk-switched Hop Pull Control Protocol (CHoPCoP) [8] is another window-based method that introduces the explicit congestion notification with random early marking and changes the window size according to the AIMD mechanism with constant value increasing in the additive increase. CHoPCoP introduces an Interest packet shaper at an intermediate router.

Recently proposed window-based methods adopt new approaches. The CUBIC-based method [9] follows the CUBIC TCP congestion control in its window control, with an explicit congestion tag in a Data packet. The Practical Congestion Control (PCON) [11] and the Window based Congestion Control Mechanism (WinCM) [12] introduces an active queue management monitoring packet-sojourn time at routers. PCON can implement a number of classic

TABLE I. RELATED WORK ON RECEIVER-DRIVEN WINDOW-BASED APPROACH

Method	Congestion handling	Window control	Comments
ICP [6]	timeout	AIMD (only in congestion avoidance phase, similar with TCP Reno)	one timer per flow
CCTCP [7]	timeout	AIMD (slow start and congestion avoidance phases, similar with TCP Reno with timestamp option)	one timer per Interest, timestamp in Interest
CHoPCoP [8]	congestion tag, timeout	AIMD (slow start and congestion avoidance phases, increase by constant value in congestion avoidance phase)	Interest shaping at routers
CUBIC-based [9]	congestion tag, timeout	CUBIC TCP	queue monitoring at routers
PCON [11]	congestion tag, NACK, timeout	TCP Reno or BIC TCP, one decrease per RTT	CoDel at routers, consider multipath
WinCM [12]	congestion tag, NACK, timeout	no congestion: BIC TCP, light congestion: TCP Reno, additive decrease	AQM at routers, consider multipath
VCP-based [13]	load factor	multiple increase mechanism dependent on load factor, multiplicative decrease	-

TABLE II. RELATED WORK ON RATE-BASED APPROACH



Method	Congestion handling	Rate control	Comments
SF [14]	NACK, timeout	heuristic, eg. AIMD	-
new SF [15]	NACK, timeout	heuristic, eg. AIMD, one decrease per RTT	-
HoBHIS [16], [17]	suppressed by rate reporting	follow reported rate	calculate rate based on Interest rate to upstream, downstream link bandwidth, # of queued Data packets, RTT with producer, # of flows
ECN [18]	suppressed by rate reporting	follow reported rate	focusing on upstream link parameters, does not use # of flows
MIRCC [19]	suppressed by rate reporting	follow reported rate	strictly focusing on upstream link, consider multipath
MNRCP [20]	suppressed by rate reporting	follow reported rate	use # of Interest and Data flows, use NACK, consider multipath

loss-based TCP algorithms, and actually, implemented TCP Reno and BIC TCP. PCON avoids unnecessary window reduction resulting from consecutive negative acknowledgments (NACKs) or congestion tags, and allows at most one window decrease within one RTT. WinCM introduces Reno and BIC TCP. During no congestion, i.e. when the cumulative count of congestion tagged Data packets is zero, the window increase follows BIC TCP, and otherwise it follows TCP Reno. As for the window decrease, it adopts the additive decrease instead of the multiplicative decrease. The Variable-Structure Congestion Control Protocol (VCP)-based method [13] introduces the load factor that indicates the status of Data packet sending queue in an individual router. The largest load factor in a flow is conveyed in a Data packet. A consumer adjusts the window increasing according to the load factor for a flow. In the case of low load, the multiplicative increase is used. When moderate load and high load, the additive increase with constant value and the Reno-like additive increase are used, respectively. The window decreasing adopts the multiplicative decrease.

It is considered that those window-based methods have a problem that the window size itself may not be optimal when Data packets are cached in different routers.

The rate-based methods are classified into the non-deterministic scheme, which uses the AIMD mechanism in determining the Interest sending rate, and the explicit rate reporting scheme, in which intermediate routers report the optimal Interest rate to a consumer. The Stateful Forwarding (SF) [14] is an example of the former scheme. It introduces an NACK packet, which has a similar packet structure with Interest, as a negative response to an Interest packet. NACK packets are generated when a router losses an Interest packet, e.g., due to the congestion detection. The new SF [15] is an extension of SF in order to avoid unnecessary rate reduction due to multiple NACKs generated during one congestion event. It reduces the rate once within one RTT, as PCON mentioned above.

In contrast with those non-deterministic methods, there are some methods that enable routers to report a maximum allowed Interest sending rate to a consumer [16]-[20]. These methods take a similar approach but have several differences in the detailed procedure. The Hop-By-Hop Interest Shaping (HoBHIS) [16], [17] is one proposed in an early stage. A router focuses on the upstream link for the Interest packet sending and on the downstream link for the Data packet sending. It also use the number of flows, whose monitoring provides significant overheads for routers. The Explicit Congestion Notification (ECN) based Interest sending rate control method [18] tries to focus only on the upstream link, but it seems to still use the Data packet queue length on the downstream link. MIRCC [19] is sophisticated compared with other methods, in the meaning that it just focuses only on the upstream link. The Interest packet queue length is used instead of the Data packet queue length in other methods. The Rate-based, Multipath-aware Congestion Control Algorithm (MNRCP) [20] is based on a similar scheme with HoBHIS, and it takes account of the numbers of Interest and Data flows separetely.

These rate-based methods with explicit rate reporting are able to control Interest transmission so as to suppress congestion, and as a result they can provide higher throughput. However, their implementation requires the precise timing control for sending Interest packets. The fine-grained clock is hard to implement in off-the-shelf computers, as discussed later. The Hop-by-hop Window-based Congestion Control (HWCC) [24] tries to resolve this problem by taking the hop-by-hop window-based approach. HWCC introduces a hop-by-hop acknowledgment (H-ACK) packet that notifies a router of the reception of an Interest packet together with the Interest rate over the next link to the producer. The per-hop window size between this router and the next router is determined according to the reported rate and the link RTT measured by the H-ACK.

B. Overhead of fine-grained clock implementation

In off-the-shelf computers, the rate control mechanism in which an Interest packet is sent in a specific interval is implemented by the interrupt handling framework. As described in the previous section, it is required to handle various Interest sending rates, and so the interrupting clock tick takes finer value. In an example given above, the clock tick is 1 µsec while the actual sending intervals are 10 µsec, 20 µsec, or 33 µsec.

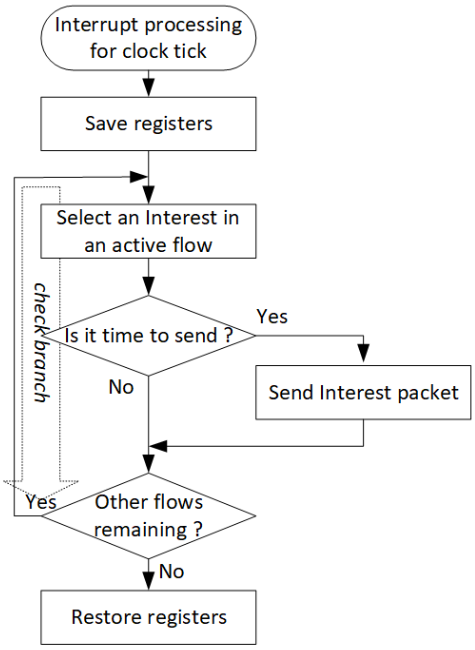


Figure 1. Schematic processing diagram of rate control

Figure 1 shows a schematic processing diagram of the rate control in NDN nodes. A timer hardware generates clock tick interrupts regularly, and then the interrupt handling shown in Figure 1 is invoked. At the beginning, the current values of registers are saved. Then, an Interest packet of an Interest-Data flow stored in a send queue is selected, and it is checked whether the time to send the Interest packet is reached or not. If the sending time is not reached, then an Interest packet of another flow is searched and the same process is done in the case there is another one, otherwise the interrupt handling routine is finished after restoring the register values.

Here, we assume that the CPU clock rate is 4GHz, and that the processing flow when the sending time is not reached (No for "Is it time to send ?" in the figure, *check branch*) requires forty CPU clocks. Then, the time for one check branch is 10 nano sec (1/4 nano sec × 40). If an NDN node handles 100 active flows, the interrupt handling routine takes 1 μsec to be executed even if there are no Interest packets to be sent at that timing. If the clock tick for the rate control is 1 μsec, one CPU executes only the check branches by its full capability. Recent CPUs have multiple cores such as 8 cores. If the number of active Interest-Data flows is 1,000, recent multi-core CPUs cannot support even the check branches.

There are some traditional rate-based schemes, but they use some hardware mechanism instead of the fine-grained clock. For example, the Asynchronous Transfer Mode (ATM) uses a kind of rate-based cell transfer [26], but ATM uses null cells discarded at a receiving side to regulate cell streams at a specific rate. [27] introduced pause packets over Gigabit Ethernet, corresponding to null cells in ATM, that are used only between end nodes and switching hubs. Those approaches are implemented by the MAC level hardware, but NDN Interest packets are handled as a higher level, which makes the hardware support difficult.

TABLE III.
LIST OF MIRCC PARAMETERS

Parameter	Definition
R(t)	Stamping rate at time *t*
C	Capacity of upstream link
N	Equivalent number of flows with full rate
T	Interval of rate calculation
q(t)	Inflated instantaneous queue size
y(t)	Incoming Interest rate during [*t-T*, *t*)
d(t)	Smoothed average RTT
β (t)	Self-tuned parameter for stability
η	Target link utilization

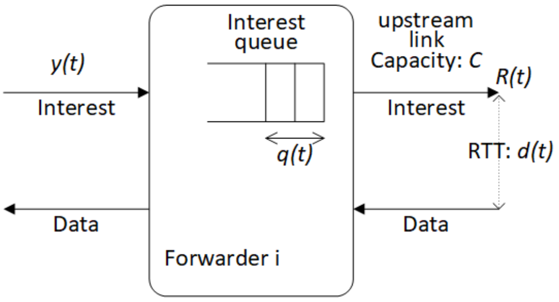


Figure 2. Forwarder model in MIRCC

C. Details of MIRCC

In order to examine the impact of coarse-grained clock to the rate reporting congestion control, we pick up MIRCC. In MIRCC, consumers and routers that forward Interest packets, called forwarders, maintain the parameters indicated in Table 3. The model of a forwarder is given in Figure 2, which explains some of the parameters. It should be noted that the parameters focus on the upstream link of a forwarder. Each forwarder calculates the Interest sending rate *R(t)* for individual flows, at each interval *T*. *R(t)* is specified as the sum of *base_rate(t)* and *excess_rate(t)*. The *base_rate(t)* is the rate to split the allowed link bandwidth among the passing flows. The *excess_rate(t)* is for filling the extra available bandwidth with traffic equally. Each of them is given in the following way.

In order to calculate *base_rate(t)*, a forwarder estimates the number of flows by equation (1).

$$N = {}^{max\left(C,\, y(t)\right)}\Big/_{R\left(t-T\right)} \qquad (1)$$

Then, *base_rate(t)* is computed as follows:

$$base\_rate\left(t\right) = \frac{\eta C - \beta\left(t\right)\frac{q(t)}{d(t)}}{N} \qquad (2)$$

Here, $\beta(t)$ is given by

$$\beta(t) = max\left(0.1, \frac{y(t)-y(t-T)}{y(t)}\right) \qquad (3)$$

As for *excess_rate(t)*, the following equation is used.

$$excess\_rate(t) = R(t-T) - {}^{y(t)}/_{N} \qquad (4)$$

In order to avoid high-frequency oscillation, an exponential weighted moving average (EWMA) is applied to both *base_rate(t)* and *excess_rate(t)* with weight 0.5. Finally, *R(t)* is given by the following equation.

$$R(t) = base\_rate\_ewma(t) + excess\_rate\_ewma(t) \qquad (5)$$

When a router receives a Data packet, it checks the stamping rate included in the packet, and if the included rate is larger than the computed *R(t)*, then *R(t)* is set in the Data packet.

III. MIRCC WITH COARSE-GRAINED CLOCK

A. Implementation of MIRCC over ndnSIM

In order to evaluate the performance of MIRCC, we implemented it over the ndnSIM simulator version 1.0 [28]. The reason we used this version is that we reused the coarse-gained clock implementation in our previous paper. We implemented MIRCC in the following way.

(1) Add R(t) parameter in Data packet

In order to convey *R(t)* in a Data packet, we defined the corresponding parameter, `m_rate`, and the methods to access and modify it, in files `model/ndn-data.{h,cc}`. Besides, the methods for formatting a Data packet, `Serialize()` and `Deserialize()`, is modified in `model/wire/ndnsim.cc`.

(2) Implement a method calculating R(t)

A method called `CalculateRate()` is implemented in `utils/ndn-limits.cc`. This method is invoked every *T* interval, and calculates *R(t)* according to equations (1) through (5) specified above. Here, we need to mention that *y(t)* is given by dividing the number of received Interest packets by *T*, that the leaky bucket size is used as *g(t)*, and that η=1 in our case.

(3) Add various functions in Interest/Data handling

Interest and Data packets are handled in file `model/fw/ndn-forwarding-strategy.cc`. We added the following functions in this file.

- Counting received Interest packets.
- Evaluating smoothed RTT at receiving Data packets.
- Setting *R(t)* in a Data packet if the corresponding value in the Data packet is larger than the calculated *R(t)*.

(4) Behaviour of consumer

A consumer sends Interest packets according to the reported *R(t)* in Data packets. We implemented this kind of consumer as a new class called `ConsumerLi`.

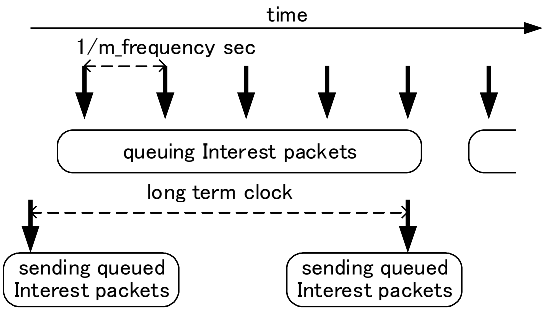

time
1/m_frequency sec
queuing Interest packets
long term clock
sending queued Interest packets
sending queued Interest packets

Figure 3. Implementation scheme of coarse-grained clock system in a consumer

B. Implementation of Coarse-grained Clock Based MIRCC

In the original NDN, the rate control in the consumer is implemented as follows. The sending of Interest packets with a specific rate is done in the `ScheduleNextPacket()` method of the `ConsumerLi` class. In this method, the `SendPacket()` method of the `Consumer` class, which is the superclass, is invoked periodically, every `1.0/m_frequency` seconds. The `SendPacket()` method sends one Interest packet actually.

We emulated a course-grained clock in the `Consumer` class in the following way (Figure 3).

- A clock system with longer tick, such as 100 msec, is implemented in the `ConsumerLi` class. It calls itself periodically with the `Schedule()` method of the `Simulator` class.
- We also introduced a queue storing Interest packets temporarily. This queue is implemented using the `list` class.
- In the `SendPacket()` method, Interest packets are stored in the queue, instead of being sent actually.
- When the longer clock tick is invoked, all the queued Interest packets are transmitted actually.

In the router side in MIRCC, we implemented a coarse-grained clock, by assigning a large value, such as 100 msec, in the interval *T*.

IV. PERFORMANCE EVALUATION MIRCC WITH COARSE-GRAINED CLOCK

A. Experimental setup

The network configuration used in this evaluation is shown in Figure 4, which is a dumbbell configuration where two consumers (C1 and C2) and two producers (P1 and P2) are connected through two routers (R1 and R2). The bandwidth and delay between a consumer and a router, and between a router and a producer are 10Mbps and 50 msec, respectively. Those between routers are 12Mbps and 100 msec, respectively. The length of a Data packet is 1,250 bytes (10,000 bits), and so the link speed 10Mbps and 12Mbps corresponds 1,000 packets/sec and 1,200 packets/sec, respectively. The depth of a token bucket for

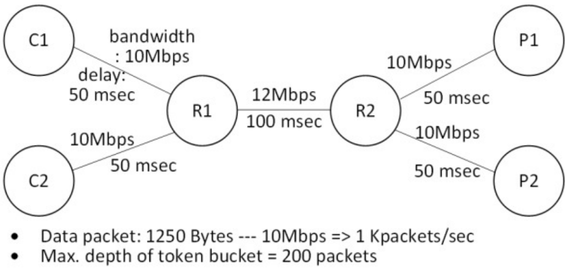


Figure 4. Network configuration for performance evaluation

policing the Interest packets is set to 200 packets in consumers and routers.

We assume that all of the nodes in Figure 4 use the same granularity for the rate control clock. That is, the long-term clock in the consumers and interval *T* in the routes uses the same time interval value. Under these conditions, we evaluated the cases that the rate control clock has 100 msec, 250 msec, 500 msec and 750 msec interval values. In all the evaluation runs, consumer C1 transmits Interest packets to producer P1 between time 1 sec and 10 sec, and consumer C2 sends Interest packets to producer P2 between time 3 sec and 8 sec. In this evaluation, cache is not used.

B. Results of performance evaluation

Figure 5 shows the results, where the time interval is set to 100 msec, 250 msec, 500 msec, and 750 msec. The graphs show the Interest sending rate at consumers C1 and C2.

The result in Figure 5(a) corresponds to the case that the time interval is 100 msec. In the beginning, the Interest sending rate at C1 takes the value around 1,000 packets/sec although there is a slight fluctuation. When C2 starts the communication, the Interest sending rates at C1 and C2 go to 600 packets/sec, with some fluctuations. This result shows that two flows from C1 and C2 share the bandwidth of the bottleneck link evenly. It is considered that MIRCC performs well, even if the time interval is relatively large.

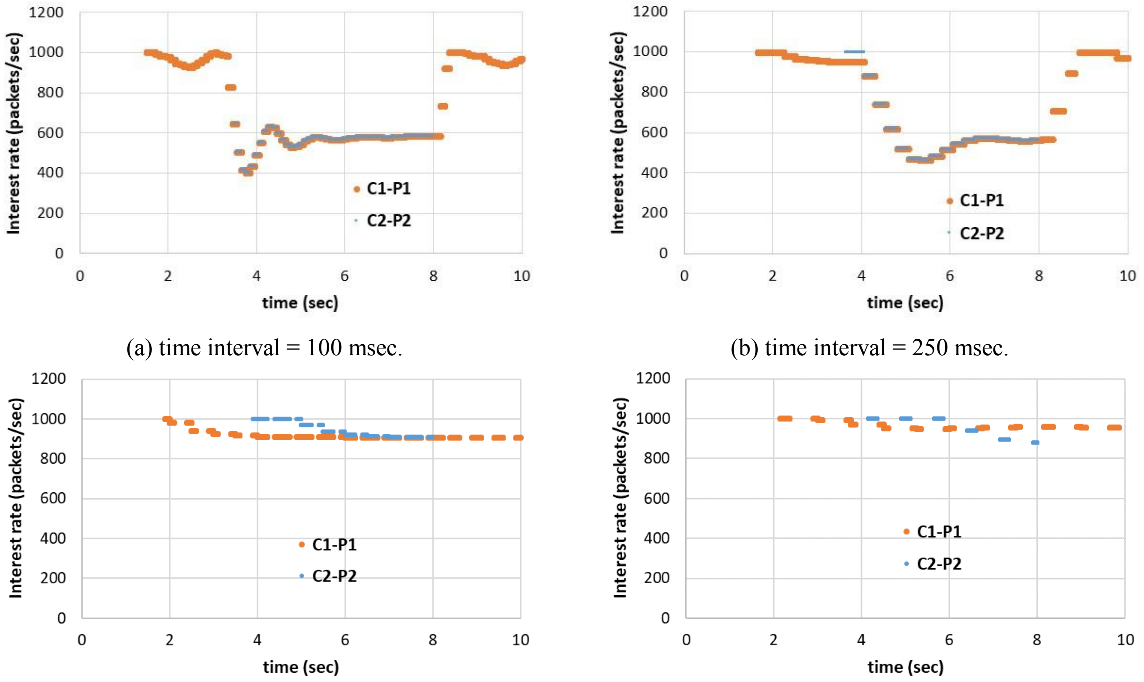
Figure 5(b) shows the result in the case that the time interval is 250 msec. The result seems to be similar with the case of 100 msec. The difference is that the convergence to 600 packets/sec when C2 starts the communication takes larger period, around 2 sec.

In contrast to these results, those in Figures 5(c) and 5(d) show a catastrophic situation. Even if C2 starts its session, the Interest sending rates in C1 and C2 keep 1,000 packets/sec as if the individual consumer communicates alone. These results indicate that the MIRCC mechanism does not work well. It should be noted that the time interval values in those cases are larger than RTT (400 msec) between the consumer and the producer.

We also examined the Data packet delivery losses. Table 4 shows the loss rate of the C1-P1 and C2-P2 flows in the four coarse-grained clock values. These values give the overall loss rate throughout individual flows. It should be

TABLE IIV. OVERALL LOSS RATE OF EACH FLOW

time interval flow	100 msec	250 msec	500 msec	750 msec
C1-P1	0.017	0.103	0.552	0.704
C2-P2	0.036	0.239	0.576	0.712



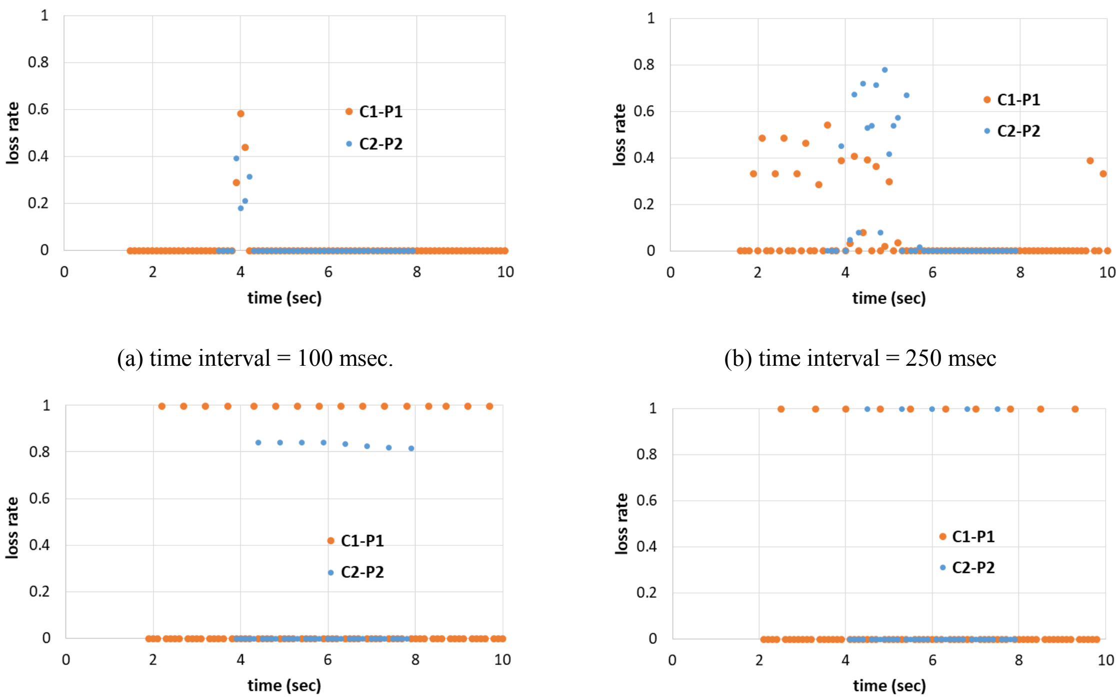
(a) time interval = 100 msec.

(b) time interval = 250 msec.

(c) time interval = 500 msec.

(d) time interval = 750 msec.

Figure 5: Time variation of Interest sending rate.



(a) time interval = 100 msec.

(b) time interval = 250 msec

(c) time interval = 500 msec.

(d) time interval = 750 msec.

Figure 6. Time variation of Data packet loss rate.

noted that our performance evaluation suppresses the retransmissions of Interest packets. So, the loss rate given here is that for original Interest packets. That is, the loss rate is given by the number of delivered Data packets divided by the number of sent Interest packets. In the case that the time interval is 100 msec, the loss rate is small, and that for 250 msec time interval is 0.1 for the C1-P1 flow and 0.24 for the C2-P2 flow. On the other hand, the loss rates for 500 msec and 750 msec time intervals is larger than 0.5. These correspond to the result of the Interest sending rate vs. time that the MIRCC congestion control does not work well under these time interval values.

Figure 6 shows the time variation of the Data packet loss rate calculated for every 100 msec. Figure 6(a) shows that there are some Data packet losses around 4 sec, both in the C1-P1 and C2-P2 flows. This time frame is just after the C2-P2 flow started. In other time frames, all Data packets are delivered. As a result, the overall loss rate is low in the case of 100 msec time interval.

When the time interval is 250 msec, there are some losses from time 2 sec to 4 sec (Figure 6(b)). This means that there are some losses in the C1-P1 flow while only this flow exists. When the second flow started, there are some losses between time 4 sec and 5.5 sec. These losses happen while the Interest sending rate of two flows decreases from 1,000 packets/sec to 600 packets/sec. The reason for these losses is that this decreasing is slower than the case of 100 msec time interval. This delay is considered to be caused by the slow behavior of consumers and routers due to longer time interval.

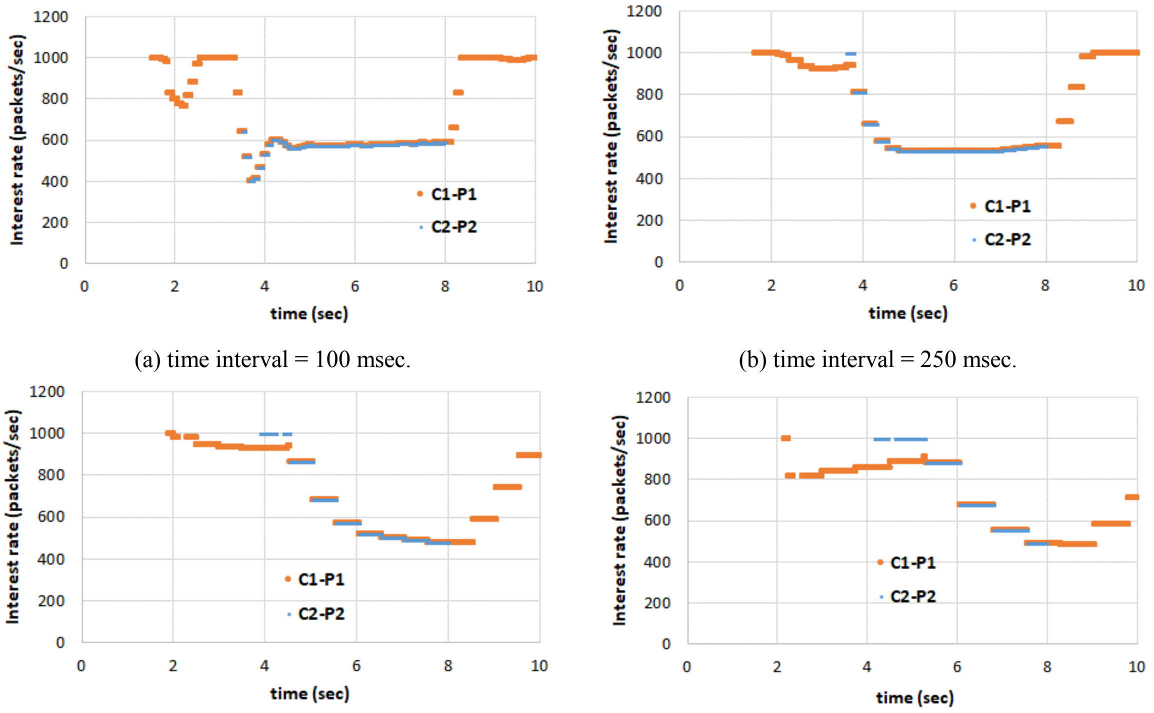
The results in Figures 6(c) and 6(d) are significantly deferent. While the C1-P1 flow is continuing by itself, there are some periods when the loss rate is 100%. This means that an MIRCC flow does not work well in the situation that the shaping clock tick in a consumer and interval T in routers have a large value. As a result, the loss rate becomes larger than 50%, and so the mechanism cannot detect the fact that there are two flows sharing the bottleneck link.

V. Smoothening Interest Sending and Its Performance Evaluation

A. Smoothening Interest sending

In this section, we propose a mechanism that smoothens the Interest packet sending without using fine-grained clock. The performance degradation shown in the previous section comes from long time intervals used in consumers and routers. The long time interval in consumers makes the Interest packet sending bursty, and that in routers makes the update of the stamping rate unfrequent. Although the rate update in routers is difficult to make frequent, the burstiness of Interest sending at consumers can be reduced by a mechanism to process the Interest sending when Data and NACK packets are received. This mechanism was useful for the Stateful Forwarding with coarse-grained clock [22].

Here, we propose an Interest control method that utilizes the Data and NACK packet receiving timing. The receiving processing of Data and NACK packets is triggered by a packet receive interrupt. This does not require large



(a) time interval = 100 msec.

(b) time interval = 250 msec.

(c) time interval = 500 msec.

(d) time interval = 750 msec.

Figure 7. Time variation of Interest sending rate with Interest sending smoothening.

processing overhead, different from the software based rate control mechanism. So, the receiving timing is a good chance to proceed the Interest packet sending. We have added the following mechanism in the coarse-grained clock based MIRCC described in Subsection III.B.

- When a consumer receives a Data or a NACK packet, it processes the received packet and then tries to send the Interest packets that need to be sent by this timing, i.e., those stored in the Interest queue in the implementation described in Subsection III.B.
- This procedure is implemented in the `OnData()` and `OnNack()` methods in the `Consumer` class.

B. Performance evaluation of coarse-grained clock based MIRCC with Interest sending smoothening

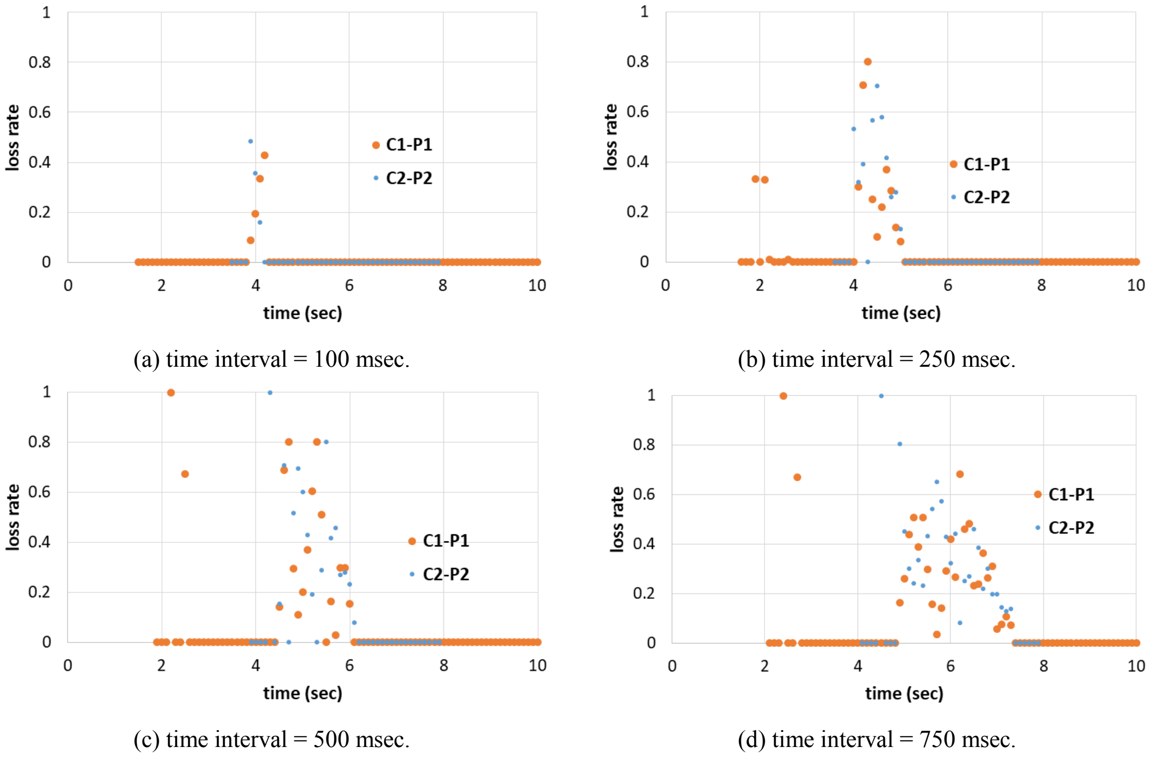
This subsection shows the result of performance evaluation of coarse-grained clock based MIRCC with Interest sending smoothening. We use the same experimental conditions described in Subsection IV.A.

Figure 7 shows the Interest packet sending rate of the proposed method, when the time interval value is 100 msec, 250 msec, 500 msec, and 750 msec. The results in Figures 7(a) and 7(b) are similar to those shown in Figures 5(a) and 5(b), respectively. This means that, when the original MIRCC is working well, the Interest sending smoothening proposed here does not contribute so much to the performance.

TABLE V. OVERALL LOSS RATE OF EACH FLOW WITH INTEREST SENDING SMOOTHENING

time interval flow	100 msec	250 msec	500 msec	750 msec
C1-P1	0.009	0.062	0.156	0.209
C2-P2	0.034	0.145	0.308	0.399

The results shown in Figures 7(c) and 7(d) differ largely from those in Figures 5(c) and 5(d), respectively. When the time interval used in consumers and routers is large, the original MIRCC cannot estimate the optimal rate. As shown in Figure 6(c) and 6(d), there are packet losses even when only one flow exists. Since the data links used by the C1-D1 flow have enough bandwidth when only this flow exists, these packet losses seem to come from the Interest packet shaping using the coarse-grained timer at the consumer. The Interest sending smoothening proposed here, on the other hand, distributes the Interest packet sending to the timing when Data or NACK packets are received. As a result, the Interest sending rates of two flows could be tuned to the half of the bottleneck link bandwidth. That is, the proposed smoothening method takes an effect similar to the Interest packet sending with smaller time intervals, although the Interest sending itself is not performed regularly in an identical interval. Comparing the results in Figure 7(c) and 7(d), the shift of Interest sending rate from 1,000 packets/sec to 600 packets/sec is slower in the case of 750 msec interval. This is because the new stamping rate is reported by routers at the configured time interval, and the consumers take longer time to change the rate in the case of 750 msec interval.



(a) time interval = 100 msec.

(b) time interval = 250 msec.

(c) time interval = 500 msec.

(d) time interval = 750 msec.

Figure 8. Time variation of Data packet loss rate with Interest sending smoothening.

Table 5 and Figure 8 shows the results of Data packet loss rate. Table 5 shows the overall loss rate of individual flows calculated throughout the sessions. Comparing with the results in Table 4, the loss rates when the time interval is 100 msec and 250 msec are similar for the original MIRCC and the MIRCC with Interest sending smoothening. More specifically, the loss rate of the smoothing is slightly smaller than the original MIRCC. For the cases of 500 msec and 750 msec time interval values, the overall loss rate in the smoothening is much better than the original MIRCC, whose loss rate was larger than 50%.

Figure 8 shows the time variation of the Data packet loss late calculated for every 100 msec. By comparing Figures 6(a) and 8(a), the smoothening reduced the Data packet loss rate slightly in the case of 100 msec time interval. Figure 8(b) shows some difference from Figure 6(b) in the case of 250 msec time interval. By introducing the smoothening, the Data packet losses are limited to the time frames of around 2 sec and from 4 sec to 5 sec.

The results in Figures 8(c) and 8(d) changed largely from those in Figures 6(c) and 6(d). By introducing the smoothening, the Data packet losses in the single flow are reduced largely in the cases of 500 msec and 750 msec time intervals. When two flows exist in the case of 500 msec and 750 msec time intervals, the smoothening also reduced the loss rate. While there were several 100 % loss rate observations in the original MIRCC, there is just one 100 % loss rate observation by use of the smoothening.

VI. Conclusion

This paper discussed about the impact of the coarse-grained clock on MIRCC, one of rate-based NDN congestion control methods with explicit rate reporting. We implemented MIRCC over the ndnSIM simulator and evaluated the performance in a dumbbell network when the rate control clocks in consumers and routers have a large tick value. The result showed that the rate control of MIRCC does not work well due to bursty Interest packet sending at consumers and rough rate detection at routers. We proposed a method which smoothens the Interest packet sending by use of the timing of Data packet receptions. The results of performance evaluation showed that the smoothening is effective in reducing the Data packet losses when the rate control clock has a large tick value.

References

[1] Cisco public, *Cisco Annual Internet Report (2018-2023)*. White paper, 2020.

[2] V. Jacobson et al., “Networking Named Content,” in *Proc. CoNEXT ’09*, pp. 1-12.

[3] Y. Ren, J. Li, S. Shi, L. Li, G. Wang, and B. Zhang, “Congestion control in named data networking - A survey,” *Computer Communications*, vol. 86, pp. 1-11, Jul. 2016.

[4] A. Afanasyev, et al., “Host-to-Host Congestion Control for TCP,” *IEEE Commun. Surveys & Tutorials*, vol. 12, no. 3, pp. 304-342, 2010.

[5] K. Ramakrishnan, S. Floyd, and D. Black, *The Addition of Explicit Congestion Notification (ECN) to IP*. IETF RFC 3168, Sep. 2001.

[6] G. Carofiglio, M. Gallo, and L. Muscariello, “ICP: Design and Evaluation of an Interest Control Protocol for Content-Centric Networking,” in *Proc. IEEE INFOCOM 2012*, pp. 304-309.

[7] L. Saino, C. Cocora, and G. Pavlou, "CCTCP: A Scalable Receiver-driven Congestion Control Protocol for Content Centric Networking," in *Proc. IEEE ICC 2013*, pp. 3775-3780.

[8] F. Zhang, Y. Zhang, A. Reznik, H. Liu, C. Qian, and C. Xu, "A Transport Protocol for Content-Centric Networking with Explicit Congestion Control," in *Proc. IEEE ICCCN 2014*, pp. 1-8.

[9] Y. Liu, X. Piao, C. Hou, and K. Lei, "A CUBIC-Based Explicit Congestion Control Mechanism in Named Data Networking," in *Proc. IEEE CyberC 2016*, pp. 360-363.

[10] K. Nichols and V. Jacobson, "Controlling Queue Delay," *ACM Magazine Queue*, vol. 10, issue 5, pp. 1-15, May 2012.

[11] K. Schneider, C. Yi, B. Zhang, and L. Zhang, "A Practical Congestion Control Scheme for Named Data Networking," in *Proc. ACM ICN 2016*, pp. 21-30.

[12] M. Wang, M. Yue, and Z. Wu, "WinCM: A Window based Congestion Control Mechanism for NDN," in *Proc. IEEE HotICN 2018*, pp. 80-86.

[13] S. Xing, B. Yin, J. Yao, H. Zhang, Q. Zhai, and H. Shi, "A VCP-based Congestion Control Algorithm in Named Data Networking," in *Proc. IEEE IAEAC 2018*, pp. 463-468.

[14] Y. Cheng, A. Afanasyev, I. Moiseenko, B. Zhang, L. Wang, and L. Zhang, "A case for stateful forwarding plane," *Computer Communications*, vol. 36, no. 7, pp. 779-791, Apr. 2013.

[15] T. Kato and M. Bandai, "Congestion Control Avoiding Excessive Rate Reduction in Named Data Network," in *Proc. IEEE CCNC 2017*, pp. 1-6.

[16] N. Rozhnova and S. Fdida, "An effective hop-by-hop Interest shaping mechanism for CCN communications," in *Proc. IEEE INFOCOM Workshops 2012*, pp. 322-327.

[17] N. Rozhnova and S. Fdida, "An extended Hop-by-hop Interest shaping mechanism for Content-Centric Networking," in *Proc. IEEE GLOBECOM 2014*, pp. 1198-1204.

[18] J. Zhang, Q. Wu, Z. Li, M. A. Kaafar, and G. Xie, "A Proactive Transport Mechanism with Explicit Congestion Notification for NDN," in *Proc. IEEE ICC 2015*, pp. 5242-5247.

[19] M. Mahdian, S. Arianfar, J. Gibson, and D. Oran, "Multipath-aware ICN Rate-based Congestion Control," in *Proc. ACM ICN 2016*, pp. 1-10.

[20] S. Zhong, Y. Liu, J. Li, and K. Lei, "A Rate-based Multipath-aware Congestion Control Mechanism in Named Data Networking," in *Proc. IEEE ISPA/IUCC 2017*, 174-181.

[21] K. Fall and W. Stevens, *TCP/IP Illustrated, Volume1; The Protocols, Second Edition*. Addison-Wesley, 1994.

[22] T. Kato, K. Osada, R. Yamamoto, and S. Ohzahata, "A Study on How Coarse-grained Clock System Influences NDN Rate-based Congestion Control," in *Proc. IARIA ICN 2018*, pp. 35-40.

[23] T. Kato, T. Enda, R. Yamamoto, and S. Ohzahata, "A Study on Performance of Explicit Rate Report Based Congestion Control under Coarse-grained Clock Management," in Proc. *INSTICC DCNET 2020*, pp. 82-88.

[24] T. Kato and M. Bandai, "A Congestion Control Method for NDN Using Hop-by-hop Window Management," in *Proc. IEEE CCNC 2018*, pp. 1-6.

[25] A. Afanasyev, I. Moiseenko, and L. Zhang, "ndnSIM: NDN simulator for NS-3," *NDN, Technical Report NDN-0005*, 2012.

[26] ITU-T, *B-ISDN asynchronous transfer mode functional characteristics, Series I: Integrated Services Digital Network*. Recommendation I.150, Feb. 1999.

[27] Y. Yamamoto, "Estimation of the advanced TCP/IP algorithms for long distance collaboration," *Fusion Engineering and Design*, vol. 83, issue 2-3, pp. 516-519, Apr. 2008.

[28] NDN, "Overall ndnSIM documentation; Forwarding Strategies," http://ndnsim.net/1.0/fw.html.

4$^{\text{th}}$ Workshop on Internet of Things—Enablers, Challenges and Applications

THE Internet of Things is a technology which is rapidly emerging the world. IoT applications include: smart city initiatives, wearable devices aimed to real-time health monitoring, smart homes and buildings, smart vehicles, environment monitoring, intelligent border protection, logistics support. The Internet of Things is a paradigm that assumes a pervasive presence in the environment of many smart things, including sensors, actuators, embedded systems and other similar devices. Widespread connectivity, getting cheaper smart devices and a great demand for data, testify to that the IoT will continue to grow by leaps and bounds. The business models of various industries are being redesigned on basis of the IoT paradigm. But the successful deployment of the IoT is conditioned by the progress in solving many problems. These issues are as the following:

The IoT technical session is seeking original, high quality research papers related to such topics. The session will also solicit papers about current implementation efforts, research results, as well as position statements from industry and academia regarding applications of IoT. The focus areas will be, but not limited to, the challenges on networking and information management, security and ensuring privacy, logistics, situation awareness, and medical care.

- The integration of heterogeneous sensors and systems with different technologies taking account environmental constraints, and data confidentiality levels;
- Big challenges on information management for the applications of IoT in different fields (trustworthiness, provenance, privacy);
- Security challenges related to co-existence and interconnection of many IoT networks;
- Challenges related to reliability and dependability, especially when the IoT becomes the mission critical component;
- Zero-configuration or other convenient approaches to simplify the deployment and configuration of IoT and self-healing of IoT networks;
- Knowledge discovery, especially semantic and syntactical discovering of the information from data provided by IoT.

Topics

The IoT session is seeking original, high quality research papers related to following topics:

- Future communication technologies (Future Internet; Wireless Sensor Networks; Web-services, 5G, 4G, LTE, LTE-Advanced; WLAN, WPAN; Small cell Networks…) for IoT,
- Intelligent Internet Communication,
- IoT Standards,
- Networking Technologies for IoT,
- Protocols and Algorithms for IoT,
- Self-Organization and Self-Healing of IoT Networks,
- Object Naming, Security and Privacy in the IoT Environment,
- Security Issues of IoT,
- Integration of Heterogeneous Networks, Sensors and Systems,
- Context Modeling, Reasoning and Context-aware Computing,
- Fault-Tolerant Networking for Content Dissemination,
- IoT Architecture Design, Interoperability and Technologies,
- Data or Power Management for IoT,
- Fog—Cloud Interactions and Enabling Protocols,
- Reliability and Dependability of mission critical IoT,
- Unmanned-Aerial-Vehicles (UAV) Platforms, Swarms and Networking,
- Data Analytics for IoT,
- Artificial Intelligence and IoT,
- Applications of IoT (Healthcare, Military, Logistics, Supply Chains, Agriculture, ...),
- E-commerce and IoT.

The session will also solicit papers about current implementation efforts, research results, as well as position statements from industry and academia regarding applications of IoT. Focus areas will be, but not limited to above mentioned topics.

Technical Session Chairs

- **Cao, Ning,** College of Information Engineering, Qingdao Binhai University
- **Furtak, Janusz,** Military University of Technology, Poland
- **Zieliński, Zbigniew,** Military University of Technology, Poland

Program Committee

- **Al-Anbuky, Adnan,** Auckland University of Technology, New Zealand
- **Antkiewicz, Ryszard,** Military University of Technology, Poland
- **Brida, Peter,** University of Zilina, Slovakia
- **Chudzikiewicz, Jan,** Military University of Technology in Warsaw, Poland

- **Cui, Huanqing,** Shandong University of Science and Technology, China
- **Ding, Jianrui,** Harbin Institute of Technology, China
- **Fouchal, Hacene,** University of Reims Champagne-Ardenne, France
- **Fuchs, Christoph,** Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE, Germany
- **Hodoň, Michal,** University of Žilina, Slovakia
- **Johnsen, Frank T.,** Norwegian Defence Research Establishment (FFI), Norway
- **Karpiš, Ondrej,** University of Žilina, Slovakia
- **Krco, Srdjan,** DunavNET
- **Laqua, Daniel,** Technische Universität Ilmenau, Germany
- **Lenk, Peter,** NATO Communications and Information Agency, Other
- **Li, Guofu,** University of Shanghai for Science and Technology, China
- **Marks, Michał,** NASK - Research and Academic Computer Network, Poland
- **Monov, Vladimir V.,** Bulgarian Academy of Sciences, Bulgaria
- **MURAWSKI, Krzysztof,** Military University of Technology, Poland
- **Niewiadomska-Szynkiewicz, Ewa,** Research and Academic Computer Network (NASK), Institute of Control and Computation Engineering, Warsaw University of Technology
- **Papaj, Jan,** Technical university of Košice, Slovakia
- **Savaglio, Claudio**
- **Ševčík, Peter,** University of Žilina, Slovakia
- **Shaaban, Eman,** Ain-Shams university, Egypt
- **Staub, Thomas,** Data Fusion Research Center (DFRC) AG, Switzerland
- **Suri, Niranjan,** Institute of Human and Machine Cognition
- **Wrona, Konrad,** NATO Communications and Information Agency

DOI: 10.15439/2020F154
ISSN 2300-5963 ACSIS, Vol. 21

Coordinated autonomic loops for target identification, load and error-aware Device Management for the IoT

Neil Ayeb*, Eric Rutten†, Sebastien Bolle*, Thierry Coupaye* and Marc Douet*

*Orange Labs, Meylan, France
*Email: firstname.lastname@orange.com

†Univ. Grenoble Alpes, Inria, CNRS, LIG, F-38000 Grenoble France
†Email: firstname.lastname@inria.fr

***Abstract*—With the expansion of Internet of Things (IoT) that relies on heterogeneous, dynamic, and massively deployed devices, device management (DM) (i.e., remote administration such as firmware update, configuration, troubleshooting and tracking) is required for proper quality of service and user experience, deployment of new functions, bug corrections and security patches distribution.**

Existing industrial DM platforms and approaches do not suit IoT devices and are already showing their limits with a few static home devices (e.g., routers, TV Decoders). Indeed, undetected buggy firmware deployment and manual target device identification are common issues in existing systems. Besides, these platforms are manually operated by experts (e.g., system administrators) and require extensive knowledge and skills. Such approaches cannot be applied on massive and diverse devices forming the IoT.

To tackle these issues, our work in an industrial research context proposes to apply autonomic computing to DM platforms operation and impact tracking. Specifically, our contribution relies on automated device targeting (i.e., aiming only suitable devices) and impact-aware DM (i.e., error and anomalies detection preceding patch generalization on all suitable devices of a given fleet). Our solution is composed of three coordinated autonomic loops and allows more accurate and faster irregularity diagnosis, vertical scaling along with simpler IoT DM platform administration.

For experimental validation, we developed a prototype that demonstrates encouraging results compared to simulated legacy telecommunication operator approaches (namely Orange).

Keywords*–*device management, multiple loop cooperation, internet of things, firmware update, configuration management.

I. Context & Motivation

A. Device Management

DM CONSISTS of remote (and potentially massive) operations on a fleet of deployed devices. Managed devices include, but are not limited to, workstations, broadband or IoT gateways and smartphones. After the wide usage of Blackberry and smartphones later-on in business context, enterprises used Mobile Device Management (MDM) for application remote installing, configuration provisioning and over-the-air (OTA) software updates [1].

Historically, DM solutions targeted workstation for configuration management, application provisioning and OS (Operating System) patching. Afterwards, with high speed (e.g., Digital Subscriber Line, Cable) internet accesses widespread, routers and modems (a.k.a Boxes) required user specific configurations and regular firmware updates to enhance Consumer Premises Equipment (CPE) lifespan. This led to the creation of an industry consortium, the Broadband Forum [2], (initially DSL Forum) that aim to standardize network technologies and protocols for internet access over phone lines, then progressively integrated *home DM* specifications and architectures among other topics.

With the rise in complexity of services and device computing power, later Telco DM solutions started incorporating remote troubleshooting features.

B. DM Features

We define DM as initial and 'in-life', remote, firmware updates (Maintenance) configuration of devices (Provisioning), probe data collection (Monitoring), and troubleshooting (Assistance) [3]. Maintenance lets system administrators push firmwares on the entirety or specific parts of a given device fleet (e.g., patching a security issue with a new firmware, deploying a new feature for beta test members). It was the core feature for early solutions. Historically, DM was mainly about remote firmware updates. With the increasing complexity of devices and maintenance costs, solutions started incorporating new features such as those we mentioned above. Provisioning incorporates new services (de)activation, and equipment behavior modification by editing a given device data-model (e.g., change its data platform URL, enable new sensors of a modular peripheral, pair with a nearby device). Monitoring grants log and runtime data to be pushed using a DM protocol to a central platform for data analysis (e.g., using QoS (Quality of Service)) measures to detect network congestion or average resource consumption per device category). Assistance empowers system administrator to remotely execute diagnostic commands on devices aiming for troubleshooting without physical intervention (e.g., triggering reboots, factory configuration resets aiming to fix a peripheral).

DM is crucial a continuous, correct functioning of devices, while ensuring proper QoS for end users or business partners. Yet faulty DM operations can cause consequent losses for companies, especially when they target a whole fleet of devices. In October 2019, Google pushed a faulty firmware to a part of its home voice assistants *Google Home* and *Google Mini* fleet. After a reboot, devices became unable to boot or were locked in an infinite loop. This incident had a negative economic impact on Google since all devices were replaced free of charge [4]. If we suppose that 1% of the fleet (i.e., 52 Millions at the end of Q4 2018) [5] was affected, it would represent 520,000 devices bricked to be potentially replaced, plus shipping costs. Moreover, HP-Enterprise (HPE) and Dell

EMC deployed a critical firmware update designed by their enterprise SSD manufacturer Western Digital. Indeed, a faulty firmware was integrated within these drives at release time (2015). Without fix, data loss become basically inevitable after 32768 hours of running [6] [7]. Beside fault mitigation, DM can be used for performance enhancement and new features deployment. Indeed, *Tesla* electric vehicles received in November 2019 an OTA software update via 4G-LTE Network that enhanced the peak output power of *Model S* engine by 37kW. Besides, Tesla's firmwares constantly improve self-driving and charging capabilities. Telecommunication operators use some DM features of their home and enterprise internet gateways for remote troubleshooting when customers call after-sales service for technical assistance. It allows pushing new firmwares and configurations to its deployed fleets of home devices (e.g., Broadband and Fiber Gateways, TV decoders, Wi-Fi Extenders). These experiences show how crucial DM is for businesses and give a peak about its evolution with Internet of Things widespread.

IoT relies on massive deployment of various devices (e.g., sensors, gateways, actuators..). However, they are usually heterogeneous regarding their computing, storage, and communication capabilities (i.e., simple sensors configured to push data vs. peripherals with high computing power) and environments (i.e., mobile on-battery devices vs. rather powerful fixed gateways). IoT Services can rely on multiple devices from distinct manufacturers and owners. This entails the need for collaborative DM Platforms for service provisioning, troubleshooting and configuration.

C. Challenges and contributions

These characteristics, compared to legacy DM, involve several challenges regarding IoT DM.

- Heterogeneity: From few types of devices to numerous ones and various environments (connectivity, nearby devices), computing, storage and networking capabilities. IoT allows services to use multiple devices in contrast with single owner traditional objects.
- Dynamicity: From internal states (battery, computing load, running services, network conditions) that are rather stable to versatile ones that changes over time.
- Interoperability: From isolated technical solutions (i.e., one DM platform per device type) to multi-platform devices enabling multiple interdependent services.
- Scalability: From few devices managed by centralized designs to massive amounts of devices managed via distributed designs.

In this study, our goal is DM platform operation automation, therefore tackling device **Heterogeneity** and environment **Dynamicity** by adjusting execution speed to capabilities of devices and infrastructure (i.e., hardware, current load and network congestion). Our adaptation strategy relies on operational metrics such as monitored DM operation execution error count or infrastructure response time. Furthermore, a step towards **Scalability** management is introduced in this paper via vertical scaling [8].

Operating DM solutions require consequent efforts and expertise. For instance, according to interviews realized with DM teams, updating a fleet of residential routers at Orange is done as follows:

First, the DM system should be given (by its administrator) a firmware and its target (i.e., subset of devices in this version of this manufacturer) that will be processed. For each DM operation, it is potentially required to add specific parameters (e.g., higher retry tolerance, variable firmwares depending on subscribed services or owners). Once launched, firmware installation is remotely done for each device. During that phase, active monitoring is performed remotely by administrators for fault detection and tracking. After complete target migration to the new firmware, a rollback can potentially be triggered if unexpected device behavior is reported by users. This typical workflow cannot be adequate for IoT DM if the previously mentioned challenges are taken into consideration due to high complexity and failure risk.

In order to address the above-mentioned IoT DM challenges, this paper makes the following architectural and experimental contributions:

- A **Coordinated multi-loop autonomic architecture** for IoT DM;
- An **Operation Generation & Target Identification** loop for automatic target (i.e., subset of devices to process) identification and operation launching;
- A **Decomposition, Enforcement and Tracking** loop for DM operations execution and monitoring;
- A **Speed Regulation** loop for batch size variation (i.e., amount of devices processed in each **Decomposition, Enforcement and Tracking** loop iteration) depending on anomalies and infrastructure load;
- A **Proof of Concept** for speed regulation and impact assessment validation.

The rest of this paper is organized as follows. Section 2 describes existing research and industrial work. Section 3 introduces our contribution regarding Autonomic IoT DM. Section 4 discusses our proof-of-concept and experimental results. Finally, Section 5 presents our conclusion and perspectives.

II. Related Work

We propose to analyze the different categories of DM solutions, i.e. Home, Mobile, Workstation and IoT, to compare their proposed features as well as technical implementations.

Our survey on existing work led us to conclude that the main objective of authors is to optimize executing firmware updates on constrained embedded boards, therefore focusing on the process itself and not fleet management or DM platform operation. These devices are often working using battery power and do not offer much computing power. Therefore, update process adaptation (e.g., optimization [9], [10], [11], securing [12], [13], [14]) is explored for such devices. Existing standard DM protocol are also studied and compared for IoT usecases [15],[16], [17].

To the best of our knowledge, no existing research work aims to automate DM platforms operation for IoT management.

TABLE I. IoT Platform DM Capabilities Survey

	Orange Live Objects	Amazon Web Services IoT	Microsoft Azure Iot Hub	Bosch IoT	IBM Watson
Firmware Update	✓	✓	✓	✓	✓
Configuration Update	✓	✓	✓	✓	✓
Standard DM Protocol	LWM2M + Custom	Custom	Custom	CWMP - OMA-DM - LWM2M	Custom
Campaigns	✓	✓	✓	✓	✓
Execution Speed Regulation	✗	✗	✗	✗	✗
Dynamic Target	Partial	✓	✓	Partial	✗
Reactivity to DM ops. errors	✗	Progress Reporting	Progress Reporting	Progress Reporting	Manual

A. Industrial Solutions Analysis

From our analysis of existing Home, Mobile, Workstation, and IoT DM solutions and their features we observe that:

- Home (e.g., Internet residential gateways and TV decoders) management does not offer application stores except Android TV devices therefore partially resembling IoT management in that point. The main difference with the IoT is that home devices are always plugged in AC power, usually connected with reliable network connections and embed some auto-diagnosis features.
- MDM differs from IoT DM by a significant firmware fragmentation [18] and user-triggered updates. Android will also allow seamless transition (i.e., no service interruption) from one firmware to another [19] in its next release. Such advanced mechanisms cannot easily be implemented in constrained IoT devices. For instance, IoT hardware limitations does not allow the separation between applications (Google Store) and kernel-OS packages (Android ROM).
- Workstation DM, (according to our internal survey at Orange), allows OS updates and configurations installation while machines are used with little acceptable performance degradation. These updates are applied outside work hours by a remote reboot command (or by waiting for end users to reboot). IoT updates could be theoretically differed but firmware update implementations usually triggers a reboot at the end. Even though some IoT devices run UNIX kernels or microkernels, they do not include package management like their desktop and server counterparts.

Configuration updates and remote troubleshooting of IoT devices have some similarities but also significant differences with home, mobile and workstation management. The key difference between those, is firmware nature (OS + applications for some, OS only for others) and update mechanisms (instant reboot, firmware developed by manufacturer or a third party). Besides, home and mobile device manufacturers keep their firmware closed source and rarely implement DM API. IoT however is going towards openness via unified standards (e.g., OneM2M) and protocols (e.g., Open Mobile Alliance Lightweight Machine to Machine LWM2M) Due to the aforementioned differences, efforts have been focused to analyze and identify limitations in telecommunication DM, and also investigate Orange current management strategy for its device fleets, in addition to surveying industrial IoT platforms capabilities and features.

1) Telecommunication Operator Home Device Management:

a) Features: Orange current Home DM solution is internally developed. Its based on CWMP [20] protocol (i.e., CPE WAN Management Protocol), also known by the name of its technical specification document: TR-069. It enables remote firmware updates, tracking, troubleshooting, and configuration of Livebox Routers (i.e., Orange's Internet Gateways) and Set-Top-Boxes (TV Decoders and Multimedia Gateways). This solution is currently centralized and manages around 20 Millions of CWMP Devices.

CWMP proprietary platforms (Arris, SagemCom, Axiros) and open-source solutions (e.g., GenieACS [21], FreeACS [22]) exist and are being operated for router and other TR-069 compliant devices management. They cover firmware update and device configuration but do not usually offer advanced mechanisms such as dynamic device groups, operation tracking and history.

b) Operation Analysis: Our study of Orange's strategy for HomeLAN DM shows this behavior: A firmware will be installed on a few devices and these will be manually observed by experts (i.e., who have the ability to interpret probe data, and trace errors). Afterwards, several thousands of devices will be migrated and will stay under observation during approximately ten days. These data will be analyzed by system administrators with other indicators such as hotline calls for malfunctioning devices. In case of very high amount of reports by users and field-technicians, firmware will be declared non-viable by system administrator, the global operation canceled and each migrated equipment is returned to its original firmware. Otherwise, firmware is set to be installed in the entire fleet.

2) IoT Device Management:

a) Features: In Table I, we compare existing technical IoT platform DM capabilities. Our criteria include minimal DM features (Firmware and Configuration Upgrade), Campaign Launching (i.e., operations on multiple devices of the fleet), error reactions (ability to react to execution abnormal behavior from a platform or devices), speed regulation (how much devices are processed in a given time slot) or dynamic target (entities that includes devices depending on their current hardware or software states).

All of existing platforms incorporate at least firmware and configuration upgrade capabilities. Depending on the platform standard DM protocols can be supported or custom ones can be used. Orange Business Services (OBS) Internet of Things platform Live Objects [23] incorporates DM capabilities. It is targeting rather constrained devices and enables remote firmware update. AWS IoT as Microsoft Azure IoT Hub and Bosch IoT Suite adds to these features dynamic group capabilities, progression tracking of operations and additional multiple standard DM protocols support for Bosch IoT Suite.

However, IBM Watson IoT only support firmware and configuration updates on multiple devices via custom proprietary DM protocols.

b) Operation Analysis: Strategies are defined by system administrators and are specific to use cases. It depends on which devices are managed, what hardware capabilities they have, their network environments and finally how much device anomaly/error/loss is tolerable. To the best of our knowledge, no public documentation regarding such strategies exists.

B. Discussion of existing work

A part of existing research work [9], [10], [11], [12], [13], [14], [15], [16], [17], aims to tackle embedded and security constraints regarding firmware update process therefore optimizing DM at a device level. None focus on large fleet management (e.g., numerous IoT gateways, sensors or actuators) and DM platform operation optimization as a whole (e.g., configuring firmware-device association, firmware deployment strategy).

Feature-wise, the previously mentioned approaches and solutions for Home DM are not suitable for IoT objects. Indeed, the former are usually very specific designs, thus very efficient for single device types, but unsuitable for heterogeneous device fleets in various and dynamic environments (i.e., constantly changing battery states, computing load, network signal, activated services). In contrast, traditional home network devices are usually plugged-in AC power, mostly wired to a wide-area-network access, and managed by a single specific owner/vendor DM solution. Industrial solutions however such as Amazon IoT [24], Azure IoT Hub [25], Bosch IoT Suite [26], and Orange Live Objects [23] offer many characteristics that suit IoT DM such as light communication cost implementation, multi-protocol and various device type compatibility.

However, performing large operations still require the intervention of experts. Operation and configuration is done manually by them (as with Home DM). Indeed, these platforms do not implement mechanisms that automate target identification in a fleet, or allow error detection and mitigation operation execution and tracking. Moreover, device states (e.g., load, network, software) are not taken into consideration for management operations and fleet firmware update. The dynamicity of device states implies several specific configurations to be deployed on subsets of a given device fleet, even in case of homogeneous set of devices. Thus, active and complex reconfiguration of such DM solutions is required for correct and optimal functioning.

These characteristics lead us towards investigating autonomic computing approaches for IoT DM platform operation and configuration while also automating anomaly detection and mitigation. We aim to add an autonomic management layer on top of existing DM platform to enhance their capabilities and automate their operation.

III. Autonomic IoT DM Managers

We propose an architecture based on three cooperating autonomic loops respectively for 'operation launching and target identification', 'DM operations speed regulation', and 'on-device execution while detecting anomalies'. Our proposal is platform and protocol agnostic, therefore masking existing DM platforms complexity and specificity. This allows integration of diverse objects while extending and interfacing with various DM solutions.

A. Overview

IoT device firmware get released by manufacturers during device's commercialization and support period. Configurations are developed and pushed when required (e.g., new service, security flaw patching etc..) by the system administrator. These are not meant to be systematically applied to every compatible device except for security updates that should be generalized as soon as possible.

Based on device and firmware information (extracted from their datamodels), it is possible to infer whether a device should be updated or not. For instance, when on low battery or poor network conditions, firmware updates should be avoided to prevent a corrupted installation that would render the device definitely out-of-order. Another example is only targeting relevant devices for a minor firmware or configuration update (i.e., fixes a bug when certain features are enabled or used). The main goal is avoiding service interruption and serious dysfunctions risks induced by DM operations when they are not critical. Besides, if faulty DM operations (e.g., bugged firmwares or configurations) are performed on a part of the fleet or all of it, consequences can be (but are not limited to) lower QoS, abnormal behavior and device rendered unable to function again without manual flashing or replacement. To avoid generalization of such firmwares and configurations, we propose active monitoring of DM Operations impact.

To sum up, we aim to integrate part of a system administrator expertise in an autonomic management layer that will enhance DM platform operation. Thus, we identify two key features for a smarter IoT DM:

- Automatic operation launching and target (devices) identification;
- Automatic error tracking on both of devices and infrastructure (i.e., hardware and network hosting the DM platform).

These features are defined to take on three of the main challenges induced by the massive deployment of IoT devices. Indeed, our architecture tackles the **Heterogeneity** of IoT by allowing multiple device types to be handled by the same autonomic managers by abstracting the concept DM Command (i.e., protocol specific elementary) and allow **heterogeneous** device types to be handled by the same autonomic managers. Besides, **Dynamicity** of IoT devices is also addressed by our proposal in two ways: using device state active checking for target identification and operation launching (e.g., computing power at a given time, running services, paired devices, network signal, movement, interference). Anomaly tracking also allow our proposal to tackle **Dynamicity** via active monitoring of defined QoS measures during progressive operation execution on the fleet. Indeed, deployment speed is regulated according to operation metrics (e.g., errors, warning). **Scalability** is managed via our approach thanks to its ability to vertically scale (i.e., use more available computing power to increase execution speed [8]) depending on infrastructure load.

B. Multi-Loop IoT Device Management Architecture

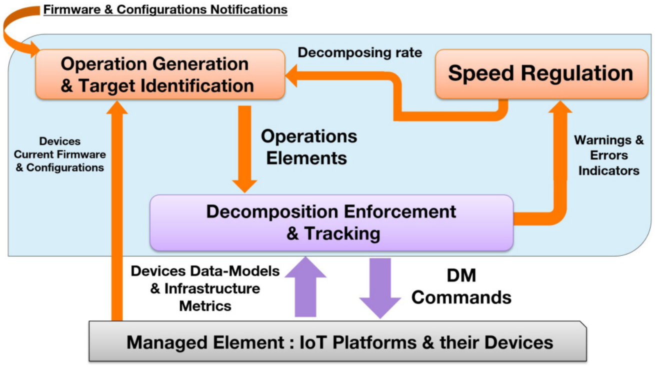

Firmware & Configurations Notifications
Operation Generation & Target Identification
Decomposing rate
Speed Regulation
Devices Current Firmware & Configurations
Operations Elements
Warnings & Errors Indicators
Decomposition Enforcement & Tracking
Devices Data-Models & Infrastructure Metrics
DM Commands
Managed Element : IoT Platforms & their Devices

Figure 1. Global multi-loop architecture for DM

We propose an autonomic control system that will automatically operate some features of DM platforms, namely operation generation, target identification, anomaly detection (e.g., execution errors, infrastructure overload), processing speed variation. It is based on multiple coordinated control loops that react to execution errors and warnings faster than existing human-based approaches.

The architecture is able to tackle IoT challenges (i.e., numerous devices with different context, states and capabilities). Indeed, it automatically triggers DM operations and identifies suitable devices to process (i.e., correct initial firmware, battery and network status). Moreover, it takes into consideration hardware load and network congestion.

The system aims towards one global goal: 'Keeping a device fleet compliant and up to date'. It is composed of three autonomic loops (see Figure 1). First, *Operation Generation & Target Identification* gets devices datamodels via IoT platforms. This allows automatic target identification and operation launching. Second, speed execution variation is operated by the *Speed Regulation* loop that decides, depending on progression, phase, errors count, and infrastructure response time, among other measures, which speed should be forwarded to the *Operation Generation & Target Identification* manager. Finally, the *Decomposition Enforcement and Tracking* manager actually sends commands to devices and collects execution data and logs that are compiled and sent to the *Speed Regulation* loop to compute.

We distinguish two levels of managers. High level ones (colored in orange in Figure 1) that are centralized and concentrate all data for decision-making (*Operation Generation & Target Identification*, *Speed Regulation*). Low level managers (colored in purple in Figure 1) are meant to be instantiated multiple times for horizontal scaling.

1) Managed Element: DM platforms and their managed device fleets, each containing one (or more) device type (e.g., Netatmo Home Weather Stations, Philips Hue Devices) represent the managed element of our proposed autonomic control system.

2) Monitoring: The autonomic management system observes these data:

- Firmware & Configuration Notifications: they contain information (e.g., Type or Criticality, Installation requirements) used for prioritization and target identification;
- Device Hardware and Software states (extracted from their datamodel instances), for error and warning detection;
- Infrastructure Metrics (e.g., DM Servers API's response times) for overloading avoidance;
- Optional: External business information such as amount of hotline calls.

Firmware information is provided by developers. It takes the form of a description file manifest that contains information such as Type (Critical, Major, Minor, Hotfix) or installation requirements (e.g., migration path: v1.1 to v1.2 to v2.0, minimum battery level). This information is used by our autonomic system to target the right subset of devices that should be receiving the DM Operation.

In order to accurately detect QoS variation for warning diagnosis and error mitigation, we defined a collection of commonly available metrics in devices datamodels. These include an average CPU usage, RAM load, and network interface occupation. In addition, for accuracy's sake, we propose a set of device-type related measures. For instance, an IP Camera should show a stable video bitrate (within margins).

Infrastructure congestion mitigation is done via response-time observation. In fact, an increase in that measure implies a size reduction of sent DM operations for execution.

3) Effectors: In order to keep a fleet up to date and well-functioning, devices that were targeted will receive a set of DM commands (i.e., elements composing a DM operation) generated by the autonomic IoT DM system.

In the following subsections, each autonomic manager has its input, output, pace and workflow detailed.

C. Decomposition, Enforcement and Tracking

This autonomic loop aims to push DM commands to IoT platforms while enforcing execution policies (e.g., asserting max parallel operations possible on servers, operation prioritization, retry approaches). It is also responsible for tracking data (e.g., logs, server response-times, probe data) aggregation.

a) Input: Three entries are necessary for this loop to run:

- The first input required for this autonomic loop to operate is a set of 'DM Operation Elements'. They target a part of identified devices (e.g., 20% of eligible devices for a firmware installation). These are computed by Operation Generation loop.
- Ongoing devices datamodels are the second input of this loop, allowing device state assessment during execution (e.g., detect whether a *DM Command* is properly executed or not). These states include current firmware version, QoS measures, hardware state (e.g., battery power, CPU load, free memory).
- The last entry is related to infrastructure metrics (e.g., response time, overload alerts, amount of network or platform errors).

We propose two sets of QoS indicators related to managed devices: commonly available probe measures in different standard protocols datamodels (LWM2M [27], USP [28], CWMP [20]) and a set optional of device-type related indicators.

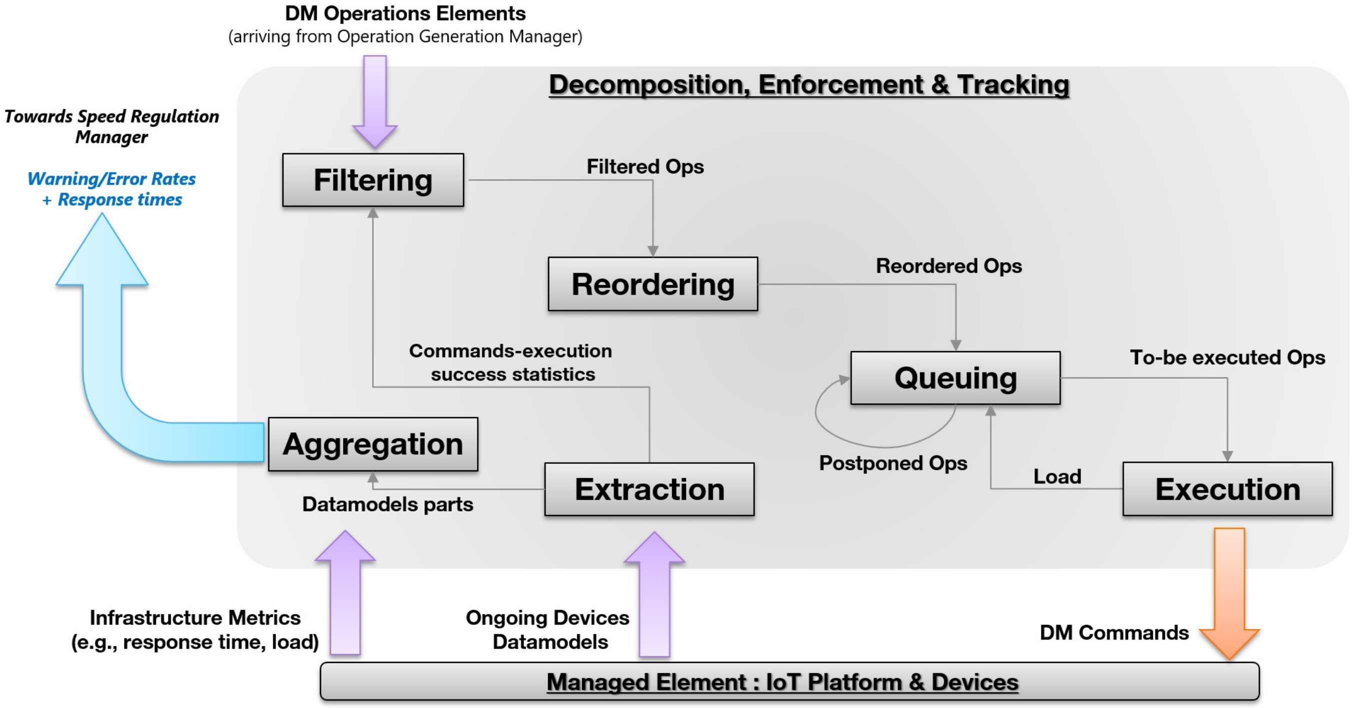

DM Operations Elements
(arriving from Operation Generation Manager)
Decomposition, Enforcement & Tracking
Towards Speed Regulation Manager
Warning/Error Rates + Response times
Filtering
Filtered Ops
Reordering
Reordered Ops
Commands-execution success statistics
Queuing
To-be executed Ops
Aggregation
Postponed Ops
Load
Extraction
Execution
Datamodels parts
Infrastructure Metrics (e.g., response time, load)
Ongoing Devices Datamodels
DM Commands
Managed Element : IoT Platform & Devices

Figure 2. *Decomposition, Enforcement & Tracking* autonomic workflow

First ones are as follows:

- Average CPU usage per time slot (e.g., 6 hours);
- Average RAM usage per time slot;
- Storage utilization;
- Amount of network packets sent, received, errors;
- Network signal strength.

Optional data include (but are not limited to):

- Application QoS measures (e.g., video bitrate, abnormal sensor data);
- CPU usage variation per time slot (e.g., for spike detection).

We aim by analyzing these data for more accurate processing of hard to detect losses of QoS (e.g., slight variations but randomly happening, big spikes on a certain type of environment). These are usually assessed via costly physical interventions from technical services following a customer care call. These trips should only be triggered for mandatory interventions (e.g., battery replacement, hardware failure).

b) Output: Two outputs emanate from this autonomic loop.

- Error and Warning percentages and rates (in blue in the scheme): they indicate if a DM Operation has a negative impact on devices, or if a firmware doesn't install properly on a part of the target. These measures are extracted from devices datamodels and aggregated before being sent to *Speed Regulation* loop. It also integrates QoS measures regarding DM Servers (e.g., infrastructure response-time).
- Protocol specific DM Commands (in orange in the scheme) inferred from ongoing DM Operations Elements. They are sent to devices via one (or more) DM Server APIs.

c) Pace: This autonomic loop keeps running while all awaiting and ongoing DM Operations are not completed or failed.

d) Workflow: Figure 2 provides a global picture of how are the DM Operations Elements processed. Its workflow is composed of the following steps:

- *Filtering*: First module filters out *DM Operations* that does not comply with retry policy. This action avoids loop's overflow due to several unsuccessful operations being queued. This module assesses if on-going *DM Operations* are successful based on *Execution Success Statistics*. For instance, an operation can expire if a certain percentage of its commands fail after several retries. These percentages are extracted from devices datamodel. Operation nature should be taken in consideration. Indeed, a network configuration patch on a fleet may be considered failed if not applied on 100% of the fleet.
- *Reordering*: Afterwards, reordering component proceeds in rearranging operations to be treated regarding their priority. It is either set by the DM Operator for a manual *Operation*, or inferred using loop's knowledge base. We identified the following order:
  1) Critical (e.g., security patch, urgent fix);
  2) Major (e.g., new feature release);
  3) Minor (e.g., non critical bugfix);
  4) Hotfix (e.g., bugfix for rare certain devices in specific environments).

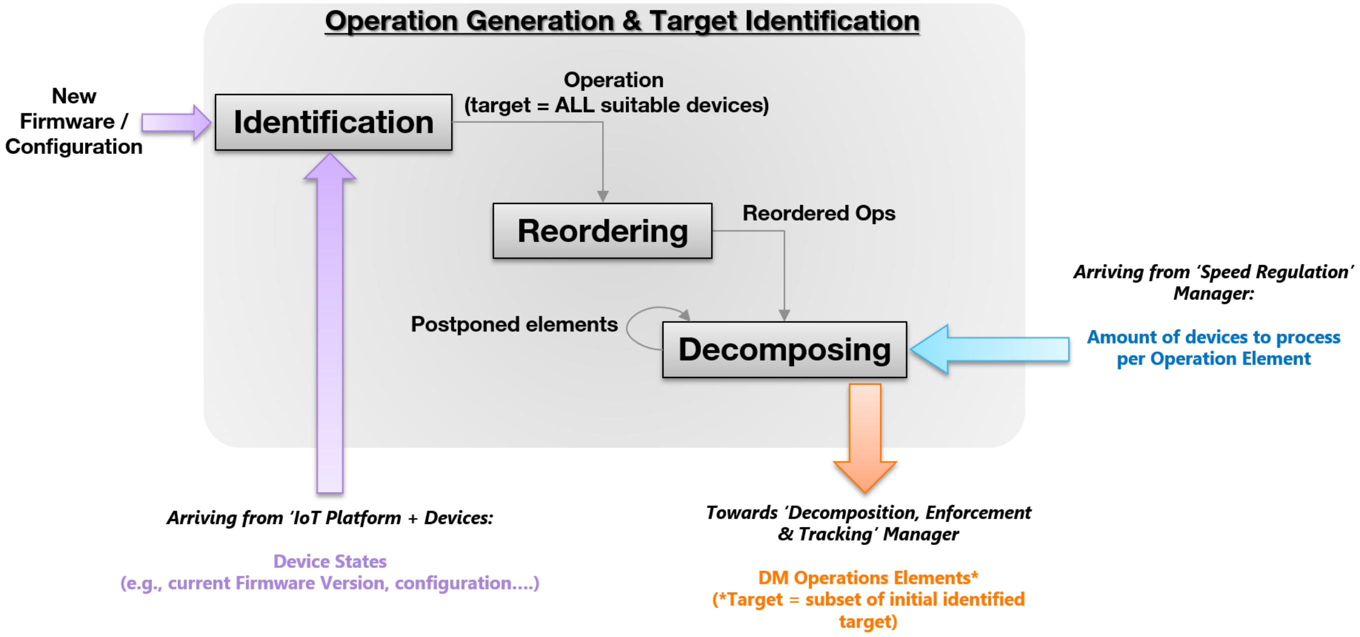

Operation Generation & Target Identification
New Firmware / Configuration
Identification
Operation (target = ALL suitable devices)
Reordering
Reordered Ops
Postponed elements
Decomposing
Arriving from 'Speed Regulation' Manager:
Amount of devices to process per Operation Element
Arriving from 'IoT Platform + Devices:
Device States (e.g., current Firmware Version, configuration....)
Towards 'Decomposition, Enforcement & Tracking' Manager
DM Operations Elements* (*Target = subset of initial identified target)

Figure 3. *Operation Generation & Target Identification* autonomic workflow

This procedure is needed in order to avoid higher priority operations to be systematically executed after on-going massive low priority ones (e.g., a security patch on a small set of devices must be pushed before applying a minor hotfix to a complete fleet of sensors).

- *Queuing*: Next module concerns operations Queuing. It sets some operation for later execution depending on node's hardware capabilities. It computes how much parallel operations can be executed and tracked. This mechanism avoids hardware overloading.
- *Execution*: Once to-be-executed operations are identified, they are translated to protocol specific *DM Commands* that are sent by a DM Server to the targeted devices (i.e., Managed Element). For a given autonomic loop, Device Type and DM Protocol are identified (e.g., Indoor Geolocalization Station, LWM2M).
- *Aggregation*: In order to track proper execution of a given *DM Command*, the loop pulls execution-data from devices then aggregates warning and error percentages. This data is used for two purposes. First, retry policy relies on it to discontinue operations when considered failed. Second, compiled execution data is pushed to *Speed Regulation* Autonomic loop allowing it to regulate DM Operations progression. Errors represent devices that do not respond after updates, while warnings incorporate DM Server response time variation and abnormal device behavior (e.g., wrong values, high memory usage, frequent registration rate).
- *Extraction*: This entity consists of querying devices datamodels and extracting required data from it, current firmware version and impacted features QoS measures for a firmware update operation. For a given device, a datamodel defines its state, hardware capabilities, and software environment.

D. Operation Generation & Target Identification

This autonomic loop is responsible for DM Operation generation (based on the managed fleet state) and decomposition in *DM Operation Elements* that will be forwarded to the *Enforcement, Decomposition & Tracking* loop. Indeed, it is in charge of target identification (i.e., defining the currently suitable devices that need to receive a given firmware or configuration) while applying computed ongoing processing speed (via decomposed operations size) based on the decision of the Speed Regulation loop.

a) Input: Two events can trigger its activation.

- New firmware or configuration notifications and their description file. This information allows target identification and priority inference.
- Manually sent DM operations (e.g., Update Compliant Weather stations to Firmware 3.1, Enable motion detection on all Proximity Sensors). These are basically notifications in which target is manually defined by DM Operator (or Administrator).

Besides, another input is required for this autonomic loop to fulfill its role. Indeed, the computed amount of devices by *Speed Regulation* loop is injected for operation speeding up or down.

b) Output: There is a single output for this autonomic loop. It consists of DM Operation that aim suitable devices from the fleet (e.g., Devices with correct network conditions).

c) Pace: This autonomic loop keeps monitoring the device fleet therefore detecting new devices that are not compliant (new or out-of-date device). If new DM operations are to be executed or ongoing, it keeps passing through all its steps until done.

d) Workflow: Figure 3 details this manager's workflow. It is composed of the following steps:

- *Identification*: Once an input received by the loop, this module triggers target identification based on included information in firmware description. For instance, a given system update may only be applied to devices in the right current version. Another example is minimum battery requirement for patching.
- *Reordering*: Next module is in charge of business SLA application through reordering. Indeed, in an open DM platform (e.g., DM as a Service), contracts with 3rd parties may induce variable SLA agreements. It is different from Decomposition, Enforcement & Tracking Loop reordering module. This one aims to do SLA-based high level prioritization of pending operations, while the lower entity reorders parts of ongoing ones.
- *Decomposing*: Last task treats how fast a *DM Operation* progresses depending on warning and errors by increasing or decreasing its batch size (i.e. amount of device processed each iteration). This amount is based on *Speed Regulation* loop that computes how much devices should to be treated according to execution anomalies rates and IoT platform infrastructure load.

E. Speed Regulation

This loop aims to decide whether an ongoing DM Operation should be accelerated, slowed, halted, or stopped. It takes its decision based on warnings, error rates on devices received by *Decomposition, Enforcement, & Tracking* loop, while also taking into account infrastructure load metrics sent by the managed IoT Platform.

DM Operations are characterized by their 'State' and 'Phase'.

- State: Pending (Not Started), Ongoing (being processed), Aborted (Stopped due to high error rates), Finished (successfully executed)
- Phase: We define three possibilities: Test (beginning of an operation). Cautious (next step, with more devices yet moderate speed). Generalization (fast phase where the goal is to process as many devices as possible). Phases are determined by processed device percentage.

We propose a rather simple algorithm making that choice based on the current phase of an operation. Depending on in which phase it is, the changes of speed will be more or less significant. Thus, error and warning tolerance are lower in *Test* phase. This allows more accurate decision-making for speed regulation: the earlier errors are detected, the more drastic is the regulation.

a) Input: Input here is error and warning rates arriving from *Decomposition, Enforcement and Tracking* autonomic loop and infrastructure metrics. Both are used for decomposition rate computation.

b) Output: This loop outputs the right number of suitable devices that should be processed in *Operation Generation & Target Identification* loop based on error and warning rates.

c) Pace: Each execution cycle of this loop is triggered by an operation batch finishing in Decomposition Enforcement & Tracking loop.

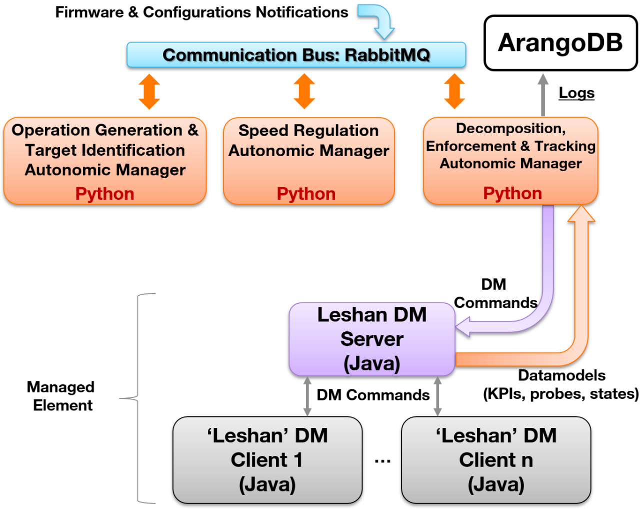

Firmware & Configurations Notifications
Communication Bus: RabbitMQ
ArangoDB
Logs
Operation Generation & Target Identification Autonomic Manager
Python
Speed Regulation Autonomic Manager
Python
Decomposition, Enforcement & Tracking Autonomic Manager
Python
DM Commands
Leshan DM Server (Java)
Datamodels (KPIs, probes, states)
DM Commands
Managed Element
'Leshan' DM Client 1 (Java)
...
'Leshan' DM Client n (Java)

Figure 4. Technical Component Architecture

d) Algorithm: This algorithm works as follows for each operation:

1) If pending: Start Operation
2) If errors > tolerated error rates, Abort
3) Update Phase (depending on progression)
4) Regulate: According to metrics variation and phase : Speedup or Slowdown

Multiple regulation strategies are possible. Depending on risk tolerance, variation can be Linear, Power, Polynomial, Exponential.

IV. Proof Of Concept & Experimental Validation

In this section we detail our experimental setup in order to evaluate our approach capability to regulate speed automatically. First, we provide details regarding the technical architecture of our setup. Afterwards, we present our environment before describing autonomic loop implementation.

A. Implementation & Experimental Results

For this experimental setup, we used Eclipse Foundation open-source DM Client and Server (i.e., Leshan Project) [29]. Each autonomic loop is implemented in the form of a Python script. Inter-loop communication is done via a messaging queue: RabbitMQ. Figure 4 details components interactions. Currently, all software modules run on a single physical server therefore no network latency is present during inter-process communication. During execution, logs are being pushed to a database for further analysis and interpretation. ArangoDB serves as a database for our experimentation, it hosts logs and past execution traces.

Our internal survey of Orange DM approach leads us to model it in three phases as follows. Initial phase aiming few devices (less than 0.01%) of the fleet lasting 48 hours of manual metrics observation. Second phase targets few percents of our fleet (i.e., 3%). It lasts 10 days of execution and surveillance time. Last step is a generalization phase that installs the update in all available devices at a rate of 6.25% of the fleet per 24 hour slot (rate defined by experts as a reasonable speed that allows manual monitoring to be done

TABLE II. Comparison Table: Existing vs Autonomic Approach

	Existing Orange Home DM	Industrial IoT Platforms	Autonomic Enhanced Platform
Operation Launching	Manual	Manual	Automatic
Target Identification	Manual	Manual	Automatic
Protocol Support	Single	Multiple	Multiple (Platform Dependent)
Execution Speed	Static	Static	Dynamic
Error Awareness	✗	Partial	✓
Vertical Scaling	✗	Up	Up-Down

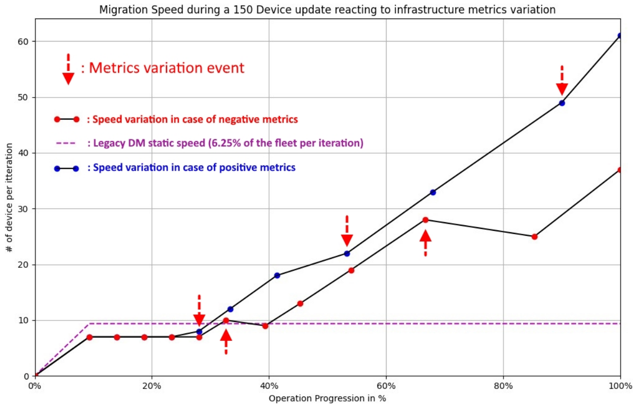


Figure 5. Autonomic DM: Reaction to DM Platform metric perturbations

and avoids call-center congestion). For comparison, we take the latter (Fastest rate) as reference regarding existing home DM solution execution speed.

B. Test Protocol

In order to test our system, we launch a predefined (150 for our tests) amount of DM Client that simulates our device fleet. Afterwards, we trigger a new firmware availability notification via the messaging queue. This leads Operation Generation & Target Identification loop to compute which device of the fleet will receive this firmware and starts the DM operation.

During each DM Operation Element execution its size is monitored and plotted (i.e., how much devices are to be processed). This allows to observe what decision how the autonomic system regulates the element size in normal iterations and perturbed ones. We introduce 2 types of perturbations (red arrows in Figure 5), positive and negative events.

- Positive variation of infrastructure response times and execution metrics.
- Negative variation of the aforementioned indicators.

If metrics improve compared to last iteration, accelerate update deployment by rising *Batch Size* (i.e., amount of devices processed in each iteration). Otherwise, it will either slow down (increasing amount of anomalies), stabilize (if little or no variation) or abort (if too many errors are detected).

For reference, Orange internal Home DM migration strategy's peak speed is represented in purple in Figure 5 (6.25% of the fleet per iteration).

In Figure 5 the number of devices per iteration is plotted as a function of overall progression. It details how our autonomic loops based approach reacts to metric variations.

Scenario 1 (Negative variation of metrics) colored in red in Figure 5 shows how autonomic management reacts to response time increase (red arrows in plot). Indeed, it slightly lowers execution speed for one iteration (as seen approx. at 35% 65%). Once infrastructure scaled up (therefore response time back to a lower value), the system increases batch size again in next iteration.

Scenario 2 (Positive variation of metrics) colored in blue in Figure 5 shows how our system manages improving metrics coming from DM Infrastructure at 27%, 55% and 90%. Slopes between these points and next ones is higher (therefore speed variation too). In fact, autonomic Management interpret lower response time (and error count) as a sign of resources availability at the IoT Platform level.

C. Discussion

Table II compares our approach with existing Orange Home DM Platform and industrial IoT platforms, DM protocol support, execution speed adaptation, reaction to anomalies and vertical scaling.

While Orange Home DM and Industrial IoT rely on system administrators for operation launching and target identification, autonomic approach monitors new firmware or configuration availability by itself and also triggers update operations when a fleet is not up-to-date anymore (e.g., new device arrival). Both of current Home and IoT DM platforms respectively support single or few DM standard protocols. Autonomic Enhanced approach relies on underlying platform for connectivity. Two key features of autonomic management are speed regulation based on error & infrastructure metrics and error awareness. These allow more accurate platform operation and avoid faulty configuration and firmware to be generalized creating service interruption. Finally, while Home DM do not offer vertical scaling [8], since industrial IoT platform are designed to run on Cloud Infrastructure such as Microsoft and Amazon ones, they allow up and down scaling. However, since they rely on the infrastructure beneath them to scale, if not hosted in an unrestricted elastic cloud environment or in case of limited resources available, no scaling is possible. Autonomic Enhanced DM enables full vertical scaling in the DM system directly by regulating execution speed.

Our tests are realized in a local server with some great compute capabilities. Yet for experimentation accuracy, real life testing on IoT platforms such as Azure, AWS or IBM could make experimental results richer. This requires to have access to source code and instances of them in order to modify their behavior and integrate an autonomic management layer on top. While Vertical Scalability is addressed in this paper, horizontal scalability is currently in investigation. It is required for massive (Millions, Billions) amount of IoT DM. Last, our proposal protocol compatibility is currently enabled via platform protocol support. Our design was thought to be protocol agnostic by design. With middleware acting as a proxy translating ourgoing orders from autonomic management to protocol specific commands this issue should be resolved.

V. Conclusion & Perspectives

In this paper, we address three of identified IoT DM challenges. **Heterogeneity**, **dynamicity** and **scalability** of devices makes existing Home, MDM and workstation DM solutions used at Orange and their industrial IoT counterparts unsuitable for IoT DM. This is due to their manual operation, static execution speed and lack of impact detection (e.g., device errors, infrastructure overload).

We propose a novel autonomic approach for IoT DM. It relies on:

- A Coordinated multi-loop architecture for IoT DM
- An **Operation Generation & Target Identification** loop that automatically targets suitable devices.
- A **Decomposition, Enforcement and Tracking** loop that executes DM operations and monitors devices and infrastructure.
- A **Speed Regulation** loop that regulates **Decomposition, Enforcement and Tracking** speed according to anomalies and infrastructure load.
- A Proof of Concept for experimental validation.

In terms of perspectives, we aim to validate autonomic target identification by interfacing our autonomic IoT DM managers to Orange Live Objects [23] cloud platform. We have ongoing work regarding IoT challenges that have not been addressed in this paper (i.e., Interoperability and Scalability). Thus, we are investigating our proposal's scaling capability through distribution at the edge of the network, that is a key requirement for massive IoT DM. This will allow numerous IoT devices management via **horizontal scalability** in addition to vertical scalability detailed in this paper [8]. We are also exploring several millions of devices simulation on Grid5000 infrastructure [30].

References

[1] "11 best practices for mobile device management (mdm), ibm security thought leadership white paper." [Online]. Available: https://www.ibm.com/downloads/cas/VENWY8OG

[2] "Broadband forum." [Online]. Available: https://www.broadband-forum.org/

[3] N. Ayeb, E. Rutten, S. Bolle, T. Coupaye, and M. Douet, "Towards an autonomic and distributed device management for the internet of things," in 2019 IEEE 4th International Workshops on Foundations and Applications of Self* Systems (FAS*W), Jun 2019, p. 246–248.

[4] "Google home firmware update is bricking some units." [Online]. Available: https://9to5google.com/2019/10/24/google-home-firmware-brick/

[5] "Rbc analyst says 52 million google home devices sold." [Online]. Available: https://voicebot.ai/2018/12/24/rbc-analyst-says-52-million-google-home-devices-sold-to-date/

[6] "Hpe support." [Online]. Available: https://support.hpe.com/hpesc/public/docDisplay?docLocale=en_US&docId=a00097382en_us

[7] "Dell emc support." [Online]. Available: https://www.dell.com/support/home/fr/fr/frbsdt1/drivers/driversdetails?driverid=8h6hj&oscode=w12r2

[8] M. Michael, J. E. Moreira, D. Shiloach, and R. W. Wisniewski, "Scale-up x scale-out: A case study using nutch/lucene," in 2007 IEEE International Parallel and Distributed Processing Symposium, 2007, pp. 1–8.

[9] N. Gligorić, S. Krčo, D. Drajić, S. Jokić, and B. Jakovljević, "M2m device management in lte networks," in 2011 19th Telecommunications Forum (TELFOR) Proceedings of Papers, Nov 2011, p. 414–417.

[10] A. A. Corici, R. Shrestha, G. Carella, A. Elmangoush, R. Steinke, and T. Magedanz, "A solution for provisioning reliable m2m infrastructures using sdn and device management," in 2015 3rd International Conference on Information and Communication Technology (ICoICT), May 2015, p. 81–86.

[11] I. Danila, R. Dobrescu, D. Popescu, R. Marcu, and L. Ichim, "M2m service platforms and device management," in 2015 9th International Symposium on Advanced Topics in Electrical Engineering (ATEE), May 2015, p. 67–72.

[12] M. N. Islam and S. Kundu, "Remote configuration of integrated circuit features and firmware management via smart contract," in 2019 IEEE International Conference on Blockchain (Blockchain), Jul 2019, p. 325–331.

[13] H. Gupta and P. C. van Oorschot, "Onboarding and software update architecture for iot devices," in 2019 17th International Conference on Privacy, Security and Trust (PST), Aug 2019, p. 1–11.

[14] S. Choi and J.-H. Lee, "Blockchain-based distributed firmware update architecture for iot devices," IEEE Access, vol. 8, 2020, p. 37518–37525.

[15] S. Rao, D. Chendanda, C. Deshpande, and V. Lakkundi, "Implementing lwm2m in constrained iot devices," in 2015 IEEE Conference on Wireless Sensors (ICWiSe), Aug 2015, p. 52–57.

[16] M. Ha and T. Lindh, "Enabling dynamic and lightweight management of distributed bluetooth low energy devices," in 2018 International Conference on Computing, Networking and Communications (ICNC), Mar 2018, p. 620–624.

[17] A. Karaagac, M. VanEeghem, J. Rossev, B. Moons, E. DePoorter, and J. Hoebeke, "Extensions to lwm2m for intermittent connectivity and improved efficiency," in 2018 IEEE Conference on Standards for Communications and Networking (CSCN), Oct 2018, p. 1–6.

[18] "Google kills android distribution numbers on the web, but we've got you covered." [Online]. Available: https://9to5google.com/2020/04/10/google-kills-android-distribution-numbers-web/

[19] "New android 11 devices will support virtual a/b for seamless updates." [Online]. Available: https://www.xda-developers.com/google-virtual-ab-seamless-updates-android-11/

[20] "Broadband forum cpe wan management protocol (cwmp) data models." [Online]. Available: https://cwmp-data-models.broadband-forum.org//

[21] "Genieacs." [Online]. Available: https://github.com/genieacs/genieacs

[22] "Freeacs." [Online]. Available: https://github.com/freeacs/freeacs

[23] "Orange live objects." [Online]. Available: https://liveobjects.orange-business.com/#/liveobjects

[24] "Amazon aws iot device management." [Online]. Available: https://aws.amazon.com/iot-device-management

[25] "Azure iot hub." [Online]. Available: https://azure.microsoft.com/en-us/services/iot-hub/

[26] "Software updates over the air - bosch iot suite." [Online]. Available: https://www.bosch-iot-suite.com/software-updates-over-the-air/

[27] "Oma specifications." [Online]. Available: http://openmobilealliance.org/wp/index.html

[28] "User services platform (usp)." [Online]. Available: https://usp-data-models.broadband-forum.org/

[29] "Eclipse leshan." [Online]. Available: https://github.com/eclipse/leshan

[30] "Grid'5000: A large-scale and flexible testbed for experiment-driven research in all areas of computer science." [Online]. Available: https://www.grid5000.fr/w/Grid5000:Home

DOI: 10.15439/2020F99
ISSN 2300-5963 ACSIS, Vol. 21

A LoRa Mesh Network Asset Tracking Prototype

Emil Andersen, Thomas Blaalid, Hans Engstad, Sigve Røkenes
Norwegian University of Science and Technology (NTNU)
Trondheim, Norway

Frank T. Johnsen
Norwegian Defence Research Establishment (FFI)
Kjeller, Norway

***Abstract*—Long Range (LoRa) is a low powered wide area communications technology, which uses radio frequencies in the industrial, scientific and medical (ISM) band to transmit data over long distances. Due to these properties, i.e., the long range and little restrictions on deployment and use, LoRa is a good candidate for building an asset tracking application on, for example targeting search and rescue operations. This paper describes the development and testing of such a prototype, using commercial off-the-shelf Internet of Things (IoT) consumer devices and a proprietary mesh protocol.**

The prototype enables distributed position tracking utilizing the Global Positioning System (GPS), a gateway to the Internet, a server for data storage and analysis, as well as a Web application for visualizing position tracking data. The devices are small, and our tests have included both personnel on foot carrying the equipment, as well as having the devices on vehicles.

***Index Terms*—Internet of Things, Wireless mesh networks, Web services**

I. Introduction

EXISTING technologies such as cellular networks offer rapid communication across fair distances, but are limited in their scope of operation. Large cellular towers have limited range and rely on extensive infrastructure to provide service, and consequently it is expensive to maintain and expand this type of network. Due to their isolated nature they are also susceptible to a single point of failure [1], [2].

A mesh network circumvents many of these limitations through its distributed design. Perhaps most crucially, a mesh communications network is not reliant on a central station or site. A distributed solution can provide connectivity in near any location, even those without any existing infrastructure. It will also be more resilient to network failure than a centralized solution, because it has no single point of failure. If a network node is destroyed or otherwise left inoperative, remaining units in the network are adaptable and will continue to provide service. Finally, the hardware required to deploy a Long Range (LoRa) mesh network is inexpensive and accessible [3]. This enables swift propagation of a large number of devices, which is essential to establish a robust distributed network. The tradeoff for the advantages of long range and low power use is primarily the low rate of data transfer that LoRa offers [4].

The properties of mesh networks make them particularly suitable for certain applications where traditional communications infrastructure is impractical or insufficient. Examples of application areas include asset tracking, sensor data dissemination, and low bandwidth communication in case of exhaustive infrastructure failure. Due to the low cost and accessibility of the necessary hardware, these, as well as many other applications, are both feasible and pragmatic using affordable commercial off-the-shelf (COTS) devices [5].

A notable example of this is search and rescue operations in areas with challenging geography, such as Norwegian mountain ranges. These areas may not have reliable access to cellular networks, which leaves crucial primary communication channels unavailable. In order to perform a successful search, for instance when looking for a missing hiker in the mountains, it is important to know which area the rescue personnel is currently in (current position) as well as which areas have been searched already (history of positions). An application for location tracking using a LoRa mesh network could provide this, while simultaneously producing a detailed map of all regions of the area searched so far. In this paper, we pursue a prototype implementation of such a search and rescue system, built on COTS Internet of Things (IoT) consumer devices and a proprietary mesh protocol.

The remainder of the paper is organized as follows: Section II outlines the scope of our work, whereas Section III gives an overview of the technologies involved in the prototyping effort. Section IV discusses the design and implementation of our software. The tests we performed using our prototype are summarized in Section V. Section VI presents the analysis of our findings, leading up to a summary of results in Section VII. Section VIII presents related work. Finally, Section IX concludes the paper.

II. Prototype scope

The purpose of our work was to develop and test an initial prototype for search and rescue operations, with the main focus being a distributed network for geographical tracking. To achieve this, we limited our scope to COTS IoT products to build an affordable, highly portable and easily deployable system. Hence, we chose to focus on LoRa, since the protocol offers long range communications while at the same time being battery efficient. LoRa is also one of the few choices out there that you can deploy with few limitations, as it is not reliant on commercial infrastructure, as opposed to e.g., NB-IoT and Sigfox [6]. Due to this, our prototype system uses the LoRa protocol to send messages between devices in a mesh network. The functionality scope is limited to each device reporting its position using its onboard Global Positioning System (GPS).

The mesh network itself is self-healing and not reliant on any infrastructure. Internet is not needed, granted that the

Note that the NTNU authors were equally involved in the work, and hence are listed in alphabetical order by surname. The work was sponsored by FFI project 1431.

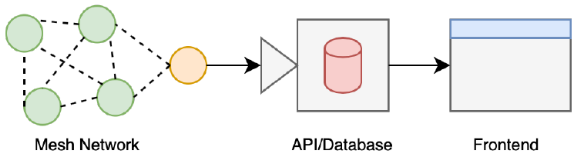


Fig. 1. Prototype system high-level architecture

back-end is available locally. However, for the prototype we deployed the back-end on the Internet, so here one of our devices needs Internet connectivity. These devices that connect to the Internet are referred to as *border routers*. Border routers should receive coordinate data from the whole mesh network, and forward it to a database web server application programming interface (API). The web server is responsible for storing the data persistently in a database. Finally, the location data should be displayed in a frontend web application. In addition to the location data, the application should show the nodes' roles within the network (for debugging), historic positions of nodes (can be toggled on or off, an important feature of the search and rescue application), as well as other relevant metadata collected by the system (used for prototype and protocol evaluation purposes).

Figure 1 illustrates the system architecture. Here, green circles are routers in the mesh network. The orange circle has the special role of the border router, and is responsible for forwarding data from the mesh network to the web server hosting the API and database. Note that even though the figure only shows one border router, there may be multiple for redundancy. In our implementation, a router that is in range of a pre-defined WiFi network SSID and manages to connect to it, automatically becomes a border router. The system needs at least one border router to function as expected, since otherwise the data flow to the database will be disrupted. The mesh network operates on the COTS IoT products over LoRa, whereas the remainder of the system is deployed on the Internet using a Web service API.

III. Technology background

The COTS products we chose were the Lopy4 [7] and Pytrack [8] from Pycom LTD., since these devices provide an inexpensive IoT prototyping platform for developing with Python. Also, the devices have GPS support as well as several protocols, including LoRa, WiFi, Bluetooth, and Sigfox. Below we present some key properties of the LoRa protocol, followed by a brief introduction to PyMesh [9], which is the mesh network implementation we used in our prototype.

A. LoRa

LoRa is a proprietary physical layer specification that enables a chip with an inexpensive crystal to have high sensitivity, and provides long range wireless transmission with low data rate and low energy usage. The industrial, scientific, and medical (ISM) frequency band that LoRa devices operate on is 915 MHz in the US and 868 MHz in Europe [10].

TABLE I
Relevant EU 868 frequency band data rates

Data rate	SF/Chirp rate	Indicative physical bit/s	Payload size
DR5	SF7/125kHz	5470	242 bytes
DR6	SF7/250kHz	11000	242 bytes

LoRa devices achieve long range due to the transceiver's ability to filter on the constant chirp signals, which enables the device to detect and lock to the LoRa signal. A chirp signal is a rapid increase or decrease in radio frequency over time. The LoRa protocol uses variations of these chirps to establish a connection and encode transmitted data [4].

1) Data rate: The data rate is a direct result of the chirp rate used for transmission. A higher chirp rate enables LoRa to encode more data in the same amount of time. A key advantage of LoRa is its ability to demodulate multiple simultaneous signals at the same frequency if the LoRa devices use different data rates [11]. This increases the capacity of a single LoRa device and enables them to communicate with a large number of devices simultaneously if necessary, as long as the adaptive data rate functionality is enabled. This could however be problematic for continuously moving devices, because higher data rates reduce their range and could prevent their signal from reaching a neighboring device in the network.

2) Spreading factor: Spreading factor (SF) is a parameter in the LoRa protocol that directly affects battery usage, range and how often a device can transmit a message. SF adjusts the number of chirps (the data carrier in the signal) that are sent per second. A lower SF indicates that more chirps are sent per second, whereas a higher SF implies a lower chirp rate. Sending data of the same length with a high SF will create a longer transmission time (known as airtime). More airtime forces the modem to run for a longer duration, and therefore consume more energy. SF is graded on a roughly exponential scale between 7 and 12, where each step is equivalent to doubling the airtime for each unit data and an approximate increase of 2.5 dB in signal strength. A SF of 12 would have the greatest range because the receiving device has more opportunities to sample the signal. Another consideration is that longer airtime result in fewer opportunities to send data (since each message takes longer to transmit). LoRa supports several different data rates (DR). A payload of 11 bytes using a DR0 configuration would only be able to send data roughly once every 2 minutes following appropriate government regulations.

For the sake of our prototype, we experimented with two different data rates: DR6 (which is the default for PyMesh) as well as DR5. Details of these DR are shown in Table I. Lower DR than 5 have decreasing payload sizes, that possibly could affect the performance of the mesh network. Our own prototype data format only has a payload of 53 bytes, and should theoretically (not tested) be usable all the way down to DR3, which offers a payload of 53 bytes.

B. PyMesh

PyMesh is a LoRa based mesh network technology consisting of a firmware and library that we obtained from Pycom

TABLE II
DEVICE ROLES IN THE PYMESH NETWORK

Role	Explanation	Color
Router	Most devices in the network will usually be routers. Routers are devices with neighbors, that are capable of forwarding data towards a border router.	Green
Leader	The leader device is responsible for distributing addresses within the network and making other devices aware of where the nearest border router is located. There is always a single Leader in each network partition, which is dynamically self-elected.	Purple
Child	The child role is given to devices located on the edge of the network graph. These devices are not located in a path to neighboring nodes and will therefore never forward data from other devices.	White
Border Router	The border router role is assigned to devices with an Internet connection. There can be multiple border routers in the mesh network. Ultimately, these devices are responsible for forwarding data from the mesh to a web server through the Internet. A border router may simultaneously act as router or leader as well, depending on the network.	Orange

under a time-limited developer's license. Their technology is implemented using OpenThread [12] which is an open source implementation of the IPv6-based networking protocol Thread [13]. PyMesh was developed by Pycom to enable LoRa MAC addresses to be used over IPv6, and therefore enable an OpenThread mesh network to operate over LoRa. PyMesh removes the need for static gateways, which decentralizes the network's infrastructure and makes it more flexible.

PyMesh will automatically assign a Pycom device to one of four different network roles — leader, border router, router or child (see Table II). The role of a device changes continuously based on several factors, the most important being the radio-link strength between devices (Received Signal Strength Indication (RSSI)). The assignment of roles creates a link local address for every PyMesh device directly connected to another device, as well as a mesh local address for every device in the same network. This enables all LoRa messages to be routed efficiently through the mesh network to a device capable of forwarding it out of the network (border routers), as well as updating mesh information on other devices.

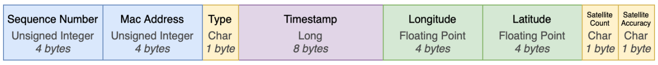
IV. Design and implementation

Our prototype uses a client-server architecture. The system has two clients, the PyMesh border router and the frontend web application, and a web server providing the API endpoints and persistent storage. The server interfaces with the clients in different ways: The PyMesh border router sends data from its devices to the server, while the frontend application requests data from the server. Both methods use a Representational State Transfer (REST) API over HTTP. Below we outline the central parts of the prototype. It should be noted that even if the Pycom devices support multiple network protocols, we are only using the LoRa and WiFi capabilities in our prototype. For the complete description of software development methodology and a more detailed system architecture, see [14].

A. PyMesh

There are eight components in the software we built on PyMesh:

1) **Unit**: The Unit component is the most important component, since it is responsible for managing and calling the other components. The methods in Unit are called from the special main.py file, which is automatically executed when a device is powered on. The component is responsible for ensuring the device is checking for an Internet connection at regular time intervals, setting up the correct role for the device, collecting and making sure data is packed correctly, and finally forwarding it to either another device if it is not a border router, or to the server API if it is.
2) **WiFi**: The WiFi component is responsible for trying to connect the device to a specified WPA2 secured WiFi network if it can find a connection. It has a method for returning if the device has a connection, that Unit uses to determine if the device should act as a border router or not. WiFi is the only protocol we support in the prototype at the moment for bridging a border router to the Internet.
3) **Setup**: The Setup component handles the configuration of the device depending on whether it is a regular router or a border router. It is also responsible for initializing the PyMesh configuration on the device making it a part of the network.
4) **Callback**: The Callback component handles message forwarding depending on whether it is a regular router or a border router when receiving a message. If it is a regular router the package will be forwarded throughout the network until it reaches a border router. When data reaches a border router the Callback component will trigger a method in Unit in order to send the package out of the PyMesh network to the server.
5) **DataPacker**: The DataPacker component is responsible for retrieving the data from the GPS component and packaging the data into a format which is suitable for sending through both LoRa and HTTP to the API. Most of the data is provided by the GPS component, but the DataPacker is also responsible for fetching the MAC address and the node type (role) of the device. The component translates the data between the static byte packet format we have defined (see Figure 2) and standard Python dictionaries in order to enable this.



Sequence Number	Mac Address	Type	Timestamp	Longitude	Latitude	Satellite Count	Satellite Accuracy
Unsigned Integer	Unsigned Integer	Char	Long	Floating Point	Floating Point	Char	Char
4 bytes	*4 bytes*	*1 byte*	*8 bytes*	*4 bytes*	*4 bytes*	*1 byte*	*1 byte*

Fig. 2. Our packet format for LoRa GPS transmissions

6) **GPS**: The GPS component handles everything related to the Pytrack GPS. When Pytrack has a GPS fix, this component returns the necessary data to the DataPacker component. The data consists of positional information like longitude and latitude, as well as satellite accuracy and the number of satellites available. It also synchronizes the GPS time with the Real Time Clock (RTC) on the device for timestamping the data packets. If the GPS component cannot find a fix, it will continuously attempt to establish a connection as long as the device is powered on.
7) **Light-emitting diode (LED)**: The LED component is a helper component which is most useful in debugging and showing the user what process is happening. It is useful for observing what role the device is assigned to or when the device is trying to establish an Internet connection. Figure 3 shows the meaning of the LED colors. The device cycle starts with the LED blinking yellow (WiFi color code) if it is not connected, or the LED being on constantly if it has a connection and is acting as a border router. The cycle ends with a constant color representing the role a device has in the PyMesh network at that time.

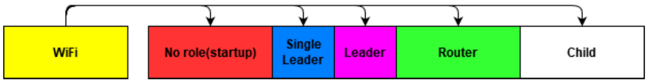


Fig. 3. LED colors assigned to roles and when WiFi is connecting/connected

8) **Node Type**: The Node Type component contains constants with the different roles in the PyMesh network and corresponding LED colors in Figure 3. It is responsible for encoding PyMesh role definitions in the data format used by the DataPacker and API (i.e., the *Type* field shown in Figure 2).

B. Backend

We designed the backend architecture to be light weight and intuitive. There are three entities in the data model, Coordinate, Node and GaiaCoordinate. The GaiaCoordinate entity is implemented on the server for testing purposes, to hold data from the Gaia app [15] we used for GPS comparison, as further discussed in Section V. Each Pycom device will have a unique MAC address, which will be linked to the Node data-model. The Coordinate entity will store each coordinate that it receives from the mesh-network, and link this coordinate to the corresponding Node. The GaiaCoordinate entity will also link its coordinates to a node to be able to visualize a reference in the frontend after testing. The backend consists of two Django-apps [16], which are called “core” and “api”. The core-app contains what can be considered the business-logic and is closer to the database. The api-app defines a REST API which both the frontend and Pycom devices can use.

TABLE III
PYCOM DEVICE FIRMWARE, SMARTPHONES, OPERATING SYSTEMS AND APP VERSIONS

Pytrack firmware	pytrack0.0.8.dfu
LoPy4 firmware	LoPy4-1.20.2.r1
Pycom devices	Pycom A (Mac 4), Pycom B (Mac 5), Pycom C (Mac 3), Pycom D (Mac 7), Pycom E (Mac 6)
Samsung Galaxy J3 SM-J330F	One UI version 1.1, Android version 9, WPA2 WiFi hotspot
Huawei P20 EML-L29	EMUI version 9.1.0, Android version 9, Gaia GPS version 2020.3
Samsung Galaxy S20+ SM-G986B/DS	One UI version 2.1, Android version 10, Gaia GPS version 2020.3
Server	Ubuntu 18.04.3 LTS, GNULinux 4.15.0-76-generic x86 64

C. Frontend

The frontend was written in React [17], and uses a component hook based architecture. Its structure can be divided roughly into two parts, the map for geographic visualization of node positions, and the supplementary user interface for orienting within the application as well as accessing more advanced functions. API hooks are responsible for providing the necessary data for components, which it fetches from the server-side REST API. In order to improve performance and reduce complexity, request parameters such as filtering and search are offloaded to the server. Responses are provided in a Java Script Object Notation (JSON) format, specifically GeoJSON [18] for coordinate data, which, with minimal processing can be visualized in our frontend.

V. TESTS

We wanted to test PyMesh functionality in practice, to see how it would perform as the data carrier for our search and rescue application. The hardware and software we used is summarized in Table III. The goal of our system testing was to gather data to investigate certain metrics:

- Average packet loss and packet loss by range for DR5 and DR6
- Average range for DR5 and DR6
- Average GPS accuracy in meters
- GPS accuracy over time in meters

To have an additional source of GPS tracks, we chose to use the Gaia GPS app [15], which includes functionality to export recorded tracks as easily interpretable Comma Separated Values (CSV) files. This made it easy to perform analysis on gathered data from our system and import Gaia tracks directly into our database and web page for comparison.

Before a test, each Pycom device participating is connected to a power source. When booting they have to be minimum two meters apart and not be started simultaneously, in order to prevent a PyMesh connectivity issue (this occurs due to LoRa transceiver saturation and subsequent mesh initialization

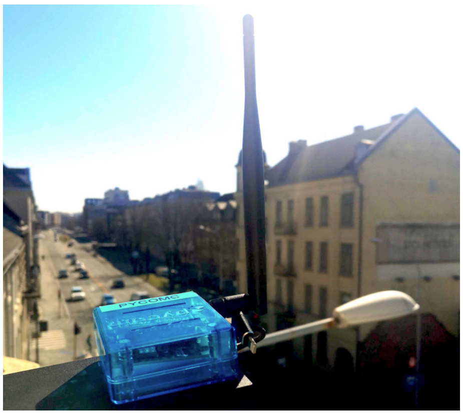


Fig. 4. Photo of our Pycom C node.

failure). A phone with 3G/4G is used as a WPA 2 WiFi hotspot, and the device is placed near the Pycom device that should become a border router. Once all devices get a GPS fix (this can take between 5 and 15 minutes depending on the location), the test can begin. The total setup time is approximately 15 minutes to ensure all devices get a GPS fix, and the test will begin when the recording on the Gaia app tracking is initialized. It should be noted that we do not use an external GPS antenna, which would likely give us a location fix more rapidly. The only external antenna used is for LoRa, both GPS and WiFi in our nodes use the built-in antennas (see Figure 4 for a picture of one of our fully assembled nodes).

At this point, testing can commence. When a *test session*[1] is finished, all Gaia GPS logs have to be exported from the phone manually, transferred to a computer and parsed into the database. This makes the trail visible in the web application. The raw log file is also uploaded to the source code repository to be accessible to our analysis scripts.

A. Limitations

The COVID-19 pandemic severely impacted our collaboration. The quarantine rules posed a major obstacle in gathering data for analysis. Initially, we had planned to test together and gather data with all five devices we had bought, but this was no longer possible. Because of the pandemic, we were not able to do testing with all the Pycom devices simultaneously. Further, we encountered a hardware issue of one of the devices. Rather than replacing the faulty node as we would have aimed to do under normal circumstances, we instead opted to attempt to attempt to repair it so that we could continue our tests. In spite of this we managed to perform some tests and gather useful data, using up to three Pycom devices at a time, which were located in Trondheim. The remaining two devices were in another city, and the team member with access to those were in quarantine, so they were used for software development, but not any of the range tests.

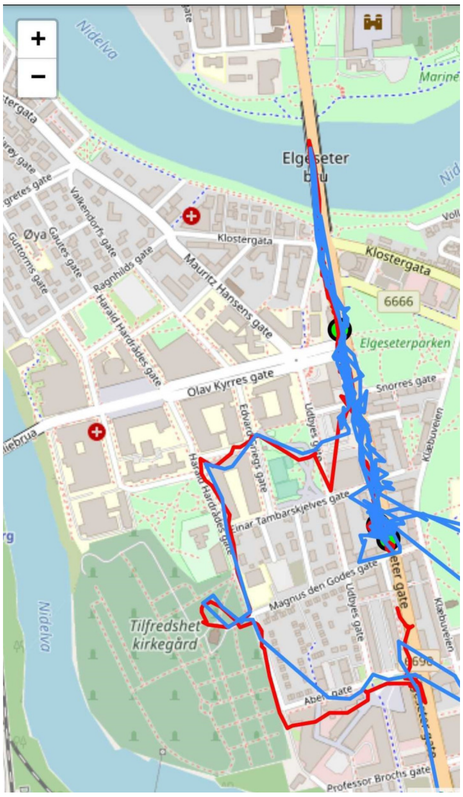


Fig. 5. PyMesh trail (blue) on top of Gaia (red) for tests 1 and 2.

Within these limitations, we performed the test series outlined below.

B. Test session 1

This session took place in Trondheim city and the priority was to measure packet loss at various distances, so as to investigate the range of the communication system. The Gaia GPS trail and Pycom trail for both tests 1 and 2 can be seen in Figure 5. All test were performed with DR6.

1) Test 1: Range: Since this was the first range test we slowly increased the distance to the border router while maintaining line of sight. After Pycom D was out of range on Elgeseter bridge and returned back in range, the devices renegotiated and switched roles (Pycom D was a router but was assigned to the leader role).

2) Test 2: Range: After the first range test was complete, a new Gaia recording was started to now measure packet loss with buildings between the Pycom devices. At the end of the test, Pycom D had an error where the LED was blinking yellow rapidly, indicating an error. The test was concluded soon after.

C. Session 2

This testing session took place in the rural outskirts of Trondheim. The testing environment was challenging for the

[1] We define a *test session* as a number of tests in an urban or rural environment with a given data rate. A test session may also include transportation to and from the given environment to gather data for GPS accuracy analysis. The main goal of a test session is to gather data for packet loss as a function of the distance to a border router, or to test the limit in range to a border router. The test sessions are summarized for brevity in this paper. For complete information on all test, see [14].

Pycom devices, since there is a lot of vegetation that blocks the line of sight. Terrain height differences made it more difficult to correctly measure the packet loss with respect to distance. All tests were performed with DR6.

1) Test 6: Rural Forest: Once the test started, both devices already had high GPS accuracy because they had been running for over 30 minutes. We walked up a forest path and lost connection due to a mountain crest in the way, but regained connection once Pycom D was in higher terrain and vegetation decreased. We experienced no issues, and Pycom D reconnected quickly once it returned in range.

2) Test 7: Rural Forest: This test started in a different direction on a steeper path with more trees between the devices. At the top of the hill the path flattened out and the network was subjected to more vegetation than just trees. The packet loss at this location was high, and we noticed that Pycom D had an issue where the LED was rapidly blinking yellow, similar to the one in test 2. It was later discovered that the Pycom D USB connector was loose. This hardware issue can be attributed to the faults seen on this node, since it would affect the power supply. Later, we soldered the connector back on, which improved the stability of this node.

D. Session 3

During this session we tested routing between three devices. The session was conducted in Trondheim city. The test consisted of two team members with one device each, as well as a border router. The border router was located outside a window on the fourth floor, and did not move during the test. The test was performed using DR6.

1) Test 8: Routing: Some issues occurred approximately 10 minutes into the test where both devices lost connection to the network, causing the test to be paused briefly. After some time both devices managed to reestablish their connection to the mesh network, and testing resumed. The two participants in the test walked together until maximum range was reached to the border router. Afterwards, one member continued to walk in the same direction with their device, while the other stayed stationary. Routing worked well between these devices. After a short while an issue occurred where Pycom D did not send its own data. In spite of this issue we were able to test routing successfully.

E. Session 4

This session took place approximately 10-15 minutes from Trondheim center, in the area previously used for session 2. Two members of the team took part in the session, each with their own device, as well as the border router. Two tests were conducted during this session. The border router was placed in a static position. All tests were performed using DR5.

1) Test 9: Rural Forest: This test used the same route as test 6 and both devices were assigned the router role. At the point furthest away from the border router, both devices lost connection to the mesh network and could not reconnect. There seemed to be a hardware or firmware issue causing the devices to be unable to reconnect. Saturation could also be a factor causing the issue, but this is uncertain – it may well be that we encountered the known issue described here [19].

2) Test 10 and 11: Rural Forest: These tests used the same route as test 7. Both devices were assigned the router role. This time the members walked in different directions. Due to an error one of the devices was powered off during some of the test, causing a gap in GPS data until the device was powered on again. Unlike test 9, the device quickly reestablished its connection to the mesh network and the rest of the test was performed without issues.

F. Session 5

The session was conducted in Trondheim city, and the goal was to test range and routing. Two members of the team participated, each with their own device, and the border router was placed at a static location. All tests used DR5.

1) Test 14: Range: This test started with both members walking together in the same direction, in an attempt to find the distance where they would lose connection to the mesh network. During the test both devices had the role of router. After walking approximately 850 meters, connection between the devices and the border router was lost. Both routers managed to reconnect to the network afterward. At this time, the border router experienced an unknown issue leading to corrupt data being sent to the server, so the test was concluded. It should be noted that this issue occurred only this once, so we were unable to further investigate this error.

2) Test 15: Range: This test took place with two devices in a car, where one had the role of router while the other was a leader. The goal was to test how far away the devices could be from the border router when buildings blocked the line of sight. The test started with a few smaller buildings where both devices managed to send data without problems. The range from the border router and size of buildings increased as the test went on, until connection was lost. This test was completed without any issues, and the range from the border router was quite good considering the amount of tall buildings blocking the signal path.

3) Test 16: Routing: This test took place in the same area where we tested routing during session 3. The test followed the same procedures as session 3, with both team members walking together for a while, and then one member remaining at a stationary position while the other walked further away. In contrast to test 8, this time both devices seemed to be able to transmit their data without any issues. The max range achieved with routing during this test was also much greater than in test 8. The device furthest away from the border router managed to reestablish its connection to the network after it was lost, and data was therefore also gathered on its way back to the border router. This is the longest and most stable test throughout all testing sessions.

VI. Analysis

The following sections present graphs, plots and images of data from our analysis of the PyMesh network. The graphs describe various relationships related to packet loss and accuracy.

A. Considerations

We used a collection of scripts that we developed to perform analysis on the data we gathered during testing. During the testing process we did not do logging on-device due to the increased workload and complexity this would cause, which meant that we lacked coordinate data for lost packets. In order to generate better distance metrics we used linear interpolation between the most recent data point before the loss, and the first one after. This allowed us to estimate the position of lost data packets. Because the devices are continuously moving, we divided the data up to 14 distance ranges to facilitate estimation of packet loss by distance. For the sizes of our data sets this number worked well for visualizing trends accurately without excessive variance. We used the haversine formula [20] to calculate distances between coordinates.

Another consideration was the use of Gaia coordinates to measure GPS accuracy of the system. The Gaia tracker did not log coordinate data frequently enough to precisely match every mesh network coordinate with a "ground truth" position. In order to work around this, we allowed a maximum of 10 seconds disparity between timestamps for most tests, and we found this acceptable due to the low distance we would usually move during this time. This allowed us to not lose too many data points while still ensuring that our analysis was valid within the restrictions of these considerations.

B. Environments

Figure 6 illustrates a clear difference between walking in a city with high buildings and limited signal strength compared to line of sight. Tall buildings influence the range and packet loss of a device. DR5 and DR6 differ about 100 to 150 meters in maximum range. DR5 has a significantly lower packet loss at distances under 700 meters.

Figure 7 shows how the environment and distance affects the packet loss of LoRa transmissions. The tests were conducted in the city (left) and the forest (right). Tests 1 and 9, as well as test 2 and 10, used the same routes, making it a better graph to compare between DR5 and DR6 in challenging environments. The spikes in this graph for forest test 1 (blue) and forest test 9 (green) have the same reduction at about a 125 meter distance, as well as a gradual increase towards 225 meters. Graphs 2 and 10 are less similar, but beyond 100 meters their trends are comparable. The graph indicates that DR5 could handle the challenging vegetation much better before a high packet loss occurred.

C. Routing

Our analysis showed interesting differences in potential max range and packet loss using multiple routing devices. The left graph in Figure 8 shows the packet loss differences between DR5 and DR6. The graph (left) shows the same spikes and dips where both tests were closing in on max range. The increase in packet loss is a result of the distance between the devices. The graph to the right shows all the moving Pycom devices participating in each test. We can see that instability of Pycom D at around 350 meters (Test 8, red line) amplified the resulting packet loss for Pycom B at around 800 meters (Test 8, green line). These spikes occurred at the same time, but at different distances.

Our DR5 tests do not display the same level of amplification as DR6 due to device instability. The increase in packet loss beyond 1000 meters for Pycom B (Test 16, blue line) is a result of the device being at a greater distance to Pycom D, than Pycom D is to the border router, Pycom C. See Table IV for accurate distances and Figure 11 for an illustration of the Pycom device positions for Test 16.

D. GPS accuracy

Figure 9 presents the difference in GPS accuracy between the Gaia GPS app and Pycom devices (note that three outliers are omitted from the graph in order to ensure readability). We see a clear trend in the increase of GPS accuracy over time for these tests, although the data set for this graph is small. For this test the Pycom devices are being transported in a vehicle that can travel long distances between each reported PyMesh location. This will influence the GPS accuracy.

Figure 10 shows the distribution and average GPS accuracy for all tests performed in urban and rural environments. The forest tests lasted shorter (up to about 20 minutes), but still had better accuracy overall. This is a result of the open environment making it possible to have more satellite fixes simultaneously. The city tests have an average accuracy of approximately 12 meters. Note that because this graph spans a duration of 50 minutes it does not show the initial increase in accuracy after booting up a device. The tests also had a 15 minute setup time to ensure higher accuracy.

VII. Results

The average packet loss with line of sight for DR5 was 35.1% (336/957) with Pycom D and B, and 32.25% (158/490) for Pycom B alone. This metric was calculated using all tests performed on DR5 (not including accuracy tests). The average packet loss for DR6 was 51.47% (263/511 packets). Due to the hardware instability of Pycom D and the lower amount of data points we consider this metric less reliable than that of DR5.

For short distances below 50 meters we observed a minimum packet loss of approximately 25% for DR5 and 30% for DR6 in forest environments. The packet loss of DR6 increased at a greater rate in shorter distances compared to DR5 in challenging environments. Our tests achieved a maximum range of 608.46 meters for DR6 and 880.88 meters for DR5. This effectively shows a 272 meter increase (44.7%) for DR5 with a direct line of sight, as shown in Table IV.

The PyMesh routing functionality creates an effective doubling in range (up to 1633.37 meters), as shown in Figure 11 and Table IV. Hierarchical routing in a PyMesh network makes it possible to have up to 2.437 square kilometer coverage per Pycom device in a flat environment, as shown in Figure 11.

The average accuracy was approximately 12 meters in urban environments and 10 meters in open forest environments.

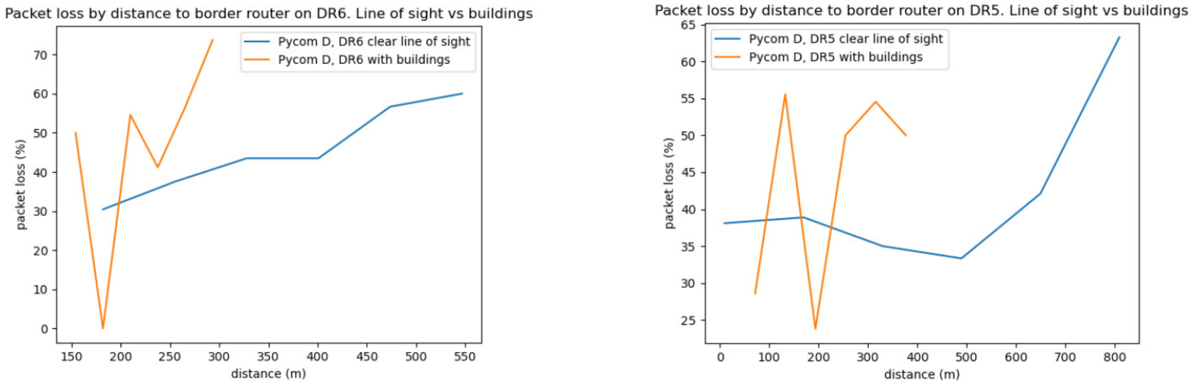


Fig. 6. Packet loss in different city environments on DR6 (left) and DR5 (right)

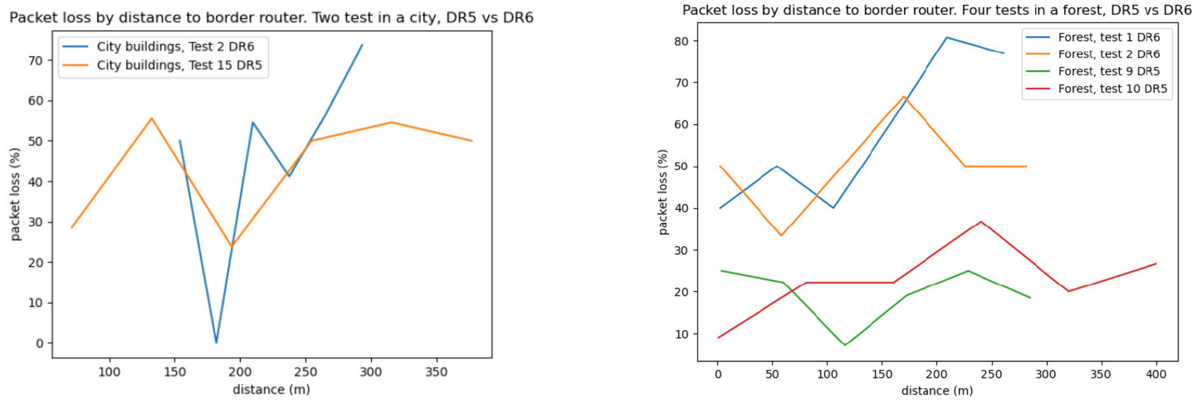


Fig. 7. Packet loss by distance for challenging urban (left) and rural (right) environments

There is a large amount of variance (Figure 10), and achieving a high accuracy usually takes around 15 minutes (Figure 9).

DR5 has provided better performance in terms of both packet loss and range compared to DR6 in our tests. The effect of different data rates on the stability of the PyMesh network itself (not the GPS tracking we perform), is difficult to conclude based on our testing due to hardware issues and amount of data gathered. We think that using DR5 shows promise, and would recommend further research with respect to its stability with a larger number of devices in a PyMesh network.

VIII. Related work

The NATO Research Task Group (RTG) IST-147 titled ”Military Application of Internet of Things” examined applying COTS civilian IoT approaches for military purposes. Typically, the use case was centered around a humanitarian assistance and disaster relief (HADR) coalition operation in a Smart City, where IoT information from the city could be used as additional sensor input to the military situational awareness and hence Command and Control (C2) systems [21]. Following that group's conclusion in 2019, this work now continues in NATO RTG IST-176 titled ”Federated Interoperability of Military C2 and IoT Systems”. That group continues work on Smart Cities, but also broadens the scope to include such use cases as our recent work on crowdsourcing and crowdsensing [22].

For instance, Mekki et al. [6] have performed a comparison of Low Power Wide Area Networking technologies, including LoRa. They point to LoRa's main strengths being battery lifetime, capacity, and cost. In their view, LoRa will serve as the lower-cost device, with very long range (high coverage), infrequent communication rate, and very long battery lifetime. Further, LoRa will also serve the local network deployment and the reliable communication when devices move at high speeds. They identify the following application areas as suitable for using LoRa: Smart farming, manufacturing automation, smart buildings, and tracking for logistics.

LoRa is a building block of LoRaWAN [10], which adds security and the means to organize a LoRa network with one or more gateways bridging LoRa to other networks, e.g., the Internet. The military application aspect of LoRaWAN has been investigated by Michaelis et al. [24], who used the USA version of LoRaWAN in the 915 MHz band to track vehicles in an urban environment (downtown Montreal, Canada). Their findings show a usable range of LoRaWAN of up to 5 Km under the conditions tested. In the case where buildings obstructed the line of sight, packet loss increased and the ef-

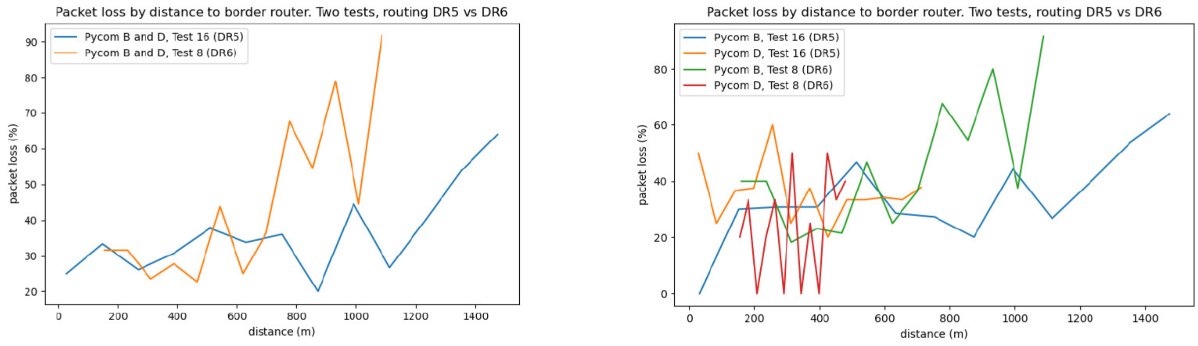


Fig. 8. Packet loss by distance routing DR5 vs DR 6 summary (left) and all devices (right)

TABLE IV
DISTANCES BETWEEN GPS POINTS (CALCULATED USING [23]). FROM LEFT TO RIGHT: PYCOM B TO D (DR5), PYCOM D TO C (BORDER ROUTER) (DR5), PYCOM B TO C (DR5), PYCOM D TO C (DR6, TEST 1).

	Pycom B to D	Pycom D to C	Pycom B to C	Pycom D to C
GPS 1 Latitude	63.43307	63.41855	63.41855	63.418652
GPS 1 Longitude	10.39128	10.39625	10.39625	10.396535
GPS 2 Latitude	63.42521	63.42521	63.43307	63.424006
GPS 2 Longitude	10.39349	10.39349	10.39128	10.394009
Distance apart	880.88 meters	753.18 meters	1633.37 meters	608.46 meters

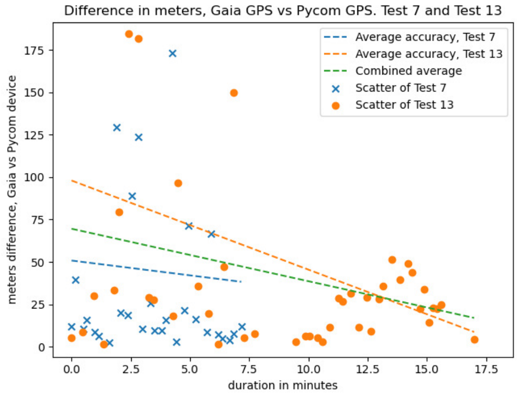


Fig. 9. GPS accuracy over time

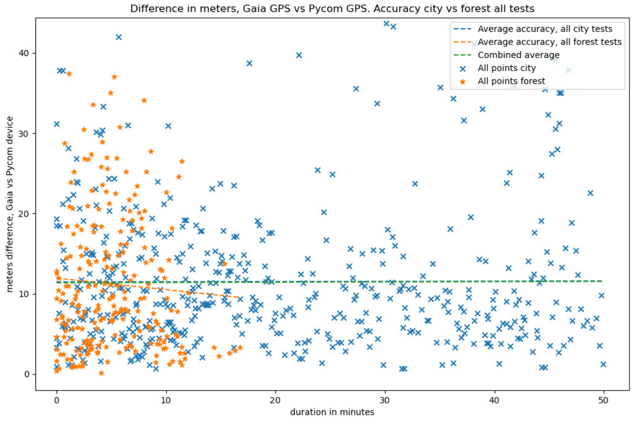


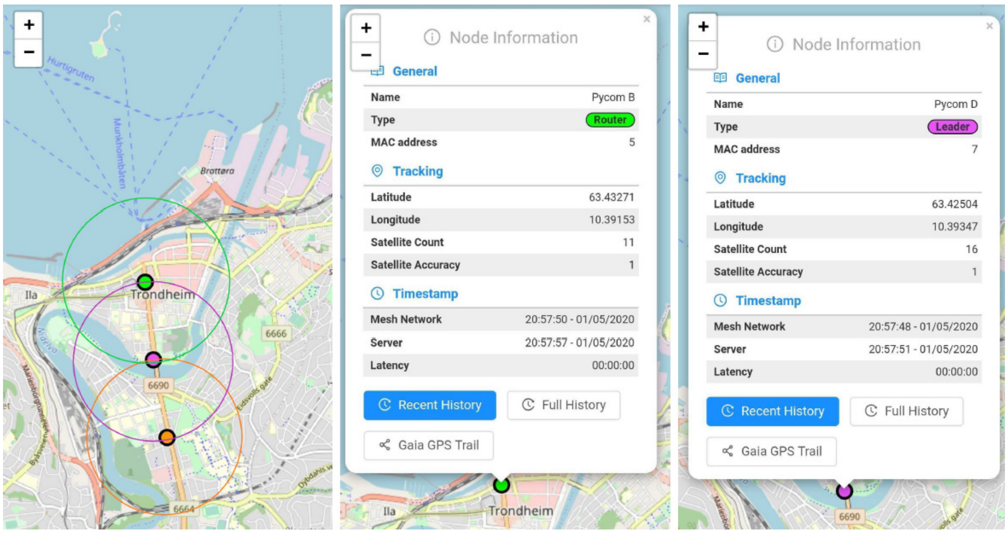
Fig. 10. Aggregated GPS accuracy over time

fective range was shorter, around 2.5 Km. However, to the best of our knowledge, this and other such studies (e.g., [25] which presents a similar experiment with comparable results) all rely on LoRaWAN. We are not aware of any prototype similar to ours as described in this paper, that performs tracking using commercial IoT devices over a LoRa mesh network.

IX. CONCLUSION AND FUTURE WORK

Our prototype uses PyMesh and LoRa to enable you to easily track the location of assets anywhere in the world, though under the limitation that one node, a border router, needs to be within range of a preexisting cellular network. The remaining nodes can operate without other coverage than the LoRa mesh network. Once the necessary software is installed the solution is easy to employ, even without technical knowledge. The prototype includes a modern web application that is intuitive and easy to use, and is supported on all major platforms including mobile. From our tests, the software performed as expected and could be a tool for asset tracking in search and rescue operations. Our findings show that though the Pycom units we used with PyMesh exhibited a shorter range than previous experiments conducted with LoRaWAN (see, e.g., [24], [25]), it should be noted that in our experiment all nodes were equally capable, whereas the LoRaWAN experiments had a dedicated, more capable gateway deployed. Still, coupling LoRa with PyMesh on the Pycom units effectively doubled the operating range, as we found, due to the multi-hop and routing capabilities of the mesh network.

For future work, it would be interesting to experiment with more units in a larger network. The scalability of the network was not specifically investigated by us so far, since we had a limited amount of nodes available. Since we found that of



Node Information

General

Name	Pycom B
Type	Router
MAC address	5

Tracking

Latitude	63.43271
Longitude	10.39153
Satellite Count	11
Satellite Accuracy	1

Timestamp

Mesh Network	20:57:50 - 01/05/2020
Server	20:57:57 - 01/05/2020
Latency	00:00:00

Node Information

General

Name	Pycom D
Type	Leader
MAC address	7

Tracking

Latitude	63.42504
Longitude	10.39347
Satellite Count	16
Satellite Accuracy	1

Timestamp

Mesh Network	20:57:48 - 01/05/2020
Server	20:57:51 - 01/05/2020
Latency	00:00:00

Fig. 11. Theoretical coverage with routing, in difference to coverage with no routing (only orange circle). Taken at approximately max distance (test 16).

the data rates we tested, DR5 has provided better performance in terms of both packet loss and range compared to DR6, we would like to perform more extensive tests with DR5. Also, it would be interesting to try adding additional features to the prototype, like short, pre-defined messages, in addition to the tracking we have implemented now. Such a messaging capability would nicely complement the tracking features, making the prototype much more versatile for asset tracking, search and rescue, and possibly HADR operations.

Acknowledgment

We would like to thank Torkjel Søndrol at FFI, Norway for sharing valuable insights on the Pycom IoT development platform. Finally, a thanks to Katerina Mangaroska at NTNU, Norway, for her efforts as supervisor for Engstad, Andersen, Røkenes and Blaalid when they worked on their Bachelor degree thesis [14], of which the prototype described in this paper is a major component.

References

[1] A. Davis. Cellular baseband security. Georgia Institute of Technology, 2012. Available at https://smartech.gatech.edu/bitstream/handle/1853/43766/davis_andrew_t_201205_ro.pdf (2020/02/05).

[2] A. Turner. Tackling cellular vulnerabilities. Available at https://misti.com/infosec-insider/tackling-cellular-vulnerabilities (2020/02/04). MISTI.

[3] Pycom. Pycom company website. Available at https://pycom.io/ (2020/02/03).

[4] Semtech. Lora modulation basics. Available at https://web.archive.org/web/20190718200516/https://www.semtech.com/uploads/documents/an1200.22.pdf (2020/04/27).

[5] N. Suri, M. Tortonesi, J. Michaelis, P. Budulas, G. Benincasa, S. Russell, C. Stefanelli, and R. Winkler. Analyzing the applicability of internet of things to the battlefield environment, 2016 International Conference on Military Communications and Information Systems (ICMCIS), 23-24 May 2016, DOI: 10.1109/ICMCIS.2016.7496574, Brussels, Belgium

[6] K. Mekki, E. Bajica, F. Chaxel, and F. Meyer, A comparative study of LPWAN technologies for large-scale IoT deployment, ICT Express Volume 5, Issue 1, March 2019, Pages 1-7.

[7] Pycom. Lopy4 specsheets. Available at https://docs.pycom.io/gitbook/assets/specsheets/Pycom_002_Specsheets_LoPy4_v2.pdf (2020/03/19).

[8] Pycom. Pytrack specsheets. Available at https://docs.pycom.io/gitbook/assets/pytrack-specsheet-1.pdf (2020/03/19).

[9] Pycom. Pymesh documentation. Available at https://docs.pycom.io/pymesh/ (2020/03/19).

[10] L. Alliance. LoRaWAN specification. Available at https://lora-alliance.org/sites/default/files/2018-04/lorawantm_specification_-v1.1.pdf (2020/02/03). Section 2, LoRaWAN Regional Parameters.

[11] Semtech. Understanding lora adaptive data rate. Available at https://lora-developers.semtech.com/library/tech-papers-and-guides/understanding-adr/ (2020/04/27).

[12] OpenThread. Openthread opensource thread implementation. Available at https://openthread.io/ (2020/02/03).

[13] T. Group. What is thread. Available at https://www.threadgroup.org/What-is-Thread (2020/03/19).

[14] H. Engstad, E. Andersen, S. Røkenes and T. Blaalid. Long Range Mesh Networks. NTNU, Department of Computer Science, May 2020.

[15] Gaia. Gaia gps company site. Available at https://www.gaiagps.com/ (2020/04/29).

[16] Django. Django framework. Available at https://www.djangoproject.com (2020/02/04).

[17] React. React framework. Available at https://reactjs.org/ (2020/04/27).

[18] Internet Engineering Task Force (IETF), The GeoJSON Format, RFC 7946 Standards Track, August 2016.

[19] P. forum. wlan causes core panic. Available at https://forum.pycom.io/topic/5765/loadprohibited-core-panic-when-initialize-wlan (2020/04/29).

[20] pypi.org. Haversine formula library for python. Available at https://pypi.org/project/haversine/ (2020/04/29).

[21] F. T. Johnsen, Z. Zielinski, K. Wrona, N. Suri, C. Fuchs, M. Pradhan, J. Furtak, B. Vasilache, V. Pellegrini, M. Dyk, M. Marks, and M. Krzyszton, Application of IoT in Military Operations in a Smart City, 2018 International Conference on Military Communications and Information Systems (ICMCIS), Warsaw, Poland, 22 - 23 May 2018.

[22] M. Pradhan, F. T. Johnsen, M. Tortonesi, and S. Delaitre, Leveraging Crowdsourcing and Crowdsensing Data for HADR Operations in a Smart City Environment, IEEE Internet of Things Magazine, Volume: 2, Issue: 2 , June 2019.

[23] gps coordinates. Gps, distance calculator. Available at https://gps-coordinates.org/distance-between-coordinates.php (2020/04/29).

[24] J. Michaelis, A. Morelli, A. Raglin, D. James, and N. Suri. Leveraging LoRaWAN to Support IoBT in Urban Environments. 207-212. 10.1109/WF-IoT.2019.8767294.

[25] B. Jalaian, T. Gregory, N. Suri, S. Russell, L. Sadler, and M. Lee. Evaluating LoRaWAN-based IoT devices for the tactical military environment, 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), 5-8 Feb. 2018, DOI: 10.1109/WF-IoT.2018.8355225, Singapore, Singapore

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 511–519 DOI: 10.15439/2020F48 ISSN 2300-5963 ACSIS, Vol. 21

A Federated Platform to Support IoT Discovery in Smart Cities and HADR Scenarios

Lorenzo Campioni[1], Niccolò Fontana[1,2], Alessandro Morelli[2], Niranjan Suri[3,2], Mauro Tortonesi[1]
[1] Distributed Systems Research Group, University of Ferrara, Ferrara, Italy
{lorenzo.campioni, mauro.tortonesi}@unife.it
[2] Florida Institute for Human and Machine Cognition (IHMC), Pensacola, FL, USA
{amorelli, nfontana, nsuri}@ihmc.us
[3] US Army Research Laboratory (ARL), Adelphi, MD, USA
niranjan.suri.civ@mail.mil

***Abstract*—Smart Cities are among the most dynamic and rapidly evolving modern environments, driven by the development of new technologies and the fast growth of the Internet of Things (IoT), which enable the acquisition and processing of very large amounts of data. However, accessing IoT assets is proving to be a challenge, as neither formal nor de facto standards to discover connected Things have emerged. Services that provide discovery and access capability for IoT resources are in the rise, but they often adopt service-specific interfaces and authorization mechanisms that hinder the development and maintainability of IoT applications. Low flexibility and interoperability become especially problematic during emergency situations, when responders might need to access resources that normally would not be allowed to access. To address these issues, this paper describes MARGOT, a distributed edge computing platform that supports domain-aware and secure discovery of IoT resources in Smart Cities. Experimental results obtained using MARGOT in an emulated network environment show that our platform can effectively reduce discovery latency and bandwidth consumption under the considered use cases and network conditions.**

***Index Terms*—Internet of Things (IoT), Humanitarian Assistance and Disaster Recovery (HADR), Resource Discovery, Distributed Information Systems**

I. Introduction

SMART Cities worldwide are thriving and evolving at great speeds, mainly driven forward by the possibility of combining innovative Information and Computing Technology (ICT) solutions with small, cheap, and yet powerful computational and sensor technology, which has paved the way for new business opportunities that are attracting large numbers of investors and industry leaders to the sector. Leveraging ICT to process large amounts of data generated by connected sensors, actuators, and other intelligent objects that are part of the Internet of Things (IoT), Smart Cities aim at improving their citizens' quality of live by enhancing government, health, transportation, security, education, and other services [1]–[4].

The accessibility of ”Things” and their interoperability with other systems are major challenges that Smart Cities have to face, as they directly impact their capability to advance at a fast pace by introducing new services or extending the ones currently offered [5]. To address this issue, public and private organizations that work in the field of smart services and IoT have been developing new services that allow applications to retrieve access information on smart Things and other connected devices managed by those organizations. In the future, with the development of new communications and computation technologies, which enable the autonomous discovery of new nodes and resources in the network, and the definition of formal standards for the IoT, we expect that more and more players will enter the market and the number of IoT services and connected devices to rise as a consequence.

These IoT services are typically Cloud-based [6], [7], offer a proprietary Application Programming Interface (API) that reflects the nature of exposed assets, and employ a multitude of different techniques to enforce security and verify clients' authorization [8]. This strongly reduces accessibility and interoperability and raises the need for solutions that can simplify the discovery and access of the IoT assets exposed by those heterogeneous services. Proposed approaches will need to match domain-specific security requirements, offer good performance to users and applications, and avoid taking a heavy toll on the Smart City's network infrastructure. In addition, those approaches will have to be able to support the cities' administration in case of Humanitarian Assistance and Disaster Relief (HADR) missions and operations, during which the network connectivity might be limited and it is vital that emergency responders are able to access critical data and resources from the stricken areas.

As a step in this direction, the present paper describes the design and architecture of MARGOT, a platform to support domain-aware and context-aware discovery of IoT resources in Smart Cities. This work extends our previous study in [9] by discussing the use of Federation Services [10] to provide an array of distributed capabilities within MARGOT. We also present the results of new experiments that demonstrate MARGOT's ability to lower IoT asset discovery latency for applications and reduce network bandwidth consumption under specific usage and network conditions.

II. Accessing IoT resources in Smart Cities

Smart Cities and other smart environments are characterized by the extensive and effective use of ICT solutions to improve human day-by-day activities, enabling smart living and the development and deployment of next generation services.

In particular, Smart Cities take advantage of the pervasive presence of connected ”intelligent objects” that interact with the environment. This capillary network of IoT devices permits to acquire large quantities of data on their surroundings, e.g. sensor measurements, images, audio, video, and so on, that can be processed and then made available to users and applications. Such capabilities enable the development of time-critical, context-, and location-aware services for the smart citizens.

However, despite the IoT being one of the main pillars of Smart Cities, it also poses several challenges, which include the following: IoT devices present computational and power limitations that might reduce accessibility; small sensors and other Things typically only support specific communication protocols for constrained devices; multiple actors are involved, as owners and administrators of the devices, with different security and access policy requirements. Given the incredibly large and continuously growing number of connected Things and their extreme heterogeneity, in terms of data produced, location, domain, accessibility, and others, it becomes crucial to provide solutions that enable users and applications to identify resources based on specific requirements and interests.

To this day, several cities and organizations have deployed solutions to enable users and applications to obtain access information about IoT resources made available within the domain (e.g., the city). Such solutions are typically Cloud-based and offer a web-based API, generally encoded using JSON or other standard formats, that allows consumers to query for access and other types of information about the managed IoT resources.

These IoT services typically allow to search for different types of devices, including webcams, weather sensors and stations, several kinds of sensors (pollution, noise, light, traffic, etc.), vehicles and other transportation-related things, and so on. Some of these services are: Windy (https://www.windy.com), OpenWeather (https://openweathermap.org), City Bikes (https://citybik.es), Thingful (https://www.thingful.net), Digitraffic (https://www.digitraffic.fi), the New York State’s 511 Traveler Information System (511NY) (https://511ny.org), LookCAM (https://lookcam.com), and Airly (https://airly.eu). Some services do not offer direct access to the managed devices, but provide an interface to query and retrieve the data generated by them.

For this study, we used data obtained from the services provided by Digitraffic, NY511, and Airly. The Finnish Transport Agency operates Digitraffic, which offers real time traffic information that covers road, marine, and rail traffic in Finland. NY511 (https://511ny.org), hosted by the State of New York, provides information tightly related to transportation services and road conditions throughout the State. Finally, Airly gathers information related to air quality around the globe using sensors that measure the concentration of PM1, PM2.5, PM10, and NO2 and O3 gases in real time.

III. Future IoT Services and HADR Scenarios

Since the term IoT was coined, the population of devices that pervade smart environments has constantly grown. Thanks to The multitude of novel communication solutions, such as LTE/4G, 5G, LoRa, LoRaWAN, and other lightweight communication protocols, more and more devices have been able to connect to the Internet and generate information that other devices and applications can consume. Scientists expect that the number of devices connected to the Internet will continue to grow strongly in the next years [11], sustained by the decreasing cost of hardware, the development of new technologies more suited to constrained devices, and the definition of official standards, such as the IEEE P2413 ”Standard for an Architectural Framework for the Internet of Things (IoT)” [12], that can simplify the interaction between devices and other network actors. Furthermore, numerous private organizations and stakeholders have been attracted by the business opportunities involved with Smart Cities and IoT and started to deploy their own private sensors, thus actively participating to the growth of the number of IoT devices connected.

This growth further increases the need for services akin to those provided by Digitraffic, 511 NY, and Airly. As an example, let us consider an application for police authorities that makes use of traffic camera feeds and image recognition software in order to track down a criminal during a chase. The application requires access (IP addresses, protocols used, and so forth) and other information (e.g., cameras’ locations) to be able to connect to cameras located in positions from which they can record the target’s movements. Without other solutions, the application must have prior knowledge of all cameras’ locations and access information, an approach that is not really suited for highly dynamic environments such as Smart Cities, where sensors can move, sleep, or fail for a number of possible reasons. In fact, this approach would require that either all these events are notified to all applications or applications are designed to handle failures nicely and fallback to other data sources (for instance, if the desired camera is offline, there might still be a lower resolution camera that can record the same area, or other cameras nearby that might still provide useful streams for the purpose of the application). Furthermore, the Smart City infrastructure is typically composed of many different domains where resources are managed and maintained by different organizations, each with potentially different security and access policies. IoT discovery services can help mitigate the complexity of discovering and then accessing IoT devices in multi-domain Smart City scenarios by providing updated information on live resources to applications, supporting sophisticated search criteria to ensure that clients are informed about all relevant resources, and requiring the authentication of clients.

We envision that the Smart Cities’ highly dynamic environment and the constant growth of connected IoT assets will lead to a new generation of IoT services that are able to autonomously discover and register new resources in the network. Such services will significantly simplify and speed up the deployment of new devices, since they will not require manual registration to be discoverable by clients. In addition, these new IoT services will provide all stakeholders with a constantly updated view of the status of the devices within the environment and information on how to retrieve data from them, e.g., via a M2M-compliant API.

These services will also offer a strategic advantage during HADR operations in Smart City environments. HADR operations take place after a disaster has severely damaged parts of the environment and/or put human lives at risk, which requires local authorities to immediately enact safety protocols to assists the victims and prevent aggravating the current situation. During these procedures, the ability to exploit local sensors and other IoT assets plays a key role in increasing the effectiveness of HADR operations by improving the situational awareness of responder teams. Moreover, the support for the automatic discovery of new resources allows those teams to deploy sensors on-the-fly (e.g. a camera-equipped drone) whenever required and retrieve the produced data using the same applications they would use to access other devices in the city.

However, the Smart City network infrastructure could also suffer damages during disasters. As a consequence, links can be severed and nodes can break, causing traffic to be rerouted, network congestion on the unstruck links to worsen, and portions of the network to become unreachable. The IoT infrastructure is especially impacted when it requires connectivity to the Cloud, as it often happens with sensors that publish collected data to remote servers or when applications periodically check the status of IoT devices by means of some form of network polling. In these conditions, IoT services and applications that run within the edge network and are able to autonomously discover available resources would offer a more robust solution. More specifically, the automatic discovery of IoT resources would enable services and applications to identify assets that have become unavailable, clients might still be able to connect to the desired nodes located in the same edge network, and emergency responders could still deploy and access new sensors dynamically during a mission.

Despite the advantages that we expect new generation IoT services to bring about, many challenges will remain. First, future IoT services will likely share information through proprietary APIs that satisfy domain-specific requirements. As a consequence, IoT applications that allow emergency responders and other personnel to access assets across multiple domains will be required to implement and maintain different interfaces to each service. From the security perspective, these software will often have to support and manage several security protocols, authentication mechanisms, and user certificates to be compatible with the requirements of different services, which require considerable coding efforts and costs. During HADR and other emergency operations, rescue teams might need to access devices that would not normally be allowed to; to address this requirement, special solutions will likely be needed that necessitate additional work from both software developers and IoT administrators responsible for the different domains. Moreover, developers of IoT applications for emergency responders will have to invest significant efforts to design software that can withstand partial network failures, for instance implementing distributed caching, supporting peer-to-peer querying and data retrieval, and discovering new sensors on-the-fly.

IV. The MARGOT Platform

Due to the considerations discussed in Section III, solutions able to locate new IoT resources autonomously and designed to be more robust in presence of HADR situations are extremely interesting. In this context, we developed a distributed edge computing platform, MARGOT, that has the goal of simplifying the development of IoT applications. MARGOT permits to discover IoT resources across separated domains and network segments and supports context-aware applications by providing them with a query interface that accepts parameters to refine the search criteria. IoT applications can interact with MARGOT via a JSON RESTful API. The architecture of MARGOT is represented in Fig. 1, which shows the major components, i.e., the Discovery Agents (DAs), the Information Processor (IP), Federation Services (or Information Management System Bridge, IMSBridge), and the ReST API, and the interactions between them. The MARGOT platform is designed in such a way that each instance is responsible for the discovery of resources within one or more domains, and each domain has one and only one MARGOT instance of reference, which will typically be deployed within the same network or close to it, i.e., geographically and/or in terms of network hops. The knowledge of all discovered IoT resources is distributed across all MARGOT instances in the system, which exchange information via Federation Services (we sometimes refer to the set of MARGOT instances connected via Federation as ”federated MARGOT instances” or ”MARGOT federation”).

DAs implement domain-specific resource discovery and store data about the discovered assets in the local MARGOT database. As shown in Fig.1, DA implementations rely on specific protocols, such as MQTT or CoAP, and the corresponding discovery procedures to detect and identify IoT assets within the local network or domain. To support the discovery of the resources made available via services like 511NY or Airly, it is enough to write a DA that implements the API of the chosen service. DAs are also responsible for complying with any security requirement imposed by the domains. For instance, an IoT domain could require DAs to be authenticated and authorized before they can access the domain resources.

DAs can perform the discovery process either proactively, if the process is executed periodically or the implemented protocol supports some form of proactive discovery, or reactively, when specific events trigger discovery, such as a request issued by a client or coming from a federated MARGOT instance. Generally speaking, proactive behavior trades query latency for the freshness of the information. Depending on the rate of user requests and the cost of the discovery process, proactive discovery could lead to either a decrease or an

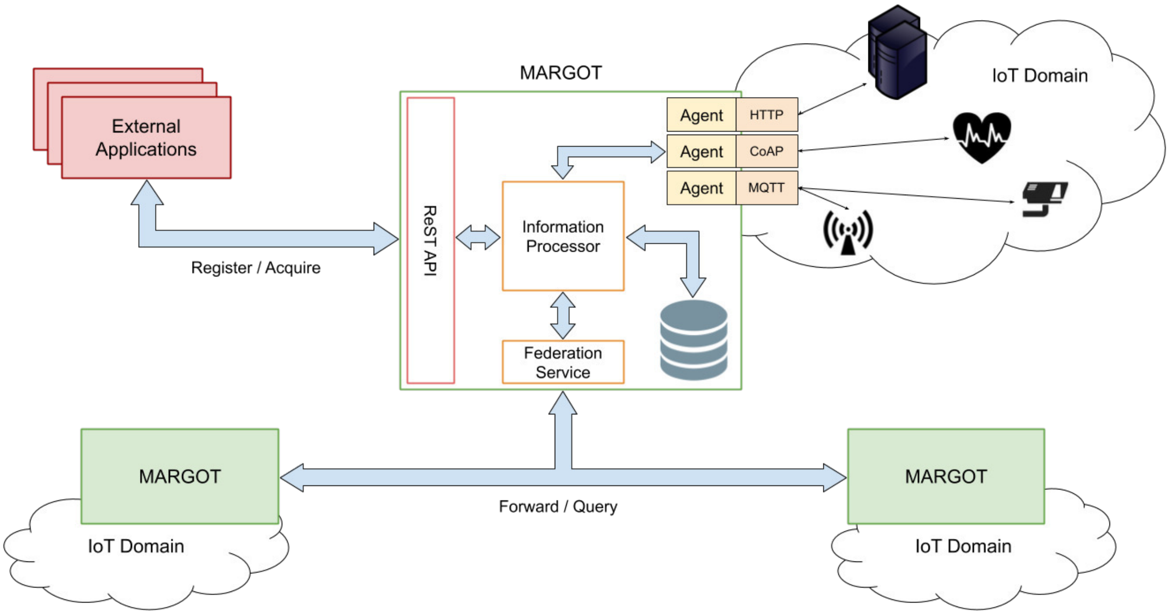


Fig. 1. The MARGOT Architecture

increase in bandwidth consumption. To tackle this matter and provide more efficient proactive discovery strategies, DAs should allow MARGOT to tune parameters that control the discovery process, e.g., the frequency of the process. Protocols that naturally support proactive discovery, such as CoAP and MQTT, can increase the efficiency of detecting new assets, but they still require some form of periodic probing to ensure that previously discovered Things are still alive and reachable.

The IP takes care of processing the data received from the DAs, storing them in the local database, and handling clients' requests. While doing this, the IP also collects statistics that characterize the domain and user requests, including the variability of the discovered IoT assets and frequently requested resources, and merges them with the statistics received from federated MARGOT instances. Finally, the IP manages data exchange via Federation Services: it decides which IoT resources to publish into the MARGOT federation, forwards user queries if necessary, replies to queries received from remote MARGOTs, and shares updated domain statistics.

The IP is also responsible for tuning the behavior of proactive DAs and perform other optimizations based on the acquired statistics. For instance, in presence of highly variable local domains, MARGOT will typically request registered DAs to increase the frequency of discovery, adjust its caching policy by decreasing the expiration time for the resources in those domains, and notify federated MARGOTs about the changes. This will affect the number of user queries that will be forwarded to federated MARGOTs against the number of requests that will be resolved using the data cached in the local database, which in turn will have an impact on the system bandwidth utilization and the accuracy of the information returned to clients.

MARGOT also supports *proactive querying*, which guarantees that the information cached in its local database about certain IoT assets is always up-to-date. MARGOT activates this mechanism if the number of user requests for the same set of resources in a given period of time is above a *"trigger"* threshold and preserves it until that number falls below a *"maintenance"* threshold. Both thresholds are configurable and can be tuned by the local MARGOT administrator. When proactive querying for a certain set of resources has been activated, whenever one of the DAs reports an update to at least one element of the set, MARGOT automatically pushes the updated information to all federates. By doing this, other MARGOT instances will be able reply to user requests that involve any elements in the set directly, without having to send queries to other federates. Proactive querying affects the amount of data exchanged by MARGOT via Federation in a way that ultimately depends on the number of assets in the set, how often they are updated, the number of MARGOT instances involved, and the amount of user requests received that can be resolved from the resources in the set.

A. Data Sharing via Federation Services

MARGOT relies on Federation Services [10] to exchange data with other MARGOT instances. Federation Services provide clients of the IMSBridge (federates) with a completely distributed publish-subscribe communications infrastructure that supports and simplify information exchange in multi-domain scenarios. *Topics* control the routing of data over the Federation network. Topics are named abstractions (i.e., identified by unique strings) that can be thought of as independent communication channels over the network; some topics are typically predefined via configuration files, but federates can also create new ones dynamically at run-time. New messages published within one topic are delivered to all federates sub-

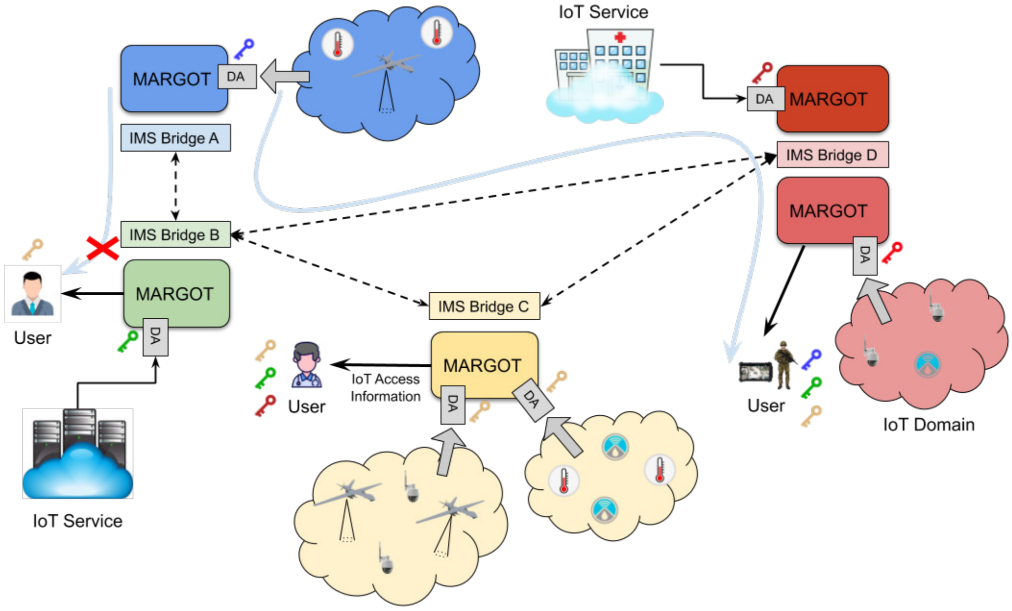


Fig. 2. MARGOT with Federation Services and ABE security

scribed to that topic, which will receive the message directly from the publisher or from another IMSBridge. Clients can join and leave these channels at any time.

Each IMSBridge instance can specify policies that determine what information it is allowed to share with every other instance. This allows one domain's administrators to define rules locally that implement that domain's security and data sharing policies, without the need to coordinate with other domains' administrators. Sharing policies can be enforced using a security system based on Attribute-based Encryption (ABE), which permits to combine different security layers on a per-message basis [13] and ensures that only clients in possess of the right keys can access the messages. Federates can use ABE to encrypt the payload and metadata of messages before publication, but the information relative to the topic of publication will not be encrypted to guarantee correct routing.

Federation Services support other useful features for distributed multi-domain environments. *Distributed queries* allow federates to send and run queries on all or a subset of IMSBridge instances, in order to retrieve published messages that match client-defined criteria. Distributed queries take advantage of metadata information specified during publication to select relevant messages. *Smart synchronization* enables two IMSBridge instances to exchange updated messages after a disconnection period, e.g., caused by network-problems or other issues. Depending on the nature of the messages, all or only the most recent updates will be exchanged.

MARGOT leverages Federation Services' capabilities for all distributed actions, including the discovery of new instances, proactive querying and IoT data replication on federated instances, remote query execution, and the exchange of control information and usage and domain statistics. This enables users and clients to discover resources available across multiple IoT domains and allows the platform to adapt its behavior, e.g., concerning caching and query forwarding, under certain circumstances. Finally, MARGOT leverages Federation Services' policy-based data sharing and ABE security to control the access to IoT assets information.

Figure 2 depicts a distributed MARGOT deployment connected via Federation Services. Each MARGOT in the Figure is connected to a single IMSBridge that enables it to exchange data with other MARGOT instances through the Federation network; dashed arrows represent connections between IMSBridge instances that compose the Federation network. Users can connect to any MARGOT instance to obtain access information about IoT assets discovered by any federated MARGOT, in accordance with the data sharing and security policies implemented by the IMSBridge instances involved. For example, on the right side of the Figure, a user with permissions to access IoT asset information for three different domains (represented by the blue, green, and yellow keys next to him) connects to its local MARGOT instance (light red box in the Figure) to retrieve information about sensors in the blue domain. MARGOT translates the request received from the user into a Federation query, which is disseminated from *IMS Bridge D* across the whole Federation network, solved locally by each federate, and finally the answers are relayed back to the node from which the query originated. MARGOT then caches the information locally for future requests and generates a response for the user with the requested information. As a second example, another user that only has permissions to access information about Things located in the yellow domain also issues a request for assets in the blue domain; however, this user does not have permissions to access IoT information from the blue domain and so he or she will not be able to decrypt the data received from MARGOT.

B. What MARGOT brings to the Table

The MARGOT platform offers a number of extremely interesting features to IoT application developers, which significantly facilitate and speed up development. Without doubt, one of the most appealing functionalities is that MARGOT grants access to all discovered IoT resources via a single API. This feature can help cutting down development and maintenance costs of applications tremendously. Additionally, the API offered by MARGOT already provides the possibility to formulate selective queries to filter out unwanted resources, for instance restricted to a specific geographical area, domain, and/or sensor type. As a consequence, developers do not have to write the code to handle filtering (or, at least, that code can be simplified considerably) and applications will save bandwidth and battery life by downloading and processing less data, which is especially good for mobile applications. Finally, since MARGOT instances will generally be deployed in locations that enable the discovery of new IoT assets (think about the case of sensors whose discovery necessitates the use of multicast over the local network segment), MARGOT will be able to give users and applications access to resources that they would not be able to discover otherwise, because of network and protocol limitations.

MARGOT can also help IoT service providers at multiple levels. First, MARGOT's caching capability can reduce the amount of traffic that service providers will need to handle. Moreover, the local MARGOT instance will typically be deployed closer, e.g., in terms of network hops, to IoT resources and service providers than users, whose location cannot be predicted or controlled easily, and lower distances tend to increase network efficiency. Finally, MARGOT essentially decouples users' requests for information on IoT resources from their discovery; this allows discovery-related traffic to become independent from the number of system users, thus generating more stable and predictable traffic loads over the sensor networks. All this translates into lower costs for infrastructure, network service, and power consumption for providers.

MARGOT can also help the Smart City administrators by simplifying the management of permissions required to access resource discovery for different domains and IoT services. We can imagine that many services that will offer access to IoT assets in future Smart Cities will require users to authenticate before being able to call their API. This is already the case today, with services like Airly, which require users to acquire an API key to pass to each call to enforce service throttling and ensure that the requesting users have the right permissions. With the rise in the number of IoT services and domains that will require authentication, managing permissions will grow increasingly complex for both users and service providers. Thanks to the combined use of MARGOT and Federation Services, it becomes possible to offload some of the complexity to the platform. For instance, they will be able to create separated federations of trusted MARGOT instances within which information can be shared securely without any external access. This would allow city officials, law Enforcement, or emergency response personnel to share access information to IoT infrastructure segments managed by the different administrations without limitations through a dedicated and secured MARGOT federation.

Finally, MARGOT can help during certain HADR situations by mitigating some of the effects of partially unavailable network infrastructures, e.g., due to damage or power failure. In these scenarios, it might become impossible for users to reach the servers of a provider, such as 511NY, but they may still have access to part of the edge and sensor networks. A distributed solution like MARGOT, which replicates data across federates, enhances the whole system's fault tolerance by leveraging redundant instances running in dispersed geographical locations. Therefore, clients are more likely to still have their queries resolved even when the network infrastructure is partially down because they can issue requests to remote MARGOT instances that were unaffected by the disaster. Once an instance receives a client request, even if the MARGOT with the responsible DA remains unreachable, it can still respond with the requested data via Federation, as long as at least one federate has cached those data in the past.

V. Experimental Results

We performed three different experiments to evaluate the effectiveness of a Federation of MARGOT in a multi-domain scenario; for an evaluation in a single domain context, the reader can refer to [9]. We conducted the experiments in an emulated network environment created using the Extensible Ad-hoc Networking Emulator (EMANE), which allows to control latency, bandwidth, and packet loss of the emulated network links. The scenario consists of 3 separated networks, which we will call Domain *A*, *B*, and *C*, respectively, connected via EMANE-controlled links. More specifically, the link between Domain *A* and Domain *B* presents a latency of 30 ms; Domain *A* is connected to Domain *C* via a 80 ms latency link; finally, the link that connects Domain *B* and Domain *C* has a latency of 100 ms.

Each domain has a node running MARGOT that can acquire information about IoT resources that the instances running in the other domains cannot directly obtain. To do so, we configured a DA in each MARGOT to interface with one of the three IoT services that we described in section II: NY511, Airly, and Digitraffic. In addition, each MARGOT is connected to a local IMSBridge that can federate with the other bridges via the EMANE-controlled links. As the results will show, MARGOT clients are able to connect to any MARGOT instance and retrieve information about IoT resources from any domain, as all MARGOT instances are federated.

In our scenario, three clients, which represent three rescue teams (Rescue Team 1 through Rescue Team 3), move from one domain to another and periodically query the local MARGOT instance to retrieve access information about devices that satisfy their interests. For simplicity reasons, each client

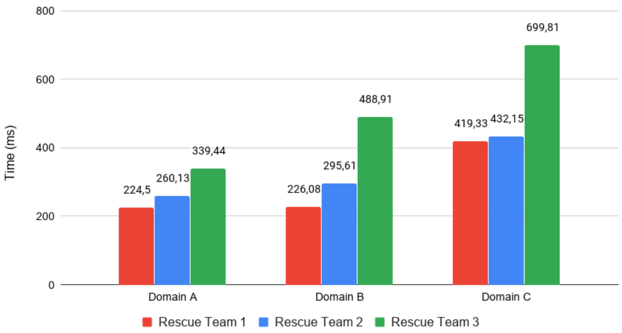


Fig. 3. Average Response Latency per Domain, MARGOT Caching Disabled

issue requests for the same subset of sensors each time: the first team generates requests for 10 sensors, the second team receives information about 50 sensors, and the third one about 250 sensors. We ran the first two experiments 100 times for each combination of Rescue Team and Domain, measuring the average response time from the client application. For the third experiment, we ran two tests of the duration of about one hour each, the first time with proactive query disabled and the second time after enabling that feature.

In the first experiment, we measured the average time that each rescue team in each domain had to wait to receive all the requested data without any use of caching within MARGOT; the goal was to measure the expected latency whenever new resources are queried for the first time or the cached entries are expired. As a consequence, for this experiment all MARGOT instances always forward the clients' queries to the other domains via Federation. Figure 3 shows the measured latency for each team receiving data from each domain. The first rescue team register the lowest latency since their requests involve less traffic, but the growth is less than linear (the number of assets queried increases five-fold when going from Rescue Team 1 to Rescue Team 2, and from Rescue Team 2 to Rescue Team 3, but the measured latency only increases by a fraction of that, with 65.4% being the highest increase), which suggests that the processing and forwarding of the queries has the largest impact on the total latency. Moreover, all teams present a higher average latency when connected to Domain *C*; this happens because the local MARGOT database contains a much larger number of entries compared to the other MARGOT instances, which increases the processing time.

The second experiment is similar to the first one, but in this case we primed the database of all MARGOT instances with the necessary data and then configured them to solve all clients' queries from the local database. The results, shown in Fig 4, are particularly relevant for scenarios in which clients request information about ”popular” resources, which would often be resolved from cache, and in cases where the requested resources are fairly static and, consequently, high cache validity timeouts have been set for those resources. As the Figure shows, caching allows the system to reduce response times by one order of magnitude. As the response latency decreases, now ranging between 10 and 20 milliseconds, other factors, such as delays in the communications caused by the TCP protocol, thread scheduling and context switches, memory access, and others, whose impact in the first experiment was negligible, now have a visible effect on the measured response times. After examining the collected data, we concluded that they are the cause of the increased delay experienced by Rescue Team 1 and 3 when connected to Domain A and B, respectively.

We designed the final experiment to show the effects of MARGOT's proactive querying on bandwidth consumption. This experiment only involves Domain *A* and Domain C: we instantiated two static clients in Domain *A* that send a total of 5 requests per minute to retrieve data on the same subset of IoT resources located across the two domains, for a total of about 150 sensors. We repeated the experiment twice: in the first run, proactive querying was disabled (reactive querying), whereas, in the second run, we changed the configuration of MARGOT in Domain A to enable it (proactive querying). In the second run, after about 10 minutes, the frequent requests for the same set of resources trigger proactive querying, which enables the MARGOT running in Domain A to send a special request to the MARGOT in Domain C, after which the receiving MARGOT will start to push proactively any update regarding any of the resources that have been requested frequently.

The results obtained are shown in Figure 5. The graph shows the amount of traffic generated by the two MARGOT instances every minute with (in red) and without (in blue) the proactive querying optimization enabled. After the first 10 minutes and until one of the clients starts making requests for different sensors (which happens about 40 minutes after the beginning of the test), the bandwidth consumption in the second run decreases to an average of 500 Kbits per minute against a baseline traffic of 2.2 Mbits per minute measured during the first run. This reduction is caused by two factors: first, the absence of query messages, which do not need to be forwarded between federates when proactive querying is active; and second, a reduction in the amount of data sent, since updates are limited to the resources that have changed state. Note that proactive querying also reduces the latency of requests to values very close to those shown in our second experiment, as all client requests for assets updated proactively will be resolved by MARGOT from the local database.

VI. Related Work

IoT resources discovery within Smart Cities is an active research topic. The authors of [14] performed a comprehensive survey of discovery technologies for IoT environments, analyzing and comparing solutions such as multicast DNS (mDNS), multicast CoAP, the Simple Service Discovery Protocol (SSDP), and others. In [15], the authors present the Smart and Power Efficient Node Discovery Protocol (SPEND), a reliable and energy-efficient discovery mechanism for IoT-Fog networking scenarios that leverages the MQTT protocol to keep track of Things, which act like publishers/advertisers in

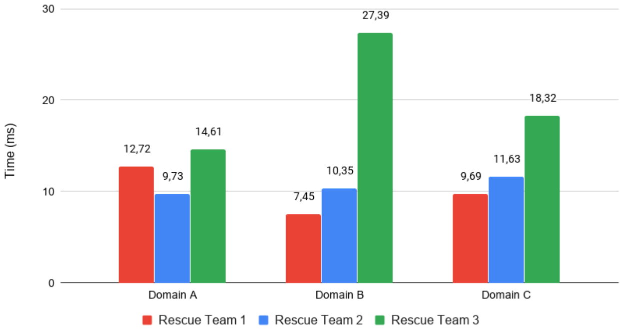


Fig. 4. Average Response Latency per Domain, MARGOT Caching Enabled

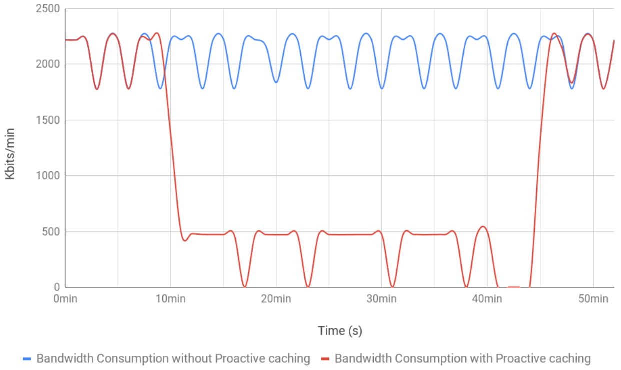


Fig. 5. Bandwidth Consumption: Reactive Querying vs. Proactive Querying

the network. Experimental results show the effectiveness and low power consumption of the protocol, thereby supporting the thesis that MQTT is a good choice for constrained devices. In another study, the authors evaluate a multi-domain, distributed discovery mechanism for the IoT that takes advantage of the CoRE Resource Directory (RD) and CoAP as the interface for discovering and accessing resources [16]. In the this solution, IoT Gateways are responsible for implementing the RD within the single IoT domain, while an approach based on a Distributed Hash Table (DHT) enables global discovery across multiple IoT domains.

A few works also evaluate the importance of resource discovery in the context of Information-centric Networking (ICN), often focusing on the network edge. The work in [17] discusses the possibility of adopting semantic matching techniques to perform IoT service discovery in ICN scenarios to increase the flexibility of the discovery processes. In [18], the authors propose the use of a service-response model to perform resource discovery at the network edge in Named Data Networking (NDN)-based scenarios. The proposed model naturally takes advantage of interest-based routing of NDN, which the authors extend by adding a deferral scheme to avoid collisions over the same service requests. The authors then extend their work in [19], proposing a solution to support resource discovery across NDN-based and IP-based networks. Their approach leverages mDNS to perform discovery inside the IP network and the Named Publish Subscribe Networking (NPSN) protocol within the NDN network; a gateway solution called Future Internet eXchange Point (FIXP) is used to bridge between the different networks and protocols.

MARGOT differs from the aforementioned solutions because it takes advantage of a distributed architecture of gateways that implement domain-specific discovery capabilities and support domain-specific security policies via Federation Services and ABE. Using Federation Services, MARGOT can forward client queries and IoT data over the Federation network based on sharing policies and replicate data to increase its availability, while ensuring that access is given only to authorized users. Furthermore, MARGOT supports pluggable DAs to simplify resource discovery and management in multi-domain scenarios, enabling the use of a wide range of open or closed, standard or non-standard discovery protocols.

VII. Conclusions and Future Work

In this paper, we analyzed the current status of IoT services and traditional solutions adopted to face the problem of IoT discovery in Smart Cities. Then, we discussed future directions of IoT and services, with a special focus on the pivotal role of effective IoT discovery solutions in HADR scenarios. Finally, we described the MARGOT platform and its features, a solution to manage the complexity of IoT asset discovery in multi-domain scenarios and maintain the availability of resource access information in HADR scenarios. Experimental results show that MARGOT can also effectively reduce latency in the IoT discovery process and lower bandwidth consumption under the considered use cases.

In the future, we will add the support for additional IoT services and discovery protocols to MARGOT. We are also planning to enhance the IP to enable more intelligent and effective optimization strategies. These will include a prediction system built upon collected statistics about users' requests and interests to further reduce response times for the users of the system, for instance by anticipating what types of sensors or geographical areas they will be interested in. Furthermore, we are considering further extending the security mechanisms within MARGOT. In particular, we are planning to implement security mechanisms to support different client authentication and authorization policies (owners, admins, guest, et cetera), in order to prevent resource access from ill-intentioned users, and also introduce an authentication system at the level of Federation Services, to avoid that malicious MARGOT instances can successfully federate and inject fake or purposely erroneous discovery data into the system.

References

[1] R. Morello, S. C. Mukhopadhyay, Z. Liu, D. Slomovitz, and S. R. Samantaray, "Advances on Sensing Technologies for Smart Cities and Power Grids: A Review," *IEEE Sensors Journal*, vol. 17, no. 23, pp. 7596–7610, Dec 2017. doi: 10.1109/JSEN.2017.2735539

[2] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, “A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications,” *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1125–1142, Oct 2017. doi: 10.1109/JIOT.2017.2683200

[3] T. Hui, R. Sherratt, and D. Díaz-Sánchez, “Major requirements for building smart homes in smart cities based on internet of things technologies,” *Future Generation Computer Systems*, vol. 76, 11 2016. doi: 10.1016/j.future.2016.10.026

[4] M. Noura, M. Atiquzzaman, and M. Gaedke, “Interoperability in internet of things: Taxonomies and open challenges,” *Mob. Netw. Appl.*, vol. 24, no. 3, p. 796–809, Jun. 2019. doi: 10.1007/s11036-018-1089-9

[5] P. Barnaghi and A. Sheth, “On searching the internet of things: Requirements and challenges,” *IEEE Intelligent Systems*, vol. 31, no. 6, pp. 71–75, Nov 2016. doi: 10.1109/MIS.2016.102

[6] R. Lea and M. Blackstock, “City hub: A cloud-based iot platform for smart cities,” in *2014 IEEE 6th International Conference on Cloud Computing Technology and Science*, Dec 2014. doi: 10.1109/CloudCom.2014.65. ISSN null pp. 799–804.

[7] A. Taherkordi and F. Eliassen, “Scalable modeling of cloud-based iot services for smart cities,” in *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, March 2016. doi: 10.1109/PERCOMW.2016.7457098. ISSN null pp. 1–6.

[8] H. Kim and E. A. Lee, “Authentication and authorization for the internet of things,” *IT Professional*, vol. 19, no. 5, pp. 27–33, 2017. doi: 10.1109/MITP.2017.3680960

[9] L. Campioni, R. Lenzi, F. Poltronieri, M. Pradhan, M. Tortonesi, C. Stefanelli, and N. Suri, “MARGOT: Dynamic IoT Resource Discovery for HADR Environments,” in *MILCOM 2019 - 2019 IEEE Military Communications Conference (MILCOM)*, Nov. 2019. doi: 10.1109/MILCOM47813.2019.9021092. ISSN 2155-7578 pp. 809–814.

[10] R. Lenzi, G. Benincasa, E. Casini, N. Suri, A. Morelli, S. Watson, and J. Nevitt, “Interconnecting Tactical Service-Oriented Infrastructures with Federation Services,” in *MILCOM 2013 - 2013 IEEE Military Communications Conference*, Nov 2013. doi: 10.1109/MILCOM.2013.123. ISSN 2155-7586 pp. 692–697.

[11] D. Evans, “The Internet of Things How the Next Evolution of the Internet Is Changing Everything,” CISCO, Tech. Rep., Apr 2011. [Online]. Available: https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf

[12] A. Pal, H. K. Rath, S. Shailendra, and A. Bhattacharyya, “Chapter 3 IoT Standardization : The Road Ahead,” 2018. doi: 10.5772/intechopen.75137

[13] F. Poltronieri, L. Campioni, R. Lenzi, A. Morelli, N. Suri, and M. Tortonesi, “Secure Multi-Domain Information Sharing in Tactical Networks,” in *MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM)*, Oct 2018. doi: 10.1109/MILCOM.2018.8599693. ISSN 2155-7578 pp. 1–6.

[14] A. Bröring, S. K. Datta, and C. Bonnet, “A Categorization of Discovery Technologies for the Internet of Things,” in *Proceedings of the 6th International Conference on the Internet of Things*, ser. IoT’16. New York, NY, USA: Association for Computing Machinery, 2016. doi: 10.1145/2991561.2991570. ISBN 9781450348140 p. 131–139.

[15] Venanzi *et al.*, “MQTT-Driven Sustainable Node Discovery for Internet of Things-Fog Environments,” in *2018 IEEE International Conference on Communications (ICC)*, May 2018. doi: 10.1109/ICC.2018.8422200. ISSN 1938-1883 pp. 1–6.

[16] G. Tanganelli, C. Vallati, and E. Mingozzi, “Edge-Centric Distributed Discovery and Access in the Internet of Things,” *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 425–438, Feb 2018. doi: 10.1109/JIOT.2017.2767381

[17] Quevedo *et al.*, “On the Application of Contextual IoT Service Discovery in Information Centric Networks,” *Comput. Commun.*, vol. 89, no. C, p. 117–127, Sep. 2016. doi: 10.1016/j.comcom.2016.03.011

[18] M. Amadeo, C. Campolo, and A. Molinaro, “NDNe: Enhancing Named Data Networking to Support Cloudification at the Edge,” *IEEE Communications Letters*, vol. 20, no. 11, pp. 2264–2267, Nov. 2016. doi: 10.1109/LCOMM.2016.2597850

[19] Quevedo *et al.*, “Internet of things discovery in interoperable information centric and IP networks,” *Internet Technology Letters*, Jul. 2017. doi: 10.1002/itl2.1

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 521–524 DOI: 10.15439/2020F143 ISSN 2300-5963 ACSIS, Vol. 21

Face Mask Detection at the Fog Computing Gateway

Srinivasa Raju Rudraraju
School of Computer and
Information Sciences
University of Hyderabad
Hyderabad, India
Email: r.srinivasaraju@gmail.com

Nagender Kumar Suryadevara
School of Computer and
Information Sciences
University of Hyderabad
Hyderabad, India
Email: nks@uohyd.ac.in

Atul Negi
School of Computer and
Information Sciences
University of Hyderabad
Hyderabad, India
Email: atul.negi@uohyd.ac.in

***Abstract*—This work proposes a fog computing-based face mask detection system for controlling the entry of a person into a facility. The proposed system uses fog nodes to process the video streams captured at various entrances into a facility. *Haar-cascade-classifiers* are used to detect face portions in the video frames. Each fog node deploys two MobileNet models, where the first model deals with the dichotomy between *mask* and *no mask* case. The second model deals with the dichotomy between *proper mask wear* and *improper mask wear* case and is applied only if the first model detects mask in the facial image. This two-level classification allows the entry of people into a facility, only if they wear the mask properly. The proposed system offers performance benefits such as improved response time and bandwidth consumption, as the processing of video stream is done locally at each fog gateway without relying on the Internet.**

I. Introduction

THE usage of face mask by the general public to impede the spread of the Corona Virus pandemic is highly essential. Wearing face mask limits the spread of virus through droplets, such as saliva or mucus [1]. Automatic entry and access control systems based on face mask detection are of immense help at several places such as workplaces, railway stations, shopping malls. These systems help in restricting the entry of persons not wearing a mask to a facility without manual intervention.

Fog Computing is a decentralized computing and storage infrastructure that brings processing closer to the data origin [2],[3]. It addresses the response time needs of real-time Internet of Things (IoT) applications, where Cloud-based IoT is not an ideal choice. Fog Computing disperses a portion of computation and storage from the cloud to devices at the network edge thereby improving response time, bandwidth consumption, data protection, and security of the IoT applications [4],[5].

The computation, storage, and bandwidth requirements of the face mask detection system increase with more number of such units being installed at several points in a facility. To address this problem, we proposed a fog computing-based face mask detection system for entry and access control. The Raspberry Pi (RPi) Camera attached to the RPi device is installed at each entrance of the facility to capture the video. The RPi fog node processes the video frames to track faces in the video stream. The *MobileNet* face mask detection models, deployed in each fog node, classify whether a person wears the mask properly or not using the facial images from the video frame, thereby controlling entry into the facility.

Our work attempts to meet the following objectives:

- A fog computing based face mask detection system to improve the response time and bandwidth consumption.
- Entry restriction to a facility based on the outcome of the face mask detection model.

II. Related Work

A. Entry and Access Control using Face Recognition

A. Nag *et al.* designed a face recognition based door access control in the IoT environment [6]. OpenCV functionality is used to detect and recognize faces of known people and thereby managing the door access automatically. P. Hu *et al.* proposed fog computing-based face identification and recognition scheme that tries to offload face recognition task from cloud to fog nodes [7].

B. Face Mask Detection Systems

Few works are developed to detect people wearing a face mask or not [8],[9]. These techniques have considered two classes of facial images for training the model: with mask and without the mask. The developed systems vary in terms of the framework and model chosen to build the model.

Our proposed system applies face mask detection models at two levels to deal with no mask, proper mask wear and improper mask wear cases separately for automatic entry and access control. The proposed system makes use of fog computing environment for the inference process to obtain performance benefits.

III. System Description

The proposed system employs a RPi fog node integrated with Raspberry Pi Camera and Relay Sensor at each entrance where entry control is required. The fog nodes are connected to the same Wi-Fi network. The basic architecture of the proposed system is shown in Fig. 1.

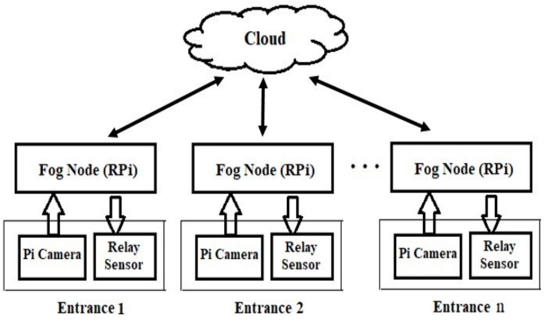


Fig. 1 The basic architecture of the proposed system

The frames in the video stream captured by Pi Camera are processed by the fog node. Whenever any face(s) is detected in the frame, the face mask detection model tries to identify whether the person(s) is wearing the mask or not. The RPi module sends a control signal to the relay to open the door, if the person wears the mask properly. The decision to open the door or not is taken completely on the fog node, and the event information can be sent to the Cloud optionally for further storage and processing. The various software used in the system design are explained below:

A. OpenCV

OpenCV (Open Source Computer Vision) is a library that has several built-in functions for performing computer vision and machine learning tasks [10]. The proposed system uses the *frontal face haar cascade* classifier from OpenCV to detect faces in the video frames.

B. Keras MobileNetV2

Keras is an open-source neural network (NN) library that provides a high-level API wrapper to TensorFlow [11],[12]. MobileNetV2 model is built in our experiment using Keras API, as it is a lightweight convolutional NN that reduces the inference cost on mobile and embedded devices [13],[14].

C. Basic Operation of the Proposed System

The proposed system employs two MobileNetV2 models for classifying whether a person is wearing the mask properly or not. The first model is a binary classifier that is trained using two classes of images – ***mask*** and ***no mask***. The ***mask*** class contains facial images of persons wearing a mask (includes both proper and improper mask wear). The second model is a binary classifier that is trained using ***proper mask wear*** and ***improper mask wear*** images. Mask is said to be properly put on if nostrils and mouth are covered. If nostril or mouth is detected even when the person wears a mask, the instance is classified as improper mask wear and entry should be restricted. In our experimental setup, these two models are trained on a single RPi fog node and deployed in all the fog gateways for inference purposes. The rationale behind choosing two binary classifiers instead of a single three-class classifier is to improve the classification accuracy, especially between *proper* and *improper mask wear* classes. The choice provides a tradeoff between classification accuracy and throughput.

Each fog node processes the video frames captured by Pi Camera and uses *Frontal Face Haar Cascade* classifier to detect the faces in those frames. When one or more faces are detected in a frame, level one binary classifier (***mask*** vs. ***no mask***) is applied on each face region of interest (ROI) in the video frame. If the person wears the mask, then level two classifier (***proper*** vs. ***improper mask wear***) is applied to identify whether the person is wearing the mask properly or not. This two-level classification restricts the entry of people with improper mask wear into the facility. The basic operation of the proposed system is shown in Fig. 2.

IV. Implementation Details

The system provides a proof-of-concept for face mask detection using fog computing gateway. The processing of the video frames is done at the source of video capture. In reality, the frames in the video stream could be sent and processed using the resources in the fog gateway.

A. Face Mask Detection Model Training

Datasets used for Model Training: As discussed in Section III, the proposed system uses two binary classifiers based on the MobileNetV2 model. Classifier-1 (model-1) is trained using dataset-1 with a total of 770 facial images divided into two classes: *with mask* and *without mask*. The "*with mask*" class included images of faces with and without proper face mask wear. Classifier-2 is built using dataset-2 with a total of 500 facial images divided into two classes: *proper mask wear* and *improper mask wear*. The average size of training images is around 5KB (Image dimensions 250x160 with 96 dpi). Fig 3 and Fig 4 show a sample set of facial images from dataset-1 and dataset-2 respectively. Even though there are few face mask datasets available online [15], we have prepared our own dataset as the existing face mask datasets lack images for *improper mask wear* class.

Training the mask detection models: The various steps followed to train the mask classification models are given below:

Step 1. Load the images from the dataset using *load_img()* function from Keras API. The loaded images are resized to 224x224 format.

Step 2. The loaded images are normalized using *preprocess_input()* function. The data (facial image) and label lists are updated with images in the dataset.

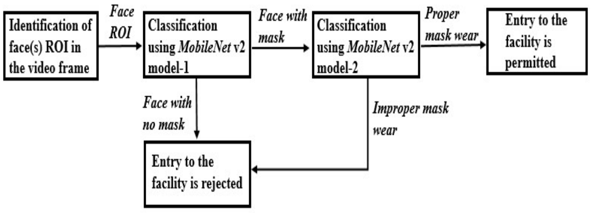

Identification of face(s) ROI in the video frame
Face ROI
Classification using MobileNet v2 model-1
Face with mask
Classification using MobileNet v2 model-2
Proper mask wear
Entry to the facility is permitted
Face with no mask
Improper mask wear
Entry to the facility is rejected

Fig. 2 The basic operation of the proposed system

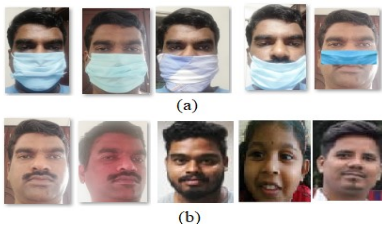


Fig. 3 A sample set of facial images used for model-1training (a) with mask (b) without the mask

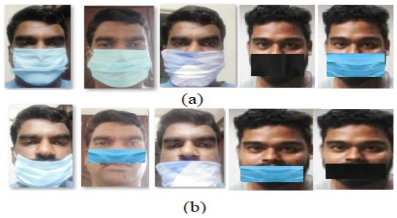


Fig. 4 A sample set of facial images used for model-2 training (a) with proper mask wear (b) without proper mask wear

Step 3. Convert the data and labels into *numpy* arrays. One-hot encoding is performed on the labels to represent them as binary vectors.

Step 4. The dataset is partitioned into training (80%) and testing (20%) sets using *train_test_split()* function.

Step 5. Instantiate MobileNetV2 model trained with ImageNet dataset. Load the model that doesn't include the classification layers at the top. Transfer learning is used in our experimental setup to transfer knowledge from the ImageNet dataset domain to our Facial dataset domain [16].

Step 6. The weights of all the layers in the convolutional base are frozen to prevent updates during training. We added the classifier on top of this base model and trained the top-level classifier.

Step 7. The model is compiled using *Adam optimizer* and *binary cross-entropy* loss function.

Step 8. The model is trained and serialized to disk for the usage during the inference process.

B. Classification of Facial Images

RPi 4 device integrated with Pi Camera is used as a fog node at each entrance to capture the video stream. The Pi Camera has 5MP resolution and can record 1080p videos at 30 frames per second. The face mask detection models trained in the previous step are loaded in each fog node for inference. The following are the various steps performed during the inference process on each video frame:

Step 1. Identification of face ROI in the frame using *frontal face haar cascade* classifier from OpenCV.

Step 2. If more than one face is detected in the frame, for each face do the following:

- Resize the face image to 224x224 RGB image.
- Convert the image into ***numpy*** array and preprocess it.
- Predict the output class (mask vs. no mask) of the image using the MobileNetV2 model-1 loaded in the node.
- If the prediction class on the image is ***mask***, then model-2 is used for further classification.
- If the prediction class by model-2 is "proper mask wear", then fog node controls the relay to open the door. Otherwise, entry is restricted.

Fig. 5 shows the screenshots of the outcome of the face mask detection model for proper and improper mask wear cases.

V. Results and Discussion

The face mask detection models (model-1 and model-2) are trained using different learning rates (LR) 0.001 and 0.0001 with two different numbers of epochs 10 and 20. The model-1 and model-2 training tasks have taken 34 minutes and 20 minutes respectively (with LR = 0.001 and #Epochs = 20) on RPi 4. The face detection and inference tasks for the given video frame have taken 0.3 seconds and 3.4 seconds respectively on average. Table I and Table II present the accuracy and loss values during training and validation phases of model-1 and model-2 respectively (with LR = 0.001 and #Epochs = 20).

From the results in Table I and Table II, we can observe that training, validation accuracy and loss values are improving during the models' training. After few number of epochs (epoch #15 for model-1 training, and epoch #10 for model-2 training approximately), we get fluctuations in the accuracy and loss values. The variations in the accuracy and loss values are due to *model overfitting* with more number of epochs. Overfitting of model could happen with the selection of small dataset for training the model. As the RPi device is resource-constrained, we have chosen small datasets of facial images for model training. One solution to address this problem is *early stopping* using *callback* mechanism.

While training model-1 on the Raspberry Pi node using dataset with 770 images, a warning message is received with respect to allocation of memory exceeding 10% of system memory. If we want to train the model with a large dataset to avoid model overfitting, we can train it on a high-end machine and deploy the model on the fog node for inference. Alternatively, we can distribute the model training to several nodes in the fog cluster. Fig. 6 presents performance metrics related to model-1 and model-2 training.

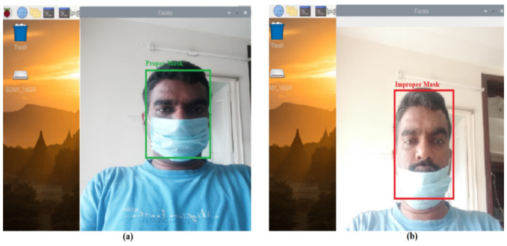
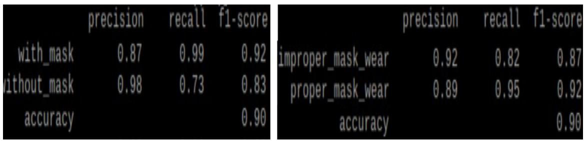


Fig. 5 Face mask detection model prediction when (a) the person wears the mask properly (b) the person wears the mask improperly



```
              precision    recall  f1-score
   with_mask       0.87      0.99      0.92
ithout_mask       0.98      0.73      0.83
    accuracy                           0.90
```

```
                   precision    recall  f1-score
improper_mask_wear       0.92      0.82      0.87
  proper_mask_wear       0.89      0.95      0.92
          accuracy                           0.90
```

Fig. 6 Performance metrics related to model-1 and model-2 training

TABLE I.
ACCURACY AND LOSS VALUES DURING MODEL-1 TRAINING

Epoch	Train_Loss	Train_Acc	Val_Loss	Val_Acc
1	0.64	0.69	0.49	0.71
2	0.39	0.82	0.39	0.82
3	0.36	0.88	0.48	0.74
4	0.35	0.88	0.47	0.74
5	0.32	0.89	0.45	0.75
6	0.24	0.9	0.47	0.74
7	0.22	0.91	0.46	0.78
8	0.21	0.92	0.44	0.79
9	0.21	0.91	0.43	0.79
10	0.19	0.93	0.44	0.78
11	0.19	0.92	0.41	0.8
12	0.18	0.93	0.34	0.82
13	0.17	0.94	0.4	0.78
14	0.17	0.93	0.35	0.83
15	0.17	0.94	0.42	0.81
16	0.18	0.93	0.47	0.77
17	0.17	0.93	0.56	0.75
18	0.16	0.94	0.57	0.74
19	0.17	0.93	0.54	0.76
20	0.16	0.94	0.55	0.75

TABLE II.
ACCURACY AND LOSS VALUES DURING MODEL-2 TRAINING

Epoch	Train_Loss	Train_Acc	Val_Loss	Val_Acc
1	0.71	0.66	0.44	0.77
2	0.5	0.72	0.34	0.79
3	0.42	0.79	0.32	0.8
4	0.39	0.8	0.28	0.85
5	0.35	0.88	0.29	0.84
6	0.32	0.89	0.29	0.9
7	0.22	0.91	0.27	0.88
8	0.22	0.9	0.26	0.9
9	0.2	0.92	0.24	0.91
10	0.29	0.85	0.25	0.9
11	0.28	0.89	0.28	0.88
12	0.25	0.9	0.29	0.88
13	0.24	0.92	0.3	0.85
14	0.25	0.9	0.32	0.84
15	0.22	0.92	0.35	0.82
16	0.21	0.92	0.38	0.78
17	0.22	0.93	0.39	0.76
18	0.21	0.91	0.38	0.77
19	0.19	0.92	0.37	0.8
20	0.19	0.92	0.38	0.78

VI. CONCLUSION AND FUTURE WORK

This work presented a proof-of-concept fog computing-based face mask detection system for automatic entry and access control into a facility. The fog gateway processes the video stream captured at the entrance to recognize whether a person is wearing a mask or not. Two *MobileNet* models are deployed in each fog node located at each entrance to the facility. The first model deals with the dichotomy between mask and no mask case. The fog node uses the second model to detect whether the person wears the mask properly or not, in case the first model detects the person wearing the mask. The proposed system allows entry into the facility, only if the person wears the mask properly. The results of the classification are encouraging with model accuracy value around 90%.

As the processing of video stream is done locally at each fog node, the proposed system offers performance benefits such as improved response time and bandwidth consumption. The proposed system could be integrated with Cloud, optionally, for further storage and processing of video stream. In the future, the system could be extended to distribute the model training among several fog nodes in the network.

REFERENCES

[1] J. Howard, A. Huang, Z. Li, Z. Tufekci, V. Zdimal, H. van der Westhuizen *et al.*, "Face Masks Against COVID-19: An Evidence Review", Preprints 2020, 2020040203, http://dx.doi.org/10.20944/preprints202004.0203.v1

[2] S. Sarkar, R. Wankar, S. Srirama and N.K Suryadevra, "Serverless Management of Sensing Systems for Fog Computing Framework", IEEE Sensors Journal, ISSN: 1530-437X, 20(3):1564-1572, 2020, http://dx.doi.org/10.1109/JSEN.2019.2939182

[3] S. R. Rudraraju, N. K. Suryadevara and A. Negi, "Face Recognition in the Fog Cluster Computing," IEEE International Conference on Signal Processing, Information, Communication & Systems, 2019, pp. 45-48, http://dx.doi.org/10.1109/SPICSCON48833.2019.9065100

[4] N. N. Khan, "Fog computing: A better solution for IoT," International Journal of Engineering and Technical Research., vol. 3, no. 2, pp. 298–300, 2015, ISSN: 2321-0869

[5] M. Aazam, S. Zeadally, and K. A. Harrass, "Fog Computing Architecture, Evaluation, and Future Research Directions", IEEE Communications Magazine (2018), pp. 46-52

[6] A. Nag, J. N. Nikhilendra, and M. Kalmath, "IOT Based Door Access Control Using Face Recognition", International Conference for Convergence in Tech., http://dx.doi.org/10.1109/i2ct.2018.8529749

[7] P. Hu, H. Ning, T. Qiu, Y. Zhang, and X. Luo, "Fog Computing Based Face Identification and Resolution Scheme in Internet of Things", IEEE Transactions on Industrial Informatics, 13(4), 1910–1920, 2017, http://dx.doi.org/10.1109/tii.2016.2607178

[8] COVID-19: Face Mask Detector, Last accessed 05 Jun 2020, URL:https://www.pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/

[9] Face Mask Detection, Last accessed 10 Jun 2020, URL: https://www.towardsdatascience.com/covid-19-face-mask-detection-using-tensorflow-and-opencv-702dd833515b

[10] OpenCV, Last accessed 19 May 2020, URL: https://opencv.org/ about

[11] Keras Guide, Last accessed 9 May 2020, URL: https://keras.io/guides/

[12] TensorFlow Tutorials, Last accessed 10 Jun 2020, URL: https://www.tensorflow.org/tutorials

[13] MobileNet, Last accessed 20 May 2020, URL: https://www.tensorflow.org/api_docs/python/tf/keras/applications/mobilenet

[14] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand *et al.*, "Mobilenets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv:1704.04861, 2017

[15] Face Mask Dataset, Last accessed 21 Jun 2020, URL: https://github.com/X-zhangyang/Real-World-Masked-Face-Dataset

[16] Transfer Learning, Last accessed 21 May 2020, URL: https://www.tensorflow.org/tutorials/images/transfer_learning

1^{st} International Forum on Cyber Security, Privacy and Trust

NOWADAYS, information security works as a backbone for protecting both user data and electronic transactions. Protecting communications and data infrastructures of an increasingly inter-connected world have become vital nowadays. Security has emerged as an important scientific discipline whose many multifaceted complexities deserve the attention and synergy of computer science, engineering, and information systems communities. Information security has some well-founded technical research directions which encompass access level (user authentication and authorization), protocol security, software security, and data cryptography. Moreover, some other emerging topics related to organizational security aspects have appeared beyond the long-standing research directions.

The International Forum of Cyber Security, Privacy, and Trust (NEMESIS'20) as a successor of International Conference on Cyber Security, Privacy, and Trust (INSERT'19) focuses on the diversity of the cyber information security developments and deployments in order to highlight the most recent challenges and report the most recent researches. The session is an umbrella for all cyber security technical aspects, user privacy techniques, and trust. In addition, it goes beyond the technicalities and covers some emerging topics like social and organizational security research directions. NEMESIS'20 serves as a forum of presentation of theoretical, applied research papers, case studies, implementation experiences as well as work-in-progress results in cyber security. NEMESIS'20 is intended to attract researchers and practitioners from academia and industry and provides an international discussion forum in order to share their experiences and their ideas concerning emerging aspects in information security met in different application domains. This opens doors for highlighting unknown research directions and tackling modern research challenges. The objectives of the NEMESIS'20 can be summarized as follows:

- To review and conclude research findings in cyber security and other security domains, focused on the protection of different kinds of assets and processes, and to identify approaches that may be useful in the application domains of information security.
- To find synergy between different approaches, allowing elaborating integrated security solutions, e.g. integrate different risk-based management systems.
- To exchange security-related knowledge and experience between experts to improve existing methods and tools and adopt them to new application areas

Topics

- Biometric technologies
- Cryptography and cryptanalysis
- Critical infrastructure protection
- Security of wireless sensor networks
- Hardware-oriented information security
- Organization- related information security
- Social engineering and human aspects in cyber security
- Individuals identification and privacy protection methods
- Pedagogical approaches for information security education
- Information security and business continuity management
- Tools supporting security management and development
- Decision support systems for information security
- Trust in emerging technologies and applications
- Digital right management and data protection
- Threats and countermeasures for cybercrimes
- Ethical challenges in user privacy and trust
- Cyber and physical security infrastructures
- Risk assessment and management
- Steganography and watermarking
- Digital forensics and crime science
- Security knowledge management
- Security of cyber-physical systems
- Privacy enhancing technologies
- Trust and reputation models
- Misuse and intrusion detection
- Data hide and watermarking
- Cloud and big data security
- Computer network security
- Assurance methods
- Security statistics

Technical Session Chairs

- **Awad, Ali Ismail,** Luleå University of Technology, Sweden
- **Bialas, Andrzej,** Research Network Łukasiewicz – Institute of Innovative Technologies EMAG, Poland

Program Committee

- **Banach, Richard,** University of Manchester, United Kingdom
- **Bun, Rostyslav,** Lviv Polytechnic National University, Ukraine
- **Clarke, Nathan,** Plymouth University, United Kingdom
- **Cyra, Lukasz,** DM/OICT/RMS (UN)

- **Daszczuk, Wiktor Bohdan,** Warsaw University of Technology, Poland
- **Felkner, Anna,** Research and Academic Computer Network NASK
- **Furnell, Steven,** Plymouth University, United Kingdom
- **Furtak, Janusz,** Military University of Technology, Poland
- **Gawkowski, Piotr,** Institute of Computer Science, Warsaw University of Technology, Poland
- **Grzenda, Maciej,** Orange Labs Poland and Warsaw University of Technology, Poland
- **Hämmerli, Bernhard M.,** Hochschule für Technik+Architektur (HTA), Switzerland
- **Hasssaballah, M.,** South Valley University, Egypt
- **Kapczynski, Adrian,** Silesian University of Technology, Poland
- **Krendelev, Sergey,** Novosibirsk State University,JetBrains research, Russia
- **MD Faisal, Mohammad,** Integral University, India
- **MD Rafiqul, Islam,** School of Computing and Mathematics|Charles Sturt University
- **Misztal, Michal,** Military University of Technology, Poland
- **Pańkowska, Małgorzata,** University of Economics in Katowice, Poland
- **Rot, Artur,** Wroclaw University of Economics, Poland
- **Stokłosa, Janusz,** WSB University in Poznan, Poland
- **Suski, Zbigniew,** Military University of Technology, Poland
- **Szmit, Maciej,** University of Lodz, Poland
- **Wahid, Khan Ferdous,** Airbus, Germany
- **Yahya, Eslam,** Ohio State University, Columbus
- **Zamojski, Wojciech,** Wrocław University of Technology
- **Zieliński, Zbigniew,** Military University of Technology, Poland

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 527–536 DOI: 10.15439/2020F21 ISSN 2300-5963 ACSIS, Vol. 21

Developing Defense Strategies from Attack Probability Trees in Software Risk Assessment

Marko Esche
Physikalisch-Technische Bundesanstalt,
Abbestraße 2-12, 10587 Berlin, Germany
Email: marko.esche@ptb.de

Federico Grasso Toro
Federal Institute of Metrology METAS,
Lindenweg 50, 3003 Bern-Wabern, Switzerland
Email: federico.grasso@metas.ch

***Abstract*—Since the introduction of the Measuring Instruments Directive 2014/32/EU, prototypes of measuring instruments subject to legal control in the European Union must be accompanied by a risk assessment, when being submitted for conformity assessment. Taximeters, water meters, electricity meters or fuel pumps form the basis for the economic sector usually known as Legal Metrology, where the development towards cheaper all-purpose hardware combined with more sophisticated software is imminent. Therefore, a risk assessment will always have to include software-related issues. Hitherto, publications about software risk assessment methods lack an efficient means to derive and assess suitable countermeasures for risk mitigation. To this end, attack trees are used in related research fields. In this paper, defense probability trees are derived from attack probability trees, well-suited to the requirements of software risk assessment and used to identify optimal sets of countermeasures. The infamous Meltdown vulnerability is used to highlight the experimental application of the method.**

I. Introduction

LEGAL Metrology covers measurements and measuring instruments used as a basis for economic transactions. In this paper, a method to assess the risks associated with these instruments is used to derive suitable countermeasures to mitigate identified risks. The Measuring Instruments Directive (MID) [1], whose aim is to establish trust in measurements for users and customers of such instruments, lays down the basic procedures for putting measuring instruments on the market. Trust in measurements plays a significant role, since the Legal Metrology sector generates an annual turnover of around 500 billion Euros for the European Union market [2].

As a first step, conformity to so-called essential requirements, laid down by Annex 2 of the MID, shall be demonstrated by the manufacturer with the help of a conformity assessment procedure. These essential requirements encompass physical properties of the measuring instrument, such as climatic operating conditions and electromagnetic compatibility testing, as well as information technology requirements on software and data protection.[1]

Over the past decade, it has become apparent that measuring instruments are going through a transformation process from simple stand-alone devices with integrated hardware and software to complex distributed systems, which use cheaper, less sophisticated hardware with more complex software. Therefore, the risk assessment to be supplied by manufacturers for conformity assessment of a prototype [1] has to specifically address the risks related to software, as well as its inherent hardware risks.

As an aid for manufacturers, PTB has published a method for software risk assessment, specifically tailored for the use in Legal Metrology [2] based on ISO/IEC 27005 [3] and ISO/IEC 18045 [4]. With the help of generic assets to be protected, derived from the MID, the method allows objective comparisons of different instruments produced by different manufacturers. Following definitions from ISO/IEC 27005, the method calculates risk as the combination of the impact produced by the realization of certain threats to assets and of the probability of occurrence of each threat. The technical steps to realize a threat are normally summarized in so-called attack vectors. An attempt at providing standardization to derive attack vectors is described in [5]. The approach is based on the attack tree concept used in related fields of research, such as the design of cryptographic protocols and access control [6].

In the present paper the attack probability trees (AtPTs) presented in [5] are explained in detail and extended towards defense probability trees (DePTs), as a formal method to derive optimal countermeasures to be used for risk mitigation. After this brief introduction, the remaining paper is structured as follows: In Section II, a literature overview is presented, which sketches a brief history of attack trees, explaining their basic functionality and applications. Section III provides a summary of the original method and recapitulates the concept and the applicability of AtPTs. Then, Section IV addresses how to find optimal sets of countermeasures, by means of DePTs derived from AtPTs. An experimental application of the method, focused on the Meltdown vulnerability [7] and its implications for Legal Metrology, is presented in Section V. Finally, Section VI provides a summary of the paper and potential further work.

II. Literature Overview

According to the international standard ISO/IEC 27005 [3] the risk assessment process can formally be divided into three

[1] One conformity assessment body within the European Union is the *Physikalisch-Technische Bundesanstalt* (PTB), Germany's national metrology institute. One conformity assessment body that follows the MID as a consequence of the bilateral agreements between Switzerland and the European Union, is the Federal Institute of Metrology METAS.

IEEE Catalog Number: CFP2085N-ART ©2020, PTI

phases: 1) The risk identification phase: It normally starts with the definition of certain assets to be protected. A derivation of assets applicable to Legal Metrology was presented in [2] and a short example is supplied in Section III. 2) The risk analysis phase: Based on the identified assets, threats are formulated, constituting an invalidation of a specific security property of an asset by an attacker with an associated impact. The technical steps needed to realize a threat, commonly referred to as attack vectors, are also identified during this phase. 3) The evaluation phase: The final stage of the risk assessment is the evaluation of the risk associated with a threat. The evaluator decides whether the estimated risk is tolerable or if countermeasures need to be implemented.

The following example from IT security clarifies these three phases: If a cryptographic key on a computer must be protected against retrieval by an attacker, during the first phase (risk identification), the key becomes one asset to be protected with confidentiality, as the associated security property. One way to protect such a key would be the read/write permissions of the operating system, which - under normal conditions - can only be changed by using the administrator's credentials. During the second phase (risk analysis), the threat can be formulated: ”By guessing the correct administrator password - an action which corresponds to one possible attack vector - an attacker would be able to modify the read/write permissions for the cryptographic key and retrieve it without being detected.“ Among other things, the likelihood of said attack vector would then depend on the window of opportunity to access the computer and on the strength of the chosen password. In the final phase, sc. risk evaluation, the calculated risks are prioritized and a cut off point for the risk assessment is defined. For all unacceptably high risks, countermeasures are then selected and implemented to repeat the assessment process until all risks are classified as tolerable. The theoretical framework for the selection of countermeasures is described in Section III and illustrative examples can be found in Section V.

This paper focuses on the efficient graphical and logical representation of these vectors and on the identification and selection of its countermeasures. Originally, attack trees were used by human evaluators to illustrate and identify vulnerabilities of a known system by graphical means. However, these trees also have a number of mathematical properties that make them well-suited for automatic processing and evaluation. The following sub-sections provide the foundations of attack trees and their applications on risk assessment of software.

A. Foundations of Attack Trees

In [6], a detailed general introduction to attack trees and their potential applications is given. According to Mauw and Oostdijk, the most basic properties of any attack tree can be summarized as follows: While the root node of such a tree constitutes an attacker's main goal, its child nodes can be seen as refinements thereof, which need to be achieved in order to reach said goal. Following this interpretation, the leaves of an attack tree constitute atomic attacks, for which no further refinement is possible. An exemplary tree that only consists of seven nodes is given in Figure 1.

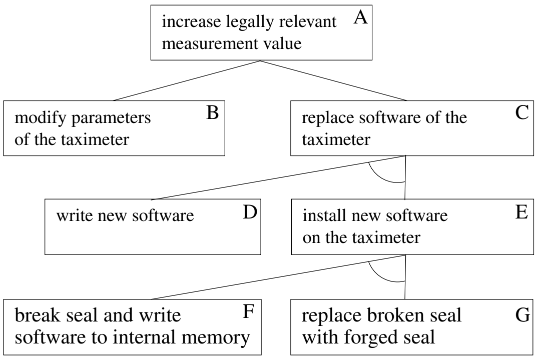

increase legally relevant measurement value
A
modify parameters of the taximeter
B
replace software of the taximeter
C
write new software
D
install new software on the taximeter
E
break seal and write software to internal memory
F
replace broken seal with forged seal
G

Fig. 1. Simple illustration of an attack tree that shows how the calculated fare/measurement value of a taximeter may be manipulated.

In the example, an attacker's possible strategies to manipulate the fare calculated by a taximeter are illustrated. Before exploring the meaning of the shown tree, it is necessary to explain the specifics of its graphical representation: Child nodes are always logically connected to either form an AND- or an OR-expression. The AND-statement is illustrated by an arc connecting the respective child nodes and indicates that all of these need to be implemented to achieve the attack associated with the parent node. On the other hand, if child nodes represent alternative ways to reach the parent objective, they are connected via an OR-statement, in which case no arc is drawn. Mauw and Oostdijk [6] interpret the two possible connections between nodes as 'conjunctive aggregation' and 'disjunctive refinement (choice)'. Certainly, there is no guarantee that an attack tree will be a binary tree. However, if more than two child nodes are identified, they can always be transformed into a binary structure by combining pairs of them into subgoals until only two child nodes remain. The examplary attack tree given in Figure 1 states that the fare can either be manipulated (node A) by changing the parameters of the taximeter (node B) or by replacing its software (node C). Other possible alternatives are not discussed here, since this example simply aims to illustrate the operation modes of attack trees. Therefore, at the root node a simple OR-connection can be found. Consequently, replacing the software requires at least two steps: 1) writing a new software (node D) and 2) installing it on the instrument (node E). Both steps need to be implemented for the attack to be executed successfully, which is expressed by an AND-statement, see Figure 1. It should be noted that AND-connected attacks do not necessarily need to be implemented simultaneously, as is the case in the illustrated scenario. The process of installing the software can once more be sub-devided into two AND-connected tasks: 1) opening the taximeter (node F) to write new software to its memory and 2) forging a new seal afterwards (node G). It can be seen that the combination of two nodes into a summary node has no influence on the mathematical properties of the local sub-tree, such as likelihood of occurrence [6]. Therefore, it is the evaluator's choice to limit the number of refinements of an attack as she sees fit. In practice, a node needs no further refinement if the associated attack constitutes a simple technical task with a known scope and easily determinable properties.

In addition to providing basic means of describing and manipulating attack trees, Mauw and Oostdijk introduce the concept of predefined characteristics of a node, e.g. possibility, cost, equipment needed. They show that the attributes of any parent node can be determined by combining the information associated with the respective child nodes. They highlight the point that these rules will be specific to the application, although they can usually be determined directly from the attributes and their properties. These rules work directly from the leaves towards the root without having to resort to trace-backs. After the application of these rules, the result is a set of attribute values, either of the root node itself or of a chosen sub-tree. In the latter case, the sub-tree may either reflect a set of promising attacks or contain information, such as its internal structure, not directly represented by the values of its attributes. It is important to note that there is no requirement for any node to only exist once within a tree [6]. Instead, nodes may have multiple copies whose attributes are linked; therefore, a change in one part of an attack tree can also affect otherwise unconnected branches. Mauw and Oostdijk also introduce the attack suite as a set of attacks that can be used to realize a goal without having to address the logical structure of the summarized sub-trees and their nodes. They show that attack trees with different structural representations may, nevertheless, contain the same logical information.

Concerning attack tree transformations, Mauw and Oostdijk formulate two rules: 1) 'associativity of conjunction' and 2) distributivity of 'conjunction over disjunction'. 'Associativity of conjunction' basically states that any sub-tree can be moved to its parent node if the root of the sub-tree is the only child node of the parent. In this case, parent node and child are logically identical. The second rule, which addresses the distributivity of 'conjunction over disjunction', states that any node with two separate sub-trees corresponds to two copies of the same node with one sub-tree each. Mauw and Oostdijk [6] provide proofs for both transformation rules.

The following additional remarks by Mauw and Oostdijk on the attributes of attack trees will be used later in Section III: The value of an attribute can only be determined if the semantics of the tree are known. With this knowledge the value of the attribute can be estimated from the properties of the equivalent attack suite. Additionally, the authors state that the attributes of a node can only depend on the attributes of its child nodes alone.

B. Threat Risk Analysis for Cloud Security based on Attack-Defense Trees

The attack trees discussed so far only focus on strategies an attacker might follow. In [8], Wang, Lin, Kuo, Lin and Wang introduce a new derivative of such trees that also offers the possibility to model defensive strategies. These Attack-Defense Trees (ADT) are specifically tailored to describe the attack profile and calculate the associated attack cost. Both can then be used to choose appropriate countermeasures even for complex attacks. This process is usually referred to as Threat Risk Analysis (TRA) which takes vulnerability information and attack profiles as input. Within the scope of a TRA, there are both the estimation of the impact of a successful attack and a precise description of attack progression. The combination of both allows the user of the method to develop adequate contermeasures/defense strategies. It should be observed that, when trying to describe all possible attack paths at a desired level of detail, attack trees quickly become complex and difficult to handle. Wang, Lin, Kuo, Lin and Wang therefore claim that expressing both attack and defensive strategies in the same tree is usually too complex and beyond the scope of the tree. In contrast to attack trees, which are used to model systemic weaknesses, protection trees migrate weaknesses and thus have the potential to help to identify protective strategies. Section III gives more theoretical background on this, while Section V provides an example of weakness migration and defense strategy selection. As explained in [8], it was usually assumed that an attacker has complete information about the system to attack and would always select the easiest available attack path. In reality, the attacker may, however, act upon an incomplete set of facts, which will affect her ability to select the easiest route. This needs to be taken into account when choosing appropriate countermeasures. Wang, Lin, Kuo, Lin and Wang then supply equations for estimating probability of occurence and other metrics for both AND- and OR-connections between attack nodes. Since the ADT is supposed to be used for both attack and defense modeling these metrics do not only include probability of success, attack cost and impact, but also revised attack cost and revised impact for the countermeasure stage, adding a forth and final phase (risk mitigation) to the previously described risk assessment process. Similar metrics are used in Section III. Once again, all metrics are estimated for the leaf nodes first and then propagated up the tree towards the root node. Whenever assessing risks of a new system, the first step consists of identifying possible vulnerabilities. To this end, public databases are one important source of information. The second step of an assessment is the collection of information on recognized attacks. This includes identification of attack vectors or, more generally, of means to realize a threat by exploiting a known vulnerability. In a third step, an ADT is built, including all identified vulnerabilities which could facilitate an attack. To construct the tree and fill it with information, rules for the transfer of the metrics mentioned above are postulated. The fourth step then consists of the systematic evaluation of the ADT. Since data on past incidents may not always be available, the calculated probability of occurrence is always affected by a level of uncertainty. Nevertheless, it is stated that there is a deterministic transfer function between attack cost and defense cost, influenced by security policies, procedures, equipment and training of personnel. The final step for the ADTs is to establish adequate countermeasures for each attack. Despite the general applicability of their concept, Wang, Lin, Kuo, Lin and Wang state that the final step would need to be individually tailored for each new scenario, since the countermeasures solely depend on the available vulnerabilities.

C. Automated Generation of Attack Trees

Mauw and Oostdijk [6] showed that several different graphical representations may exist for one logical attack tree. Since the layout of the tree itself is thus subject to the evaluator's decisions, an automated approach could ensure that all attack trees adhere to the same design principles. One such approach was presented by Vigo, Nielson and Nielson [9]. They overcome the stated problem by inferring attack trees from the process logic otherwise used to describe them. According to Vigo, Nielson and Nielson both the scientific community and the wider public in general profit from attack trees since they are easily quantifiable as well as easily comprehensible. In the described implementation, the root node again reflects a threat to be implemented and child nodes describe sub-goals to be combined to realize the threat. To build the attack tree the attack process is first translated into propositional formulae from which the tree is then automatically inferred. The result subsequently does not suffer from human interpretation errors and is thus easily reproducible. Once the attack tree has been derived, the leaves, which are interpreted as atomic attacks, are labeled with individual costs that propagate up the tree. In line with the process-oriented approach, attack and defense actions are seen as communication processes from signal theory and are treated accordingly. From these, the cheapest set of atomic attacks is selected, corresponding to the most likely attack path, and the resulting attack cost is calculated. This idea is reused in Section IV to derive optimal defense strategies.

III. Software Risk Assessment in Legal Metrology

The software risk assessment method, used here to construct a method for countermeasure identification and selection, was originally published in [2]. The method is currently used by both examiners and manufacturers during conformity assessment of measuring instruments within the EU. A number of modifications and additions, to this method have been published [5]. In this section these publications are quickly summarized to lay the groundwork for the countermeasure derivation introduced in Section IV.

A. Fundamentals of Risk Assessment Method

In line with the formal requirements and definitions for risk assessment laid down in the international standard ISO/IEC 27005 [3], risk can be defined as a 'combination of impact and probability of occurrence attributed to a threat'. A threat is seen as 'an adverse action carried out by an attacker/threat agent upon the security properties of an asset'. In principle, the standard offers examiners a choice between quantitative and qualitative interpretations of the associated terms. Here, the quantitative approach is used to make results more easily reproducible and algorithmically processable.

TABLE I
Mapping between TOE-resistance and attack probability score [2].

Sum of Points	TOE Resistance	Probability Score
0-9	No rating	5
10-13	Basic	4
14-19	Enhanced Basic	3
20-24	Moderate	2
> 24	High	1

As a first building block, assets derived from legal requirements in Annex I of the MID are detailed [2]. Within this paper, measurement data with their associated security properties integrity and authenticity will be used as an example. Nevertheless, the following observations are applicable to all other assets as well. Availability of data is not required, since in absence of measurement data no commercial transaction can take place and no financial damage can be inflicted. Subsequently, an example for a possible threat could be formulated as follows:

An attacker falsifies the authenticity of measurement data.

In line with the definition from ISO/IEC 27005, values for impact and probability of occurrence are needed to calculate a numerical risk score. Here, only two impact values are used ($\frac{1}{3}$ if a single measurement is affected and 1 if all future or past measurements are affected). To evaluate the probability of occurrence, technical details (so-called attack vectors) needed to implement a threat are evaluated according to the vulnerability analysis described in ISO/IEC 18045. For example, guessing an administrator's password would be one attack vector to obtain access to protected operating system features. The evaluation of all possible attack vectors for an attack by means of the ISO/IEC 18045 vulnerability analysis consists of assigning point scores in five different categories: 1) time required; 2) expertise required; 3) knowledge needed about the target of evaluation (TOE); 4) window of opportunity; and 5) equipment needed to implement the attack. An expertise score of 0, for instance, reflects the fact that the attack can be carried out by a layman. If an expert is needed instead, the score is set to 6. Incidentally, the very same scores can also be used to describe the cost of implementing a countermeasure to an attack since attacker and defender have to perform similar tasks, see Section V. More detailed examples that address all five scores may be found in Section V of this paper and in ISO/IEC 18045 [4]. Once the scores for all five categories have been assigned, the sum score is calculated. In the original ISO/IEC 18045 this sum score with a theoretical upper limit of 51 points is mapped to a TOE resistance between 'no rating' and 'high'. It follows that an attack is less likely to happen if the sum score is higher. As described in [2], this resistance can be transformed into an attack probability score between 1 and 5, where 5 corresponds to a high likelihood, see Table I. The probability score is then multiplied with the determined impact to calculate the numerical risk score, which is again in the range between 1 and 5. An overview of the entire risk assessment workflow is given in [2].

B. Extension of Risk Assessment method for Attack Probability Trees

In [5] the attack tree concept from Mauw and Oostdijk was extended by augmenting the attack nodes with attributes such as time and expertise detailed above. The resulting attack probability trees (AtPTs) both represent the attack logic and the probability of occurrence (and subsequently risk) associated with a threat. This means that each attack vector is no longer evaluated individually, but only the atomic attacks at the leaf nodes are assessed. This reduces the possibility for misjudging an attack and makes it possible to re-use atomic attacks for different threats.

To propagate the attributes up the tree a number of rules specifically tailored for the characteristics of each attribute were introduced. Since these rules are used extensively in the experimental example in Section V, the following description is needed:

- Time
  - **AND**: Time representation in point scores is logarithmic (1 for more than a day, 2 for one to two weeks, 19 for half a year). Adding up times for two attacks can, therefore, be approximated by selecting the maximum of the two.
  - **OR**: The time score connected to the smaller sum-score is chosen.
- Expertise
  - **AND**: Normally, the maximum of both scores is chosen. Should expertise in both hardware and software (HW and SW) be needed, scores are added with a maximum value of 8, see ISO/IEC 18045.
  - **OR**: The expertise score connected to the smaller sum-score is chosen.
- Knowledge of the TOE
  - **AND**: The maximum of both knowledge scores is chosen.
  - **OR**: The knowledge score connected to the smaller sum-score is chosen.
- Window of opportunity
  - **AND**: A smaller window of opportunity (higher score) for one node will also affect the other node. Therefore, the maximum is selected.
  - **OR**: The window of opportunity score connected to the smaller sum-score is chosen.
- Equipment
  - **AND**: The maximum of both equipment scores is chosen unless equipment from different areas is required (HW or SW), in which case the scores are added with a maximum of 9 according to ISO/IEC 18045.
  - **OR**: The equipment score connected to the smaller sum-score is chosen.

IV. Selecting countermeasures

In the context of this paper, a countermeasure shall be any technical modification of a measuring instrument or organizational measure that results in the reduction of the overall risk associated with the instrument below a predefined threshold. However, finding a set of countermeasures to prevent an attack from happening is more complex then simply countering each individual atomic attack.

A. Connection between AtPTs and Logic Networks

In the case of an AND-statement, increasing the attack cost, i.e. the sum score of one leaf node, always affects the sum score of the parent node. For OR-connected nodes, modification of one node may be sufficient, if the other node's sum score is already above a predetermined level and the associated risk is subsequently low enough. In other cases, however, it may be necessary to counter both child nodes of an OR-connected node to increase the cost associated with the parent node sufficiently, see Figure 2.

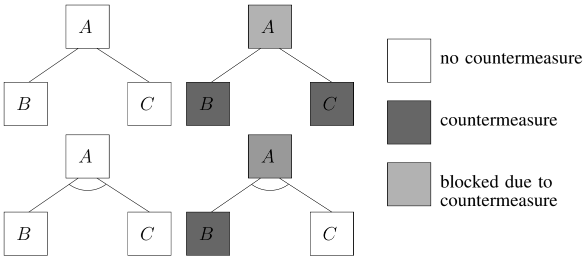


Fig. 2. Two seperate kinds of atomic AtPTs exist which correspond to OR-statements (*above*) and AND-statements (*below*) in boolean logic. For an OR-statement it may be necessary to counter both leaf attacks i.e. to increase their individual costs to mitigate the threat while one blocked leaf always has the same affect for an AND-statement.

It should be noted that the two atomic trees correspond directly to the fundamental building blocks of any logic network. It follows that even more complex trees can be converted into logic networks. This is the inverse principle of the method described in [9]. Of course, the interpretation of AtPTs as logic networks, see Figure 3, lacks all the information concerning individual attributes and the probability of occurrence or risk. However, this simple Boolean view of an AtPT constitutes a practical step towards identifying potential sets of countermeasures to be evaluated. To derive practical countermeasures to be implemented, any AtPT can first be transformed into the equivalent Boolean equation. The corresponding term for the exemplary tree given in Figure 1 is given in Equation (1).

$$A = B \vee C = B \vee (D \wedge E) = B \vee (D \wedge (F \wedge G)) \qquad (1)$$

It should be noted that nodes C and E disappear in this representation since they only act as intermediate summary nodes. The final equation only contains the root and leaf nodes. By calculating the logical inverse of Equation (1) the Boolean expression of the Defense Probability Trees (DePT) can be found, see Equation (2).

$$\begin{aligned} \overline{A} &= \overline{B \vee (D \wedge (F \wedge G))} = \overline{B} \wedge \overline{(D \wedge (F \wedge G))} \\ &= \overline{B} \wedge \overline{(D \wedge (F \wedge G))} = \overline{B} \wedge (\overline{D} \vee \overline{(F \wedge G)}) \\ &= \overline{B} \wedge (\overline{D} \vee (\overline{F} \vee \overline{G})) \end{aligned} \qquad (2)$$

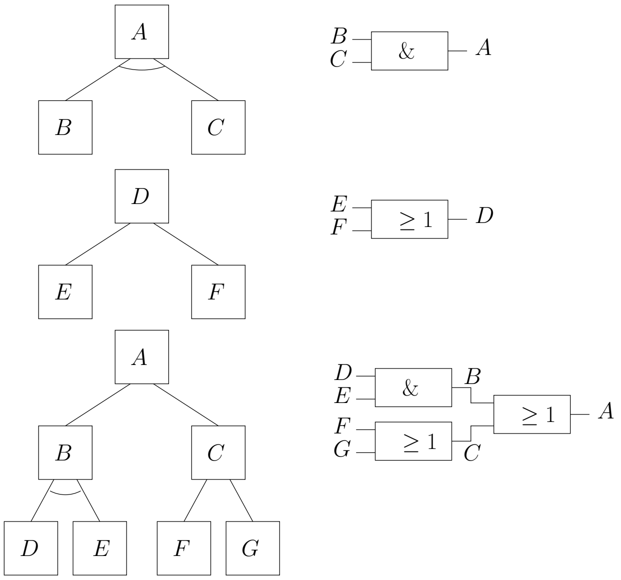


Fig. 3. Any given AtPT can be transformed into a logic network. Three examples (*top to bottom:* AND-connection, OR-connection, complex AtPT) are shown here using electric gate symbols.

The inverse of the logical representation of an AtPT represents all possible countermeasure scenarios, since an attack can only be accomplished if the attack associated with the root node is realized. The logical inverse, therefore, describes all combinations of realized/prevented sub-goals that prevent the root node and its attack from being achieved. Once the Boolean representation has been transformed back into a tree, referred to as a DePT, the connections between AtPT and DePT can be easily seen. Should the DePT contain ambiguous leaf nodes where a range of countermeasures could be selected, this is due to a badly defined AtPT, where the atomic attacks could be further refined by additional subdivisions.

Figure 4 contains the DePT for the taximeter example given in Figure 1. At every leaf node, the DePT automatically offers a possible countermeasure to prevent that atomic attack from being carried out: Node G, for instance (replacing an old seal with a forged new seal) can be countered by improving the quality of the applied seals. Other suggested countermeasures are less practical, however. To prevent new software from being installed (node F), a possible countermeasure is to make the procedure of installing software more difficult, e.g. by creating a non-standard programming interface. Unfortunately, such a measure would also affect the legal intended modification of software. Countermeasures to the remaining leaf nodes (B and D) should also be obvious. Node B (modification of parameters), can be prevented from happening by increasing parameter protection. Node D (writing new (fake) software for the taximeter) can be made more difficult by increasing the complexity of the API. To reiterate, this countermeasure would affect both attacker and original programmer of the software and it is, thus, impractical. Based on the DePT, it can be concluded that one way to prevent fare manipulation is implementing increased parameter protection measures and better physical seals. Other combinations of countermeasures are, of course, also possible. To select the optimal constellation of countermeasures (cheapest countermeasure set to lower the risk below a defined threshold) the attributes examined during the original risk assessment are needed.

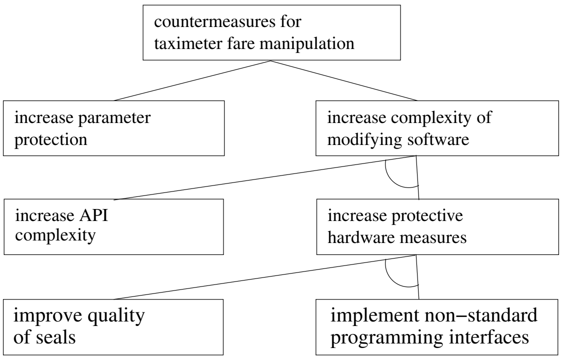

countermeasures for taximeter fare manipulation
increase parameter protection
increase complexity of modifying software
increase API complexity
increase protective hardware measures
improve quality of seals
implement non−standard programming interfaces

Fig. 4. DePT for the countermeasures needed to prevent the calculated fare/measurement value of a taximeter from being manipulated.

B. Usage of Defense Probability Trees

As mentioned above, the simple Boolean view on AtPTs and DePTs lacks all information on attack attributes, probabilities of occurrence and subsequently risk. However, the DePT can be used in the same way as the AtPT to estimate defense costs and resistance to attacks: Here, it will be assumed that the impact of a realized threat is 1 (larger number of measurement results affected) and that a total risk of 3 or lower (probability score ≤ 3, sum score ≥ 14) is acceptable. Each leaf node in the DePT can then be assigned scores for time, expertise, knowledge, window of opportunity and equipment that represent the cost of implementing a countermeasure that reduces the risk of the corresponding leaf node in the AtPT to 3 or lower. The knowledge score now refers to the amount of research needed to realize a countermeasure. If all necessary information is easily available (for instance, because the defender is also the original developer) the knowledge would be considered readily available (score of 0). The window of opportunity then reflects the fact that a defender may not always have access to the attacked instrument. It should be noted that, just as for the attack scenario, not all countermeasures need to be implemented. Instead, the DePT will help to identify the set of countermeasures that produce an acceptable risk reduction at minimal cost/effort. The rules of attribute propagation within the DePT remain the same as listed in Section III-B, since the defender will try to minimize her efforts as well. The process of finding an optimal set of countermeasures can be summarized as follows:

1) Construct an AtPT according to the rules described in [5].
2) Derive a DePT from the resulting Boolean formulae and discard all branches that constitute impractical measures that affect both attacker and intended user.

3) Select implementations for each atomic countermeasure (leaf node in the DePT) that reduce the resulting risk for the specific node to 3 or lower and label the corresponding leaf node with the respective attribute scores.
4) Propagate these weights up the tree to identify the optimal (cheapest) set of countermeasures.

For the attribute scores for implementation of the countermeasure, the same values as for the implementation of attacks can be used, since they reflect the cost of implementation adequately. The only difference is the perspective of the party attempting an implementation of the countermeasure. For example, the window of opportunity for implementation of a countermeasure should almost always be easy (score of 0) since the user of the instrument will normally allow access to the instrument to close its vulnerabilities. A practical example that illustrates the four steps listed above is given in the following section.

V. Experimental Example: Assessment of the Meltdown vulnerability

The following assessment of the Meltdown vulnerability and its impact on measuring instruments was performed at PTB in late 2017. Manufacturers of affected instruments were afterwards contacted to close the vulnerability, when necessary. At the moment of writing this paper, all necessary software patches have been implemented and re-certified, preventing susceptible instruments to remain in the field.

A. Problem Description

The address space of all physical memory including the address space of the kernel of most operating systems such as Linux, Windows and macOS is mapped in the page table of each individual user process. To prevent processes from accessing certain addresses without permission, these are marked with a supervisory bit. If a process attempts to write data to or read data from such an address, an exception is triggered. However, between the actual access violation and the triggering of the exception, a certain time elapses during which the processor may have read data from the forbidden address and performed additional transient commands [7]. These actions affect the contents of the processor cache, which itself may be read out by the well established Flush-Reload or Evict-Reload algorithms [10]. Both algorithms depend on a precise time reference to determine the contents of the cache. Since this process can be repeated as often as needed, it is feasible to read out the entire contents of the RAM with a data rate of around 500kByte/s.

From a technical point of view, the Meltdown vulnerability can initially only be used to read data from the RAM without permission and thus violate the confidentiality of a specific piece of information. Confidentiality, however, does not belong to the security properties of the assets worthy of protection of a measuring instrument, which only address integrity, authenticity and availability. However, the vulnerability can be exploited to spy on secondary sensitive attributes, such as private keys used for asymmetric cryptographic signatures. For different types of measuring instruments, such private keys are used to cryptographically sign determined measurement results and thus to ensure that the measurement results can no longer be changed unnoticed after an export from the measuring instrument. In addition, the signature ensures that the results have been actually generated by the calibrated measuring instrument.

Within this example, only threats on the authenticity of the measurement data will be examined. The formal threat definition can therefore be stated as follows:

An attacker generates false measurement results.

B. Attack probability tree for the Meltdown scenario

The Attack Probability Tree (AtPT) shown in Figure 5 breaks down this threat into individual partial attacks / attack vectors, which must be combined to realize the threat. As before, attack vectors associated with an arc are read as AND-connected, all other attack vectors are OR-connected.

The attacks A01, A04, A07, A09, A10, A12, A13, and A14 represent atomic attack vectors that are no longer subdivided. From the AtPT shown in Figure 5, the following two basic requirements can be derived, which are necessary for exploiting a Meltdown attack on a measuring instrument to realize the threat:

- The measuring instrument's processor is affected by the Meltdown vulnerability.
- The measuring instrument has an open physical interface or a network connection that is protected by means of the operating system.

The attack is therefore unworkable if a hardware security module is used or if the measuring instrument has no open physical or logical interfaces.

C. Evaluation of atomic attack vectors

The score for atomic attack vectors according to the procedure described in [2] is shown in Table II. A brief description of all atomic attack vectors in order to execute the attack is presented below:

A13, A14: To introduce executable code into the instrument (A11), an attacker must use a physical interface (A13) or a network interface (A14).

A12: Once A11 has been achieved (attacker introduces executable code into the instrument), and the attacker is set to use a vulnerability of the operating system (A12), she has the ability to execute the Meltdown attack on the processor (A08).

If code is to be executed on the attacked measuring instrument with limited privileges, basically two variants are conceivable to differ, in terms of what detailed knowledge of the system architecture must be available to the attacker. The more probable scenario results from the lower sum score:

1) Known system architecture (known location of the key): Knowledge of the system: 11, Time: 1, Expertise: 3 (Total: 15)
2) Unknown system architecture (unknown location of the key):

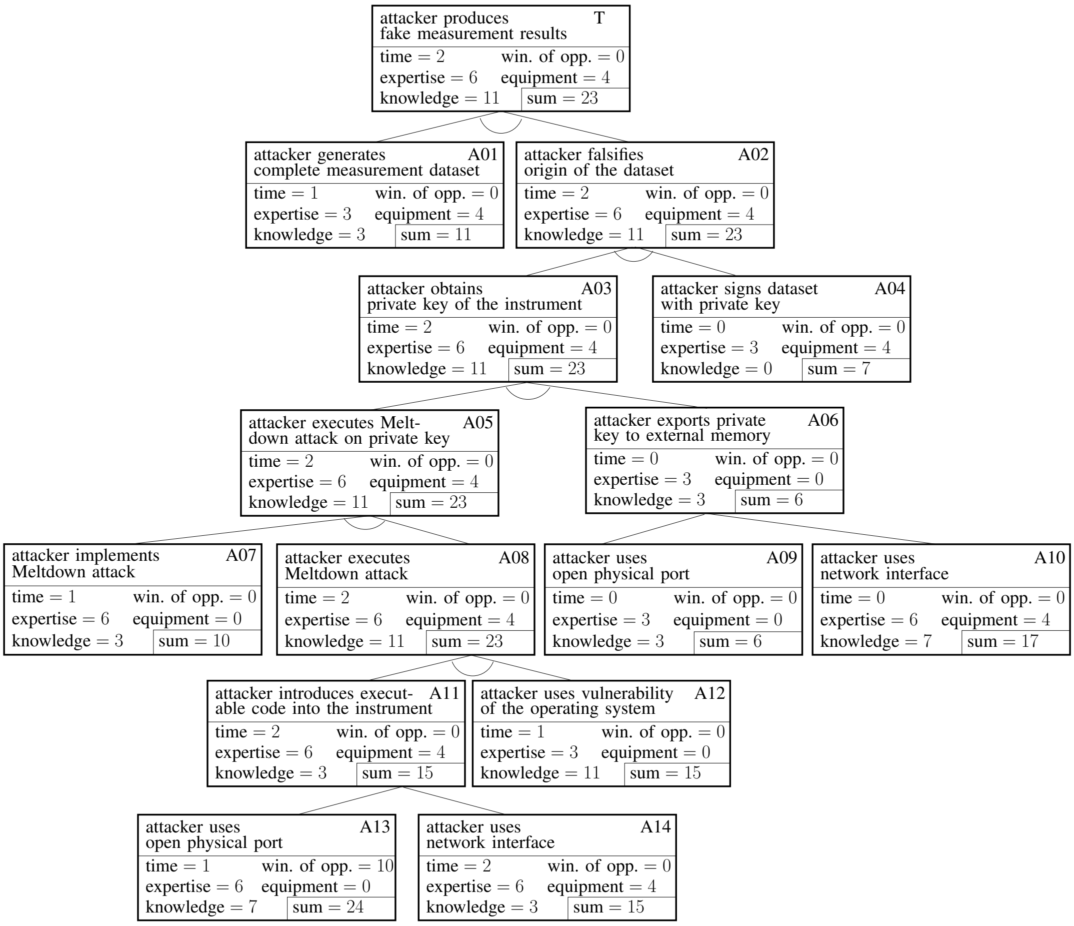

attacker produces fake measurement results T
time = 2 win. of opp. = 0
expertise = 6 equipment = 4
knowledge = 11 sum = 23
attacker generates complete measurement dataset A01
time = 1 win. of opp. = 0
expertise = 3 equipment = 4
knowledge = 3 sum = 11
attacker falsifies origin of the dataset A02
time = 2 win. of opp. = 0
expertise = 6 equipment = 4
knowledge = 11 sum = 23
attacker obtains private key of the instrument A03
time = 2 win. of opp. = 0
expertise = 6 equipment = 4
knowledge = 11 sum = 23
attacker signs dataset with private key A04
time = 0 win. of opp. = 0
expertise = 3 equipment = 4
knowledge = 0 sum = 7
attacker executes Meltdown attack on private key A05
time = 2 win. of opp. = 0
expertise = 6 equipment = 4
knowledge = 11 sum = 23
attacker exports private key to external memory A06
time = 0 win. of opp. = 0
expertise = 3 equipment = 0
knowledge = 3 sum = 6
attacker implements Meltdown attack A07
time = 1 win. of opp. = 0
expertise = 6 equipment = 0
knowledge = 3 sum = 10
attacker executes Meltdown attack A08
time = 2 win. of opp. = 0
expertise = 6 equipment = 4
knowledge = 11 sum = 23
attacker uses open physical port A09
time = 0 win. of opp. = 0
expertise = 3 equipment = 0
knowledge = 3 sum = 6
attacker uses network interface A10
time = 0 win. of opp. = 0
expertise = 6 equipment = 4
knowledge = 7 sum = 17
attacker introduces executable code into the instrument A11
time = 2 win. of opp. = 0
expertise = 6 equipment = 4
knowledge = 3 sum = 15
attacker uses vulnerability of the operating system A12
time = 1 win. of opp. = 0
expertise = 3 equipment = 0
knowledge = 11 sum = 15
attacker uses open physical port A13
time = 1 win. of opp. = 10
expertise = 6 equipment = 0
knowledge = 7 sum = 24
attacker uses network interface A14
time = 2 win. of opp. = 0
expertise = 6 equipment = 4
knowledge = 3 sum = 15

Fig. 5. Attack probability tree for generating false measurement results with a stolen private key by means of the Meltdown vulnerability. Attack vectors linked with an arc are AND-connected. All other attack vectors have an OR-connection. Note: The attack can only be implemented if the private key is available in the instrument at runtime and if the processor is affected by Meltdown.

Knowledge of the system: 7, Time: 4, Expertise: 6 (Total: 17)

A07: Once A08 has been achieved (attacker executes Meltdown attack), and the attacker has written the required code for spying out the private key, she can use the Meltdown attack on the private key (A05). When the attacker implements the actual attack to exploit the Meltdown vulnerability, she will probably create a full dump of all memory. Therefore, no special knowledge of the system is needed. Once again, the more probable attack scenario results from the lower sum score:

1) Attacker's own implementation:
Knowledge of the system: 3, Time: 2, Expertise: 8 (Total: 13)
2) Attacker uses third parties' implementations:
Knowledge of the system: 3, Time: 1, Expertise: 6 (Total: 10)

A09, A10: The writing or exporting of data is already provided in many measuring instruments, so that no increased effort has to be set for this. Either of these two subgoals allows exporting of the private key to external memory (A06). Once A06 has been achieved, and the attacker has executed the Meltdown attack on the private key (A05), the attacker can obtain the private key of the instrument (A03).

A04: Once the attacker has obtained the private key (A03), she is able to sign new datasets (A04); if A03 and A04 are achieved, the attacker can falsify the origin of fake datasets (A02).

A01: Finally, once the capacity for falsifying the origin of datasets is achieved (A02), and the attacker has generated a new (and fake) measurement dataset, the analysed Threat (T) can be realized.

In the case of an operating system configured in accordance with legal requirements, no code can be executed from an external medium. However, it cannot be ruled out that the Meltdown bug will not lead to speculative execution. It should be noted that attacks A01 (generation of a measurement dataset) and A04 (signing of a measurement dataset) must be repeated for each individual measurement result and therefore should be assigned a reduced impact of $\frac{1}{3}$. However, since these are attacks with comparatively little effort, the overall impact rating does not change. Once the rules for attribute

TABLE II
EVALUATION OF ATOMIC ATTACK VECTORS - THE ATTACK VECTORS A01 AND A04 DIFFER FROM ALL OTHERS IN THAT THEY MUST BE REPEATED FOR EACH INDIVIDUAL REALIZATION OF THE THREAT. ACCORDINGLY, THE DAMAGE HERE IS REDUCED TO A VALUE OF 1/3.

Attack ID	attack vector	time	expertise	knowledge	window of opp.	equipment	sum	impact
A13	Attacker uses open interface to bring code into instrument.	1	6	7	10	0	24	1
A14	Attacker uses network connection to bring code into instrument.	2	6	3	0	4	15	1
A12	Attacker expoits operating system vulnerabilites to execute code with limited privileges.	1	3	11	0	0	15	1
A07	Attacker implements Meltdown attack.	1	6	3	0	0	10	1
A09	Attacker uses open interface to export private key.	0	3	3	0	0	6	1
A10	Attacker uses network connection to export private key.	0	6	7	0	4	17	1
A04	Attacker signs measurement result with private key.	0	3	0	0	4	7	$\frac{1}{3}$
A01	Attacker generates complete measurement dataset.	1	3	3	0	4	11	$\frac{1}{3}$

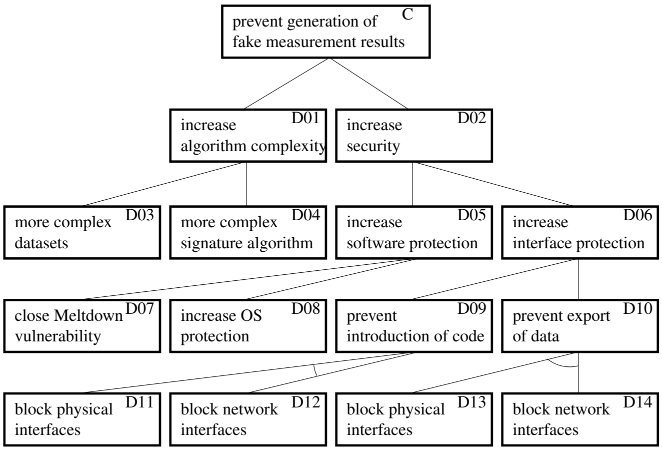

prevent generation of fake measurement results C
increase algorithm complexity D01
increase security D02
more complex datasets D03
more complex signature algorithm D04
increase software protection D05
increase interface protection D06
close Meltdown vulnerability D07
increase OS protection D08
prevent introduction of code D09
prevent export of data D10
block physical interfaces D11
block network interfaces D12
block physical interfaces D13
block network interfaces D14

Fig. 6. Defense probability tree for generating false measurement results with a stolen private key by means of the Meltdown vulnerability. Defense vectors linked with an arc are AND-connected. All other defense vectors have an OR-connection.

propagation have been applied, the root node has a sum score of 23, which is equivalent to a probability score of 2 and a risk score of 2. Such a rating would be acceptable for most instrument classes except when the law requires even more stringent protective measures, in which case the risk has to be reduced to 1.

D. Identification and Selection of Suitable Countermeasures

As described in Section IV the DePT (see Figure 6) can be inferred by converting the AtPT to a Boolean expression and applying the inverse operation on the expression. The resulting graphical representation is given in Figure 6. It should be noted that the structure of the tree has been modified according to the transformation rules laid out in [6] for better comprehensibility. From this initial DePT all nodes can be removed that constitute impractical defensive measures, i.e. making the signature algorihm more complex (node D04), deactivating all hardware and software interfaces (node D11/D13 and D12/D14) since all three are needed for the actual intended operation of the instrument. Afterwards, a number of nodes remain that have only one child node. According to [6], such a parent node can be replaced by the child node since its attributes must be identical to those of the child node. After the removal of all impractical defensive measures, the DePT is reduced to a tree of three nodes, see Figure 7. The remaining DePT states that retrieving the private key from within a measuring instrument via exploitation of the Meltdown vulnerability to generate false measurement data can be prevented by one of two alternatives:

- external code is prevented from being executed through increased protective operating system measures (node D08),
- a patch is applied to the operating system to prevent speculative code execution, thus closing the vulnerability (node D07).

When evaluating these alternatives, all assigned scores (time, expertise, knowledge, window of opportunity and equipment) have to be based on the assumption that the implementation is done by a white hat developer who has access to both the instrument, as well as all manufacturer's documentation.

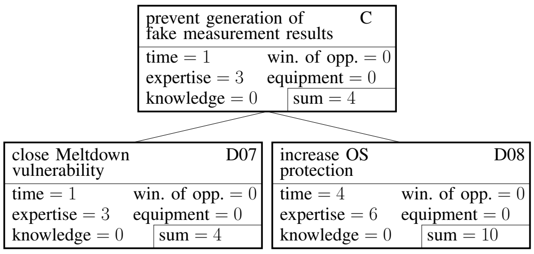

prevent generation of fake measurement results C
time = 1 win. of opp. = 0
expertise = 3 equipment = 0
knowledge = 0 sum = 4
close Meltdown vulnerability D07
time = 1 win. of opp. = 0
expertise = 3 equipment = 0
knowledge = 0 sum = 4
increase OS protection D08
time = 4 win. of opp. = 0
expertise = 6 equipment = 0
knowledge = 0 sum = 10

Fig. 7. Reduced DePT for the Meltdown vulnerability. Nodes with only one child node have been replaced by the child node in accordance with the rules laid down in [5].

Concerning the implementation of increased protective operating system measures (D08), a programming expert (score of 6 for expertise) should be able to implement, test and release a solution within a month (score of 4 for time). Since all internal manufactuer's documentation is available to the white hat developer, the knowledge score shall be set to 0 (readily available). The same is true for the window of opportunity. Furthermore, no special equipment will be needed to implement a software solution. Similarly, applying an operating system patch to close the vulnerability (D07) should be quite easy.

Since many operating system manufacturers provide patches and instructions for their installation on their website, any proficient user (expertise score of 3) should be able to patch the operating system within a week. All necessary information should be readily available (knowledge score of 0), while access to the instrument is always granted. Also, as was the case for node D08, only standard equipment is needed for the implementation of the countermeasure.

E. Conclusion of Experimental Example

Once the two leaf nodes of the DePT have been labeled with these attributes, see Figure 7, the rules laid down in Section III-B are applied to yield the attributes for the root node. This results in the root node becoming a copy of leaf node D07, which indicates that application of an operating system patch is the easiest countermeasure against the defined threat. Once the Meltdown vulnerability has been closed as described, the only way to get the private key of the instrument (node A03 in Figure 5) is to correctly calculate the key by a brute-force method, taking longer than half a year for any state-of-the-art signature algorithm. Since this will set the time score for node A03 to 19 points, the root node will be similarly affected, increasing its sum score to 40 (equivalent to a probability score and risk score of 1). Therefore, the proposed countermeasure is suitable to prevent the examined threat.

The same procedure can be applied to other threats with larger DePTs. The logic behind choosing a countermeasure should be, in general, identical to the logic of attack implementations, since both attacker and defender will aim to achieve their goals with minimal effort. Therefore, all previously performed evaluations of AtPTs will hold true also for DePTs. To validate the usefulness and efficiency of the proposed countermeasure identification and selection method, more exemplary applications are needed, of course. Nevertheless, this investigation should be seen as a first proof of concept. Within the scope of this paper, the action associated with a node has either been assumed to have a permanent effect (impact score of 1) or its influence on the overall cost of implementing an attack/countermeasure was assumed to be negligible. Since this assumption will eventually fail in certain scenarios, an approach to deal with differing impact scores for individual nodes is still needed.

VI. Summary

Evaluating IT threats to assets and selecting appropriate countermeasures will form a cornerstone of Legal Metrology in the near future. While other IT sectors are already using such risk-based evaluation schemes, the present paper describes a suitable method for risk mitigation tailored for measuring instruments subject to legal control. To this end, AtPTs with calculation rules for their attributes have been transformed into equivalent DePTs. Since the method is based on established international standards, it is anticipated that it can easily be applied in other economic sectors as well. An application of the method [5] on the Meltdown vulnerability has been used to demonstrate the workflow and usage of the presented countermeasures selection procedure. In the future, the impact of vulnerabilities in common IT products on Legal Metrology will certainly need to be investigated more frequently. The risk analysis and countermeasure derivation methods discussed in this paper represent a selection of tools, at the disposal of notified bodies, manufacturers and market surveillance authorities, to asses the risks associated with such IT-related incidents. One piece that is still missing for AtPTs and DePTs to be generally applicable is a method to deal with attacks/countermeasures with different impact scores (i.e. permanent and repetitive) within one specific AtPT/DePT alike. Further work will address a theoretical analysis of the influence of impact scores on the overall risk and a proposal to reflect such scores in AtPTs and DePTs. Finally, a harmonized guideline for using AtPTs and DePTs should be written to be applied to new technologies, to test the efficiency of both methods for streamlining innovations within Legal Metrology.

References

[1] EC, "Directive 2014/32/EU of the European Parliament and of the Council of 26 February 2014 on the harmonisation of the laws of the Member States relating to the making available on the market of measuring instruments," European Union, Council of the European Union ; European Parliament, Directive, February 2014.

[2] M. Esche and F. Thiel, "Software risk assessment for measuring instruments in legal metrology," in *Proceedings of the Federated Conference on Computer Science and Information Systems*, Lodz, Poland, September 2015. doi: http://dx.doi.org/10.15439/978-83-60810-66-8 pp. 1113–1123.

[3] ISO/IEC, "ISO/IEC 27005:2011(e) Information technology - Security techniques - Information security risk management," International Organization for Standardization, Geneva, CH, Standard, June 2011.

[4] ——, "ISO/IEC 18045:2008 Common Methodology for Information Technology Security Evaluation," International Organization for Standardization, Geneva, CH, Standard, September 2008, Version 3.1 Revision 4.

[5] M. Esche, F. Grasso Toro, and F. Thiel, "Representation of attacker motivation in software risk assessment using attack probability trees," in *Proceedings of the Federated Conference on Computer Science and Information Systems*, Prague, Czech Republic, September 2017. doi: http://dx.doi.org/10.15439/2017F112 pp. 763–771.

[6] S. Mauw and M. Oostdijk, "Foundations of attack trees," in *Proceedings of the 8th international conference on Information Security and Cryptology*. Seoul, Korea: IEEE, December 2005. doi: http://dx.doi.org/10.1007/11734727_17 pp. 186–198.

[7] M. Lipp, M. Schwarz, D. Gruss, T. Prescher, W. Haas, A. Fogh, J. Horn, S. Mangard, P. Kocher, D. Genkin, Y. Yarom, and M. Hamburg, "Meltdown: Reading kernel memory from user space," in *27th USENIX Security Symposium, USENIX Security 2018, Baltimore, MD, USA, August 15-17, 2018.*, 2018, pp. 973–990. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/lipp

[8] P. Wang, W.-H. Lin, P.-T. Kuo, H.-T. Lin, and T. C. Wang, "Threat risk analysis for cloud security based on attack-defense trees," in *Proceedings of the International Conference on Computing Technology and Information Management*. Seoul, Korea: IEEE, April 2012, pp. 106–111, ISBN: 978-89-88678-68-8.

[9] R. Vigo, F. Nielson, and H. R. Nielson, "Automated generation of attack trees," in *Proceedings of the IEEE Computer Security Foundations Symposium*. Seoul, Korea: IEEE, 2014. doi: http://dx.doi.org/10.1109/CSF.2014.31 pp. 337–350.

[10] Y. Yarom and K. Falkner, "FLUSH+RELOAD: A high resolution, low noise, L3 cache side-channel attack," in *Proceedings of the 23rd USENIX Security Symposium, San Diego, CA, USA, August 20-22, 2014.*, 2014, pp. 719–732. [Online]. Available: https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/yarom

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 537–540 DOI: 10.15439/2020F43 ISSN 2300-5963 ACSIS, Vol. 21

Voice authentication based on the Russian-language dataset, MFCC method and the anomaly detection algorithm

Anna Sidorova
MEPHI Cryptology and cybersecurity department
Kashirskoe Sh. 31 Moscow, Russia
Email: saa075@campus.mephi.ru

Konstantin Kogos
MEPHI Cryptology and cybersecurity department
Kashirskoe Sh. 31 Moscow, Russia
Email: kgkogos@mephi.ru

Abstract—**Almost all people's data is stored on their personal devices. There is a need to protect information from unauthorized access. PIN codes, passwords, tokens can be forgotten, lost, transferred, brute-force attacked. For this reason, biometric authentication is gaining in popularity. Biometric data are unchanged for a long time, different for users, and can be measured. This paper explores voice authentication due to the ease of use of this technology, since obtaining voice characteristics of users doesn't require an equipment in addition to the microphone. The method of voice authentication based on an anomaly detection algorithm has been proposed. The software module for text-independent authentication has been implemented on the Python language. It's based on a new Mozilla's open source voice dataset "Common voice". Experimental results confirmed the high accuracy of authentication by the proposed method.**

I. Introduction

VOICE authentication is a biometric authentication method that uses the user's voice as an identifier. It is based on determining the belonging of a given speech signal to some speaker. The voice of the speaker and then the speech signal entering the authentication system are unique. This causes to an interest in him as a biometric object [1].

Most studies on this topic identifies two main types of voice biometric systems: text-dependent and text-independent.

Text-dependent ones are more often used to control access to the system: during authentication, a certain phrase is pronounced, which is compared with the model of user registered in the system. The vulnerability of such systems is obtaining unauthorized access by recording a passphrase using modern tools of acoustic eavesdropping and providing it to the system.

In text-independent systems, almost any fragment of sounding speech can be used. Their accuracy is less, and the complexity of the implementation is greater. When using this mode, along with the mechanism for verifying statements as two-factor authentication, repetition attacks are almost impossible. For this reason, text-independent voice authentication is of special interest.

II. Related Works

During the work, the analysis of existing research in the field of voice authentication has performed.

The authors of the work [3] propose a method of text-dependent user authentication according to the phrase, pronounced by him. To extract features, the MFCC (mel-frequency cepstral coefficients) method is used. To compare records, the distance between two sets of coefficients is calculated. If the distance between the two MFCC sets is less than the specified threshold, the entries are considered the same. The best result in experiments based on dataset, collected by the authors: FAR=16,66%, FRR=0%.

The author of the work [4] has developed a text-independent speaker identification system based on GMM (Gaussian mixture models). To calculate features, the author uses MFCC and inverse MFCC. During the recognition, for each segment of the record, the degree of its similarity with the phonemes of each speaker is calculated. A value of the proximity of the speech segment's feature vector (MFCC) and the phoneme is the average logarithmic probability GMM of the phoneme, calculated for this MFCC-vector. When the decision is made on a closed set of speakers, the nearest neighbors method is used. During experiments on the “TIMIT” and “LibriSpeech" speech corpora, the accuracy of 98% was achieved.

The research [5] also considers MFCC and DMFCC to calculate features, and in addition to this method, vector quantization is used. The study uses the "AN4" dataset, which is publicly available on the Carnegie Mallon University website. The highest accuracy obtained by the authors of the work is 89.20%.

The authors of the work [6] have organized a text-independent authentication system "VoizLock" based on their own voice database. The system can check not only the coincidence of votes, but also that the user says what he should. The authors use LPCC — linear predictive cepstral coefficient — to extract features and HMM — the hidden Markov model — method to build models. The highest accuracy in experiments is 86.25%.

In [7], a text-independent authentication system has implemented. The system compares the result of a user's pronunciation with a previously saved voice profile. If the deviation is below the threshold, authentication passes. For feature extraction, instead of LPCC, the authors preferred MFCC as a feature extraction method. It is justified by the fact that it is more reliable and works better. Also integrated in the system is CMS — the cepstral average subtraction — to subtract the estimated channel noise in the spectrum, and DDMFCC to improve voice recognition accuracy. At the last stage, the Euclidean distance between the MFCC is calculated with the help of the DTW— dynamic time warping algorithm. The best accuracy that was obtained on a self-assembled base is 90%.

The peculiarity of work [8] is that authors propose an approach using interactive voice response (IVR) with speaker recognition based on neural networks. After entering the correct password, the user is prompted to enter his voice instance. Since both factors are applied simultaneously, the probability of authentication errors is reduced. In the work characteristics of speakers are extracted using MFCC, and MLP is used to compare the characteristics. Algorithm — Gradient boosting method. The best result in experiments based on dataset, collected by the authors: FAR=14%, FRR=18%.

The authors of the research [9] identify the problem of reducing recognition performance due to strong background noise. It explores the i-vector (the improvement of GMM) system in noisy conditions. Best accuracy in experiments on voice datasets "Switchboard-1" and "NIST SRE" is 80,54%.

The Russian group of companies "Center for Speech Technologies" and its assistant, author of [10], [11] and other works about voice identification have implemented hybrid GMM–JFA–SVM and GMM–TV– SVM systems using MFCC, DMFCC and DDMFCC for calculating features. It based on the «CST-Microphone». The best result in experiments: FRR = 0,3%, FAR = 10%. In [10], the informativeness of speech features for automatic speaker identification systems was studied. The methods MFCC, LPCC, PLP are considered in details, and it is concluded that combining features always provides the lowest EER.

The paper [12] also describes the GMM, proposes an algorithm based on the k-means algorithm, for estimation GMM parameters (Gauss component numbers, etc.) and an algorithm based on vector quantation for initiating GMM parameters to maximize the likelihood function. Experiments based on the statement proposed by the "Oregon Institute of Science and Technology, Centre of Spoken Language Understanding" show that the system has a high recognition rate: about 94%.

There are few works on voice authentication among the publicly available works. Most of the work relates to the classification of many speakers by voice, which is a different task and usually is solved on the basis of algorithms of classification. These systems determine whose voice from the many speakers the new added record is similar most of all. Thus, for the high accuracy of most such systems you need to have a large number of records of people. In practice, usually one person owns some device, one person is registered in the authentication system, and only the data of the genuine user is available to the system. It is not practical to store large voice bases on a device, even if they are collected and stored safely. The most appropriate solution to the authentication problem is to train on the data of a one user and use the resulting model to identify attackers. The above is known as anomaly detection. This paper decided to first investigate the difference between people by voice characteristics, implement a classification method based on proven methods in most works, achieve high accuracy and then implement a voice authentication based on the method of detecting anomalies.

III. DATASET AND FEATURE EXTRACTION

First of all, the implementation of the voice authentication software module requires a dataset to form training and test samples.

Most of works use English-language datasets: "LibriSpeech", the "TED-LIUM" collection, "AN4"; the "TIMIT" dataset.

The multilingual open base "Common Voice", created by Mozilla Corporation, is of greatest interest. It has been decided to use it. Its advantage is that it has a large volume. It is also suitable for authentication because it contains user IDs and it is as close to reality as possible, because Internet users participate in its compilation, and the recording conditions and sound quality are far from ideal. During the work, the Russian-language part of the base was taken.

Data to implement the classification module needs to be pre-processed. Files have been converted from mp3 format to a more universal wav format. The next stage of the work was the detection of speech activity of records, which is necessary to reduce the volume of uninformative data. Speech activity detectors are used for this purpose. The filter filter_silence from the audiosegment software module of the Python language for this purpose is applied, excluding areas of silence. This filter works on the basis of the publicly available WebRTC VAD project.

Similar pre-processing is done on data to implement the authentication module, except that the training sample is a set of legitimate user records, and the test is a set of records of legitimate user and users, who are considered to be attackers.

As a result of the analysis of existing works it is noted that the most frequently used, promising features are MFCC. The reason for this is the simplicity of their calculation and good approximating ability by taking into account the different human perception of the tone of sound depending on its frequency. In addition, improvements are implemented for them, such as: using their time derivatives DMFCC and DDMCC, reverse MFCC, normalization.

The main stages of feature extraction are as follows [13]:

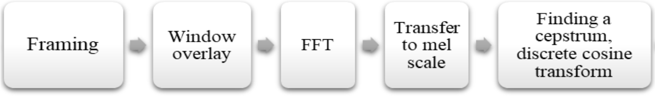


Fig. 1 Stages of feature extraction from input speech signal — MFCC

At the stage of preprocessing the input speech signal, high frequencies are amplified, noise is attenuated. Filters are applied: triangular, Gaussian, etc. The speech signal is non-stationary, so framing is done to get stationary frames of the signal. Sometimes the speech phrase signal is divided into frames of length N with an overlap, for example, half. When you overlay windows, a window weight function is applied to each individual frame. The Hamming window is used for this purpose. In the next step, the fast Fourier transform is applied. Each frame of N samples is converted from the time domain, in which the signal representation shows the dependence of the amplitude on time, to the frequency domain. Then the frequency is translated into the mel scale. The last step of the method is a discrete cosine transform. It translates the signal back to the time domain, converting it into kepstral coefficients. Cepstral mel coefficients are calculated in each frame.

The MFCC vector of a single speaker audio record is a 13-dimensional vector whose length depends on the length of the record, since the coefficients are calculated from the frames into which the record is split along its length.

IV. Classification

To solve the classification task it was decided to use the GMM method, it has proven itself in many works as a promising method, for which authors achieve improvements. To make a decision about the authorship the maximum likelihood estimation method is used. A detailed mathematical description is given in the work [11]. Thus, the module for classifying speakers in Python is implemented.

V. Authentication

As mentioned earlier, this paper solves the problem of authentication, it is proposed to focus on a legitimate user and when adding new voice records to the system, that is, when the authentication process begins, to determine whether the voice record is abnormal, that is, does not belong to a legitimate user.

When investigating the applicability of anomaly detection algorithms to the problem of voice authentication, various methods were considered. The best results were obtained by algorithms for detecting anomalies when working directly with MFCC. It is important to note that to do this, it was necessary to average the coefficients over time, since each set of coefficients has a different dimension due to different lengths of audio recordings. Among well-known algorithms, the Local Outlier Factor showed the best accuracy on the same data. A software authentication module has implemented using it. It is shown in figure 3.

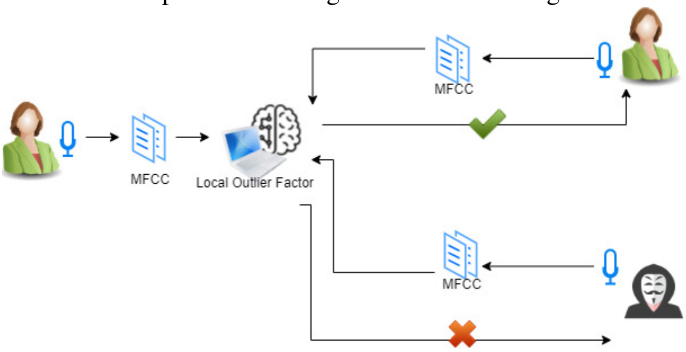


Fig. 3 The proposed scheme of the voice authentication process

To organize the text-independent authentication, it is must to add a speech recognition. This is necessary to check the uttered phrase. The paper considers the existing well-known open mechanisms of speech recognition: DeepSpeech from Mozilla, Kaldi, Vosk, Live Transcribe Speech Engine from Google, etc.

This work the Vosk engine for speech recognition are used due to its ease of use and embedding in the project, and the higher performance of the Kaldi engine that it based on. The re-trained model in Russian is publicly available. Thus, the speech recognition software module is implemented using the above technologies.

VI. Experiment Results

To investigate the possibility of classifying the speakers of the selected dataset by voice characteristics, the classification accuracy was evaluated using the selected methods for features extraction and speaker model construction. The accuracy score was calculated.

The first result obtained was an accuracy of 86%. During experiments the accuracy increases. The best accuracy is equal to 0.95. Table 1 shows the accuracy values when changing the parameters of the algorithms used.

Table I.
Dependence of accuracy on the value of the parameters

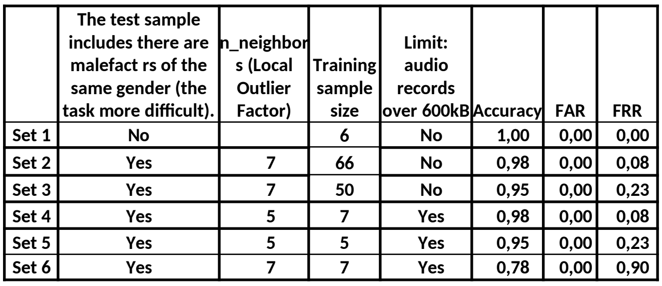
	Number of gaussians in GMM	The width of the frame in MFCC (FFT)	The value of the threshold, the sound is quieter than which is taken for silence in VAD, %	Accuracy
Set 1	32	1024		0,86
Set 2	16	1024		0,87
Set 3	16	1600		0,88
Set 4	16	1600	6	0,90
Set 5	16	1600	10	0,91
Set 6	16	1600	4; 5	0,95
Set 7	16	1600	1	0,94

After confirming that users are distinguishable by voice characteristics, the software module of voice authentication has implemented. An experimental assessment of authentication accuracy based on another part of the "Common voice" dataset has performed.

As training data and part of the test sample, 66 audio records of a female user who was accepted as legitimate were taken. The test sample also contains 41 records of both male and female attackers for the most reliable experiment.

It should be noted that the number of training sample records was being decreased during the experiment, because it is difficult for the user to record decades of phrases at the registration stage. However, it turned out that high authentication accuracy can be achieved if user records multiple audio records when registering a user, but the minimum recording size is limited. Table 2 shows the results of the experiment.

TABLE II.
RESULTS OF THE EXPERIMENTS, THE ACCURACY



	The test sample includes there are malefact rs of the same gender (the task more difficult).	n_neighbors (Local Outlier Factor)	Training sample size	Limit: audio records over 600kB	Accuracy	FAR	FRR
Set 1	No		6	No	1,00	0,00	0,00
Set 2	Yes	7	66	No	0,98	0,00	0,08
Set 3	Yes	7	50	No	0,95	0,00	0,23
Set 4	Yes	5	7	Yes	0,98	0,00	0,08
Set 5	Yes	5	5	Yes	0,95	0,00	0,23
Set 6	Yes	7	7	Yes	0,78	0,00	0,90

As a result, a software authentication module has obtained. It is clear from the experiment that the value of a type II error is usually higher and in some cases especially large: when the user speaks with a very different intonation, volume, than when registering. For example, the system may not recognize it if the training sample is very small and monotonous. This is not dangerous for the data stored on the device, but it may cause inconvenience for a legitimate user. This problem can be solved by setting the special architecture of authentication service: for example, after a certain time, you can ask the device user to supplement the training sample.

III. CONCLUSION

This paper explores existing approaches to voice authentication of users. The methods of feature extraction from the input signal and training of speaker models, proposed in open works on voice biometrics, are analyzed for the implementation of own system. Open datasets were analyzed and the "Common Voice" base has been selected for the implementation of its own system. It has been decided to first implement the speaker classification module based on the selected Russian-language dataset to make sure that users are distinguishable by voice characteristics, and then to propose and implement an approach based on detecting anomalies. A software module for text-independent identification or classification of speakers has been implemented, and its accuracy is 95%. An approach to voice authentication based on the Local Outlier Factor anomaly detection algorithm, which was not previously used in existing open research, is proposed. During the experiments, the accuracy of 98% was obtained. In the future, it would be interesting to try to implement the developed system in some real application and test it in real conditions.

REFERENCES

[1] Bernstein S.I., Kolokoltsev N.K., Ermolaeva V.V. Voice Authantication. Molodoy uchenyy[Young scientist], 2018, no.25. pp. 93-94. Available at: https://moluch.ru/archive/211/51686/ (accessed 24 April 2019)

[2] Ermilov A.V. Metody, algoritmy i programmy resheniya zadach identifikatsii yazyka i diktora[Methods, algorithms and programs for solving problems of language and speaker identification]: Extended abstract of PhD dissertation (physics and mathematics), 2014, 22 p. (in Russian)

[3] Ivanov D.A., Nikitin A.P. Text-dependent voice authentication method. Istoriya I arkhivy [History and archives], 2016, no. 3 (5). (in Russian)

[4] Zakharova V.V. Razrabotka tekstonezavisimoy sistemy identifikatsii diktora na osnove fonemnogo razbieniya i GMM [Development of a text-independent speaker identification system based on phonemic partitioning and GMM]. Proceedings of the Science Conference of undergraduate and graduate students, Belorusskiy gosudarstvennyy universitet, Minsk, 2017, pp. 28-32. (in Russian)

[5] Nogikh A.A., Solomatin D.I. Text-independent authentication. Sbornik studencheskikh nauchnykh rabot fakul'teta komp'yuternykh nauk VGU [Collection of student research papers of the faculty of computer science of VSU], 2016. pp. 117-123. (in Russian)

[6] Jayamaha R. G. Voizlock-human voice authentication system using hidden markov model. Proceedings of the 4th International Conference on Information and Automation for Sustainability. IEEE, 2008. pp. 330-335.

[7] Yan Z., Zhao S. A Usable Authentication System based on Personal Voice Challenge. Proceedings of the International Conference on Advanced Cloud and Big Data, 2016. pp. 194-199. DOI:10.1109

[8] Shah, S. A. A., Shah S.W., A. ul Asar. Interactive Voice Response with Pattern Recognition Based on Artificial Neural Network Approach. NWFP University of Engineering and Technology, Peshawar, Pakistan, 2007. pp. 249-252.

[9] J. Chang, D. Wang. Robust speaker recognition based on DNN/i-vectors and speech separation. Proceedings of the Acoustics, Speech and Signal Processing (ICASSP) IEEE International Conference, 2017. pp. 5415-5419.

[10] Matveev Yu.N. Research of information content of speech signs for automatic speaker identification systems. Vestn. MGTU im. N. E. Baumana. Ser. Priborostroenie. Spetsial'nyy vypusk. Biometricheskie tekhnologii [Bulletin of the Bauman Moscow state technical University. Series "Instrument making"], 2013. no. 2. pp. 47—51. (in Russian)

[11] Matveev Yu.N. Technologies for biometric identification of an individual by voice and other modalities. Vestn. MGTU im. N. E. Baumana. Ser. Priborostroenie. Spetsial'nyy vypusk. Biometricheskie tekhnologii [Bulletin of the Bauman Moscow state technical University. Series "Instrument making". Special issue. "Biometric technology".], 2012. № 3(3). pp. 46—61. (in Russian)

[12] Sadykhov R.Kh., Rakush V.V. Models of Gaussian mixtures for speaker verification based on arbitrary speech. Doklady BGUIR [BSUIR reports], Minsk, 2003. no. 4 pp. 95-103. (in Russian)

[13] Hundal J.K, Dr. Hamde S. T. Some Feature Extraction Techniques for Voice based Authentication System. Proceedings of the Power, Control, Signals and Instrumentation Engineering (ICPCSI) IEEE International Conference, 2017. pp. 419-421.

[14] Documentation of python-speech-features. Available at: https://python-speech-features.readthedocs.io/en/latest/#welcome-to-python-speech-features-s-documentation (accessed: 20 March 2019).

Advances in Information Systems and Technology

THE advancement of information systems and technologies creates new possibilities for the development of economies and societies. AIST constitutes a global forum for researchers and practitioners to present and discuss the recent research on information systems and technologies for business, governments, and society.

AIST invites papers covering the most recent innovations, current trends, professional experiences and new challenges in the several perspectives of information systems and technologies, i.e. design, implementation, stabilization, continuous improvement, and transformation. It seeks new works from researchers and practitioners in business intelligence, big data, data mining, machine learning, cloud computing, mobile applications, social networks, internet of thing, sustainable technologies and systems, blockchain, etc.

The main topics covered are:

- Advanced information systems and technologies for business;
- Advanced information systems and technologies for governments;
- Advanced information systems and technologies for education;
- Advanced information systems and technologies for healthcare;
- Advanced information systems and technologies for smart cities; and
- Advanced information systems and technologies for sustainable development.

Technical Session Chairs

- **Ziemba, Ewa,** University of Economics in Katowice, Poland
- **Cano, Alberto,** Virginia Commonwealth University, Richmond, United States
- **Wątróbski, Jarosław,** University of Szczecin, Poland

Program Committee

- **Assuli, Ofir Ben,** Ono Academic College, Israel
- **Beimel, Dizza,** Ruppin Academic Center, Israel
- **Bialas, Andrzej,** Institute of Innovative Technologies EMAG, Poland
- **Chmielarz, Witold,** University of Warsaw, Poland
- **Christozov, Dimitar,** American University in Bulgaria, Bulgaria
- **Cios, Krzysztof,** Virginia Commonwealth University, United States
- **Deshwal, Pankaj,** Netaji Subash University of Technology, India
- **Dias, Gonçalo Paiva,** Universidade de Aveiro, Portugal
- **Kania, Krzysztof,** University of Economics in Katowice, Poland
- **Konys, Agnieszka,** West Pomeranian University of Technology in Szczecin, Poland
- **Kovacheva, Eugenia,** University of Library Studies and Information Technologies, Bulgaria
- **Luna, Jose,** University of Cordoba, Spain
- **Michalik, Krzysztof,** University of Economics in Katowice, Poland
- **Pastuszak, Zbigniew,** Maria Curie-Sklodowska University, Poland
- **Raban, Daphne,** University of Haifa, Israel
- **Rakus-Andersson, Elisabeth,** Blekinge Institute of Technology, Sweden
- **Rechavi, Amit,** Ruppin Academic Center, Israel
- **Rizun, Nina,** Faculty of Management and Economics, Gdansk University of Technology, Poland
- **Rouibah, Kamel,** College of Business Administration, Kuwait University, Kuwait
- **Rusho, Yonit,** Shenkar College, Israel
- **Santiago, Joanna,** ISEG - University of Lisbon, Portugal
- **Sałabun, Wojciech,** West Pomeranian University of Technology in Szczecin, Poland
- **Sikorski, Marcin,** Gdańsk University of Technology, Poland
- **Szołtysek, Jacek,** University of Economics in Katowice, Poland
- **Tomczyk, Łukasz,** Pedagogical University of Cracow, Poland
- **Travica, Bob,** University of Manitoba, Canada
- **Velazquez, Isaac Triguero,** University of Nottingham, United Kingdom
- **Wątróbski, Jarosław,** Faculty of Economics and Management University of Szczecin Mickewicza 64, 71-101 Szczecin, Poland, Poland
- **Ziemba, Paweł,** University of Szczecin, Poland

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 543–552 DOI: 10.15439/2020F68 ISSN 2300-5963 ACSIS, Vol. 21

Peculiarities of Modern Street Addressing System toward to Implementation of the Smart City Conception

Dmitriy Gakh
Institute of Control Systems
Bakhtiyar Vahabzadeh str. 9
Baku, Azerbaijan
Email: dgakh@sinam.net

***Abstract*—Implementation of Smart City conception is direction of developing many modern cities. Rapid urbanization leaded to constant increasing urban population. At the same time technological growth is cause of increasing of density and complexity of urban infrastructure. Street Addressing system is an essential component of city management, quality of citizens' life, and other spheres of city's economy.**

This paper considers peculiarities of Street Addressing system from human and technologies perspectives to be an essential part of the Smart Cities concept implementation.

I. Introduction

Street Addressing (SA) system is an essential component of the city infrastructure. All problems concerning the SA are not currently reflected by proper research or sufficiently described. Such lack is caused by several factors which include, first of all, the shortage of SA in cities of poor countries and the spontaneous growth of urbanization and technosphere. The cities try to solve serious problem in urban management and adopt or upgrade the SA (by implementing the SA program) according to well-known and established methodologies.

However due to spontaneous growth of technologies, these methodologies do not consider new requirements and trends in required extent. The lack of research and resources is observed especially in SA issues relating to implementation of conception of so-called Smart City (SC). This lack is result of fact that the SC conception is young - publications about smart cities started to explode in 2012 [1].

There are several solid methodologies, proposed by the World Bank for implementing the SA programs, aimed for building or upgrading the SA in different cities [2]. These methodologies were based on practical research and implementation of SA programs in dozen of inhabited areas. At the same time the World Bank experts consider SA as a crucial component in initiatives to develop SC [3].

More and more cities are embarking on SA programs [3]. But almost all these programs are not SC oriented. There are several reasons why many cities do not focus on the SC concept in SA programs. On the one hand, there is no methodology for building/updating SA for SC. On the other hand, it seems that ICT is the solution of existing problems, and at the moment the issues of creating SA for ICT needs and, accordingly, for SC is not urgent. The absence of SC definition also creates obstacles to deeply consider it in SA programs.

SA program is complex, expensive, and fundamental: it is city-wide, impacts many urban services, and quality of citizens' life. A look to the future and taking into consideration features of SC can help build SA that would be convenient for developing SC and use of its abilities completely. This paper introduces determination and definition of SA peculiarities that can be used for planning, implementing, and maintaining the SA system with the focus on implementation of SC conception (or shortly SA for SC).

The peculiarities of SA for SC are based on literature analyses, practical experience of developing the SA for Kabul (6.5 mln. inhabitants' area, Afghanistan) [4], [5] and developing the Geographical Information System for Baku (4.6 mln. inhabitants' area, Azebaijan / http://www.-gomap.az). The research includes synthesis of knowledge in urban management, engineering, and ICT.

II. What is Smart City ?

The first question that should be answered is "what is SC?". Cities worldwide play a prime role in social and economic aspects, and have a huge impact on the environment [6]. SC seems to be a direction to where modern cities are being developed, but the final stage of this development is still unclear. Two main drivers are underlying this development: solution of existing problems and applying the new abilities due to availability of new technologies. And as the study shows, Information and Communication Technologies (ICT) are the key players. Meanwhile researchers note that SC should satisfy to people and community needs [6].

Cities worldwide have started looking for solutions to enable transportation linkages, using mixed land, and high-quality urban services with long-term positive effects on the economy. Many of the new approaches related to urban services have been based on harnessing technologies, including ICT, helping create what some call "Smart Cities" [6].

There is no established definition of what a SC is. The SC concept has been developed in three main areas: Academic, Industrial, and Governmental. Although the meaning of Smart City is not settled yet, there is an agreement on the significant role of ICTs in smart urban development [7]. ICT application requires digitalization of data and procedures

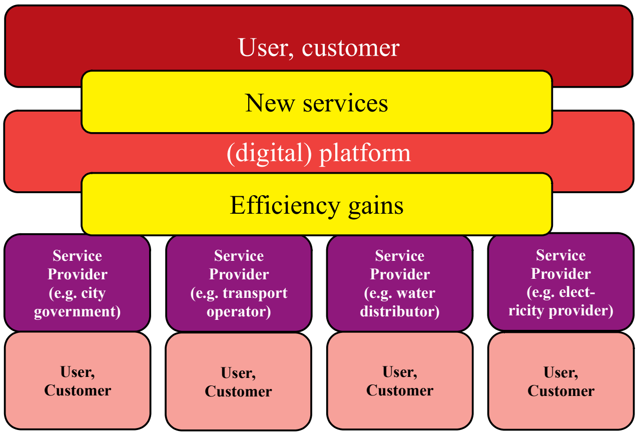

User, customer
New services
(digital) platform
Efficiency gains
Service Provider (e.g. city government)
Service Provider (e.g. transport operator)
Service Provider (e.g. water distributor)
Service Provider (e.g. electricity provider)
User, Customer
User, Customer
User, Customer
User, Customer

Fig 1. The potential of digitalization for cities

that brings opportunities to introduce new services and get efficiency gains. The potential of digitalization for cities is presented by Fig. 1.[1]

Several terms, such as “Digital”, “Instrumented”, “Interconnected”, “Intelligent” are used to describe “Smart City” phenomena [8]. At the same time the SC concept is not limited to the diffusion of ICT, but it looks at people and community needs. Diffusion of ICT in cities has to improve the way every subsystem operates, aiming to enhance the quality of life [6].

Scientists from different disciplines contribute research of SC. As a result, SC is presented from different perspectives, such as engineers’ perspective, economists’ perspective, innovation economists’ perspective, public managers’ perspective, sociologists’ perspective, human ecologists’ perspective. The following practices, all of which come under the “Smart City” label can be identified: Smart Transportation, Smart Environment, Smart Energy, Smart Water, Smart Building, Smart Safety and Security, Smart Health Care, Smart Government/City-Services, Smart Participation, and Connectivity [1].

According to [9], the following components constitute SC:

- Inter-operability of systems (key component);
- Sustainability (energy and water efficiency);
- City-wide connectivity;
- Security;
- Effective transportation;
- Development of private/public partnerships.

There are also several mature models allowing to assess how “smart” the city is. Low level of maturity of the SC, the lack of sufficient and real-time data, and the lack of standardization in previous years hampered development of such models. Citizens are considered as “prosumers” of geotagged data and content affecting cities’ everyday norms and interactions [10].

The study shows that SC can be considered as a sociotechnical system. This statement is studied in details in [8]. In this way SA for SC should be considered from two perspectives: to serve people and to serve technologies. However, it should be mentioned that the technologies are intended to serve people and can support SA. So, we can say that considering SC as a socio-technical system is the solid starting point for determination and formulation of the SA peculiarities.

III. What is Street Addressing ?

Although SA seems as well-known and well-studied conception that does not require research, this is not true. First of all, as mentioned above, there are many cities worldwide that are implementing SA program [3]. Experience gained in these programs forms solid base for research. Secondly, implemented SA programs are targeted mainly to solve existing problems, but not focus to the future technologies. Indeed, the absence of SA has greater negative effect to the urban economy than positive impact that could bring orientation towards the SC. Cost of SA and planning the faster Return of Investments (ROI) are additional factors leading to avoid such orientation.

A. *Purpose of Street Addressing*

SA is quite old and well-established conception. But conception of SC introduces not only new requirements to SA, but also new abilities to its utilization/maintenance.

The main purposes of SA are to [2]:

- enable people to get around the city more easily;
- facilitate the delivery of emergency health, fire, and police services;
- locate urban facilities;
- improve planning and managing municipal services;
- assist with urban planning and programming of investments;
- improve local tax collection;
- enable utility concessionaires to manage their networks more effectively.

These purposes can be grouped as determination of current location and determination of the delivery point.

Development of ICT and other modern technologies introduces new challenges, i.e.:

- Development of Geographical Positioning Systems (GPS), Geographical Information Systems (GIS) and Geocoding services introduces alternative ways of determination the location;
- Decentralization of city services has led to emergence of a wide range of small-sized fixtures, such as ATMs, POS terminals, electricity/water/gas distribution units, and so on. SA should be able to address these facilities (World Bank methodologies consider addressing such fixtures [2]). Many SA systems do not intend to address such small facilities;
- Increasing the density of buildings, premises, facilities etc. leads to raise of volatility (for example joining several sale points to one, then in some years divide up them back) that require SA to be

updated according to the changes. Many SA systems are unable to be updated and reflect real conditions in time;

- Automation of services, such as delivery, transportation, and so on requires integration of SA with the computer systems. Mismatch of SA data with the real situation leads to the loss of computer systems effectiveness.

Such challenges demonstrate the necessity and on the other hand abilities to implement SA programs taking into consideration development of ICT-related technologies, density and granularity of city infrastructure, and proper quality assurance.

Urban information database, which, in conjunction with a SA plan and a street index, can be used for various applications and benefit the population as a whole, local governments, and the private sector. One of actual case studies of SA application is monitoring of epidemics in Maputo, Mozambique [2]. It is obvious that involvement of ICT in SA applications leads to significant improvements.

Implementation of ICT in modern cities creates new environment to process city-related data. Optimization of physical city infrastructure, growth, concentration, and decentralization of this infrastructure introduce new requirements to identify location of the infrastructure units (urban fixtures). An example of problems relating to increase of density in city areas are issues relating to the street vendors. In poor areas problems with street vendors are complex and include lack of territorial management, poor taxation, legal violations [11]. Proper ICT application and SA system could mitigate the problems by increasing manageability of territories where street vendors can be deployed.

New construction technologies are changing shape of cities today. Smart infrastructure and architecture are dominating the shift in the construction industry. In the Smart Cities Connect Conference and Expo in 2017, Skansa USA Chief Sustainability Officer Elizabeth Heider highlighted resilient design and sustainable construction as key aspects of how the construction industry can adapt to the needs of the Smart City initiatives in Santa Monica. A recent survey cited at ReadWrite notes that at least 60% of builders are now aware of Internet of Things (IoT) technologies and another 43% say that IoT will shape how they build in the future. By integrating IoT into the building process, including data capture and analytics, the new infrastructures built are future proof [12]. These facts prove increasing of density and granularity of urban infrastructure and leading role of ICT in construction.

Today in the most of inhabited areas such urban fixtures are identified only by appropriate maintenance organizations. But increase in city density and rapid technological changes lead to intensive modernization in the city infrastructure and require integrated approach in city management. The integrated approach in its turn requires coordination and uniform identification of the fixtures belonging to different organizations. SA for SC can provide here the following abilities:

- Uniform identification of the urban fixtures and places by street addresses, that is convenient for both service workers and ordinary people. That provides them an opportunity to refer to the urban fixtures in case of setting an appointment, accident, or just navigation;
- Information systems of different organizations can be easily integrated resulting in simpler coordination of their activities;
- Urban planning and management on city level can be simplified.

B. *Notation in Street Addressing*

Well-known SA systems are based on human understanding of address notation and meaning. Street addressing is an exercise that makes it possible to identify the location of a plot or dwelling on the ground, that is, to "assign an address" using a system of maps and signs that numbers or names the streets and buildings [2]. As was mentioned above, the current SA programs are not SC oriented. As a result, the address notation is also planned to be convenient for humans, but convenience for technical applications is not considered.

Such disregard often affects not only the external form of visual representation of the address, but also the internal structure of the address. For example, sequential addressing system in developing cities leads to increased fragmentation in addresses resulting in the raise of number of database records, that eventually increases memory consumption and reduces speed of search. This also leads to increase in the likelihood of error and the amount of work to maintain the system up to date. But the address notation can be optimized to be convenient both for humans and for representation in computer systems.

Development of international transportation and globalization processes enhance physical communication between countries and, therefore, increase tourism. Comfort of the city for tourists depends on people's ability to navigate within unfamiliar areas. Moreover, the size, changes in city structure and life style create favorable conditions for the citizens to know only part of their city. Such category of citizens is quite large now and sometimes they become guests in their own city when they visit unfamiliar areas.

SA system has a direct impact on convenience of navigation within the city. It should be mentioned that SA system should be easy to use by humans with no additional means. This requirement makes the city not only more comfortable, but also safer because it allows them to navigate quicker in case of emergency.

SA System should be uniformly applicable to the inhabited localities of all sizes, from smallest settlements up to huge mega-cities. The ability of SA System to be uniformly applied for settlements of wide range of sizes and types makes management of urbanization processes easier throughout the whole country.

IV. *Street Addressing as a Service*

As it was mentioned before, the SA's well-known purpose is to enable finding the required geo-object (or location) and navigate within the city. The nature of this purpose is its function. Thus, the SA can be considered as a service. At a simple look the SA could be just a labeling system for the urban geo-objects. This is a system's passive function. ICT introduces active approach to use the SA and make it the active service. One example of considering the SA as an active service is Location Based Services (LBS). In LBS SA lies under other services.

Conception of SC implies integration of city services. Being a fundamental, the SA system plays an important role in integration of these services and can be an underlying system for many of them. These facts demonstrate that the SA plays more and more significant role not only for humans, but also for functioning of different automated services and systems.

A. Geocoding

Geocoding is the process of converting addresses (like a street address) into geographic coordinates (like latitude and longitude), which can be used to place markers on a map, or position the map. Reverse geocoding is the process of converting geographic coordinates into a human-readable address [13].

Geocoding is a part of many systems that relate to urban management, navigation and similar applications. Quality of geocoding directly depends on the SA system. Geocoding is one of early and well-known geo-informatic services that relates to SA and can be considered even as a part of SA system. Indeed, single match between street addresses, location and geographical coordinates allows to state this.

B. Geographic Information Systems

A Geographic Information System (GIS) is a computer system for capturing, storing, verifying, and displaying data related to positions on Earth's surface. GIS can show many different kinds of map data, such as streets, buildings, and vegetation. This enables people to see, analyze, and understand patterns and relationships more easily [14]. This makes GIS the main computer application type for implementation of SA program or developing/maintaining of geographical services.

C. Street Addressing Programs

Rapid urban growth and expansion have made cities unmanageable, especially in the outskirts. Informal settlements have mushroomed. However, with the advent of e-commerce, increased efforts in building disaster resilience and advancements in geo-spatial technologies, street addressing methods and applications are undergoing a revolution today [3].

Studies of the World Bank show that there is a hundred of cities implementing the SA Programs [3]. There are three types of cities: (I) having a well-established SA system; (II) having a SA system that is not satisfies to the requirements; (III) having no a SA System. Speaking strongly, well-established SA system in this statement means a SA system in their classical understanding, without orientation to the SC conception.

The SA Programs in their turn can be divided into two categories: establishing a new SA system and upgrading the existing one. Since the SA system is a city-wide one, changing or creating of SA system is expensive. Thus, the SA Programs should take into account both benefits and costs. Because the high cost of implementation of SA Programs, it makes sense to take into account their orientation towards SC early at the planning stage.

V. Peculiarities of Modern Street Addressing System

As it was shown above, the SC can be considered as socio-technical system. This implies that SA can be viewed from two perspectives - the use by humans and the use by technical solutions. Since the purpose of the SA is provision of information about location or point of delivery, in second case we can speak about perspective of use of it in Information Systems (IS).

Considering the quality of IS we can confirm that it depends on its constituent parts. This means that quality of SA contributes the whole quality of IS which uses it. Thus, reviewing the methodology of quality assurance for IS can help get clue how quality of SA for SC can be evaluated. The IS peculiarities can be assessed in accordance with so-called "-ilities": reliability, usability, safety, etc. [15]. Similar approach can be used to assess the SA peculiarities.

A. Human Use Perspective

The SA system should be able to provide services directly to people in convenient manner without special means (this requirement was considered above in this paper). There are cases when existing SA Systems are obsolete and no longer meet such requirements. For example, such problems occurred in Baku, Azerbaijan where old SA System was being upgraded by the SA Program [3].

Quality requirements to SA from human perspective are intuitively comprehensible because everybody has been using SA in their lives. Major SA "-ilities" from human use perspective are presented in the list below:

- **Readability**. Ability to read address notation (on the wall, postal envelope, package label and so on), remember it and retell it to other people. This ability relates both to one address and to range of addresses (for example several entrances on the same street). This feature also relates to the same requirements for foreign people. (Kabul Street Addressing Program supposes to use labels on 3 languages: Dari, Pashto, and English [5]);
- **Navigateability**. Ability for people to navigate within the unfamiliar areas of city without additional tools. This quality is especially important for tourists and city visitors. Generally, navigateability includes easy determination of current location, lo-

cation of desired geo-object, and ability to plan route of movement;

- **Addressability**. Ability to assign address to wide range of geo-objects, including, but not restricted:
  - ○ Country, Region of the country / Province, City;
  - ○ District of the city, Plot;
  - ○ Thoroughfare (street, road, alley, canal, railway, …);
  - ○ Dwelling, Complex Construction;
  - ○ Points: Entrance, Simple Construction, Installation, Urban Fixture, ATM, Historical Monument/Landmark, and so on.;
- **Correctness**. Basically this feature requires absence of the same addresses for different geo-objects and addresses with possible ambiguity, absence of "empty" addresses, that are pointing to nothing, and absence of geo-objects without addresses;
- **Maintainability**. Basically this feature requires ability to keep the system up to date with the lowest operational cost. Maintainability shall not impair all other qualities;
- **Cost**. This feature relates to capital cost of creating or upgrading the system. It can be decisive factor to start SA program or not.

B. Information System Perspective

The modern SA system is intended to be used in/by technical solution. Due to informational nature of SA, it is enough to consider use of SA in/by IS. Thereby it can be reviewed through IS perspective. Due to the fact that SA is considered as a service, its functionality can be realized as an ICT service (Web online service for example) providing data to other services (consumers). It is obvious that if SA service satisfies the principles of ICT quality, it will contribute total quality for the final IS.

The SA system implemented as an ICT service can be considered as a solution, containing software and hardware components. We can assume that SA system is a software solution whereas hardware can be placed behind the software. Hardware in this way can be considered just as a means to provide the possibility to run the software. Finally, we can assume that IS quality can be equal to software quality, and software quality assurance methodologies can be applied to provide quality of SA service. Requirements to the architecture and hardware can be formulated according to requirements of mission-critical systems (in this case emergency services can relay on it).

David Chappell selected three aspects of software quality: functional, structural, and process [16]. According to this division the following qualities can be highlighted:

- **Functional quality**, that make the product satisfying to the customer needs;
- **Structural (or internal) quality**, that is software structural quality refers to how it meets non-functional requirements;
- **Process quality**, that relates to the management, development, operating and maintaining (O&M), and other processes.

Taking into account this reasoning of SA service quality from the ICT perspective, the following components can be considered:

- Functional quality – SA services software;
- Structural (or internal) – SA services software and hardware;
- Process quality – SA services software and hardware, maintenance, administration, and development processes.

As it was mentioned above, quality of software can be defined by different so called "-ilities": usability, reliability, availability etc. Theoretically, there may be a lot of "-ilities" [15]. But in practice many of them can be combined. The purpose of this research is to show the universal approach that could solve specific problems relating to SA for SC development. More deep analyses can be carried out, more detail level of "-ilities" can be determined for each practical case. Such research can be done within SA programs for different cases and form base of practical experience.

Software quality attributes and trade-offs are described in [17]. So, generally the quality of a SA service solution can be characterized by providing the following "-ilities" (these are additional requirements that should be combined with corresponding well-known software quality "-ilities" in each specific case):

- **Reliability**. The system should be correct, persistent, and consistent. Data should be updated according to changes in urban development and reflect actual situation;
- **Efficiency**. Efficiency of using the solution for easy use by humans and in integration with computer systems;
- **Usability**. Generally, this feature is the same as "Readability" from Human Use Perspective (see chapter above) but supplemented by abilities to be used by IS;
- **Integrity**. All SA parts regardless the size, type and location of an inhabited area should meet all SA requirements. It means that SA applied in such areas as village, town, city, city district should be built by the same methodology and meet all requirements;
- **Maintainability**. The same as "Maintainability" from Human Use Perspective (see chapter above) but supplemented by requirements for IS. For example data integrity might require implementation of maintenance operations according to Atomicity, Consistency, Isolation, Durability (ACID) principles;
- **Flexibility**. The system should be easy to use in any real-life application. "Flexibility" is the main feature that determines in which extent the SA could be used in SC;

- **Testability**. The system should be easy to check. ICT provides additional possibilities to provide this feature. For example, sequences of house numbers can be tested according to their location along the street;
- **Reusability**. Ability to reuse parts of the addresses - could also relate to compactness of the addressing data because reuse reduces the data size within the SA system;
- **Interoperability**. Ability to be integrated with different services and systems. Generally, it could be considered as the same as "Flexibility";
- **Survivability**. Ability to keep qualities and remain intact as environment changes, i.e. reconstruction, emergency situations, natural disasters, and so on;
- **Safety**. The system should be safe to use. It also contains resistance to possible malicious activities, such as vandalism;
- **Manageability**. This feature is practically the same as one to measure/evaluate the SA peculiarities quantitative or qualitative, since it allows planning and monitoring SA related activities;
- **Functionality**. Ability to satisfy all human and IS needs according to the purpose;
- **Scalability**. Ability to be applied to inhabited places of any size;
- **Portability**. Could be considered as the same as scalability, but it relates more to different types of area and urban fabric;
- **Expandability**. Ability to add additional functionality;
- **Supportability**. Practically the same as "Maintainability".

All considered "-ilities" relate to all three components of David Chappell's aspects of quality: functional, structural, and process. At the same time all of them are tightly interdependent. For example, if SA is incorrect, it immediately becomes useless. Improving the readability can reduce addressability. Thus, trade-offs between the features should be achieved for each specific case. Specific cultural features of the area of SA program should also be taken into account.

C. Vertical vs. Horizontal Referencing

Classical SA systems have tree-like structure where leaf level is presented by house/entrance numbers and root is the country. As all elements of SA system have actual direct reference to the geo-objects located on the Earth, we can say that these references are vertical (for example country->district->city->street->house). As these references are obvious, we can consider them as explicit, or absolute references by their nature.

But there are also latent references presented in SA that can be named as relative or implicit. These references are between geo-objects grouped into structures where address/name/number of one object can give information about location of another one. The main type of such geo-objects is door numbers. For example, each door number provides information that a door with next number is located nearby. Streets or city districts can also provide such references if they are named according to their location. Because these references are between geo-objects of the same level, we can say that these references are horizontal.

Vertical references to the geographic location, in fact, relate to the SA's main purposes mentioned above. At the same time the horizontal references mainly determine the "Navigateability". For example, if somebody is on the street with numbered doorways he/she can obviously find the direction and estimate distance to the door with specific number located on the same street.

There are several other means to increase the navigateability that can include the directions to the geographic location or even culture specific features. The World Bank methodology proposes natural geographic objects, such as coastal line, rivers, mountains and so on as references determining starting points of numbering or naming of elements of the SA system [2].

Generally, there are three main components of classical SA that may refer to the geo-object:

- Regions, if they divide bigger area according to the cardinal points or other geographical features;
- Thoroughfares, if they are located along real geographical object, such as alley, boulevard, river;
- House/entrance numbers linked to the corresponding thoroughfare.

And these main components of classical SA can be referenced horizontally as the follows:

- Regions, if they logically divide bigger area according to some rule;
- Thoroughfares, if they are ordered according to some rule;
- House numbers if they constitute a sequence or ordered set.

Generally, the vertical reference helps find geo-object or location right away while horizontal reference helps travel. The SA systems are based on both vertical and horizontal references. SA for SC should provide these features in the same or greater extent. When planning SA special GIS applications could help to design the SA in way it has a the best possible navigateability by calculating names/ numbers/directions of SA elements and optimize their horizontal relations. Such research could have place in SA program but it is out of scope of current research.

Vertical and horizontal references of SA system relate to the structural component of David Chappell's aspects of quality. They are not related to addressing objects according to their physical elevation above or below to the ground. The growth of cities in height and depth (underground) is also an actual problem of addressing. But at the moment such problems relate to in-building navigation/addressing, while SA relates to objects accessible from the ground. At the same time expandable SA systems could include also in-building objects and physical vertical addressing. For example addressing by the height is seem actual in near future allowing automate delivery by the drones.

D. Modern Postal Addresses

Many people consider SA and Postal Addresses (PA) as the same conceptions. But this is not true. As it was stated above, purposes of SA system can be specified as determination of the location and the point of delivery. PA can serve the same functions, but not in all cases. PA are delivery-oriented conception and should be considered within Postal Services (PS).

Due to deep penetration of ICT into PS, their modern implementation should be also considered. Purpose of PS is to deliver the package to specific person, organization, or location. If the package is sent to specific person or organization, PS does not always need the street address to be stated as a destination. If destination uniquely identifies to whom or where the package should be delivered, it does not require to indicate the street address.

For example, the destination can be indicated as "Ministry of Education" of specific country. The PS can determine location of the ministry using underlying SA service or even provide delivery without determination of the address because the destination in this case is well-known. Another example is money transfer. It could be considered as a bank operation, but from PS point of view this operation can be considered as a delivery that does not need street address at all.

Modern PS should include ability to address one point of delivery by several methods. A requirement to use only one address for each delivery target introduces a limitation. Considering PS and SA as different services was very useful for Kabul, where the SA was broken and even ministries had no Street Addresses. This separation allowed addressing organizations by their names [4], [5].

So, the modern PS should be based on SA system and consume it as a service. And modern PS can be classified as a Location Based Service (at least its informational part) [18], [19] and use Geocoding (which was mentioned above) because it determines location by the information about the geo-object. Such vision of PS looks very convenient for SC, but the main point here is that the PS is not the same as the SA and strongly depends on it.

E. Role of Street Addressing in inter-operability of systems

SA information is part of data used in many of urban systems. There is number of SA applications that use addresses as key data (some examples are listed in description of "Flexibility" feature in chapter "Studies of the World Bank" below). Being ground data for services and systems, SA plays main role in inter-operation and integration of the systems. Inter-operation of the systems as the key component constituting a SC [9].

The feature of modern SA system to address each geo-object with one-to-one relationship allows considering the street address as an identifier (ID) for corresponding geo-object or as a part of the identification information. Moreover, ID in this situation can be considered as a globally unique identifier (GUID) because full street address is globally unique. Therefore, the modern street address can be considered as a GUID and provide identification of geographical objects and constitute a reference system (role of primary/ secondary keys) at inter-system level in distributed IT solutions.

In terms of databases street address could be considered as type of the primary key. For integrated services and external referencing to geo-objects it could be considered as a secondary key. But it should be noted that this assumption must be confirmed at modeling stage for each specific solution because SA changes and features can affect the data integrity.

So, the modern SA has abilities that should not be easily shrugged off when designing architecture of ICT applications. Avoiding of non-required additional ID's can help reducing complexity of the solutions, database size, and communication traffic. At the same time such ability shows how important the SA's role for SC can be.

F. Other Street Addressing systems

Several attempts were made to develop an universal SA system or SA system for SC. These attempts were not successful by several reasons. First of all, as was mentioned above, SA and PA are different conceptions. Authors describe their systems as SA although their systems were based on requirements of PS. Another problem is attempt to build complex code for the address that reduces its readability.

For example the MappGuru that is positioned as an universal addressing system [20] looks like codification system based on requirements of PS. Building a SA on base of this solution most probably will show poor readability because long codes are not memorable, and poor navigateability because people navigate along the streets more convenient than according to proposed zoning. Anyway this is a brief observation and specific research should be implemented to evaluate this solution. Considering model seems a good tool for the evaluation this and other solutions.

There are number of standards that are relate to the SA. In [21] a number of national and international address standards are presented. This paper concludes that addresses do not have a single common feature but rather a 'family resemblance' in the Wittgenstein sense: a complicated network of overall and common similarities of detail. An overarching abstract address standard comprising different parts, each describing a specific set of these similarities would contribute towards a better understanding of these similarities and improve correct address usage and data exchange.

Considering model is not proposed as part of standard. But it could growth to become a practice, allowing evaluate of SA program for each specific case.

VI. Use Cases

The system of peculiarities/features, proposed in this paper is built on experience of analyzing the studies and implementation of real SA-related projects. Overview of this experience seems to be useful supplement to the current study.

A. Studies of the World Bank

A very rich experience in studies and implementation of the SA programs for different cities worldwide was collected and published by the World Bank in [2]. The value of this study includes not only SA itself, but also its application to solve the actual urban problems. Many SA applications use ICT and form good use cases for research. A first look at the methodologies and experience, presented in [2] from perspective of considering model gives the following features:

- **Readability**. Is provided by codification and street signs installed;
- **Navigateability**. Is provided by dividing the city by zones, ordering of streets and doorways numbering;
- **Addressability**. SA programs assign a number to buildings (houses, places of business, facilities) and doorways. Addressing may be extended to urban fixtures such as public standpipes, fire hydrants, or waste transfer points;
- **Correctness**. Is provided by codification system and GIS;
- **Maintainability**. Is provided by organizational measures, such as regular inspections;
- **Cost**. Many SA programs are intended for poor cities. Thus cost plays here decisive role. At the same time optimization of the cost is actual for all cities;
- **Reliability**. Is provided by codification system;
- **Efficiency**. Considered mainly from human perspective;
- **Usability**. Considered mainly from human perspective;
- **Integrity**. Is provided by codification system;
- **Flexibility**. Using in different SA applications, such as civic identity, urban information (GIS, Geocoding), health care, street system management, household waste collection, inventory of municipal built assets, investment planning, improving the performance of the existing tax system, land tenure, slum upgrading, concessionary services, postal services, economic development, etc;
- **Testability**. Metric and decametric systems seems more testable than sequential;
- **Portability**. Application to cities with different shapes and landscapes;
- **Reusability**. Is provided by codification system and tree-like addressing (city, district, street, doorway);
- **Interoperability**. Can be used in GIS/Geocoding systems and integrate with different services and applications;
- **Survivability**. Metric and decametric systems are more survivable than sequential because doorway number correlate to their actual position;
- **Safety**. Is provided by installed or painted street signs and plaques;
- **Manageability**. Is provided by codification system;
- **Functionality**. Is provided by street signs and GIS/Geocoding;
- **Scalability**. SA programmes has been implemented for cities with population no more than 1.3 mln.;
- **Expandability**. Is provided by integration with IS;
- **Supportability**. Is depends mainly on technologies of manufacturing of street plaques, installation/ painting street signs, and integrated technologies.

Several SA types and methodologies, presented in [2] can be a basis for their adaptation to some specific cases. So, the SA based on metric house numbering was used for development of the SA program for Kabul, Afghanistan.

B. Case of Kabul, Afghanistan

The SA program being implemented in Kabul is an outstanding example. The city SA was completely destroyed during the long war. The SA program covers area populated by 6.5 mln. and area of New City designed for additional 3 mln. inhabitants. The uniqueness of this case is concluded in fact that it is a huge city, capital that required implementation of the SA system from the scratch. But existing methodologies of building new SA system were developed for small developed cities. Kabul required specific SA considering different specific factors.

The methodology for the SA program was developed by the Ministry of Communications and Information Technologies (MCIT) within development of LBS. This demonstrates that development of ICT can be a driver for development of the SA program. As a result, the methodology of developing the SA system intended to be used in/by IS was elaborated.

Another challenge of development and implementation of the SA program in Kabul was deep difference in city infrastructure in center of city and slums within the outskirts. The SA should keep all its features at minimal cost when applying in slums. Specific challenge was integration of established SA of New City with new SA being developed for Kabul and Afghanistan as total.

A brief study of the Kabul case from perspective of considering model demonstrates specific features that should be emphasized:

- **Readability**. Use street signs in three languages: Dari, Pashto, English;
- **Addressability**. SA system supports addressing of Central Business District (CBD) area and slum area. Urban installations can also be addresses;
- **Maintainability**. Considers different approaches for CBD area and for slum area;
- **Cost**. Considers different approaches for CBD area and for slum area;

- **Efficiency**. Considers support for humans and LBS;
- **Usability**. Considers support for humans and LBS;
- **Testability**. Metric system was selected for doorway numbering;
- **Portability**. Application to other Afganistan cities was considered;
- **Interoperability**. Integration with partially existing addressing and SA of New Kabul is considered;
- **Survivability**. Metric system was selected for doorway numbering;
- **Safety**. Addressing of slum areas was considered in depth because these areas are the most exposed to different threats;
- **Manageability**. Division to areas with different city fabric increases manageability;
- **Scalability**. Specific approaches allows to develop SA programme for this large city. Application for smaller places is supported also;
- **Supportability**. Slum areas were studied in depth to provide high level of Supportability.

The experience of SA program development for Kabul allowed focusing on integration of the SA with ICT.

C. Case of Baku, Azerbaijan

Baku is an example of city where former addressing system could no longer satisfy the modern needs. The study covers experience of working with new SA in Baku including processing of geographical data and building electronic maps for in-house GIS, online map, and Portable Navigation Devices (PND) within the project www.GoMap.Az (http://www.gomap.az). Developers faced the problems of packing the SA information into data structures and automatically provide data quality control. The problem was rooted in a large number of entrance signs where digits are supplemented by the alphabetical letter. This significantly increases the data fragmentation that is a sign of fact that structural quality of the SA system is not enough.

After reviewing of SA standard applied [22] some conclusions were made. In terms of considering SA features, the SA program for Baku was elaborated with some trade-offs. Readability was selected as the primary SA feature, but other features were weakened. And it seems that root cause of current situation with the SA is necessity to transfer from the former SA system to a new one. Achievement of the best possible readability required keeping the sequential entrance numbering. But changes in economic situation have led to constant combining or splitting premises resulting changes in the city fabric patterns and finally ended in fragmentation in entrance numbers.

Thereby Baku SA program can be considered as a case where the SA was impacted by changes in economy and urban infrastructure. This means that the SA programs worldwide should consider development and implementation with the expectation of constant changes in urban infrastructure, because development of technologies is a global process. This requirement is also true for realization of SC conception. Focusing on SC in its turn can mitigate troubles that arise due to changes in city infrastructure.

D. Commerce-Science Problem

As it was mentioned before, the SA program is expensive and cities plan the ROI for a quite short period. Planning of SA for SC in one hand promises strategic advantages in the future, but in other hand introduces additional expenses. This problem is obvious. But there is another fundamental problem being discussed in Artificial Intelligence (AI) world but also relates to all modern technologies.

In his open letter signed by many prominent researchers and developers Max Tegmark shows that a small change in technology is available at the business level, which is interested in the speedy implementation of the solution without an in-depth laboratory analysis of the consequences. Meanwhile, a change in technology can have a tremendous impact on both human life and the environment [23]. One of examples demonstrating that the problem is actual for the SA, is a case with “what3words” service (https://what3words.com). The service was introduced and implemented without proper research. Avoiding proper peer-review before implementation resulted in criticism, negative feedback, and loss of market positions [24].

The root of problems with “what3words” service is that the service was introduced as a SA system, but it is actually an LBS. If the service was analyzed for SA features, the results would show that it cannot serve as an example of a good SA system. According to the proposed system of the SA pecularities, the three most poorest qualities of “what3-words” service as a SA are: Addressability – because it can address only point geo-objects, Navigateability – because it does not support horizontal references, and Structural Quality, because the function that provides vertical references is complex and close (based on proprietary closed source codes), and there is no horizontal references making internal structure more solid.

At the same time, the convenience of service use and its advantages in specific use cases, as well as positioning it as an LBS, but not a SA, could give it a chance to become one of the integral parts of the SC services.

VII. Conclusion

Rapid development of technologies leads to intensive changes that are available at the business level and has a significant impact to human life and environment. It means that all changes in technologies should be deeply researched before being implemented. Such well-known methodologies supported by established sphere as Street Addressing requires research for each specific case in context of available technologies and future trends. Role of Street Addressing in implementation of Smart City conception is considered primarily in provision of quality services for people and in integration of different city services.

Simple system allowing assessment of Street Addressing qualities targeted to the future development is proposed in this paper. This system is considered as a starting point to carry out more deep research within the Street Addressing Programs. Experience in such research in several use cases would allow updating existing Street Addressing Program methodologies and making it possible to implement them for wide range of cities moving them closer to Smart City level.

Acknowledgment

The paper has been completed with the kind support of www.GoMap.Az project implemented by SINAM Ltd.

References

[1] M.Finger, “Smart City – Hype and/or Reality?”, IGLUS Quarterly, vol. 4, issue 1, June 2018.

[2] C. Farvacque-Vitkovic, L. Godin, H. Leroux, F. Verdet, R. Chavez, “Street Addressing and the Management of Cities”, The World Bank, Washington DC, ISBN 0-8213-5815-4, 2005.

[3] S. Dharmavaram, C. Farvacque-Vitkovic, “Street addressing - a global trend”, 2017 world bank conference on land and poverty, The World Bank, Washington DC, 2017.

[4] D. Gakh, “Analysis of World-Class Location Based Services, Survey Report of Kabul City, and Assessment on the requirements of the Numbering and the Addressing Systems”, Ministry of Communications and Information Technology, Islamic Republic of Afghanistan, Kabul, 2014.

[5] D. Gakh, “Essentials of Street Addressing Programme for Kabul City”, Ministry of Communications and Information Technology, Islamic Republic of Afghanistan, Kabul, 2014.

[6] V. Albino, U. Berardi, R. Maria Dangelico, “Smart Cities: Definitions, Dimensions, Performance, and Initiatives”, Journal of Urban Technology, vol. 22, no. 1, 2015, pp. 3–21.

[7] F. Mosannenzadeh and D. Vettorato, “Defining Smart City. A Conceptual Framework Based on Keyword Analysis”, TeMA, May 2014.

[8] H. Kopackova and P. Libalova, "Smart city concept as socio-technical system", 2017 International Conference on Information and Digital Technologies (IDT), Zilina, 2017, pp. 198-205, doi: 10.1109/DT.2017.8024297.

[9] P. Simpson, "Smart cities: understanding the challenges and opportunities", SmartCitiesWorld in association with Philips, Jan 2018.

[10] V. Moustaka, A. Maitis, A. Vakali, L. Anthopoulos, "CityDNA Dynamics: A Model for Smart City Maturity and Performance Benchmarking", WWW '20: Companion Proc. of the Web Conf. 2020, pp. 829–833, doi: 10.1145/3366424.3386584.

[11] "Do street vendors have a right to the city ?", Centre for civil society, IGLUS, October 2019.

[12] J. Brad, "Examining the Role of the Construction Industry in Building Smart Cities", IGLUS, Apr 2020.

[13] Google Developers / Geocoding API / Get Started, https://developers.google.com/maps/documentation/geocoding/start, accessed in April, 2020.

[14] National Geographic, GIS (geographic information system), https://www.nationalgeographic.org/encyclopedia/geographic-information-system-gis, accessed in April, 2020.

[15] J. Willis, S. Dam, “The Forgotten “-ilities””, SPEC Innovations, 2011.

[16] D. Chappell, “The three aspects of software quality: functional, structural, and process”, David Chappell & Associates, 2011.

[17] P. Berander, L.-O. Damm, J. Eriksson, T. Gorschek, K. Henningsson, P. Jönsson, S. Kågström, D. Milicic, F. Mårtensson, K. Rönkkö, P. Tomaszewski, “Software quality attributes and trade-offs”, Blekinge Institute of Technology, 2005.

[18] S. Steiniger, M. Neun, A. Edwardes, "Foundations of Location Based Services", University of Zurich, 2011.

[19] A. Edwardes, “Geographical Perspectives on Location for Location Based Services”, LOCWEB '09, in Proc. of the 2nd International Workshop on Location and the Web, April 2009, article no.: 5 pp. 1–4, doi: 10.1145/1507136.1507141.

[20] V. Rwerekane, M. Ndashimye, "The MappGuru, a universal addressing system", SCA '18: Proc. of the 3rd International Conf. on Smart City Applications, October 2018, article no. 19, pp. 1–7, doi:10.1145/3286606.3286796.

[21] S. Coetzee, A. Cooper, P. Piotrowski, M. Lind, M. Mccart Wells, E. Wells, N. Griffiths, M. Nicholson, R. Kumar, J. Lubenow, J. Lambert, C. Anderson, S. Yurman, R. Jones, "What address standards tell us about addresses", South African Journal of Science, 2007, 103 (11), pp. 449-458.

[22] “Nəqliyyat infrastrukturu obyektlərinə adlarin verilməsi, daşinmaz əmlak obyektlərİnə ünvan nömrələrinin təyini, ünvan lövhələrinin yerləşdirilməsi”, Azərbaycan Respublikası standartlaşdırma, metrologiya və patent üzrə dövlət komitəsi, AZS 749-2013, 2013.

[23] M. Tegmark, “An Open Letter: Research Priorities for Robust and Beneficial Artificial Intelligence”, https://futureoflife.org/ai-open-letter, accessed in April, 2020.

[24] D. Gakh, “A review of street addressing systems within the realization of conception of Smart City”, in Proc. Internet-Education-Science 2020, Vinnytsia, May 2020, pp. 96-98.

DOI: 10.15439/2020F102
ISSN 2300-5963 ACSIS, Vol. 21

Potentials of digital approaches in a tourism industry with changing customer needs – a quantitative study

Felix Häfner
Aalen University of Applied Sciences, Beethovenstr. 1, 73430 Aalen, Germany
Email: kmu@hs-aalen.de

Ralf-Christian Härting
Aalen University of Applied Sciences, Beethovenstr. 1, 73430 Aalen, Germany
Email: ralf.haerting@hs-aalen.de

Raphael Kaim
Aalen University of Applied Sciences, Beethovenstr. 1, 73430 Aalen, Germany
Email: raphael.kaim@hs-aalen.de.

***Abstract*—In recent years, the tourism industry has been undergoing a period of transformation, not least due to innovative and disruptive business models entering the market. Companies are increasingly focusing on customer needs and try to meet them with digital approaches. The early recognition of rapidly changing needs and their reactions is challenging. This paper examines four factors influencing the potentials of the changing customer needs and ongoing digitization in the tourism industry. A hypothesis model developed in previous qualitative research is examined in this paper with a quantitative approach using structural equation modeling. 157 responses from the target group were analyzed to test the factors digital marketing, data mining, digital services and online travel communities. The results of this paper show that digital marketing and data mining have a positive as well as highly significant influence on the potentials of digital approaches in a tourism industry with changing customer needs.**

I. Introduction

NOWADAYS innovation brings fewer material products and much more digital services and processes which optimize and expand the value-added process as a result of digitization. The Internet serves as a primary source of information, with great importance also in the tourism industry. Modern digital business models are establishing on the market and contain significant advantages over traditional business models [1]. Electronic data transmission is particularly suited for this purpose, allowing processes to be presented with lower costs and more efficiently [2].

Digital booking channels and services make it easier for people to access a wide variety of offers directly via digital channels. These channels are important to get in touch with the client right at the beginning of the customer journey [11]. In terms of changing customer needs through digitization [4], mobile devices have taken an essential position. They enable not only the gathering of information, but also booking and purchasing of touristic products during the journey [5].

This study essentially refers to the changing customer needs due to digitization in tourism in German-speaking countries. Digitized processes in tourism are progressively influences by new technological concepts as Big Data, Cloud- and Mobile Computing, Internet of Things (IoT) or Social Software [6]. In order to fully exploit the potential of digitization, enterprises in tourism have to adapt new digital approaches. The expansion of digital technologies enables the emergence of new business processes or even entire business models [7].

In addition, digitization has a large number of definitions, depending on the dimensions. For example, the presentation and provision of content on websites, sales channel functions (e-commerce) or business process integrations, so-called e-business, right up to the already mentioned new business models with virtual products or services are specifically defined [6]. The World Tourism Organization (UNWTO) defines tourism as "activities of persons travelling to and staying in places outside their usual environment for a period not exceeding one year without interruption for leisure, business or certain other purposes". The definition shows that tourism involves a wide range of activities, including travel preparation, arrival and stay at the destination itself as well as the return. The follow-up of the trip, even if it takes place at home again, is part of the concept [8]. An additional feature, which touches the tourism industry especially, is the emotionality of its customers [3]. For a better understanding of this change, the needs of a tourist can be systematically divided into categories [9]. One of the most widespread and fundamental considerations emerges from Maslow needs pyramid. The classification is made according to their urgency [10]. The current change, mostly influenced by social media, shifts the structures with regard to the satisfaction of needs, so-called group dynamic effects. This leads to an increased need and subsequent purchase. It also has an impact on tourism industry [11]. In most cases, the form of tourism today is pleasure and thus the highest level of the Maslow pyramid as a motive of recognition and prestige [8].

The paper is structured as follows: In the next section the research design and methodology are presented. Then the results are presented and compared with the previous qualitative study [12]. The paper concludes with the limitations and an outlook on potential further research.

This work was not supported by any organization

II. Research and Methodology

In this chapter the authors present hypotheses derived from current international literature and an existing qualitative study [12]. The data collection of the preceding qualitative study is based on semi structured expert interviews. Using the Grounded Theory approach, important determinants could be worked out. Based on this, a quantitative study using structural equation modelling is conducted to validate the previous results. After the design the research method is defined with its data collection and analysis process.

A. Research Design

In order to work out the potentials of changing customer needs in a digital world in tourism, the investigated influences were collated in a hypotheses model. Potentials in tourism can arise from new technologies and thus a change in customer needs. On the one hand, the overall view is important to see how customers assess the situation. On the other hand, various probability-based programs can be used to calculate the influence of differently selected determinants on the research question. The model with its results is shown in figure 1.

On consideration to the mentioned research design, four determinants were identified and evaluated which are described as follows. The investigation of the influence on the research question shows that various digital technologies offer significant potential. It seems to be possible in the future to meet changing customer needs on the basis of digitization in tourism. The results of the structural equation model are explained in the following chapter using the determinants considered and based on the existent qualitative study by Reichstein and Härting [12].

H1: Digital Services contain potential due to changing customer needs in a digital world in tourism.

Thanks to mobile technologies, actions can be carried out, information retrieved or trips booked and changed at any time and any location. The more individually the customer receives the information he is looking for, the faster and better he can be advised by digital services. Broad contact and information possibilities on the part of the companies offer the customer the satisfaction of individual needs [13]. In every stage of a journey tourists needs information, which is usually available on the Internet. For companies, very good digital services for providing information can be an advantage on the market. In combination with new technologies (e.g. Virtual Reality) to provide the best digital information and collecting data in the same time, digital services will be an important element in the future [14].

H2: Digital Marketing holds potential due to changing customer needs in a digital world in tourism.

With regard to research on digital marketing in tourism, some online platforms deserve special mention. The result of a survey from 2017 examined the most important social media for companies. With over 60%, Facebook is the most important social media platform, followed by LinkedIn and Instagram [15]. In this context, the trend is also towards persuasive influencers and bloggers. But also people how know each other from real life influence another through social media. The phenomenon of Web 2.0 is accompanied by a change in the market in which customers are increasingly integrated into processes of companies in order to participate actively.

H3: Data Mining shows potentials due to changing customer needs in a digital world in tourism.

Data mining is the process of identifying correlations and patterns from collected data that provide new insights. This requires large amounts of data. The MIT Technology Review sees data mining as one of the ten emerging technologies that can change the world [16]. The absolute majority of tourism customers (93%) visit tourism specific websites and leave evaluable data before the trip [17]. Companies in tourism sector have huge amounts of data at their disposal which can be used in order to generate helpful insights. With intelligent use of the gained information, there is a possibility to improve strategies and achieve competitive advantages.

H4: Potential due to changing customer needs in a digital world in tourism is contained in Online Travel Communities.

Before a trip, tourists like to inquire in designated online travel forums in order to get information about a certain destination. Impressions, experiences and exchange with other travelers seem to be very credible and are often collected in these forums. Providers can use the information on the platform to make the tourist experience increasingly more pleasant. The exchange between customers creates trust and clarity [18].

According to the literature review, all selected determinants have a positive effect on the research question and thus represent potential.

B. Research Methods and Data Collection

The research method in the form of a quantitative survey was chosen for various reasons. A subsequent quantitative study can be optimally used for testing and validating hypotheses [19]. The open source software Lime Survey was selected to conduct the survey. The study is intended to provide information in order to identify and exploit potentials in tourism.

There is a sample of 157 usable answers collected from the respondents (n=157). The data was collected over a period of three months in summer 2019. Participants for the survey were contacted personally and by telephone before a link to the survey was sent. Furthermore, social networks and platforms were used to draw the attention of groups with appropriate topics. 70% of the participants in the online survey were younger than 38 years. Only just under 8% belong to the age group older than 58 years and tend to have less affinity with the Internet. The different genders are evenly distributed among the participants of the survey (60% female and 40% male). What may well affect tourism is the family situation. 30% of the interviewees have children in the household, which influences the holiday planning and possibly leads to other priorities.

In order to investigate the effects of the determinants, the respondents could apply with five different characteristics. The Likert scale has a range from one to five (1 total agreement to 5 total disagree). To analyse the set-up hypotheses and the collected data sets, structural equation modelling (SEM) was performed. With the analysis tool SmartPLS the influence of constructs on the research question, whether there is potential of changing customer needs in tourism against the background of a digital world, can be tested. Before hypotheses can be confirmed or disproved, typical quality criteria must be checked. Indicators of the quality are Composite Reliability (CR), Cronbach's Alpha (CA) and Average Variance Extracted (AVE) [21].

III. Results

The relationships between different variables can be calculated with SEM [20]. SEM is seen as a second generation of multivariate analysis to provide a deeper insight into the analysis of the different relationships. The measurement model validates the latent variables and the structural equation model analyses the relationships between the research model and the latent variables [20].

A. Results of the SEM

After the data sets of the empirical sample had been evaluated with the analysis tool SmartPLS, the following results turned out (Figure 1).

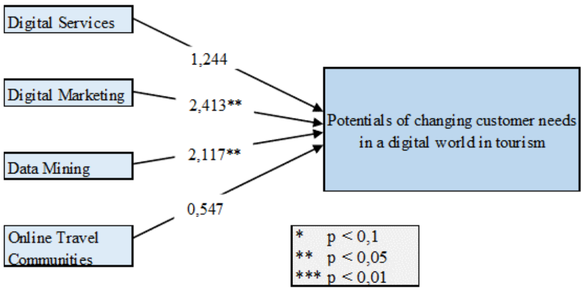


Fig. 1 Structural Equation Model

Due to the very diverse tourism activities described at the beginning, some minor influences that are not included in the model influence the result of the evaluation. Nevertheless, the coefficient of determination R² is given as 0.247 for the model and is therefore declared sufficient according to Chin [22]. According to the analysis with SmartPLS, a significant potential for changing customer needs in tourism is found in two of the four evaluated determinants which have a P-Value with p < 0,05. In summary, the determinants digital marketing (2,413) and data mining (2,117) can be described as significant potentials (p < 0,05) with a highly positive impact. The determinant digital marketing shows an original sample of 0,184 and an average sample value of 0,185. These values are slightly below the target of 0,200 which describes the indicator relevance and significance [22]. As the standard deviation is the lowest in the whole results, respondents have nearly always the same opinion of the hypotheses. Global networks of social media platforms have the potential to change the business-to-customer relationship. Therefore, they offer great opportunities in terms of digital possibilities to meet changing customer needs. Data mining has even better results in original sample and average sample value. With results of 0,212 and 0,236 the target of 0,200 is reached, which shows good modeling and confirms the significance level. Data mining forms the basis for the recording of changes in customer needs and the use of technologies that can provide the customer with specifically tailored models.

The determinants digital services (1,244) and online travel communities (0,547) miss a high significant level in the results. Digital services miss the target value in original sample and average sample value with 0,097 and 0,126. The quality of these values is somewhat weak. The reason for this could be that the definition of digital services is not understood uniformly. The values of the online travel communities are even weaker with 0,043 and 0,042. The weak result in the T-statistics and the relatively high P-value confirm the weak influence on the research question. Detailed information about the structural equation model and the influence of the determinants is shown in table 1.

A consideration of the quality criteria is important to prove the consistency of the model. Overall, the structural equation model achieves good values with respect to the Homburg quality criteria [21]. Cronbachs Alpha (CA) measures the internal consistency of the model. The target value is > 0,7. Composite Reliability (CR) shows the reliability of the determinants on the research question. It is declared as good with a value > 0,6. Also the Average Variance Extracted (AVE) shows the quality of different paths. It measures the fitting of the indicators with the determinants. If the indicators describe the associated construct well, the value is > 0,5. The investigated determinants, digital services, digital marketing and online travel forums, show very good values across all three quality criteria, which are attributed to the high quality of the model. The construct digital services has high values with 0,824 for CR and 0,721 for CA. The value for AVE is also in the target with 0,545. Even better values in quality criteria is reached in the construct digital marketing. Values 0,869 for CR, 0,805 for CA and 0,628 for AVE shows a clean path from the indicators over the construct to the research

TABLE I.
RESULTS OF THE STRUCTURAL EQUATION MODEL

Determinants	Original Sample	Average Sample Value	Standard Deviation	T-Statistics	Turing complete
Digital Services	0,097	0,126	0,078	1,244	0,214
Digital Marketing	0,184	0,185	0,076	2,413	0,016
Data Mining	0,212	0,236	0,100	2,117	0,035
Online Travel Communities	0,043	0,042	0,079	0,547	0,585

question. The construct online travel communities has the highest value for CR with 0,902 and for CA with 0,855. The AVE with a value of 0,698 shows an excellent description of the construct through indicators. Only data mining weakens the model somewhat in the values of Cronbachs Alpha with 0,604 and the recorded average variance extracted which is slightly below the target value with 0,430. The value of CR matches the quality with 0,735. The structural equation model of this study can thus be confirmed with good quality. All values of the quality criteria are summarized in table 2.

The answers given by the interviewees are very clear. Almost 100 people (61%) fully agree with the hypothesis of using digital services as a source of information. Almost 40% of the respondents said that they are strongly influenced by social media when planning their holidays. Half of them agree completely. The majority of respondents indicated that they were aware of the benefits of collecting data from companies. The analysis of the survey showed that almost half of the respondents are active in online travel forums before, during or after their holidays.

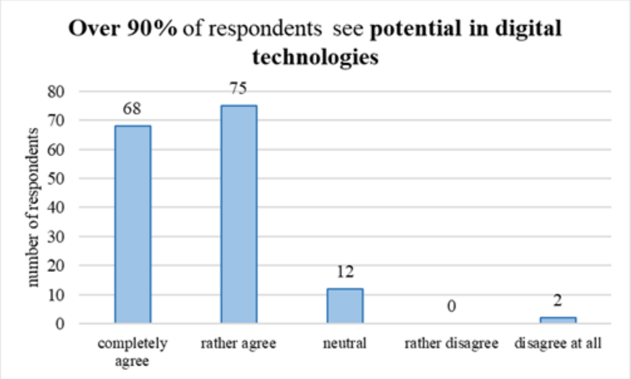


Fig. 2 Potential in digital technologies

Figure 2 refers directly to the research question. The respondents were asked to answer the question, if they see potential in digital technologies in a tourism industry with changing customer needs. More than 90% of the survey participants agree with the hypothesis that there is a potential for digital technologies in tourism. The distribution of the answers makes clear that the majority agrees with this major hypothesis. Only just under 8% of those surveyed disagree with the hypothesis, but do not reject it either.

B. Comparison with the qualitative study

The elaborated qualitative study confirms all four determinants as influential and significant. The differences between the two studies lie in the fact that experts from various companies were interviewed on the one hand and customers on the other. Two sides, each of which takes a different view at the situation. Differences in the studies can be seen in the determinants considered in the field of digital services and online travel forums. Digital marketing and data mining were perceived as significant in both studies. The difference that digital services in the quantitative study do not show a significant influence on the research question could have these reasons: On the one hand, companies provide their own services that can be used by customers. Society uses general tools to obtain information. On the other hand, companies might attach importance to the channel through which customers obtain information, e.g. accommodation. Accordingly, customers attach great importance to a good information supply and selection of offers. The channel ultimately used to contact the company is not a priority. The situation is somewhat different for the determinant of online travel forums. Companies place a strong emphasis on the data that can be generated because it is usually honest and reflects a comprehensive picture of the customer experience. In contrast, the quantitative study found that while customers use online travel forums for information purposes, very few respondents share their opinions and experiences.

IV. CONCLUSION

The results show specific potentials in the field of digital marketing and data mining. Both determinants have a significant influence on the research question. There are new opportunities to exploit market potential and competitive advantages. Digital marketing provides different approaches to address specific target groups and satisfy customer needs at the right place in the right time. Data mining forms an essential basis for the existence of modern companies. With large amounts of data (Big Data) innovative potentials of digital approaches can be exploited effectively. Once the future customer needs have been identified, the challenge lies in the fast reaction time of the companies. The design of the processes themselves has a great influence. Digitization promotes flexibility and fast response cycles. The better the

TABLE II.
QUALITY CRITERIA OF THE STRUCTURAL EQUATION MODEL

Determinants	Composite Reliability (CR)	Cronbachs Alpha (CA)	Average Variance Extracted (AVE)
Digital Services	0,824	0,721	0,545
Digital Marketing	0,869	0,805	0,628
Data Mining	0,735	0,604	0,430
Online Travel Communities	0,902	0,855	0,698

implementation of new digital approaches is promoted in companies; the more potential is available to identify and respond to customer needs.

The elaboration of this study is subject to a certain limitation by an empirical investigation in the form of quantitative research. The data sets that could be used for the investigation were limited to n = 157 persons, who act as customers in the tourism industry and were raised online. That is why the results do not allow unrestricted conclusions about the population. This can happen, even though a balanced data collection has been ensured. Results in other European countries or international may differ from the evaluations in this investigation.

Further studies in the field of digitization in tourism are quite conceivable if the responsible parameters are extended. A data collection in Europe could show some differences from this evaluation. A consideration of customer needs in a digital world in tourism, which collects data with the help of quantitative research in companies, could represent an increased added value with a different point of view. It would be interesting to compare the results in order to find out, how much potential in communication between customers and companies has not yet been fully exploited. The separate consideration of digital and conventional business models is also conceivable. That might give the opportunity of developing various, as yet untapped potentials in the tourism industry.

REFERENCES

[1] Bharadwaj, Anandhi et al. (2013): "Digital Business Strategy: Toward a Next Generation of Insights". In: MIS Quarterly, vol. 37, no. 2, 2013, pp. 471–482.

[2] Härting, R.-C.; Reichstein, C.; Schad, M. (2018): "Potentials of Digital Business Models – Empirical investigation of data driven impacts in industry", in: Robert J. Howlett et. al. (2018), Knowledge-Based and Intelligent Information & Engineering Systems, KES-2018, Elsevier B.V. 2018, Vol. 126, pp. 1495-1506.

[3] Buhalis, D. (2003): „eTourism – Information technology for strategic tourism management", Essex: Prentice Hall.

[4] Johnson, D. S., Bharadwaj B. (2005): "Digitization of selling activity and sales force performance: An empirical investigation". Journal of the Academy of Marketing Science, 33(1), pp. 3-18.

[5] Kim, H.; Law, R. (2015): "Smartphones in Tourism and Hospitality Marketing. A Literature Review". In: Journal of Travel & Tourism Marketing 32 (6), pp. 692–711.

[6] Härting, R.-C.; Reichstein, C.; Härtle, N. (2017): "Potentials of Digitization in the Tourism Industry – Empirical Results from German Experts". In: Abramowicz, W., Lecture Notes in Business Information Processing, Springer 2017, Vol. 288, pp 165-180.

[7] Härting, R.-C.; Reichstein, C.; Haarhoff, R.; Härtle, N.; Stiefl, J. (2019): "Driver to Gain from Digitization in the Tourism Industry – Insights from South African Tourism Experts", in: Yang, X.-S., et. al. (Eds.), Advances in Intelligent Systems and Computing, ICICT 2018, AISC Springer 2019, Vol. 3, pp. 293-306.

[8] Leiper, N. (1979): "The framework of tourism." In: Annals of Tourism Research 6 (4), pp. 390–407.

[9] Timoshenko, A.; Hauser, J. (2019): "Identifying Customer Needs from User-Generated Content". In: Marketing Science 38 (1), pp. 1–20.

[10] Mitchell, V.; Moudgill, P. (1979): "Measurement of Maslow's need hierarchy". In: Organizational Behavior and Human Performance 16 (2), pp. 334–349.

[11] Gretzel, U. (2011): "Intelligent systems in tourism". In: Annals of Tourism Research 38 (3), pp. 757–779.

[12] Reichstein, C.; Härting, R.-C. (2018): "Potentials of changing customer needs in a digital world – a conceptual model and recommendations for action in tourism". In: Procedia Computer Science 126, pp. 1484–1494.

[13] Thirumalai, S.; Sinha, K. (2011): "Customization of the online purchase process in electronic retailing and customer satisfaction". An online field study. In: Journal of Operations Management 29 (5), pp. 477–487.

[14] Polukhina, A.; Arnaberdiyev, A.; Tarasova, A. (2019): "Leading technologies in tourism. Using blockchain in TravelChain project". In: Proceedings of the 3rd International Conference on Social, Economic, and Academic Leadership (ICSEAL 2019). Prague, Czech Republic, 23.03.2019 - 24.03.2019. Paris, France: Atlantis Press.

[15] Rabe, L. (2019): „Welche Social-Media-Plattform ist für Ihr Unternehmen am wichtigsten?" Ed. Social Media Examiner. Online available at https://de.statista.com/statistik/daten/studie/463928/umfrage/wichtigste-social-media-plattformen-fuermarketingverantwortliche, last checked on 21.08.2019.

[16] Larose, D. (2005): "Discovering knowledge in data. An introduction to data mining". Hoboken, N.J: Wiley-Interscience.

[17] Dittert, M.; Härting, R. C.; Reichstein, C.; Bayer C. (2017): "A Data Analytics Framework for Business in Small and Medium-Sized Organizations", In: I. Czarnowski, R. Howlett, L. Jain. (Ed.), Smart Innovation, Systems and Technologies – Proceedings of the 9th KES-IDT 2017 – Part II, Springer 2017, Vol. 73, pp.169-181.

[18] Cheng, X.; Fu, S.; Sun, J.; Bilgihan, A.; Okumus, F. (2019): "An investigation on online reviews in sharing economy driven hospitality platforms. A viewpoint of trust". In: Tourism Management 71, pp. 366–377.

[19] Kothari, C. R. (2004): "Research methodology. Methods & techniques". 2nd rev. ed. New Delhi: New Age International (P) Ltd. Publishers.

[20] Rhemtulla, M.; Brosseau-Liard, P.; Savalei, V. (2012): "When can categorical variables be treated as continuous? A comparison of robust continuous and categorical SEM estimation methods under suboptimal conditions". In: Psychological Methods 17 (3), pp. 354–373.

[21] Homburg, C.; Baumgartner, H. (1995): „Beurteilung von Kausalmodellen. Bestandsaufnahme und Anwendungsempfehlungen". In: MAR 17 (3), pp. 162–176.

[22] Chin, W. (1998): "The Partial Least Squares Approach for Structural Equation Modeling". In: Modern Methods for Business Research, Psychology Press, pp.295-336.. E. Haskell and C. T. Case, "Transient signal propagation in lossless isotropic plasmas (Report style)," USAF Cambridge Res. Lab., Cambridge, MA Rep. ARCRL-66-234 (II), 1994, vol. 2.

DOI: 10.15439/2020F77
ISSN 2300-5963 ACSIS, Vol. 21

Toward Digital Transformation of Processes in Legal Metrology for Weighing Instruments

Alexander Oppermann, Samuel Eickelberg, John Exner
Physikalisch-Technische Bundesanstalt (PTB)
Abbestr. 2-12
10587 Berlin, Germany
Email: {alexander.oppermann, samuel.eickelberg, john.exner}@ptb.de

Abstract—**The digital transformation of sovereign processes is a driving force to streamline and innovate processes for measuring instruments under legal control. Providing trust is the essential purpose of Legal Metrology and still a challenging task in the digital domain. Taking the strict legal framework into account, a distributed software architecture is presented that offers privacy, security and resilience. At the same time, the platform approach seamlessly integrates existing public and private infrastructures. Furthermore, a service hub is created with interdependent services that support the digital transformation of paper-based processes, such as verification and software update. Exemplary, these two central use cases are introduced, and its requirements and implementation approach are described. The main goal is to provide the same level of trust and security, by developing new digital concepts, infrastructure and remote processes for a unified digital single market.**

I. Introduction

THIS paper focuses on the progress of innovating sovereign tasks within Legal Metrology by the digital transformation of paper-based procedures. The field of Legal Metrology comprises around 160 million measuring instruments in Germany, which contribute about 157 billion Euros to the national GDP [1]. The main purpose of Legal Metrology is to establish and provide trust among all stakeholders such as customers, manufacturers, and users of measuring instruments. By law a Notified Body, e.g. the PTB in Germany, is obligated to carry out a conformity assessment of measuring instruments. The essential requirements of the Measuring Instrument Directive (MID) [2], such as reproducibility, repeatability, durability and protection against corruption of measuring instruments and measurements, have to be fulfilled before entering the market.

AnGeWaNt[1] is a joint research project to address the automation of measuring instruments and their challenges for society. The six associates are ranging from different areas like commercial partners in the weighing and construction industry to Notified Body in Germany, such as the Physikalisch-Technische Bundesanstalt (PTB), a local innovation hub (Zenit GmbH) [3] and the Institute for applied labor science (Ifaa) responsible for aspects within social-economic and human factors [4].

In this paper, the digital transformation of two main procedures, such as verification application as well as the software update, are described and their progress in the digital transformation are outlined. Section II puts the related work into perspective with the AnGeWaNt project. In Section III, an overview of the requirements and the novel approach of the use cases is given. Section IV presents the approach to map the processes, their requirements, design decisions and shared infrastructures.

II. Related Work

A. European Metrology Cloud

The AnGeWaNt project can be considered as a national spinoff of the *European Metrology Cloud Project* (EMC) focusing only on weighing instruments.

The EMC project circumcises 16 different measuring instrument classes in a supranational setting within a European legislation context. One of the major goals is to support the unified digital single market that the European Commission has issued. While the EMC project aims to distribute hardware nodes [5] that will provide essential parts of the envisioned infrastructure, the AnGeWaNt project can be hosted, split up and distributed on any Platform as a Service provider (PaaS). The European Metrology Cloud concept is also in line with the General Data Protection Regulation (GDPR) [6].

B. GAIA-X

The GAIA-X project started as a joint initiative of Germany and France in 2019 with the goal to build a sovereign digital European Cloud-Ecosystem that is efficient, secure and highly distributed. Major key features are data sovereignty, privacy by complying with the GDPR, transparency and openness by supporting open-source principles and being flexible by building a modular and highly inter-operable platform for a broad spectrum of industry partners. These include Small and Medium Enterprises (SME) as well as government bodies [7].

III. Requirements and Use Cases

Due to the harmonization efforts in the EU, the respective national regulations in Europe are derived from the MID. It

The presented results are part of the project "AnGeWaNt - Arbeit an geeichten Waagen für hybride Wiegeleistungen an Nutzfahrzeugen" (FKZ: 02L17B050). We would like to thank the German Federal Ministry of Education and Research (BMBF) and the European Social Fund (ESF) for funding this research project. Both projects are implemented by the Project Management Agency Karlsruhe (PTKA). The authors are responsible for the content of this publication.

[1] https://www.angewant.de/

defines regulations until the measuring instrument is placed on the market. Regulations for processes after the measuring instrument has been placed on the market, such as re-verification, are not regulated by the MID. In Germany, the Measurement and Verification Act (MessEG) [8] and the Measurement and Verification Ordinance (MessEV) [9] are applied.

The following recommendations are taken into account: the International Organization of Legal Metrology (*OIML*) [10], the *MID*, software recommendations of European Cooperation in Legal Metrology (*WELMEC Guide 7.2*) [11], recommendations of *WELMEC Guide 7.3* Reference Architectures [12], and technical guidelines of the Federal Office for Information Security (*BSI TR 03109*) [13].

Additional requirements are sufficient flexibility, the protection of personal data, planning of verification appointment, and transparency of the processes (status information for processing). Furthermore, there should be standalone services for user management, a revision-safe archive store, verification application submission, scheduling of verification appointments, and a revision-safe logbook.

A. Digital Administration Shell

Cornerstone of the digital transformation will be the *Digital Administration Shell*. This concept will be developed gradually. It evolves from a *device pass*, that stockpiles instrument specific data and thus creates a unique mapping to a measuring instrument. Based on this, the *digital type plate* will be created, that holds verification-specific information, e.g. the number of the EC type-examination certificate and the accuracy class of a scale. It will result in a general *document store* that hosts all documents related to a specific measuring instrument. A unique filter mechanism will be implemented to present information depending on the specific perspective of users, roles and access rights. For the first time, this administration shell will gather all relevant data over the life span of a measuring instrument and in addition to it fulfill the legal archiving obligations.

B. Digital Verification Application

After the verification period has expired, measuring instruments under legal control must be re-verified. Without valid verification, measuring instruments may no longer be used. This helps to maintain trust in officially verified measuring instruments. The flow-chart of the German verification process is made available [14].

A key requirement is the digital transformation of the paper-based verification application process. The prototype platform aims to provide a convenient, web-based user interface. Furthermore, it facilitates and streamlines the entire process across all 16 German federal states. The target platform DEMOL requires the verification applications in a specified XML-based data structure. AnGeWaNt collects the application data and checks for plausibility before submission.

C. Software Update Application

Measuring instruments under legal control consist of legally relevant and non-relevant software. The legally relevant part comprises the metrological characteristics of the measuring instruments [9]. Only updates of the legally relevant software part are regulated. According to §37 MessEG, a software update requires the approval of the responsible verification authority.

The verification authority must be informed about which devices should be updated. Instead of applying for a software update for each device individually, a bulk application is feasible. The flow-charts of the complex process are made available for the standard appeal [15] and for the emergency appeal [16].

D. Remote Authenticity

In the field, user and market surveillance verifies measuring instruments. With increasing usage of software in measuring instruments, a further decoupling of hardware and software is inevitable. As a result only small parts of the measuring instrument, such as the sensor and a communication unit, will remain in the field. Software will be outsourced into data centers (see [17]). Thus, it is in the utmost interest of the authorities to verify legal relevant software parts remotely. Despite its location and physical access. Oppermann et al. [18] already addressed this problem and created a virtual verification monitor that fulfills the need of user and market surveillance. The authorities must have the possibility to remotely verify the authenticity of used meter, processing unit, and associated logbook. The legal relevant software has to provide a form of identification. An acceptable solution according to the WELMEC Guide [11] would be: a *software name* consisting of a string of numbers, letters, or other characters; a string added by a *version number*; and a *checksum* over the code base.

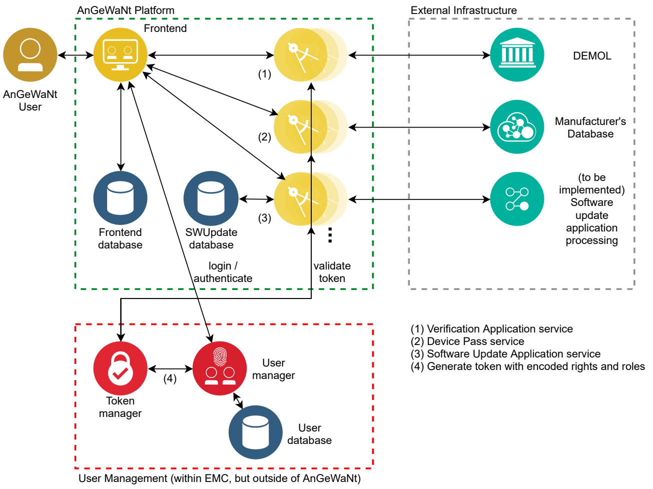

AnGeWaNt Platform
Frontend
External Infrastructure
AnGeWaNt User
(1)
(2)
(3)
DEMOL
Manufacturer's Database
(to be implemented) Software update application processing
Frontend database
SWUpdate database
login / authenticate
validate token
Token manager
(4)
User manager
User database
User Management (within EMC, but outside of AnGeWaNt)
(1) Verification Application service
(2) Device Pass service
(3) Software Update Application service
(4) Generate token with encoded rights and roles

Fig. 1. Overview of the platform architecture concept

IV. Architecture of the Platform

The envisioned platform is a central hub, that will offer access to all connected infrastructures and their provided services. All stakeholders can take advantage of the offered services and data within the platform domain. Consequently, this service hub is a first step to a interdependent service ecosystem. It will provide opportunities to develop new data-driven business cases beyond the classical domain of manufacturers and increases the innovation for future success.

Furthermore, by its modular approach, the architecture allows new services to be added with minimal effort. To increase flexibility and ease later expandability, the project strives for standardized and harmonized REST interfaces across all services. The distributed architecture offers independent deployment as well as operation of services. However, a web-based user front-end must be able to find and communicate with all provided services within the platform ecosystem.

The setup of the platform (see Fig. 1) consists of three independent modules. The main module is the AnGeWaNt Platform (see upper-left dashed box). It offers a web-based user interface and services, such as verification application and software update. The user management module (lower-left dashed box) consists of the user and token manager services. That offers a secure, stateless and flexible authentification and authorization layer. The External infrastructure module (right dashed box) ties third-party systems to AnGeWaNt platform, such as DEMOL or manufacturers' systems, to provide verification applications or device passes.

A. Common Principal Data Service

All stakeholders, devices, and device types are using the same core data across different types of processes and documents. Thus, a service which handles all common principal data is implemented, and accessible through the front-end to fill in e.g. a software update application or a verification request.

From the data model perspective, essential requirements are the mapping of the processes between stakeholders, and a document-centered view of the processes in legal metrology. A step in a process often creates a document, such as a validation application, or a software update application. At the same time, these documents must be saved in an audit-proof manner. For this reason, documents and stakeholders are being loosely coupled in the current state of development. A lot of information can be held in an abstract class `Party` rather than to specific types, such as user or manufacturer.

B. Software Update Application Service

The implemented service has specified endpoints for submission of an application, updating status, receiving a hearing, as well as placing a response to a hearing. These are the only steps in the request processing that require interaction via the AnGeWaNt platform. The software update request is processed according to the following steps (see Fig. 2):

C. Role and User Management

The project aims to separate user credentials as well as rights and roles management from the AnGeWaNt platform. First, the User manager and Token manager services can be re-used in other projects and services across the EMC. Secondly, all user-related data is not coupled to a specific application, increasing the security of the platform. Another design paradigm is to avoid sessions, because they cannot be held in a highly distributed architecture across potentially different domains. Instead, tokens are being generated and assigned to either a user that logs in successfully, or a service to prove its authenticity.

1) The user manager: handles entities which contain the user name, the password as encrypted byte sequence, granted rights, assigned roles, as well as flags whether or not the user account is enabled, locked, or its credentials are disabled. All communication with the User manager takes place via REST. The user authentication process deals with *authenticity* and *authorization*. By communicating with stateless tokens, a session which contains currently logged-in user information is not required.

2) The token manager: provides JSON Web Tokens (JWT) containing the authorization, which in turn depends on rights and roles from user manager, for a user to access specific services. The service evaluates the request header of a token to determine a valid authorized request. Like the User manager, all communication with the Token manager takes place via REST. This service only provides authenticity.

D. Security Aspects

Risk assessment of distributed metrological software has already been addressed by Oppermann et al . [19] and Esche et al. [20]. As the prototype platform is concerned with legally relevant processes regarding measuring instruments, the following attack vectors (according to [11]) must be taken into account when choosing a publicly available, widespread application framework, such as Spring Boot: *A_WEB_XSS* (Cross-site scripting attack), *A_WEB_DOS* (Denial-of-Service attack), and *A_WEB_SOCKET* (introducing malicious code via web socket). Spring Boot addresses the aforementioned security

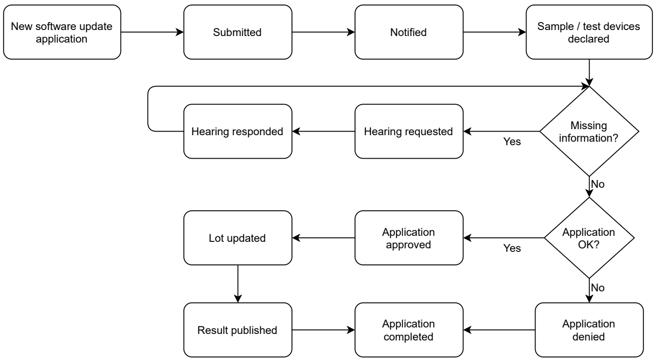


Fig. 2. Processing flowchart of the software update request process from the AnGeWaNt perspective

concerns, given that the appropriate implementations and application server configurations are being carried out.

V. Conclusion And Future Work

In this short paper the AnGeWaNt project introduced its approach to seamlessly integrate existing public and private infrastructure. The legal framework with its security requirements are briefly described and also its implication for the platform design. Legally regulated processes, such as verification application and software update, are central use cases. Their digital transformation is the driving force to innovate and streamline the underlying infrastructure.

By creating a central platform with a service hub, the digital transformation of paper-based procedures is supported. The platform is modular and built to integrate independent services. This provides new opportunities beyond the classical realm of all stakeholders and increases data-driven innovation for future success. Moreover, other research projects can easily profit from the implementation progress, e.g. the user management can be integrated into the EMC project. Furthermore, each service can be easily separated and integrated into another context. This will assure that a lot of progress made in this project can be reused and avoid time intensive and costly reimplementation. The highly modularized nature of this platform allows a greater flexibility, adaptability and eases the burden of distributing the services across different domains. This increases indirectly the resilience of the platform, because several instances of a service can be run at the same time in different domains. At last, this platform paves the way for a unified digital single market.

Furthermore, the AnGeWaNt platform is designed from the beginning with *Multi-tenancy* concept in mind. The need for different stakeholders, such as manufacturers, market and user surveillance authorities, and notified bodies to access only its own relevant data is crucial. This is guaranteed by the *User Manager*. With the introduction of *Token Manager*, a single sign on solution is created that can be used to verify a user and grant access for documents across infrastructures. This is especially helpful, e.g. to send verification applications to external systems like DEMOL or to import device passes from manufacturers.

In the next iteration, *Multi-factor Authentication* will be introduced, to increase security and enhance trust in the platform. The user manager will be extended to use a second factor for authentication. This can be a mobile device running an authenticator app, or a physical device, e.g. a fingerprint scanner attached to a client computer. Furthermore, the support and integration of OpenID connect frameworkis intended.

In the long run, the *Digital Administration Shell* will be extended to import an XML-based Certificate of Conformity (CoC) in the near future. This will offer new services, e.g. remote verification, issuing and revoking certificates. The PTB is working to establish the same level of trust for digital certificates as conventional paper-based certificates guarantee nowadays.

References

[1] N. Leffler and F. Thiel, “Im Geschäftsverkehr das richtige Maß - Das neue Mess und Eichgesetz, Schlaglichter der Wirtschaftspolitik,” 2013.

[2] European Parliament and Council, “Directive 2014/32/EU of the European Parliament and of the Council,” *Official Journal of the European Union*, 2014.

[3] M. Guth, H. Hoffzimmer, and N. Ottersböck, “Entwicklung hybrider Geschäftsmodelle vor dem Hintergrund der Digitalisierung,” *Betriebspraxis & Arbeitsforschung*, 2020.

[4] N. Ottersböck, M. Frost, T. Jeske, and V. Hartmann, “Systematischer Kompetenzaufbau als Erfolgsfaktor zur Etablierung hybrider Geschäftsmodelle,” *GfA (Hrsg) Digitale Arbeit, digitaler Wandel, digitaler Mensch? Bericht zum 66. Kongress der Gesellschaft für Arbeitswissenschaft vom 16. – 18. März 2020*, 2020.

[5] M. Dohlus, M. Nischwitz, A. Yurchenko, R. Meyer, J. Wetzlich, and F. Thiel, “Designing the European Metrology Cloud,” *OIML Bulletin*, vol. 61, no. 1, pp. 08–17, 2020.

[6] F. Thiel and J. Wetzlich, “The European Metrology Cloud: Impact of European Regulations on Data Protection and the Free Flow of Non-Personal Data,” in *International Congress of Metrology*, Array, Ed., 2019. doi: 10.1051/metrology/201901001 p. 01001.

[7] Federal Ministry for Economic Affairs and Energy (BMWi), “Project GAIA-X - A Federated Data Infrastructure as the Cradle of a Vibrant European Ecosystem - Executive Summary,” *Official Journal of Federal Ministry for Economic Affairs and Energy*, Oct. 2019.

[8] “Gesetz über das Inverkehrbringen und die Bereitstellung von Messgeräten auf dem Markt, ihre Verwendung und Eichung sowie über Fertigpackungen (Mess- und Eichgesetz - MessEG),” Nov. 2019. [Online]. Available: https://www.gesetze-im-internet.de/messeg/

[9] “Verordnung über das Inverkehrbringen und die Bereitstellung von Messgeräten auf dem Markt sowie über ihre Verwendung und Eichung (Mess- und Eichverordnung - MessEV),” Apr. 2020. [Online]. Available: https://www.gesetze-im-internet.de/messev/

[10] O. de Métrologie Légale, “General requirements for software controlled measuring instruments,” 2008.

[11] “WELMEC 7.2 Software Guide,” *WELMEC European cooperation in legal metrology, Welmec Secretariat, Delft, Standard*, 2019.

[12] “WELMEC 7.3 Guide Reference Architectures - Based on WELMEC Guide 7.2,” *WELMEC European cooperation in legal metrology, Welmec Secretariat, Delft, Standard*, 2019.

[13] BSI, “Technische Richtlinie BSI TR-03109-1 Anforderungen an die Interoperabilität der Kommunikationseinheit eines intelligenten Messsystems,” *Bundesamt für Sicherheit in der Informationstechnik, Bonn*, 2013.

[14] J. Exner and A. Oppermann, “German verification process,” May 2019. [Online]. Available: https://www.angewant.de/wp-content/uploads/2020/06/Eichantrag.pdf

[15] ——, “German software update emergency appeal,” May 2019. [Online]. Available: https://www.angewant.de/wp-content/uploads/2020/06/Standardverfahren_Softwareaktualisierung.pdf

[16] ——, “German software update process,” May 2019. [Online]. Available: https://www.angewant.de/wp-content/uploads/2020/06/Eilverfahren_Softwareaktualisierung.pdf

[17] A. Oppermann, A. Yurchenko, M. Esche, and J.-P. Seifert, “Secure cloud computing: Multithreaded fully homomorphic encryption for legal metrology,” in *International Conference on Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments*. Springer, 2017, pp. 35–54.

[18] A. Oppermann, F. G. Toro, F. Thiel, and J.-P. Seifert, “Secure Cloud Computing: Reference Architecture for Measuring Instrument under Legal Control,” *Security and Privacy*, vol. 1, no. 3, p. e18, 2018. doi: 10.1002/spy2.18. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/spy2.18

[19] A. Oppermann, M. Esche, F. Thiel, and J.-P. Seifert, “Secure Cloud Computing: Risk Analysis for Secure Cloud Reference Architecture in Legal Metrology,” *accepted in Federated Conference on Computer Science and Information Systems (FedCSIS), IEEE*, 2108.

[20] M. Esche and F. Thiel, “Software Risk Assessment for Measuring Instruments in Legal Metrology,” *Proceedings of the Federated Conference on Computer Science and Information Systems*, pp. 1113–1123, 2015. doi: 10.15439/2015F127

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 563–566 DOI: 10.15439/2020F139 ISSN 2300-5963 ACSIS, Vol. 21

Map Matching Algorithm Based on Dynamic Programming Approach

Alexander Yumaganov
Samara National Research University
Samara, Russia
yumagan@gmail.com

Anton Agafonov
Samara National Research University
Samara, Russia
Email: ant.agafonov@gmail.com

Vladislav Myasnikov
Samara National Research University
Samara, Russia
Email: vmyas@geosamara.ru

Abstract—**GPS sensors embedded in almost all mobile devices and vehicles generate a large amount of data that can be used in both practical applications and transportation research. Despite the high accuracy of location measurements in 3-5 meters on average, this data can not be used for practical use without preprocessing. The preprocessing step that is needed to identify the correct path as a sequence of road segments by a series of location measurements and road network data is called map matching. In this paper, we consider the offline map matching problem in which the whole trajectory is processed after it has been collected. We propose a map matching algorithm based on a dynamic programming approach. The experimental studies on the dataset collected in Samara, Russia, showed that the proposed algorithm outperforms other comparable algorithms in terms of accuracy.**

I. Introduction

WIDESPREAD deployment of Global Positioning System (GPS) provides a large amount of data describing movement trajectories of pedestrians, bicycles, vehicles, etc. The trajectories are observed as a sequence of GPS records. Each record usually contains ID, latitude and longitude of the GPS sensor and timestamp of the record. GPS sensors usually provide location data with high accuracy up to 5 meters on average, but in some cases, the measurement errors can be much higher, especially in the urban environment. In any case, to use the GPS data in many practical applications and transportation research we first need to perform the preprocessing step that is called the map matching process. Map matching algorithms are applied to identify the correct path as a sequence of road segments by a series of location measurements (GPS records) and road network data. Processed trajectories are an important data source for intelligent transportation systems that can be in such applications as traffic estimation and prediction [1], [2], traffic modelling [3], developing navigation services, user preferences elicitation and training of transportation recommendation systems [4], [5], and so on.

As mentioned earlier, the GPS-trajectories have measurement errors because of multiple factors: atmospheric phenomena, interference from ground-based radio sources, high-rise urban development, vegetation, imperfect hardware and the embedded processing algorithms, and others. We considered a large number of GPS tracks collected by several mobile devices and identify several typical errors:

1) Large geolocation error. The GPS sensor gives several records with the measurement error significantly higher than the usual 3-5 meters.
2) Large time gaps. The GPS sensor does not provide any data for a long time. There are gaps in the track in several minutes or more between high-quality recorded fragments.
3) Continuous deviations from the ground truth path. The GPS sensor for a long time provides coordinates with a low error, the track line looks smooth, however, the deviation of the track line from the true path is several times higher than the average measurement error.
4) Loops when stopped after fast movement.

Given the above measurement errors, the map matching process can be quite challenging. As a result, a number of map matching algorithms have been developed to solve this problem. The algorithms can be categorized by different criteria. In this paper, we consider the online/offline classification. Online map matching methods [6], [7], [8] process positions when the trajectory is still collecting. Offline map matching methods [9], [10] compute the path after the whole trajectory has been collected. In this paper, we consider the offline map matching problem.

Classification and comparative study of map matching algorithms were presented in [11]. In [12], the authors reviewed existing map matching algorithms with the aim of highlighting their qualities, unresolved issues, and provide directions for future studies. The algorithms were compared with respect to positioning sensors, map qualities, assumptions, and accuracy.

In [13], the authors developed a topological point-to-curve map matching algorithm integrated with a Kalman filter. A local incremental algorithm that matches consecutive portions of the trajectory to the road network was proposed in [14].

Weighted-based topological map matching algorithms was proposed in [15], [16]. In [16], the authors integrated raw measurements from GPS, dead-reckoning sensors, and a digital elevation model using an extended Kalman filter in order to increase the accuracy of the map matching process.

In later works, advanced map matching algorithms was proposed. In [17], [18], the authors discussed the possibility of applying Hidden Markov models (HMM) in map matching

The work was partially supported by RFBR research projects nos. 18-07-00605 A, 18-29-03135-mk

algorithms. In the proposed methods, the authors used HMM to find the most likely road route taking into account the measurement noise and the layout of the road network. In [19], the authors proposed a feature-based map matching algorithm that estimates the cost of a candidate path based on both GPS observations and a behavioral model. In [9], the authors presented a map matching algorithm based on Dijkstra's shortest path method that is applicable for large scale datasets. The authors focused on reducing the computational complexity of the algorithm. In [20], the authors also concentrated on designing efficient and scalable map-matching algorithms. They presented an algorithm integrating hidden Markov model with precomputations techniques.

Despite the large number of papers devoted to the map matching problem, the proposed solutions do not allow achieving high accuracy or, in some cases, can not find the correct path at all. In this paper, we focus on developing the map matching algorithm that allows us to identify the correct path with high accuracy. The proposed algorithm consists of two steps: calculating the shortest paths and estimating the paths using a dynamic programming approach.

The paper is organized as follows. In Section II, the main notation and problem statement given. The proposed map matching algorithm with the dynamic algorithm of the shortest path assessment are presented in Section III. Section IV describes the experimental setup and results of experimental studies. Finally, we give a conclusion and possible directions for further research.

II. Problem Statement

A road network is represented as a directed graph $G = (V, W)$, where V is the set of nodes that represent road intersections, W is the set of edges denote road segments. Each node $v \in V$ has the coordinates $\overline{x}_v = (x_v, y_v)$. Each edge $w_{ij} \in W, i, j \in V$ is described by the tuple:

$$w_{ij} = \left(l^w, v^w_{max}, X^w\right),$$

where l^w is a length of the road segment w, v^w_{max} is the maximum allowed speed, X^w is the geometry of the road segment w presented as a set of points.

Define a GPS trajectory as the set of GPS records obtained during the observation:

$$\{\overline{x}_i, t_i\}_{i=\overline{0,I-1}},$$

where I is the number of GPS records, $\overline{x}_i = (x_i, y_i)$ is the coordinates of the tracked objects (latitude and longitude), t_i is the timestamp of i-th GPS record.

Define the ground truth path P as the sequence of the edges (road segments) that was traversed by the vehicle during the observation.

Given the introduced notation, the map matching problem can be formulated as follows:

Given a graph $G = (V, W)$ and a GPS trajectory $\{\overline{x}_i, t_i\}_{i=\overline{0,I-1}}$ find the ground truth path P traversed by a vehicle in a road network.

III. Proposed Approach

A. Map Matching Algorithm

The proposed map matching algorithm can be described as a sequence of the following steps:

1) Determine the start and end nodes by the coordinates of first and last GPS records.
2) For all edges $w \in W$ located at a distance to the GPS records not exceeding $R = 100$ meters, the edge weight is set using the following equation:

$$\varphi(w) = \left(1 - \frac{1}{K}\sum_{k=0}^{K-1} \exp\left(-\alpha \|\overline{x}_k - \overline{x}^p_k(w)\|^2\right)\right) l^w$$

where K is the number of GPS records matched with the edge w, $\overline{x}_k$ are the coordinates of the matched GPS record, $\overline{x}^p_k$ is the coordinates of the GPS record projection on the edge w, α is the coefficient.
For not matched edges the weight is set as follows:

$$\varphi(w) = \beta l^w,$$

where $\beta = 10$ is the coefficient.
3) In the graph with the edge weights set as described above, the shortest path is searched from the start to end node corresponding to the first and last GPS records.
4) The found shortest path is estimated using a dynamic algorithm described in subsection III-B.
5) The algorithm for sequential removal of edges from the path is performed.
Input data: path, assessment of the path, graph. For an edge from the list of path edges:
   a) The edge is removed from the graph.
   b) The shortest path search from the start node of the removed edge to the end node is performed. If there is no such path in the graph, go to step e).
   c) The resulting path is estimated using a dynamic algorithm.
   d) If the resulting assessment is greater than the assessment of the original path, then save the resulting path to the list of best paths;
   e) Restore the deleted edge and move on to process the next edge.

   Output data: the path from the list of best paths, select the path with the maximum assessment value, or an empty path if the list of best paths is empty.
6) The algorithm for sequential removal of edges runs in a loop until an empty path is returned. The last non-empty path will be a solution of the map matching algorithm.

In order to reduce the impact of typical errors in the GPS records, the path assessment in the algorithm for sequential removal of edges is calculated by the following quality criterion:

$$J^* = \begin{cases} J_{p\_cur} - \gamma \frac{l_{p\_cur} - l_{p\_base}}{J_{p\_cur} - J_{p\_base}}, J_{p\_cur} - J_{p\_base} > 0; \\ J_{p\_cur}, \mathit{otherwise}, \end{cases}$$

where J_{p_cur} is the current path assessment, J_{p_cur} is the base path assessment, l_{p_cur} is the current path length, l_{p_base} is the base path length, γ is an empirically selected coefficient.

In the next subsection, we describe the algorithm for the reconstructed path estimation.

B. Dynamic Algorithm for Reconstructed Path Estimation

As a criterion for the reconstruction quality, we use the following:

$$J_p = \sum_{i=0}^{I-1} \exp\left(-\alpha \|\overline{x}_i - \overline{x}_i^p\|^2\right).$$

We need to match the points $\{\overline{x}_i, t_i\}_{i=\overline{0,I-1}}$ with the path p. Firstly, we discretize the path into N points with the discretization step $\Delta = 2$ meters: $p(n) = \overline{x}_n^p, p = \overline{0, N-1}$. Next, we calculate I arrays of proximity similarities between the point $\overline{x}_i$ and the path p as:

$$\varphi_i(n) = \exp\left(-\alpha \|\overline{x}_i - p(n)\|^2\right).$$

The optimization problem is to find the sequence

$$n(i)_{i=\overline{0,I-1}} : \sum_{i=0}^{I-1} \varphi\left(n\left(i\right)\right) \to \max.$$

The main recurrence relation (for the dynamic programming algorithm) has the following form:

$$\max_{n(i)} \sum_{i=0}^{I-1} \varphi_i\left(n\left(i\right)\right) = \max_{n(i_l)=\overline{n(i_{l-1}),N}} \left[\varphi_i\left(n\left(i\right)\right) + \right.$$
$$\left. + \max_{n(i)\leq n(i_l)} \sum_{i=0}^{i_l-1} \varphi_i\left(n\left(i\right)\right) + \max_{n(i)\geq n(i_l)} \sum_{i=i_l-1}^{I-1} \varphi_i\left(n\left(i\right)\right)\right].$$

Introduce the additional notations. Let $\widetilde{\varphi_i}(n)$ be the maximum integral similarity:

$$\widetilde{\varphi_j}(n) = \max_{n(i):i\leq j} \sum_{i=0}^{j} \varphi_i\left(n\left(i\right)\right).$$

Let $\pi_i(n)$ be the list of point positions.

The dynamic programming algorithm to solve the recurrence relation can be described as follows (Algorithm 1).

The result path assessment and the list of point positions are stored in $\widetilde{\varphi_0}(0)$ and $\pi_0(0)$. Using this path assessment, the best path by the specified criteria will be selected in the map matching algorithm.

IV. Experiments

Experimental studies of the map matching algorithms were carried out for a large-scale transportation network of Samara, Russia, consisting of 47274 road segments (edges) and 18582 nodes. As a source dataset, we used 20 manually collected tracks recorded by two mobile devices.

We compare the proposed dynamic-based algorithm (DBA) with the FMM algorithm [20] and the HMM-based algorithm [17] implemented as a part of the GraphHopper library [21].

Algorithm 1 Path estimation algorithm

```
for $i = I-1, 0$ do
  for $n = N-1, 0$ do
    if $i == I-1$ then
      if $n == N-1$ then
        $\widetilde{\varphi_i}(N-1) = \varphi_i(N-1); \pi_i(N-1) = \{N-1\}$
      else
        if $\varphi_i(n) > \widetilde{\varphi_i}(n+1)$ then
          $\widetilde{\varphi_i}(n) = \varphi_i(n); \pi_i(n) = \{N-1\}$
        else
          $\widetilde{\varphi_i}(n) = \widetilde{\varphi_i}(n+1); \pi_i(n) = \pi_i(n+1)$
        end if
      end if
    else
      if $n == N-1$ then
        $\widetilde{\varphi_i}(N-1) = \varphi_i(N-1) + \widetilde{\varphi_{i+1}}(N-1)$
        $\pi_i(N-1) = \pi_{i+1}(N-1); \pi_i(N-1).add(N-1)$
      else
        if $\varphi_i(n) + \widetilde{\varphi_{i+1}}(n) > \widetilde{\varphi_i}(n+1)$ then
          $\widetilde{\varphi_i}(n) = \varphi_i(n) + \widetilde{\varphi_{i+1}}(n)$
          $\pi_i(n) = copy(\pi_{i+1}(n)); \pi_i(n).add(n)$
        else
          $\widetilde{\varphi_i}(n) = \widetilde{\varphi_i}(n+1); \pi_i(n) = \pi_i(n+1)$
        end if
      end if
    end if
  end for
end for
```

To evaluate the map matching accuracy, we used two metrics: Route Mismatch Fraction (RMF) introduced in [17] and the accuracy metric (A) was used in [20].

The RMF is computed as:

$$RMF = \frac{1}{M} \sum_{m=0}^{M-1} \frac{l_{gt}^m}{l_+^m + l_-^m},$$

where M is the number of tracks, l_{gt}^m is the m-th ground truth path length, l_+ is the length of the road segments in the m-th matched path that are not in the m-th ground truth path (erroneously added), l_- is the length of the road segments in the m-th ground truth path that are not presented in the m-th matched path (erroneously subtracted).

The accuracy metric is the average of the overlapping ratio between the ground truth path (GT) and the matched path(MP):

$$Accuracy = \frac{1}{M} \sum_{m=0}^{M-1} \frac{|GT[m] \cap MP[m]|}{|GT[m] \cup MP[m]|}.$$

Table I presents a comparison of the accuracy of the selected map matching algorithms by the described criteria.

The accuracy of the proposed algorithm is higher than the accuracy of the baseline map matching algorithms.

To visually evaluate the quality of the algorithms, ground truth paths and matched paths were displayed on the map.

TABLE I
ACCURACY MEASUREMENTS OF MAP MATCHING ALGORITHMS

	RMF	**Accuracy**
DBA	**0.135**	**0.876**
FMM	0.245	0.836
HMM	0.744	0.43

Fig. 1 shows an example of the map matching result. The FMM matched path is shown by the green line, the DBA matched path is shown by the dash black line.

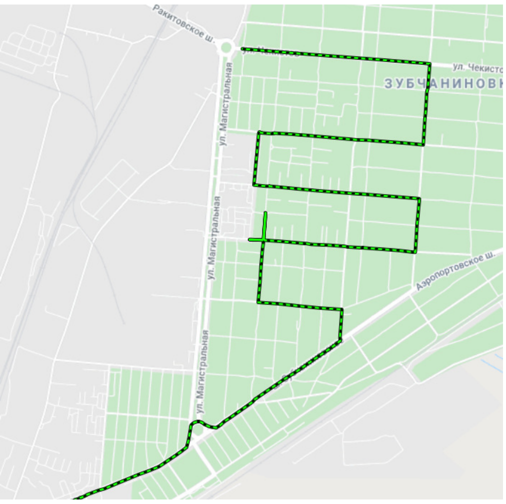


Fig. 1. Example of the map matching result

As can be seen from the picture, both algorithms provide good results, but the FMM algorithm sometimes has large errors on intersections.

V. Conclusion

In this paper, we consider the offline map matching problem in which the whole trajectory is processed after it has been collected. The proposed algorithm consists of two steps performed in a cycle: path estimation and sequential removal of edges from the path. To estimate the matching path, we presented a map matching algorithm based on a dynamic programming approach. Experimental studies conducted on manually collected tracks in Samara, Russia, allow us to conclude that the proposed algorithm has high accuracy and superior other baseline methods by selected criteria.

In future studies, we will investigate our algorithm on publicly-available datasets and focus on improving the computational efficiency of the algorithm.

References

[1] A. Agafonov and A. Yumaganov, "Short-term traffic flow forecasting using a distributed spatial-temporal k nearest neighbors model," in *Proceedings - 21st IEEE International Conference on Computational Science and Engineering, CSE 2018*, 2018. doi: 10.1109/CSE.2018.00019 pp. 91–98.

[2] A. Nagy and V. Simon, "Identifying hidden influences of traffic incidents' effect in smart cities," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018*, 2018. doi: 10.15439/2018F194 pp. 651–658.

[3] Y. Amara, A. Amamra, Y. Daheur, and L. Saichi, "A GIS data realistic road generation approach for traffic simulation," in *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems, FedCSIS 2019*, 2019. doi: 10.15439/2019F223 pp. 385–390.

[4] D. Da Silva, J. Torres, A. Pinheiro, F. De Caldas Filho, F. Mendonca, B. Praciano, G. De Oliveira Kfouri, and R. De Sousa, "Inference of driver behavior using correlated IoT data from the vehicle telemetry and the driver mobile phone," in *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems, FedCSIS 2019*, 2019. doi: 10.15439/2019F263 pp. 487–491.

[5] V. Myasnikov, "Reconstruction of functions and digital images using sign representations," *Computer Optics*, vol. 43, no. 6, pp. 1041–1052, 2019. doi: 10.18287/2412-6179-2019-43-6-1041-1052

[6] M. Kubička, A. Cela, H. Mounier, and S. Niculescu, "On designing robust real-time map-matching algorithms," in *2014 17th IEEE International Conference on Intelligent Transportation Systems, ITSC 2014*, 2014. doi: 10.1109/ITSC.2014.6957733 pp. 464–470.

[7] C. White, D. Bernstein, and A. Kornhauser, "Some map matching algorithms for personal navigation assistants," *Transportation Research Part C: Emerging Technologies*, vol. 8, no. 1-6, pp. 91–108, 2000. doi: 10.1016/S0968-090X(00)00026-7

[8] H. Wei, Y. Wang, G. Forman, Y. Zhu, and H. Guan, "Fast Viterbi map matching with tunable weight functions," in *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*, 2012. doi: 10.1145/2424321.2424430 pp. 613–616.

[9] D. Fiedler, M. Čáp, J. Nykl, P. Žilecký, and M. Schaefer, "Map Matching Algorithm for Large-scale Datasets," *arXiv:1910.05312 [cs, eess]*, Sep. 2019, arXiv: 1910.05312. [Online]. Available: http://arxiv.org/abs/1910.05312

[10] Y. Li, Q. Huang, M. Kerber, L. Zhang, and L. Guibas, "Large-scale joint map matching of GPS traces," in *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*, 2013. doi: 10.1145/2525314.2525333 pp. 214–223.

[11] M. Kubicka, A. Cela, H. Mounier, and S.-I. Niculescu, "Comparative Study and Application-Oriented Classification of Vehicular Map-Matching Methods," *IEEE Intelligent Transportation Systems Magazine*, vol. 10, no. 2, pp. 150–166, 2018. doi: 10.1109/MITS.2018.2806630 Conference Name: IEEE Intelligent Transportation Systems Magazine.

[12] M. Hashemi and H. A. Karimi, "A critical review of real-time map-matching algorithms: Current issues and future directions," *Computers, Environment and Urban Systems*, vol. 48, pp. 153–165, Nov. 2014. doi: 10.1016/j.compenvurbsys.2014.07.009

[13] D. Srinivasan, R. Cheu, and C. Tan, "Development of an improved ERP system using GPS and AI techniques," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, vol. 1, 2003. doi: 10.1109/ITSC.2003.1252014 pp. 554–559.

[14] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk, "On map-matching vehicle tracking data," in *VLDB 2005 - Proceedings of 31st International Conference on Very Large Data Bases*, vol. 2, 2005, pp. 853–864.

[15] H. Yin and O. Wolfson, "A weight-based map matching method in moving objects databases," in *Proceedings of the International Conference on Scientific and Statistical Database Management, SSDBM*, vol. 16, 2004, pp. 437–438.

[16] L. Li, M. Quddus, and L. Zhao, "High accuracy tightly-coupled integrity monitoring algorithm for map-matching," *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 13–26, Nov. 2013. doi: 10.1016/j.trc.2013.07.009

[17] P. Newson and J. Krumm, "Hidden Markov map matching through noise and sparseness," in *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*, 2009. doi: 10.1145/1653771.1653818 pp. 336–343.

[18] R. Raymond, T. Morimura, T. Osogami, and N. Hirosue, "Map matching with Hidden Markov Model on sampled road network," in *Proceedings - International Conference on Pattern Recognition*, 2012, pp. 2242–2245.

[19] Y. Yin, R. Shah, and R. Zimmermann, "A general feature-based map matching framework with trajectory simplification," in *Proceedings of the 7th ACM SIGSPATIAL International Workshop on GeoStreaming, IWGS 2016*, 2016. doi: 10.1145/3003421.3003426

[20] C. Yang and G. Gidófalvi, "Fast map matching, an algorithm integrating hidden Markov model with precomputation," *International Journal of Geographical Information Science*, vol. 32, no. 3, pp. 547–570, 2018. doi: 10.1080/13658816.2017.1400548

[21] "GraphHopper library," Jul. 2020, original-date: 2014-12-10T09:38:00Z. [Online]. Available: https://github.com/graphhopper/map-matching

2$^{\text{nd}}$ Special Session on Data Science in Health, Ecology and Commerce

DATA Science in Health, Ecology and Commerce is a forum on all forms of data analysis, data economics, information systems and data based research, focusing on the interaction of those four fields. Here, data-driven solutions can be generated by understanding complex real-world (health) related problems, critical thinking and analytics to derive knowledge from (big) data. The past years have shown a forthcoming interest on innovative data technology and analytics solutions that link and utilize large amounts of data across individual digital ecosystems. First applications scenarios in the field of health, smart cities or agriculture merge data from various IoT devices, social media or application systems and demonstrate the great potential for gaining new insights, supporting decisions or providing smarter services. Together with inexpensive sensors and computing power we are ahead of a world that bases its decisions on data. However, we are only at the beginning of this journey and we need to further explore the required methods and technologies as well as the potential application fields and the impact on society and economy. This endeavor needs the knowledge of researchers from different fields applying diverse perspectives and using different methodological directions to find a way to grasp and fully understand the power and opportunities of data science.

This is a joint track by WIG2, the Scientific Institute for health economics and health service research, the Information Systems Institute of Leipzig University and the Helmholtz Environmental Research Institute.

Topics

We embrace a rich array of issues on data science and offer a platform for research from diverse methodological directions, including quantitative empirical research as well as qualitative contributions. We welcome research from a medical, technological, economic, political and societal perspective. The topics of interest therefore include but are not limited to:

- Data analysis in health, ecology and commerce
- (Health) Data management
- Health economics
- Data economics
- Data integration
- Semantic data analysis
- AI based data analysis
- Data based health service research
- Smart Service Engineering
- Integrating data in integrated care
- AI in integrated care
- Spatial health economics
- Risk adjustment and Predictive modelling
- Privacy in data science

Technical Session Chairs

- **Franczyk, Bogdan,** University of Leipzig, Germany
- **Militzer-Horstmann, Carsta,** WIG2 Institute for health economics and health service research, Leipzig, Germany
- **H&auml;ckl, Dennis,** WIG2 Institute for health economics and health service research, Leipzig, Germany
- **Bumberger, Jan,** Helmholtz-Centre for Environmental Research – UFZ, Germany
- **Reinhold, Olaf,** University of Leipzig / Social CRM Research Center, Germany

Program Committee

- **Alpkoçak, Adil,** Dokuz Eylul University
- **Cirqueira, Douglas,** Dublin City University
- **da Rocha Cirqueira, Douglas,** Dublin City University
- **Dey, Nilanjan,** Techno India College of Technology, India
- **Kossack, Nils,** Head Mathematics and Statistics, WIG2 Institute for Health Economics and Health Service Research
- **Kozak, Karol,** Fraunhofer and Uniklinikum Dresden, Germany
- **Popowski, Piotr,** Medical University of Gdańsk, Poland
- **Sachdeva, Shelly,** National Institute of Technology Delhi, India
- **Wasielewska-Michniewska, Katarzyna,** Systems Research Institute of the Polish Academy of Sciences, Poland
- **Wende, Danny,** WIG2 Institute for Health Economics and Health Service Research And Technical University Dresden

DOI: 10.15439/2020F175
ISSN 2300-5963 ACSIS, Vol. 21

Instance Segmentation Model Created from Three Semantic Segmentations of Mask, Boundary and Centroid Pixels Verified on GlaS Dataset

Peter Malík, Štefan Krištofík
Institute of Informatics
Slovak Academy of Sciences
Dúbravská cesta 9, 845 07 Bratislava, Slovakia
Email: {p.malik, stefan.kristofik}@savba.sk

Kristína Knapová
Faculty of Informatics and Information Technologies
Slovak University of Technology
Ilkovičova 2, 842 16 Bratislava, Slovakia
Email: knapova.kristina@gmail.com

***Abstract*—Segmentation is the key computer vision task in modern medicine applications. Instance segmentation became the prevalent way to improve segmentation performance in recent years. This work proposes a novel way to design an instance segmentation model that combines 3 semantic segmentation models dedicated for foreground, boundary and centroid predictions. It contains no detector so it is orthogonal to a standard instance segmentation design and can be used to improve the performance of a standard design. The presented custom designed model is verified on the Gland Segmentation in Colon Histology Images dataset.**

I. Introduction

SEMANTIC segmentation is the most important computer vision task in biomedical applications and any improvement of it may result in saved lives [1], [2]. Combining multiple models is a well known technique to improve segmentation. Creating an ensemble of the trained models can significantly increase the single model performance [3], [4], [5], [6], [7]. It is a favorite method of many models which helped them to be placed high in competition leader-boards. The high structural diversity within an ensemble is very beneficial; therefore, varied models are usually used within an ensemble [4], [6]. Another standard method is to use a multiple loss function with at least one element entirely focused on the boundary pixels [8], [9], [10]. The boundary pixels are harder to correctly classify and using a part of the loss function focused on them can significantly improve the overall results. More advanced method is to use a separate model (or at least a separate architectural branch) to learn boundary pixels and its results combined with the standard segmentation model [11], [12].

Instance segmentation is a more complex CV task capable to differentiate classes and objects within classes on the pixel level. The advantage of instance segmentation is a capability to count objects (even objects in contact or partial occlusion) which is very beneficial in many applications [13]. The standard approach to create an instance segmentation model is to combine a semantic segmentation model with a detection model [11], [12]. Joint training of the models improves the overall results. It also improves the single model performance [14] in detection or segmentation tasks. Newer instance segmentation models combine multiple models. One model is usually a detector, one is semantic segmentation and one is dedicated to boundary pixels [15], [16].

Our proposed method is inspired by all the mentioned techniques. We combined three semantic segmentation models into a model capable to perform the instance segmentation task. One model is semantic segmentation of foreground, one is dedicated for boundary pixels and the last model is focused on the most internal pixels (near object centroids) of all segmented objects. Our method is orthogonal to the standard instance segmentation technique because there is no detector. It is also orthogonal to the ensemble technique because the models are dedicated to the different operational tasks. Our method is tested on the GlaS dataset [17]. It is an instance segmentation dataset that provides annotations with the clear differentiation of each object and the background. The presented results are from our custom designed model based on the U-Net general structure [18] incorporating ResNet [19] blocks with spacial [3], [4] and depth-wise [20] separable convolutions. The novelty and contribution of our work is:

- new technique for designing instance segmentation models composed of three semantic segmentation models,
- the custom designed instance segmentation model,
- verification of our techniques on the GlaS dataset,
- verification that our technique improves semantic segmentation in general.

Our motivation lies in finding a novel way to create instance segmentation models that is orthogonal to currently used techniques so it can be used in combination with them to further improve the state-of-the-art models.

The rest of the paper is organized as follows. The GlaS dataset and related biomedical models are discussed in section II, our model is described in section III, training is discussed in section IV, evaluation of the predicted results are presented in section V, and section VI concludes the paper.

This work has been supported by Slovak national project VEGA 2/0155/19

II. The GlaS dataset and related biomedical models

Colorectal adenocarcinoma originating in intestinal glandular structures is the most common form of colon cancer. Patient prognosis and a treatment plan is devised by pathologists based on the morphology of intestinal glands, including architectural appearance and glandular formation. Achieving good inter-observer as well as intra-observer reproducibility of cancer grading is still a major challenge. The Gland Segmentation in Colon Histology Images Challenge Contest (GlaS) held at MICCAI'2015 has been organized with the goal to find and improve an automated approach which quantifies the morphology of glands [17]. The GlaS dataset was made public as part of this challenge. It consists of 165 images derived from 16 H&E stained histological sections (each from different patient) of stage T3 (tumour has grown into the outer lining of the bowel wall) or T4 (tumour has grown through the outer lining of the bowel wall) colorectal adenocarcinoma. The images are divided into 3 parts: training set, test A, test B containing 85, 60 and 20 images respectively.

Modern biomedical models utilize or are based on some U-Net [18] like architecture. The work [21] uses a structure learning approach to segment instances of glandular structures from colon histopathology images. The authors combined hand-crafted, multi-scale image features with features computed by a U-Net like model trained to map images to segmentation maps. The results are improved with post-processing and they reached better GlaS challenge rank (combined metric) than the challenge winner. The work [15] improves the results further. Authors created their model as a combination of 4 models. The first one segments foreground, the second one with U-Net like structure segments edges, the third one is a detector and the last one fuses these results into the instance segmentation map.

Instance segmentation is very popular in recent years. A novel hierarchical neural network comprising object detection and segmentation modules to accurate cell instance segmentation of neural cells is presented in [22]. Another work oriented to precise instance nuclei segmentation [23] presents a deep multi-scale neural network, with a novel loss function that is sensitive to the Hematoxylin intensity. The work [16] presents an instance segmentation model that segments translucent overlapping objects. Authors combined segmentation and detection models with multiple branches that allowed output transformation from 2D to 3D. The work [10] presents an instance segmentation improvement of cluttered cells by using a novel multiclass weighted loss function. The work [5] uses an ensemble of mask R-CNN models to segment polyps in colonoscopy images.

III. Instance segmentation model design

We were working with the very limited computation power and had to make some compromises. The GlaS dataset contains images in resolutions 574 × 433, 589 × 453 but most images are 775 × 522. Using high resolution inputs is highly computation intensive. Therefore, we transform all these images to 256 × 256. It is a well known fact that training with higher resolution improves prediction results in general. So, we do not expect to reach the state-of-the-art results with the reduced resolution of inputs. We also made some choices to select architectures with more efficient computation during designing of our model. More details will be mentioned later.

TABLE I
Experiments with different types of input data augmentations

Augmentation types	Loss function	F1	IoU
None	0.5	0.61	0.61
Rotation	0.38	0.84	0.55
Rotation & crop & shear	0.59	0.73	0.36
All 7 types	0.86	0.71	0.26

Medical datasets rarely contain many images. It is also true for the GlaS dataset which contains 165 images. It is a well known fact that using bigger training sets improves prediction results, allows to use higher capacity models and reduces overfitting occurrence in general. We opted for data augmentation which is a standard practice with small datasets. We designed a custom augmentation scheme that uses random combination of rotation, crop, salt & pepper noise, blurring by mean filter, shear deformation in x axis and/or y axis, horizontal and/or vertical flip, and color channel shift. The rotation is in 60 degree steps, the crop size is within 20–80%. We experimented with different combinations. Some results are in table I. All experiments with different augmentation improved the results but the effect varies. We found out that combining many augmentation types in a step is detrimental. For further experiments, we reduced the probability of multiple augmentations in a step and reduced the augmentation types to rotation, crop and salt & pepper noise.

Our design is based on the U-Net plus model. The input resolution is 256 × 256. The encoder part is composed of 5 blocks. Each block is composed of two 3×3 convolutions and 2×2 max pooling so the input resolution of the next block is halved. Each convolution is followed by the normalization and ReLU. The convolutions in the first block have 32 channels and the number of the channels is doubled with the reduced resolution. The decoder block is a mirrored image of the encoder block. It starts with transposed convolution to two-times increase the resolution and is followed by concatenation that adds outputs of the encoder block with the same resolution. The center block has resolution 8 × 8, 2 convolutions with 1024 channels and no pooling. It is considered as a part of the encoder. The last encoder block is followed by a 1×1 convolution with the sigmoid. The performance of this model is shown in the first line of table II.

We made experiments with different types of convolutions used instead of standard 2D convolutions and the results are shown in table II. We used space separable convolutions, depth-wise separable convolutions and space and depth-wise separable convolutions. Each convolution was transformed into a sequence of the selected type of separation. There are no normalization and no activation functions between

TABLE II
EXPERIMENTAL RESULTS WITH DIFFERENT CONVOLUTION TYPES

Convolution types	Loss function	F1	IoU	Time	Parameters
Standard	0.58	0.8	0.47	9.04s	31 126 563
Space separable	1.7	0.49	0.02	2.7s	26 923 811
Depth-wise separable	0.5	0.86	0.66	9.3s	14 386 881
Space and depth--wise separable	0.51	0.85	0.6	3.3s	14 386 815

TABLE III
EXPERIMENTAL RESULTS OF INCREASING THE ENCODER CAPACITY

Number of Convolution sequences in the encoder block	Loss function	F1	IoU	Time	Parameters
2	0.5	0.85	0.6	3.3s	14 386 881
4	0.57	0.83	0.58	10.13s	17 200 623
6	0.98	0.8	0.44	10.05s	20 032 431

separated convolutions. Space separable convolution uses a sequence of 3×1 and 1×3 convolutions. Depth-wise separable convolution uses a sequence of 3×3 depth-wise (applied to each channel separately) and 1×1 convolutions. The space and depth-wise separable convolution uses a sequence of 3×1 depth-wise, 1×3 depth-wise and 1×1 convolutions. The best results were reached by depth-wise separable convolutions, but they consume the most computation time to train an epoch. Therefore, we decided to use space and depth-wise separable convolution instead which has only slightly worse results but is significantly faster.

We also experimented with increasing the model capacity by doubling and tripling the number of convolutions used in the encoder block. To reduce the computational requirement, we focused only on the encoder. The results are in table III. The table shows that increasing the encoder worsened the results.

Our instance segmentation model is composed of a single encoder and 3 decoders dedicated to segment foreground, boundaries and centroid pixels of all objects (glands). Its block architecture is shown in Fig. 1. Segmentation of the boundaries helps to improve the overall segmentation and allows to separate the glands that are in a contact. Segmentation of the centroids allows to filter out the noise and to focus on the true glands.

IV. Training

Our earlier experiments were done with training from the scratch. We used default random seeding offered by Tenforflow and Keras libraries. It is well known that pretraing improves the overall results. Due to our limited computation power, we selected small biomedical datasets for pretrainig. At the beginning, we used one dataset (95 images) from Nuclei Segmentation In Microscope Cell Images dataset composition [24]. Later, we used a combination of Colorectal Adenocarcinoma Gland (CRAG) dataset (173 images) [25] and PATH-DT-MSU dataset (120 images) [26], [27] which both contain images with the cervical glands. Pretraining slightly improved the results by approximately 1 % and using slightly bigger and topically close datasets improved the results slightly further.

We used the weighted binary cross-entropy loss function for the most of our experiments because it produced the best results. We experimented with our custom designed loss function that allowed more precise weight control and focus on the boundary and centroid pixels, but it was always outperformed by weighted binary cross-entropy.

Segmentation of the boundary and centroid pixels required to create extra annotations from the ground truth masks. The annotation boundaries were separated by canny algorithm and the centroids were calculated as the center positions of tight bounding boxes. To improve the imbalance of foreground and background pixels, we increased the width of boundaries and centroids by a dilatation filter. We varied the size of the dilatation filter. The experiments showed that the best prediction results were produced when the width of the annotation boundaries was approximately 11 pixels and the width of the annotation centroids was approximately 14 pixels.

As the main metric was selected F1 score and as the second evaluation score was used Intersection over Union (IoU). F1 score correlated more with visual quality inspection of segmentation results in comparison to IoU. F1 and IoU were also used for the evaluation of boundary and centroid segmentation results. However, they were calculated from their respective annotations.

Our instance segmentation model produces 3 separate output maps that have to be combined into the final instance segmentation map. It is done by simple postprocessing. The first step uses threshold values to transform predicted values to binary numbers. The second step tightens the boundary and centroid prediction by erosion filters. To improve prediction, the wider annotations were used. The erosion transforms the prediction into tight boundaries and centroids. The third step slightly denoises the foregrounds masks by using the dilatation and erosion filter in a sequence. The fourth step subtracts the boundaries from foreground masks to find the true separation between glands in contact. The fifth step removes the objects that do not have segmented centroids. This step significantly removes the noise. The thresholds of the first step are set to lower values (approximately 0.33) so most of the true foreground pixels are segmented. It can be done this way because the fifth step removes most of false objects still present in the mask.

Hyperparameters overview. The combination of grid and line searches was used to find the optimal parameters. We selected more computation efficient solutions due to the limited computational power. Selected experiments were discussed in section III. We used Adam optimizer and the default setting achieved sufficiently good results in our experiments. The default setting is represented by beta_1 = 0.9 (the exponential decay rate for the 1st momentum estimate), beta_2 = 0.999

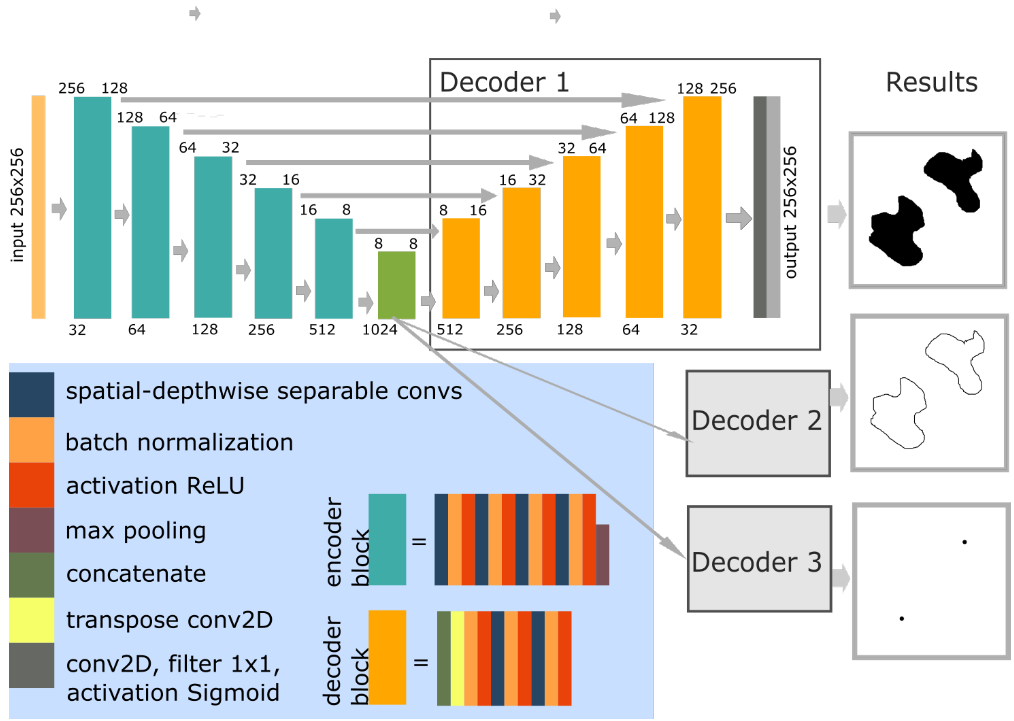


Fig. 1. Block architecture of our final instance segmentation model

(the exponential decay rate for the 2nd momentum estimate), epsilon = 1e-7 (a small constant for numerical stability). Most of our training is done with default learning rate of 0.001. We used batch size = 8 and max epochs = 150 but usually training was stopped sooner. We used early stopping with patience = 20. Input resolution = output resolution = 256 × 256, the main evaluation metric is F1 score, the additional metric is IoU and the loss function is weighted binary cross-entropy.

TABLE IV
COMPARISON THE STATE-OF-THE-ART MODELS

Model name	**F1**
CUMedVision2	0.912
ExB3	0.896
Work [15]	0.893
ExB2	0.892
Work [21]	0.892
ExB1	0.891
Our model	**0.874**
Freiburg2	0.870
CUMedVision1	0.868
CVIP Dundee	0.863
Xu et al.	0.858
Freiburg1	0.834
LIB	0.777
CVML	0.652
vision4GlaS	0.635

V. Evaluation of the Predicted Results

The GlaS dataset was used in Colon Histology Images Challenge Contest held at MICCAI'2015 and therefore there are a lot of great performing models listed in the challenge leader-board, see table IV. Our model reaches the 7th best place in F1 score while using only 256 × 256 input resolution. With reduced input resolution we did not expect to improve the state-of-the-art. Our results prove that precise instance segmentation can be done with only segmentation models, no detector is necessary. Our presented instance segmentation designed technique can be used also with a detector to improve instance segmentation further and push the state-of-the-art.

Our instance segmentation design technique can be also used to improve standard semantic segmentation. As was described in the previous section, the boundaries improve the object separation and the centroids improve the noise reduction (in the form of reduction of false predictions). The comparison of our best instance segmentation model with its only foreground (mask) segmentation branch is shown in table V. Adding boundary and centroid segmentation branches can significantly improve the performance of standard semantic segmentation.

Visual evaluation of predicted results of our best instance segmentation model and its only foreground segmenting branch can be seen in Fig. 2, Fig. 3, Fig. 4 and Fig. 5. Fig. 2 and Fig. 3 show easy samples represented by regular structure and good contrast. Fig. 4 and Fig. 5 show hard samples represented by more complex structure (irregularities, high deformations) and less contrasted texture details. Instance

TABLE V
EVALUATION OF THE IMPACT OF BOUNDARY AND CENTROID SEGMENTATION AS AN ADDITION TO THE FOREGROUND PREDICTION

Model	F1	IoU
Only foreground segmentation branch of our best IS model	0.737	0.601
Our best IS model	0.874	0.784

segmentation model clearly improves the separation between glands and helps to remove false positives.

VI. Conclusion

This work presents a novel way to design an instance segmentation model that is composed of 3 semantic segmentation models. Because it does not include a detector, it is orthogonal to standard instance segmentation design methods and can be used together with them to further improve the state-of-the-art. The presented results clearly show that adding 2 segmentation branches with foreground segmentation improves the segmentation results significantly. The boundary and centroid segmentation branches improve the separation between objects and remove false positives.

Our best performing instance segmentation model reached the 7th best result in F1 score when compare to recent works and the MICCAI'15 contest leader-board while using only 256 × 256 resolution. The model is custom designed with space and depth-wise separable convolutions and basic U-Net like structure. The segmentation models share single encoder while they use their own separate decoders.

Our model can be improved by better postprocessing in the form of fusing neural network model which we are planing to add in the future.

References

[1] T. Adams, J. Dörpinghaus, M. Jacobs, and V. Steinhage, "Automated lung tumor detection and diagnosis in ct scans using texture feature analysis and svm," in *Communication Papers of the 2018 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 17. PTI, 2018, pp. 13–20. [Online]. Available: http://dx.doi.org/10.15439/2018F176

[2] M. Li, Q. Yin, and M. Lu, "Retinal blood vessel segmentation based on multi-scale deep learning," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 15. IEEE, 2018, pp. 117–123. [Online]. Available: http://dx.doi.org/10.15439/2018F127

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1409.1556

[4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7–12, 2015*. IEEE Computer Society, 2015, pp. 1–9. [Online]. Available: https://doi.org/10.1109/CVPR.2015.7298594

[5] J. Kang and J. Gwak, "Ensemble of instance segmentation models for polyp segmentation in colonoscopy images," *IEEE Access*, vol. 7, pp. 26 440–26 447, 2019. [Online]. Available: http://dx.doi.org/10.1109/ACCESS.2019.2900672

[6] A. O. Vuola, S. U. Akram, and J. Kannala, "Mask-rcnn and u-net ensembled for nuclei segmentation," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, Venice, Italy, April 2019, pp. 208–212. [Online]. Available: http://dx.doi.org/10.1109/ISBI.2019.8759574

[7] L. Podlodowski, S. Roziewski, and M. Nurzyński, "An ensemble of deep convolutional neural networks for marking hair follicles on microscopic images," in *Position Papers of the 2018 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 16. PTI, 2018, pp. 23–28. [Online]. Available: http://dx.doi.org/10.15439/2018F389

[8] B. D. Brabandere, D. Neven, and L. V. Gool, "Semantic instance segmentation with a discriminative loss function," *CoRR*, vol. abs/1708.02551, 2017. [Online]. Available: http://arxiv.org/abs/1708.02551

[9] J. Dai, K. He, Y. Li, S. Ren, and J. Sun, "Instance-sensitive fully convolutional networks," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 534–549. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-46466-4_32

[10] F. A. Guerrero-Peña, P. D. Marrero Fernandez, T. Ing Ren, M. Yui, E. Rothenberg, and A. Cunha, "Multiclass weighted loss for instance segmentation of cluttered cells," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Athens, October 2018, pp. 2451–2455. [Online]. Available: http://dx.doi.org/10.1109/ICIP.2018.8451187

[11] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, April 2018. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2017.2699184

[12] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, June 2018, pp. 8759–8768. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2018.00913

[13] J. Respondek and W. Westwańska, "Counting instances of objects specified by vague locations using neural networks on example of honey bees," in *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 18. IEEE, 2019, pp. 87–90. [Online]. Available: http://dx.doi.org/10.15439/2019F94

[14] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 2980–2988. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2017.322

[15] Y. Xu, Y. Li, Y. Wang, M. Liu, Y. Fan, M. Lai, and E. I. Chang, "Gland instance segmentation using deep multichannel neural networks," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 12, pp. 2901–2912, December 2017. [Online]. Available: http://dx.doi.org/10.1109/TBME.2017.2686418

[16] A. Böhm, A. Ücker, T. Jäger, O. Ronneberger, and T. Falk, "Isoodl: Instance segmentation of overlapping biological objects using deep learning," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, Washington, DC, April 2018, pp. 1225–1229. [Online]. Available: http://dx.doi.org/10.1109/ISBI.2018.8363792

[17] K. Sirinukunwattana, J. P. Pluim, H. Chen, X. Qi, P.-A. Heng, Y. B. Guo, L. Y. Wang, B. J. Matuszewski, E. Bruni, U. Sanchez, A. Böhm, O. Ronneberger, B. B. Cheikh, D. Racoceanu, P. Kainz, M. Pfeiffer, M. Urschler, D. R. Snead, and N. M. Rajpoot, "Gland segmentation in colon histology images: The glas challenge contest," *Medical Image Analysis*, vol. 35, pp. 489 – 502, January 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1361841516301542

[18] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-24574-4_28

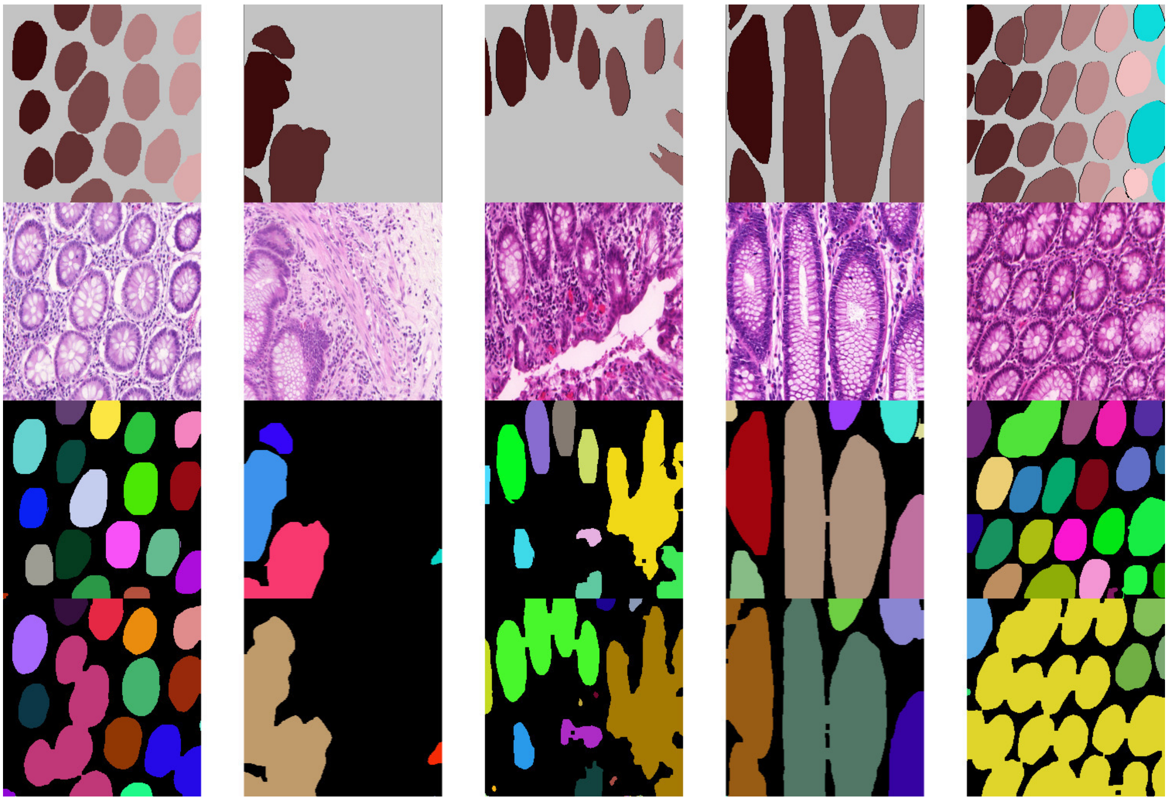


Fig. 2. Visual evaluation of easy samples. From top to bottom: annotation, input, IS prediction, only foreground prediction branch.

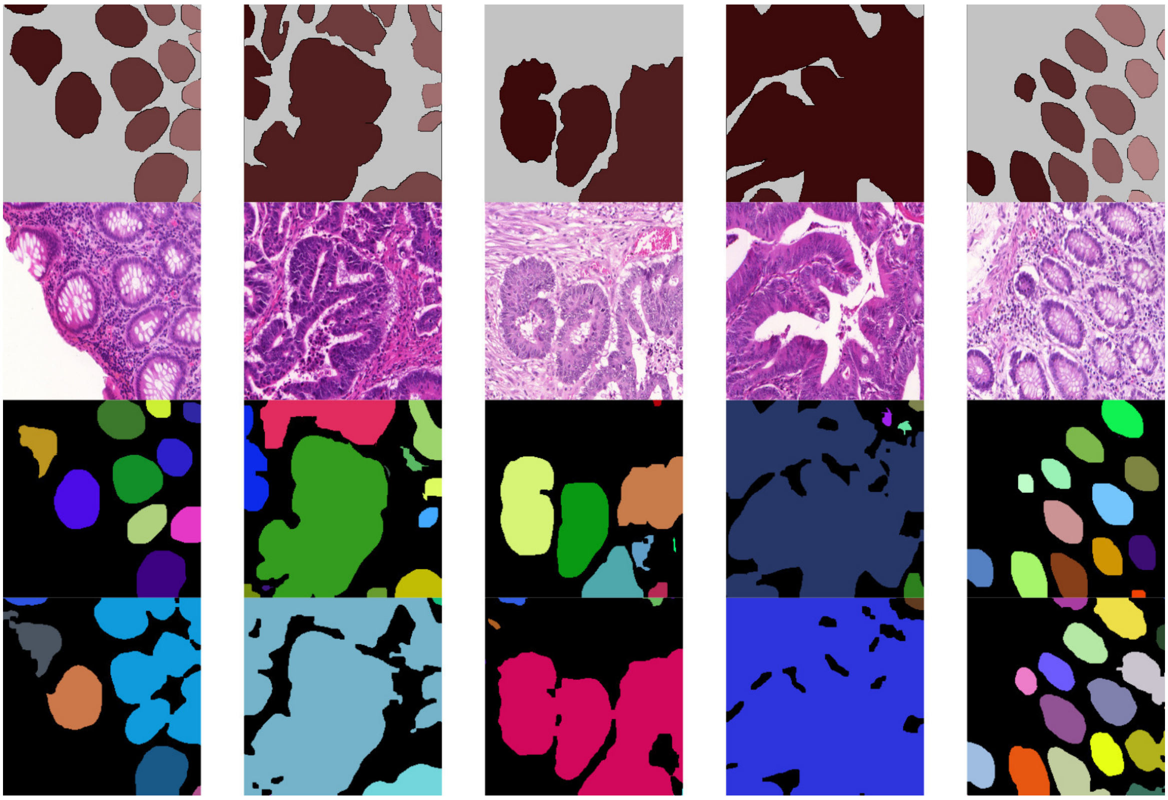


Fig. 3. Visual evaluation of easy samples. From top to bottom: annotation, input, IS prediction, only foreground prediction branch.

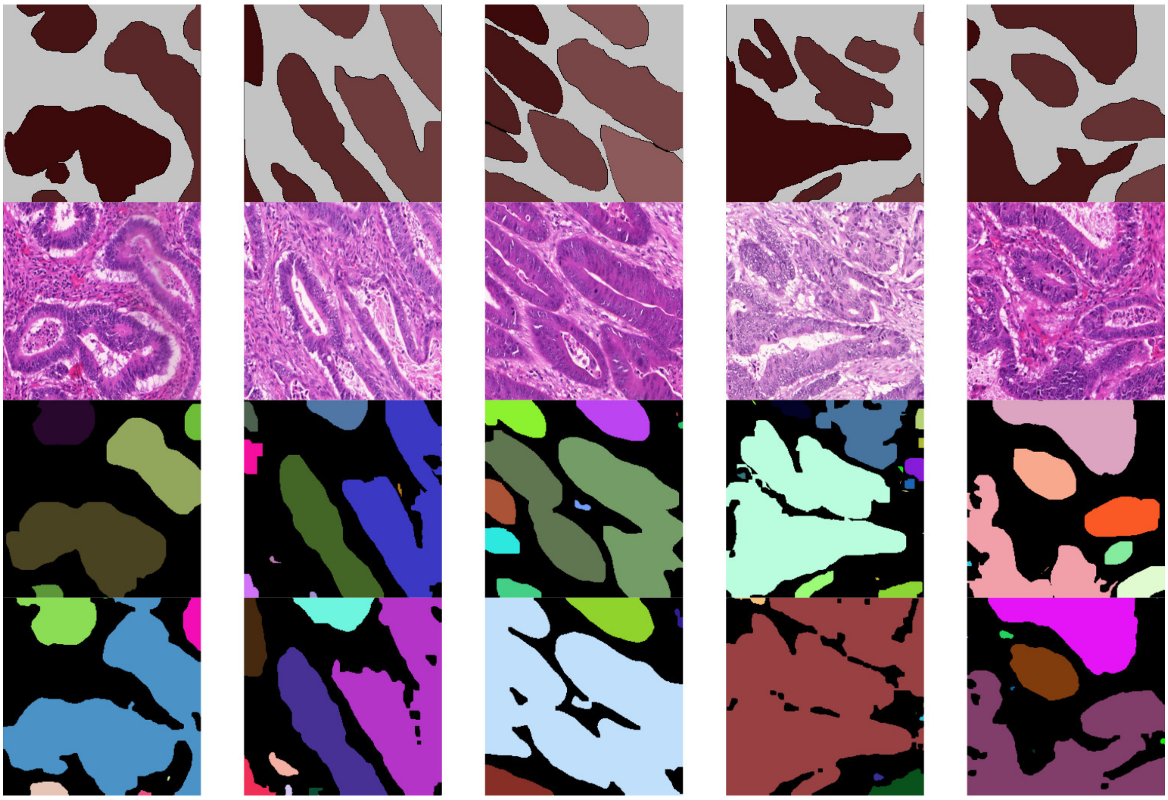


Fig. 4. Visual evaluation of hard samples. From top to bottom: annotation, input, IS prediction, only foreground prediction branch.

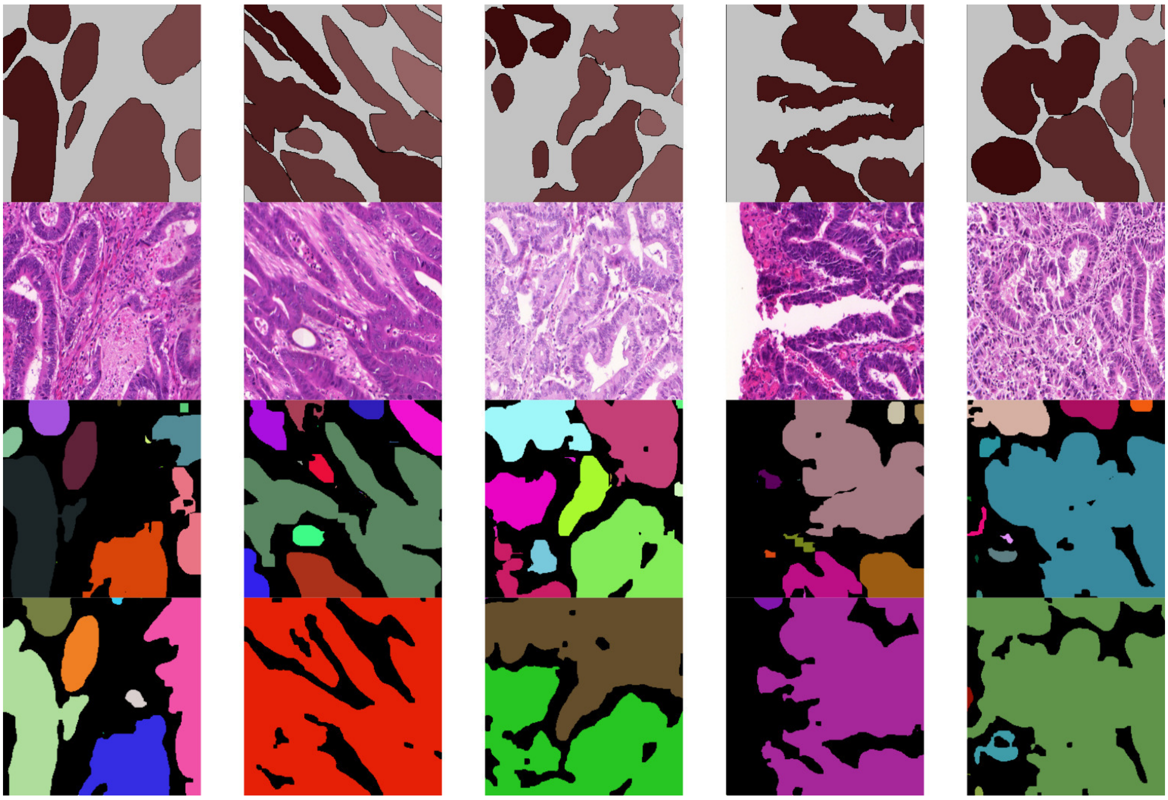


Fig. 5. Visual evaluation of hard samples. From top to bottom: annotation, input, IS prediction, only foreground prediction branch.

[19] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, June 2016, pp. 770–778. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2016.90

[20] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, July 2017, pp. 1800–1807. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2017.195

[21] S. Manivannan, W. Li, J. Zhang, E. Trucco, and S. J. McKenna, “Structure prediction for gland segmentation with hand-crafted and deep convolutional features,” *IEEE Transactions on Medical Imaging*, vol. 37, no. 1, pp. 210–221, January 2018. [Online]. Available: http://dx.doi.org/10.1109/TMI.2017.2750210

[22] J. Yi, P. Wu, D. J. Hoeppner, and D. Metaxas, “Pixel-wise neural cell instance segmentation,” in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, Washington, DC, April 2018, pp. 373–377. [Online]. Available: http://dx.doi.org/10.1109/ISBI.2018.8363596

[23] S. Graham and N. M. Rajpoot, “Sams-net: Stain-aware multi-scale network for instance-based nuclei segmentation in histology images,” in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, Washington, DC, April 2018, pp. 590–594. [Online]. Available: http://dx.doi.org/10.1109/ISBI.2018.8363645

[24] G. Payyavula, “Nuclei segmentation in microscope cell images,” 2018. [Online]. Available: https://www.kaggle.com/gangadhar/nuclei-segmentation-in-microscope-cell-images/

[25] S. Graham, H. Chen, Q. Dou, P.-A. Heng, and N. M. Rajpoot, “Mild-net: Minimal information loss dilated network for gland instance segmentation in colon histology images,” *Medical Image Analysis*, vol. 52, pp. 199–211, 2019. [Online]. Available: http://dx.doi.org/10.1016/j.media.2018.12.001

[26] A. Khvostikov, A. Krylov, I. Mikhailov, O. Kharlova, N. Oleynikova, and P. Malkov, “Automatic mucous glands segmentation in histological images,” *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-2/W12, pp. 103–109, 2019. [Online]. Available: https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W12/103/2019/

[27] N. Oleynikova, A. Khvostikov, A. Krylov, I. Mikhailov, O. Kharlova, N. Danilova, P. G. Mal’kov, N. Ageykina, and E. Fedorov, “Automatic glands segmentation in histological images obtained by endoscopic biopsy from various parts of the colon,” *Endoscopy*, vol. 51, 04 2019. [Online]. Available: http://dx.doi.org/10.1055/s-0039-1681188

15th Conference on Information Systems Management

THIS event constitutes a forum for the exchange of ideas for practitioners and theorists working in the broad area of information systems management in organizations. The conference invites papers coming from three complimentary directions: management of information systems in an organization, uses of information systems to empower managers, and information ssytems for sutainable development. The conference is interested in all aspects of planning, organizing, resourcing, coordinating, controlling and leading the management function to ensure a smooth operation of information systems in organizations. Moreover, the papers that discuss the uses of information systems and information technology to automate or otherwise facilitate the management function are specifically welcome. Papers about the influence of information systems on sustainability are also expected.

TOPICS

- Management of Information Systems in an Organization:
  - Modern IT project management methods
  - User-oriented project management methods
  - Business Process Management in project management
  - Managing global systems
  - Influence of Enterprise Architecture on management
  - Effectiveness of information systems
  - Efficiency of information systems
  - Security of information systems
  - Privacy consideration of information systems
  - Mobile digital platforms for information systems management
  - Cloud computing for information systems management
- Uses of Information Systems to Empower Managers
  - Achieving alignment of business and information technology
  - Assessing business value of information systems
  - Risk factors in information systems projects
  - IT governance
  - Sourcing, selecting and delivering information systems
  - Planning and organizing information systems
  - Staffing information systems
  - Coordinating information systems
  - Controlling and monitoring information systems
  - Formation of business policies for information systems
  - Portfolio management,
  - CIO and information systems management roles
- Information Systems for Sustainability
  - Sustainable business models, financial sustainability, sustainable marketing
  - Qualitative and quantitative approaches to digital sustainability
  - Decision support methods for sustainable management

TECHNICAL SESSION CHAIRS

- **Arogyaswami, Bernard,** Le Moyne University, USA
- **Chmielarz, Witold,** University of Warsaw, Poland
- **Jankowski, Jarosław,** West Pomeranian University of Technology in Szczecin, Poland
- **Karagiannis, Dimitris,** University of Vienna, Austria
- **Kisielnicki, Jerzy,** University of Warsaw, Poland
- **Ziemba, Ewa,** University of Economics in Katowice, Poland

PROGRAM COMMITTEE

- **Alonazi, Mohammed,** University of Sussex Informatics Department Brighton, UK, United Kingdom
- **Bicevskis, Janis,** University of Latvia, Latvia
- **Bontchev, Boyan,** Sofia University St Kliment Ohridski, Bulgaria
- **Borkowski, Bolesław,** University of Warsaw, Poland
- **Cano, Alberto,** Virginia Commonwealth University, United States
- **Carchiolo, Vincenza,** University of Catania, DIEEI, Italy
- **Czarnacka-Chrobot, Beata,** Warsaw School of Economics, Poland
- **Damasevicius, Robertas,** Kaunas University of Technology, Lithuania
- **Deshwal, Pankaj,** Netaji Subash University of Technology, India
- **Duan, Yanqing,** University of Bedfordshire, United Kingdom
- **Eisenbardt, Monika,** University of Economics in Katowice, Poland, Poland
- **El Emary, Ibrahim,** King Abdulaziz Univetrsity, Saudi Arabia
- **Espinosa, Susana de Juana,** University of Alicante, Spain
- **Fantinato, Marcelo,** University of Sao Paulo, Brazil

- **Gabryelczyk, Renata,** University of Warsaw, Poland
- **Geri, Nitza,** The Open University of Israel, Israel
- **Kania, Krzysztof,** University of Economics in Katowice, Poland
- **Kobyliński, Andrzej,** Warsaw School of Economics, Poland
- **Leyh, Christian,** University of Technology, Dresden, Germany
- **Malgeri, Michele,** University of Catania, DIEEI, Italy
- **Merkt, Oana,** Hohenheim University, Chair of Business Informatics II (530D), Germany
- **Muszyńska, Karolina,** University of Szczecin, Poland
- **Nuninger, Walter,** Polytech'Lille, Université de Lille, France
- **Rizun, Nina,** Faculty of Management and Economics, Gdansk University of Technology, Poland
- **Rozevskis, Uldis,** University of Latvia, Latvia
- **Sobczak, Andrzej,** Warsaw School of Economics, Poland
- **Swacha, Jakub,** University of Szczecin, Poland
- **Symeonidis, Symeon,** Democritus University of Thrace, Greece
- **Szczerbicki, Edward,** University of Newcastle, Australia
- **Szumski, Oskar,** University of Warsaw, Poland
- **Travica, Bob,** University of Manitoba, Canada
- **Wątróbski, Jarosław,** University of Szczecin, Poland
- **Wielki, Janusz,** Opole University of Technology, Poland
- **Zaitsev, Dmitry,** Odessa State Environmental University, Ukraine

DOI: 10.15439/2020F114
ISSN 2300-5963 ACSIS, Vol. 21

Pick-up & Deliver in Maintenance Management of Renewable Energy Power Plants

Carchiolo Vincenza*, Di Dio Francesco‡, Alessandro Longheu†,
Michele Malgeri†, Giuseppe Mangioni†, Antonio Romeo†, Natalia Trapani†

*Dipartimento di Matematica ed Informatica - Universitá di Catania - Catania - Italy
†Dipartimento di Ingegneria Elettrica Elettronica Informatica - Universitá di Catania - Catania - Italy
‡ Development and Support Center - BaxEnergy - Catania, Italy

***Abstract*—Logistic optimization is a strategic element in many industrial processes, given that an optimized logistics makes the processes more efficient. A relevant case, in which the optimization of logistics can be decisive, is the maintenance in a Wind Farm where it can lead directly to a saving of energy cost. Wind farm maintenance presents, in fact, significant logistical challenges. They are usually distributed throughout the territory and also located at considerable distances from each other, they are generally found in places far from uninhabited centers and sometimes difficult to reach and finally spare parts are rarely available on the site of the plant itself. In this paper, we will study the problem concerning the optimization of maintenance logistics of wind plants based on the use of specific vehicle routing optimization algorithms. In particular a pickup-and-delivery algorithm with time-window is adopted to satisfy the maintenance requests of these plants, reducing their management costs. The solution was applied to a case study in a renewable energy power plant. Results time reduction and simplification and optimization obtained in the real case are discussed to evaluate the effectiveness and efficiency of the adopted approach.**

I. Introduction

THE maintenance of wind power plants is a complex problem with several critical issues, whose optimization plays a significant role in determining the final costs of the energy produced [1].

The essence of improving wind turbine reliability is to reduce downtime and increase availability by optimizing its design and prescribing a well-organized maintenance schedule. These strategies require a full understanding of the system and a detailed analysis of its failure mechanisms and causes.

Several strategies have been devised for this purpose, like the Supervisory Control and Data Acquisition System (SCADA) that provides rich information about the plant itself, giving both error signals as well as components' performance information[2][3][4][5][6]. SCADA can connect individual turbines, the substation, and the meteorological stations to a central computer which allows the operator to supervise the behavior of the single wind turbine as well as the whole wind farm. Several research works exist using these systems as a primary source and using power-curve and temperature analysis[7]; they achieved good results in reporting failures and problems. Some of these research outcomes have been recognized by industry and turned into applications [8][9]. The performance of a wind turbine can be monitored systematically through a proper analysis of the collected SCADA information that covers all its sub-assemblies. However, other researches focused on a different input that involves the use of natural language and the analysis of maintenance reports compiled by operators. This approach tries to extract meaningful information from the semi-structured text and raw notes provided by maintenance operators using Natural Language Processing (NLP) techniques. To the best of our knowledge none of the existing research related to NLP aims at detecting failures related to a wind turbine, rather they just identify technology trends [10][11] In [12] the authors present a strategy using both monitoring and historical data to optimize maintenance, trying to predict the failures in order both to plan the interventions of maintenance team as well as the need of spare parts.

Whenever several wind farms must be managed, especially if they are geographically distributed on a large scale, it is necessary to pay attention to elements related to the logistic service (correct spare parts, component footprint, timing, routing efficiency, to name a few). Effectiveness and efficiency are the keys to the success of many companies, leading to reduction of losses and high service levels. A wind turbine consists of 15-20,000 components and many affect each other even if they are not directly connected. Furthermore, hard market competition and high obsolescence of components lead to a context where demand is volatile and unpredictable, therefore traditional operating strategies as creating inventories or increasing the dedicated response time consumers are not enough to gain a competitive advantage.

This paper reports some of the results of the WEAMS project [13]. WEAMS project concern with the development of an innovative asset management platform for the wind industry. One of the aims of the this project was the engineering of the platform to manage predictive maintenance strategies in wind farms. The project analyzed some logistic matters, considering different strategies to reduce costs and downtime due to routine and emergency maintenance. Specifically, this paper presents an algorithm to optimize maintenance scheduling that takes into account the location of spare parts and distributed intervention areas also located in different places, sometimes even at great distances from each other.

Section II introduces maintenance issues, focusing on logistics matters also referring to existing literature. The case study is presented in Section III where it is detailed the pickup-and-delivery algorithm used to manage wind turbine spare parts delivery. Section IV presents a couple of experiments in different scenarios. Finally, Section V briefly discusses advantages and disadvantages of the proposed approach, as well as open questions.

II. Maintenance & Logistic in a Wind Farm

A successfully predictive maintenance program, mainly in the context of wind farms, should take into account both visit scheduling and spare parts storage and delivery[14][15]. While the former issue can be tackled in traditional ways, the latter is quite complex due to the large geographical distribution of plants, often hard to reach. Moreover, spare parts are usually very large and heavy objects, difficult to move from storage to plant. Single components or sub-systems represent very different levels of the overall maintenance cost for a wind turbine. Other components exhibit very low-cost but they result in expensive in the life cycle perspective of the turbine because they can cause turbines to fail and thereby reduce the production (e.g. bearings, sensors). Moreover, the planned maintenance visits can be limited by external events such as snow or wave motion in the case of offshore wind farms.

A. *Maintenance of wind farms*

Reactive maintenance of complex and high-value installation such as wind farms is only possible if there are both a distributed spare parts storage and an intelligent scheduling algorithm that permits to reduce costs and shutdown time.

Among other typical problems of maintenance of power plants, wind farms managers must also tackle the travel times of the workers needed to reach the site and the transport of spare parts in a location often far and difficult to reach. Moreover, the maintenance providers of a wind farm are often highly specialized and they focus only on specific parts of the product, thus generating high operational expenditures (OPEX). Therefore, to stock a lot of large and heavy spare parts in several places could result in high capital costs (CAPEX). An adequate predictive maintenance strategy must take into account not only multiple stakeholders and locations in the production processes themselves but also the movements of parts - for instance in offshore plants - and reduction of the indirect cost of parts stored in a warehouse.

The classical optimization of maintenance spans over six main categories: Facility location and demand allocation, Vehicle Routing Problems, Warehouse and stock management, Goods delivery strategy, Logistic network complexity analysis, and Network performance measurement. Facility location and demand allocation study where to place the facilities (distribution centers, regional depots, collection points, etc.) and what is the optimal number of each type of facility for the location of the customers. Note how the geographical distribution typical of Wind park changes the perspective of the problem. This matter is strictly connected with Vehicle Routing Problems (VRP) which is one of the most complex combinatorial optimization problems. It consists in finding a route sets so that the vehicles can optimally serve customers' requests (according to a specific function to be optimized) while respecting constraints. The interest in solving VRP problems is motivated by their practical relevance and their inherent difficulty. Of course, the difficulties grow up in the presence of great distance and hard to reach places.

Warehouse and stock management: the goal is to determinate the correct level of stocks to be kept in the warehouses, to guarantee business continuity, in choosing the warehouse allocation policy (centralized or distributed), in determining which component will be stored in each warehouse, which should be eliminated and in general, the procurement strategies. As mentioned above, the type of spare parts and their dimension and cost make this problem more and more difficult to solve. The same problem impacts Goods delivery strategy optimization which studies the modality of movement of spare parts among the various facilities of the logistics network, including the calculation of transport costs and any outsourcing decision.

Finally, Logistic networks complexity analysis deals with the techniques and methods for studying the complexity of networks, their growth dynamics and weaknesses, to understand their level of competitiveness and performance and the Network performance measurement that aims at identifying and measuring the metrics to evaluate the system performance.

However, the context of Energy Power Plant based on Wind turbines poses new and interesting challenges to each of the previous categories. Then the design and/or optimization of a logistics network involves different aspects and many decisions which can also be sometimes conflicting. Indeed, it is rarely possible to find a solution that optimizes all aspects. More realistically, a trade-off between different key factors must be defined to balance the costs (CAPEX and OPEX) and the overall networks performance.

As said above, one of the most investigated problems concerns vehicle routing (VRP)[16]. To overcome the problem complexity various heuristics have been developed to produce good solutions with tractable computational complexity.[17][18][19]. The problem indeed presents significant computational challenges by admitting, in its more general formulation, further constraints such as the respect of time windows on both customers and deposits or by imposing a maximum vehicle transport load capacity and a maximum speed.

In the case presented in this paper, the optimization of the logistic network of green energy production, like wind and solar plants, aiming at reducing the overall cost (i.e. number of vehicles, maintenance team dimension, etc.) and complying with the time constraints. The problem to be addressed is twofold; modeling the network using complex network theory, and optimizing costs while respecting constraints through the use of VRP optimization techniques.

B. Related work

The effort typical of maintenance tasks in wind farms is due to several factors, as both space-related constraints, e.g. the difficulty of reaching off-shore (but also many on-shore) locations as well as time-related constraints, e.g. when trying to accomplish maintenance on the "right" day (as indicated by optimization algorithms) but a wind storm just hit that area. The work [14] provides a conceptual classification framework for the available literature about maintenance strategy optimization and inspection planning of wind energy systems.

An additional related matter is the logistics of spare parts, whose management significantly affects maintenance tasks; having the right part at the right time in the right location is critical to guarantee business continuity and maintenance performance.

To the best of our knowledge, no works are addressing this specific issue, e.g. in [20], authors focus mainly on weather conditions to determine the best time window and execution order for optimal intervention. Similarly, [17] proposes a hybrid heuristic optimization of maintenance routing and scheduling in particular for offshore wind farms, where optimal vessel allocation scheme is crucial (though spare parts are not considered). In [18], offshore wind farms are also addressed, in this case finding the best routes for the crew transfer vessels. Conversely, the work [21] focus on on-shore wind farms and considers forecast wind-speed values, multiple task execution modes, and daily restrictions on the routes of the technicians to determine optimal maintenance operations scheduling.

All these works tackle the question with different approaches, for instance [18] is based on the Large Neighbourhood Search meta-heuristic, whereas [21] adopts linear programming formulations and branch-and-check approach. In [20] the optimization is achieved simply through brute force whereas [17] adopts a hybrid optimization using first mixed particle swarm optimization to determine an optimal vessel allocation scheme and then discrete wolf pack search (DWPS) to optimize the maintenance route according to all constraints. A common feature most works share is the exploitation of real historical datasets to achieve realistic optimizations.

III. Pickup and Delivery Vehicle Routing Problems with Time Windows

As discussed above, in this work we address the maintenance plan optimization problem by mapping it on a specific VRP problem. To be more detailed, we employ a pickup and delivery VRP with time windows algorithm to take into account all the constraints imposed by our specific problem. It is known that determining the optimal solution to VRP is NP–hard, hence to approach such a problem many heuristics have been developed. Here we employ the algorithm proposed in [22], which consists of two phases. Indeed, it is recognized that in a typical VRP minimizing the objective function directly might not be the most efficient way to decrease the number of routes and vehicles. This because the objective function leads many times to solutions with low travel costs and this could make it difficult to reach solutions with few routes but with a higher travel cost.

To avoid this problem, the above-mentioned algorithm uses a two-stage algorithm consisting in

1) The minimization of the number of routes through the use of a Simulated Annealing algorithm.
2) The minimization of the total travel cost by using a Large Neighborhood Search algorithm.

In the following, we present the Pickup and Delivery Vehicle routing problem with time windows (PDPTW) by first introducing some definitions (taken from [22]).

Customers: The problem is defined in terms of the N customers, represented by the numbers $1, ..., N$, and a deposit, represented by the number 0. In general, with the term site, we identify the N customers and the deposit as well, i.e. sites ranges from 0 to N.

- $Customers^p$ denotes the set of withdrawal points (pickup customers).
- $Customers^d$ indicates the delivery points (delivery customers).

Travel Cost: The cost of the path between the generic sites i and j is indicated with c_{ij}. It is supposed that such a cost must satisfy the triangular inequality: $c_{ij} + c_{jk} >= c_{ik}$. The normalized travel cost $c^{'}_{ij}$ is also defined as the cost c_{ij} between sites i and j divided by the max cost among all couple of sites.

Service time: A service time is also associated with every customer i, together with a demand $q_i \geq 0$. If i is a pickup customer, the delivery counterpart is denoted by $@i$. Given that, the demand of $@i$ is $q_{@i} = -q_i$.

Vehicles: In this problem, we suppose to have m identical vehicles of capacity Q each.

Routes: In general, a route starts from the depot, visits a certain number of customers at most once, and finally returns to the depot, i.e. a route is a sequence $\{0, v_1, ...v_n, \}$, where v_i is the generic vertex of the path. Note that in a route all v_i are different, i.e. each vertex is touched only once (excluding the depot). Given a route $r = \{v_1, ...v_n, \}$, we denote with $cust(r)$ the set of of its customers, i.e. $cust(r) = \{v_1, ...v_n, \}$, With $route(c)$ we denote the route the customer c belongs to. For a given route r, its length is indicated by $|r|$, while the number of visited customers is denoted by $|cust(r)|$. The travel cost of a route is indicated by $t(r)$ and represents the cost of visiting all of its customers; it is defined as:

$$\begin{cases} t(r) = c_{0v_1} + c_{v_1v_2} + ... + c_{v_{(n-1)}v_n} + c_{v_n0} & \text{if route !=}\emptyset \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Routing plan: it is a set of routes $\{r_1, ..., r_m\}$ with $(m \geq N)$ visiting all customers exactly once:

$$\begin{cases} \bigcup_{i=1}^{m} cust(r_i) = Customers \\ cust(r_i) \cap cust(r_j) = \emptyset \qquad (1 \leq i < j \leq m) \end{cases} \tag{2}$$

A routing plan assigns a single successor and predecessor to every customer. Given a routing plan σ and a customer i, $succ(i,\sigma)$ and $pred(i,\sigma)$ are respectively the predecessor and the successor of i in the routing plan σ (shortly indicate as i^+ and i^- in the following).

Time Windows: Each site is associated with a temporal window $\{e_i, l_i\}$, where e_i represents the earliest arrival time and l_i the latest arrival time. This means that a vehicle can arrive on a site i before e_i, but it must wait e_i to start the service. Vehicles must arrive at any site i before the end of the time window l_i. In the specific case of the depot, its temporal window $[e_0, l_0]$ individuate the time e_0 in which all vehicles leave the depot and the time l_0 when all vehicles return to the depot. The departure time δ_i of a given customer i is defined as:

$$\begin{cases}\delta_0 = 0 \\ \delta_i = max(\delta_{i^-} + c_{i^-i}, e_i) + s_i \quad (i \in Customers)\end{cases} \tag{3}$$

The Earliest Service Time a_i of a given customer i is defined as:

$$a_i = max(\delta_{i^-} + c_{i^-i}, e_i) \quad (i \in Customers) \tag{4}$$

The Earliest Arrival Time $a(r)$ of a route r is defined as:

$$a(r) = \begin{cases}\delta_{v_n} + c_{v_n 0} & if\ (route! = \emptyset) \\ e_0 & otherwise\end{cases} \tag{5}$$

For a customer i the time window constraint is satisfied if $a_i \leq l_i$ and, in particular the time window constraint for the deposit is satisfied if $a(r) \leq l_0 \ \forall r \in \sigma$.

Capacities: Let us define the demand of a route r at customer c as:

$$q(c) = \sum_{i \in cust(r)\ \&\ \delta_i \leq \delta_c} q_i \tag{6}$$

With the constraint that for a customer c, $q(c) \leq Q$.

PDPTW: A solution to the PDPTW is a routing plan σ that satisfies all these constraints:

$$\begin{cases} q(i) \leq Q \\ a(r_j) \leq l_0 \\ a_i \leq l_i \\ route(i) = route(@i) \\ \delta_i \leq \delta_{@i} \end{cases} \tag{7}$$

where $i \in Customers$ and $1 \leq j \leq m$.

A solution to the PDPTW consists in finding a routing plan σ satisfying the above-mentioned constraints that also minimizes the number of vehicles and, in case of ties, the total travel cost. In formal terms σ minimizes the following objective function:

$$f(\sigma) = \langle |\sigma|, \sum_{r \in \sigma} t(r) \rangle \tag{8}$$

The algorithm used to find a solution to the PDPTW is that proposed in [22], consisting in two stages. The first one performs the minimization of the number of routes via a simulated annealing algorithm. As a classical simulated annealing algorithm, it starts from a solution and then produces a new random solution that is accepted with a probability that depends on the value produced by a domain-specific evaluation function. In particular, a new solution is produced by using a random pair relocation method (see [22] for details), while the evaluation function it uses is a lexicographic ordering function defined as in the following:

$$e(\sigma) = \langle |\sigma|, -\sum_{r \in \sigma} |r|^2, \sum_{r \in \sigma} t(r) \rangle \tag{9}$$

where the first term is the number of routes, the second term tends to favor solutions with many customers and the last term takes into account the travel cost of the routing plan.

The second stage of the algorithm proposed in [22] consists in minimizing the total travel cost by using a large neighborhood search (LNS) method. It consists of exploring the neighborhood of a given solution to find a better one, i.e. one that produces a minor value of the objective function 8. We refer the reader to [22] for additional details on the above mentioned algorithms.

IV. Experiments

As pointed out in the above sections, the optimization of VRP in the context of maintenance of Wind turbines is very complex whilst it has a high-impact on the costs and efficiency of the whole system. Indeed, the wind farms are unevenly distributed over the territory of a country as far as the spare parts deposits. Furthermore they are located in different and often distant sites.

In this section we present two examples, the former aiming at testing the effectiveness of the algorithm with a toy example and the latter to mimic a simplified real scenario. Both examples concerns a *"single spare parts store"* scenario.

A. Basic setup

The topology chosen for the first example is based on 10 nodes and 1 depot with a symmetric topology. The cost between each pair of nodes is assumed equal to 1. In particular, this setup encompasses five plain routes connecting five places, called A, B, C, D, E, each route contains only one delivery and one dispatch point. Therefore, the problem is described by 10, where $\mathcal{P}$, $\mathcal{D}$, $\mathcal{C}$ are the set of Pickups, Deliveries and Customers respectively, and by the set of routes described in eq. 11.

$$\begin{aligned}&\mathcal{P} = \{A, B, C, D, E\} \\ &\mathcal{D} = \{@A, @B, @C, @D, @E\} \\ &\mathcal{C} = \mathcal{P} \cup \mathcal{D} = \{A, B, C, D, E, @A, @B, @C, @D, @E\}\end{aligned} \tag{10}$$

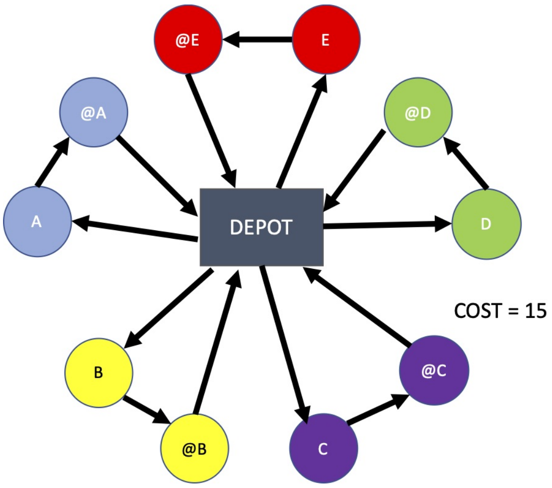


Fig. 1. Example 1: topology

$$
\begin{aligned}
r_1 &= \{depot, A, @A, depot\}\\
r_2 &= \{depot, B, @B, depot\}\\
r_3 &= \{depot, C, @C, depot\}\\
r_4 &= \{depot, D, @D, depot\}\\
r_5 &= \{depot, E, @E, depot\}
\end{aligned}
\tag{11}
$$

Figure 1 illustrates all the routes that start from *depot*. Each connection ha the same cost and it is equal to 1, the earliest arrival time to the *depot* is 40 hours and the time needed to get each customer is 2 hours. The global cost is then equal to 15.

The algorithm tries to optimize the solution according to the following two steps:

SA the route is reduce with the Simulated Annealing
LNS the route is optimized with the *Travel Cost Minimize* Function

Note that the LNS step may not converge; when this happens, it is advisable to change some values of setup and restart from scratch. The setting values are summarized in Table I.

TABLE I
FIRST EXAMPLE SETTING

Step	Item	Value
SA	Temperature value	28
	Temperature Limit value	15
	α	0.5
	Max Iterations	1
	β	1
LNS	Max Searches	5
	Max Iterations	1
	beta	2

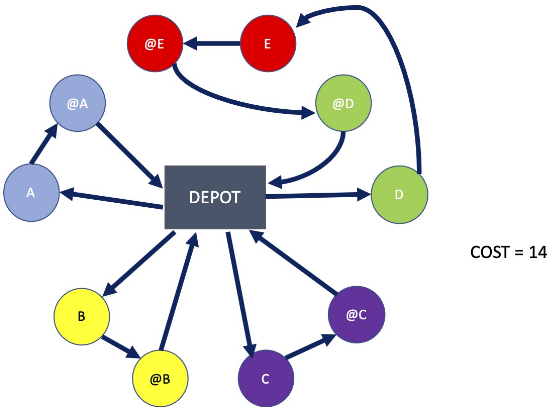


Fig. 2. Example 1: Routes after Simulation Annealing

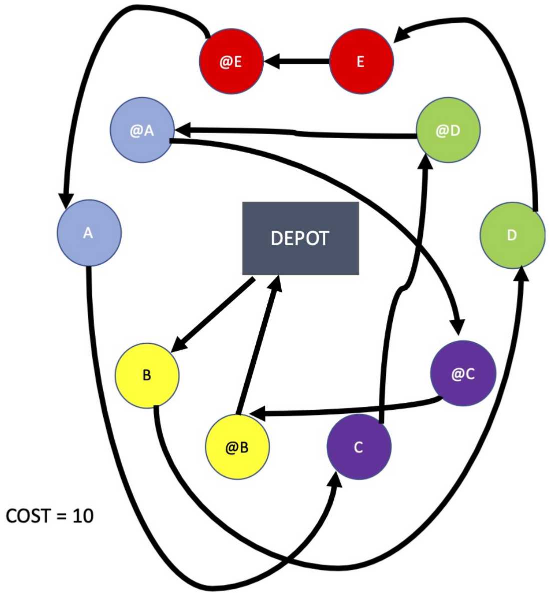


Fig. 3. Example 1: Routing after LNS

SA step looks for new routes with a better cost, in the example four routes exist with a global cost of 14.

$$
\begin{aligned}
r_0 &= \{depot, A, @A, depot\}\\
r_1 &= \{depot, B, @B, , depot\}\\
r_2 &= \{depot, C, @C, depot\}\\
r_3 &= \{depot, D, E, @E, @D, depot\}
\end{aligned}
\tag{12}
$$

Figure 2 illustrates all the routes after simulation annealing.

Finally, the LNS optimization produces a single route 13, shown in 3, which has a cost equal 10 that is much better the initial value.

$$
r_0 = \{depot, B, D, E, @E, A, C, @D, @A, @C, @B, depot\} \tag{13}
$$

B. Experiments on real routes

The supply of spare parts usually deals with two different scenarios: the former concerns with a single plant where each node represents a single wind turbine and the latter scenario concerns with the portfolio of a producer where a single node represents a whole farm. However, the problem to optimize is quite the same since we look for the cheaper path connecting *depot* with several *nodes*.

In this paper we present a case study belonging to second scenario and we suppose that a single warehouse (the *depot* node) provides all the farm with the spare parts.

We search for solutions that satisfy all the constraints defined for the algorithm. Therefore, we define for each site i a time window $[e_i, l_i]$ representing the lower and upper limits to perform an effective maintenance. That is, the spares must not arrive before e_i and not after l_i, if they arrives before e_i they must wait at least until e_i before starting maintenance.

The experiments deals with the functional maintenance of 10 farms located in Italy covering the management of very expensive and large spare parts. Table II defines all involved nodes (pickup or delivery node), the distance among nodes was build using Google map services and Table III summarizes the setup parameters.

TABLE II
PICKUP AND DELIVERY POINTS

Pickup	Name	Delivery	Name
Enna	A	Brindisi	@A
Florence	B	Genova	@B
Catania	C	Bari	@ C
Taranto	D	Naples	@ D
Milan	E	Pompei	@ E
Bologna	F	Bozen	@ F
Rome	G	Cagliari	@ G
Sassari	H	Pirri	@ H
Agrigento	I	Mele	@ I
Viterbo	L	Palese	@ L

TABLE III
EXAMPLE SETTING

Step	Item	Value
SA	Temperature value	28
	TemperatureLimit value	3
	α	0.3
	MaxIterations	2
	β	1
LNS	MaxSearches	2
	MaxIterations	2
	β	2

In order to complete the setup we selected four routes 14

$$\begin{aligned} r_0 &= \{depot, A, @A, B, @B, depot\} \\ r_1 &= \{depot, C, @C, D, @D, depot\} \\ r_2 &= \{depot, E, @E, F, @F, G, @G, depot\} \\ r_3 &= \{depot, H, @H, I, @I, L, @L, depot\} \end{aligned} \tag{14}$$

We assume that each km have a cost of 1, then the global cost of this four route is calculate using the distance matrix is equal to 7791.

After SA step the new four routes are shown in 15

$$\begin{aligned} r_0 &= \{depot, A, @A, B, @B, depot\} \\ r_1 &= \{depot, C, @C, D, @D, depot\} \\ r_2 &= \{depot, E, @E, F, @F, depot\} \\ r_3 &= \{depot, H, G, , @G, @H, I, @I, L, @Ldepot\} \end{aligned} \tag{15}$$

Unluckily, In this case the LNS optimization is failed, therefore changed the setup according to Table IV.

TABLE IV
NEW SETTING

Step	Item	Value
SA	Temperature value	30
	TemperatureLimit value	9
	α	0.5
	MaxIterations	2
	β	3
LNS	MaxSearches	5
	MaxIterations	2
	β	1

Finally, after both steps, we find the following three routes (see 16) that save more than 15% of the cost.

$$\begin{aligned} r_1 &= \{depot, A, C, D, @D, @C, @A, B, @B, depot\} \\ r_2 &= \{depot, E, @E, F, @F, G, @G, depot\} \\ r_3 &= \{depot, H, @H, I, @I, L, @L, @D, @L, depot\} \end{aligned} \tag{16}$$

V. Conclusions and Future Work

In this paper we described a case study concerning Logistic optimization, in particular the maintenance in a Wind Farm, where many challenges exist, from wind turbines location (sparse and sometimes difficult to reach, especially off-shore ones), to spare parts management (from stock to wind farm), to vehicle routing optimization algorithms.

We introduced a pickup-and-delivery algorithm with time window in a renewable energy power plant scenario, and results show that both effectiveness and efficiency are achieved.

Further works concern the extension of the proposed approach to the case of multiple stocks and the adoption of machine-learning based algorithm to manage and refill these stocks, with the same purpose of optmizing procurement time and costs. Moreover, other case studies can be examined to validate the proposed approach, also considering features as multi-site and multi-team in addition to multi-stock.

Acknowledgment

This work was partially supported by WEAMS N. F/050145/00/X32 project

References

[1] F. Castellani, D. Astolfi, P. Sdringola, S. Proietti, and L. Terzi, "Analyzing wind turbine directional behavior: Scada data mining techniques for efficiency and power assessment," *Applied Energy*, vol. 185, pp. 1076 – 1086, 2017. doi: 10.1016/j.apenergy.2015.12.049 Clean, Efficient and Affordable Energy for a Sustainable Future. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306261915016220

[2] X. Jin, Z. Xu, and W. Qiao, "Condition monitoring of wind turbine generators using SCADA data analysis," *IEEE Transactions on Sustainable Energy*, pp. 1–1, 2020. doi: 10.1109/TSTE.2020.2989220

[3] L. Chen, G. Xu, Q. Zhang, and X. Zhang, "Learning deep representation of imbalanced scada data for fault detection of wind turbines," *Measurement*, vol. 139, pp. 370 – 379, 2019. doi: 10.1016/j.measurement.2019.03.029. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0263224119302386

[4] J. Dai, W. Yang, J. Cao, D. Liu, and X. Long, "Ageing assessment of a wind turbine over time by interpreting wind farm scada data," *Renewable Energy*, vol. 116, pp. 199 – 208, 2018. doi: 10.1016/j.renene.2017.03.097 Real-time monitoring, prognosis and resilient control for wind energy systems. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960148117302896

[5] P. B. Dao, W. J. Staszewski, T. Barszcz, and T. Uhl, "Condition monitoring and fault detection in wind turbines based on cointegration analysis of scada data," *Renewable Energy*, vol. 116, pp. 107 – 122, 2018. doi: 10.1016/j.renene.2017.06.089 Real-time monitoring, prognosis and resilient control for wind energy systems. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960148117305931

[6] P. Bangalore and M. Patriksson, "Analysis of scada data for early fault detection, with application to the maintenance management of wind turbines," *Renewable Energy*, vol. 115, pp. 521 – 532, 2018. doi: 10.1016/j.renene.2017.08.073. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960148117308340

[7] D. Astolfi, F. Castellani, and F. Natili, "Wind turbine generator slip ring damage detection through temperature data analysis," *Diagnostyka*, vol. 20, no. 3, pp. 3–9, 2019. doi: 10.29354/diag/109968. [Online]. Available: http://dx.doi.org/10.29354/diag/109968

[8] C. Sequeira, A. Pacheco, P. Galego, and E. Gorbena, "Analysis of the efficiency of wind turbine gearboxes using the temperature variable," *Renewable Energy*, vol. 135, no. C, pp. 465–472, 2019. doi: 10.1016/j.renene.2018.12. [Online]. Available: https://ideas.repec.org/a/eee/renene/v135y2019icp465-472.html

[9] Y. Qiu, Y. Feng, J. Sun, W. Zhang, and D. Infield, "Applying thermophysics for wind turbine drivetrain fault diagnosis using SCADA data," *IET Renewable Power Generation*, vol. 10, no. 5, pp. 661–668, 2016. doi: 10.1049/iet-rpg.2015.0160

[10] E. Máximo and V. Pinheiro, "XMILE - an expert system for maintenance learning from textual reports (S)," in *The 30th International Conference on Software Engineering and Knowledge Engineering, Hotel Pullman, Redwood City, California, USA, July 1-3, 2018*, Ó. M. Pereira, Ed. KSI Research Inc. and Knowledge Systems Institute Graduate School, 2018. doi: 10.18293/SEKE2018-197 pp. 492–491. [Online]. Available: 10.18293/SEKE2018-197

[11] C. Bertero, M. Roy, C. Sauvanaud, and G. Tredan, "Experience report: Log mining using natural language processing and application to anomaly detection," in *2017 IEEE 28th International Symposium on Software Reliability Engineering (ISSRE)*, 2017. doi: 10.1109/ISSRE.2017.43 pp. 351–360.

[12] V. Carchiolo., A. Longheu., V. di Martino., and N. Consoli., "Power plants failure reports analysis for predictive maintenance," in *Proceedings of the 15th International Conference on Web Information Systems and Technologies - Volume 1: WEBIST,*, INSTICC. SciTePress, 2019. doi: 10.5220/0008388204040410. ISBN 978-989-758-386-5 pp. 404–410.

[13] V. Carchiolo, G. Catalano, M. Malgeri, C. Pellegrino, G. Platania, and N. Trapani, "Modelling and optimization of wind farms' processes using BPM," in *Information Technology for Management: Current Research and Future Directions*, E. Ziemba, Ed. Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-030-43353-6_6. ISBN 978-3-030-43353-6 pp. 95–115.

[14] M. Shafiee and J. D. SÃžrensen, "Maintenance optimization and inspection planning of wind energy assets: Models, methods and strategies," *Reliability Engineering & System Safety*, vol. 192, p. 105993, 2019. doi: 10.1016/j.ress.2017.10.025 Complex Systems RAMS Optimization: Methods and Applications. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S095183201630789X

[15] Y. Dalgic, I. Lazakis, I. Dinwoodie, D. McMillan, and M. Revie, "Advanced logistics planning for offshore wind farm operation and maintenance activities," *Ocean Engineering*, vol. 101, pp. 211 – 226, 2015. doi: 10.1016/j.oceaneng.2015.04.040. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0029801815001213

[16] S. Jbili, A. Chelbi, M. Radhoui, and M. Kessentini, "Integrated strategy of vehicle routing and maintenance," *Reliability Engineering & System Safety*, vol. 170, pp. 202 – 214, 2018. doi: 10.1016/j.ress.2017.09.030. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0951832017303174

[17] D. Fan, Y. Ren, Q. Feng, B. Zhu, Y. Liu, and Z. Wang, "A hybrid heuristic optimization of maintenance routing and scheduling for offshore wind farms," *Journal of Loss Prevention in the Process Industries*, vol. 62, p. 103949, 2019. doi: 10.1016/j.jlp.2019.103949. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950423019304462

[18] C. A. Irawan, M. Eskandarpour, D. Ouelhadj, and D. Jones, "Simulation-based optimisation for stochastic maintenance routing in an offshore wind farm," *European Journal of Operational Research*, 2019. doi: 10.1016/j.ejor.2019.08.032. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221719307027

[19] Y. T. Sasmi Hidayatul, A. Djunaidy, and A. Muklason, "Solving multi-objective vehicle routing problem using hyper-heuristic method by considering balance of route distances," in *2019 International Conference on Information and Communications Technology (ICOIACT)*, 2019. doi: 10.1109/ICOIACT46704.2019.8938484 pp. 937–942.

[20] N. Y. Yurusen, P. N. Rowley, S. J. Watson, and J. J. Melero, "Automated wind turbine maintenance scheduling," *Reliability Engineering & System Safety*, vol. 200, p. 106965, 2020. doi: 10.1016/j.ress.2020.106965. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0951832019306209

[21] A. Froger, M. Gendreau, J. E. Mendoza, E. Pinson, and L.-M. Rousseau, "A branch-and-check approach for a wind turbine maintenance scheduling problem," *Computers & Operations Research*, vol. 88, pp. 117 – 136, 2017. doi: 10.1016/j.cor.2017.07.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S030505481730165X

[22] R. Bent and P. V. Hentenryck, "A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows," *Computers & Operations Research*, vol. 33, no. 4, pp. 875 – 893, 2006. doi: 10.1016/j.cor.2004.08.001 Part Special Issue: Optimization Days 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0305054804001911

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 587–590 DOI: 10.15439/2020F95 ISSN 2300-5963 ACSIS, Vol. 21

Data Mining for Process Modeling: A Clustered Process Discovery Approach

Renato Cirne, Caio Melquiades, Renan Leite, Eronita Leijden, Alexandre Maciel, Fernando Buarque de Lima Neto
University of Pernambuco (UPE)
Recife, Brazil
Email: {rbc3, casml, rfl, emlvl, amam, fbln}@ecomp.poli.br

***Abstract*—Process mining has emerged as a new scientific research topic on the interface between process modeling and event data gathering. In the search for process models that best fit to reality, the process discovery approach of creating referential processes from observed behavior. However, despite these methods showing relevant results, when faced with noisy and divergent tendencies they end up producing limited results. This work proposes the application of process discovery technique, combined to cluster technique k-means, to generate new process models, considering its conformance checking measures. The proposed solution is applied to an *ad hoc* workflow. And as a result, the use of the clustering techniques coupled with process discovery showed significant gains in the generation of process models, unlike the standard approach.**

I. Introduction

DUE to the challenges posed to companies arising from the difficulty of managing complex process flow networks, various types of problems can occur, such as delays, rework, and waste of resources. Business process management methods have been introduced to maximize process that ensure alignment of business strategies with customer and stakeholders aims [1]. Typical examples of improvement include cost savings, runtimes and failure reductions.

Most traditional areas such as Data Mining (DM), Business Intelligence (BI), and Machine Learning (ML) focus on data without considering end-to-end process models. To reduce the gap between these fields of study, process mining techniques have been successfully used [2].

The challenge of process mining is to turn big data into valuable insights related to process performance and compliance. Process mining results can be used to identify and understand bottlenecks, inefficiencies, deviations, and risks [3]. Furthermore, its techniques have been applied in several real-world system, such as [4].

One of the main focuses of the study of process mining and the object of this study is process discovery, where, based on observed behavior, a process model capable of reproducing event logs is inferred [5].

It is worth pointing out that, according to Bose et. al [6], most real-life logs tend to be granular, heterogeneous, voluminous, incomplete, and noisy. Some of the most advanced process discovery techniques try to address these problems. Therefore, one of the categories of process mining data quality problems is noisy data or outliers. Most process mining techniques are misled by the presence of outliers, which impacts the quality of the mined results [6].

This work proposes the use of the process discovery technique, using the α-Algorithm, a technique strongly impacted by noise, combined with the technique of group selection of instances from data clustering (k-means) to generate new process models, taking into consideration their compliance function in the pursuit of the best process models.

II. State of Art

A. Process Mining

Process mining is a bridge between data mining and business process modeling [3]. To this end, it provides a process analysis method based on models and data-oriented analysis techniques. Through real datasets and algorithms, the approach provides scientific knowledge that can be applied directly to analyze and improve processes in a variety of domains [2].

An event log can be any ordered list of records known as events. Every event has at least a case identifier, an activity identifier, and some additional property such as a timestamp that can be considered to put the events into some deterministic order. This mechanism allows us to point to a specific event or a specific case. A case identifier is used to group events belonging somehow into some common contexts [7]. Therefore, such objects are important for the area of process mining and are defined by van der Aalst [3] as follows.

Definition 1 (*case*) Let C be *case universe*, i.e., a set of all possible cases identifier. Cases have attributes. For any case $c \in C$ and $n \in AN$: $\#_n = \perp$ is the value of attribute *n* for case *c* ($\#_n(c) = \perp$ if case *c* has no attribute named *n*).

Definition 2 (*event log*) Let *L* be a set of cases, i.e., $L \subseteq C$, such that each event appears once in the entire log at most, i.e., for any c$_1$, $c_2 \in L$ such that $c_1 \neq c_2 : \partial set(\widehat{c}_1) \cap \partial set(\widehat{c}_2) = \emptyset$. If an event log contains timestamps, then the ordering in a trace should respect these timestamps, i.e., for any case $c \in L$, *i* and *j* such that $1 \leq i < j \leq |\widehat{c}| : \#_{time}(\widehat{c}(i)) \leq \#_{time}(\widehat{c}(j))$.

Thus, for every event, an unambiguous case can be identified, which represents a collection of events belonging to the same process. The events for a case are represented in the form of a trace, i.e., a sequence of unique events.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001

Definition 3 (*trace*) Let σ be a finite sequence of events and $\sigma \in E^*$, such that each event appears only once, i.e., for $1 \leq i < j \leq |\widehat{\sigma}| : \sigma(i) \neq \sigma(j)$. Each case has a special mandatory attribute trace, $\#_{trace}(c) \in E^*.\widehat{c} = \#_{trace}(c)$ is a shorthand for referring to the trace of a case.

It is worth mentioning, in addition to the properties listed above, every event can also include any number of additional event attributes. Among the many existing ones, the process mining purpose to this project is 'Discovery'. This technique uses an event log (Definition 2) and produces a process model without using any prior information.

Process discovery output is the process model, describing events and flows. This model serves to check if events are occurring according to the proposed description [5], which is useful for compliance assurance. Therefore, compliance addresses events that should happen and are not occurring, and events that happen and are not described in the model.

In this context, the α-Algorithm is widely accepted and used [3]. It aims to extract an event log and produce a process model explaining the behavior recorded in the log [5]. So, the α-Algorithm is a process discovery algorithm that aims to build a process model through the mutual occurrences of a set of scenarios, using log-based ordering relations (Definition 4). This algorithm has as input a set of events and results in a Petri Net, defined by van der Aalst [3] and conforms to the example shown in Figure 1. In this case, events form sequences that relate to various scenarios, and the path of each scenario reports on the network.

Definition 4 (*Log-based ordering relations*) Let L be an event log over A, i.e., $L \in B(A^*)$. Let $x, y \in A$:

- $x >_L y$ if and only if there is a trace $\sigma = \langle t_1, t_2, ..., t_n \rangle$ and $i \in 1, 2, ..., n-1$ such that $\sigma \in L$ and $t_i = x$ and $t_{i+1} = y$;
- $x \rightarrow_L y$ if and only if $x >_L y$ and $x \not>_L y$;
- $x \#_L y$ if and only if $x \not>_L y$ and $y \not>_L x$;
- $x \|_L y$ if and only if $x \not>_L y$ and $y \not>_L x$.

Through a conformance checking purpose, it is possible to evaluate the existence of divergences between the model and the base, and assign a value to it [2]

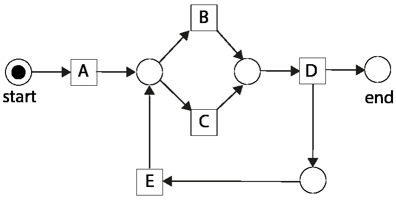


Fig. 1. Example of model in representation of Petri Net

Conformance checking [3] relates events in the event log to activities in the process model and compares both. The objective is to find similarities and discrepancies between the modeled behavior and the observed behavior.

The *token-based replay* is a refined conformance checking method that assigns a fitness value to each scenario. It allows somehow to discover the fraction of the scenario that conforms to the model and the fraction that does not [2]. The fitness calculation is done by counting the missing network tokens (m), and the remaining (r), the produced (p) and the consumed (c) ones, according to Definition 5 [8].

Definition 5 (*Fitness-token-based replay*). Let E be an event log and *PN* process model represented by a Petri Net. For each trace $\sigma \in S = \alpha(E)$ (simplified log, i.e., every event in *E* is replaced for activity attribute), consider m_σ the number of missing tokens, r_σ the number of remaining tokens, c_σ the number of consumed tokens, and p_σ the number of produced tokens during *E* reproduction in *PN*, Fitness-token-based replay (*Tbr*) is defined by:

$$Tbr = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in S} S(\sigma).m_\sigma}{\sum_{\sigma \in S} S(\sigma).c_\sigma}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in S} S(\sigma).r_\sigma}{\sum_{\sigma \in S} S(\sigma).p_\sigma}\right)$$

Thus, according Rozinat and van der Aalst [9], the number of tokens that had to be created artificially (that is, the transition belonging to the registered event was not activated and therefore could not be successfully executed) is counted and the number of tokens that were left in the model, which indicates that the process was not completed correctly. From Definition 5 it can be concluded that the closer to 1, the higher the model conforms to the reference event logs.

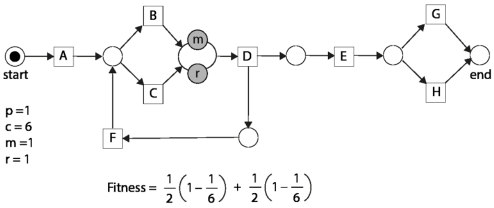

B
m
r
A
start
D
E
G
H
end
C
p=1
c=6
m=1
r=1
F
Fitness = $\frac{1}{2}\left(1-\frac{1}{6}\right) + \frac{1}{2}\left(1-\frac{1}{6}\right)$

Fig. 2. Conformance checking example [3]

By means of example, Fig. 2 presents the calculation of conformance checking over a given Petri Net and a trace equal to $\langle A, D, C, E, H \rangle$.

B. K-means

K-means is a clustering algorithm based on Euclidean distance in vector spaces. This algorithm searches for center points (or centroids) that group the input vectors into sets. Each cluster has a centroid, and the k parameter determines the number of centroids (and thus, the number of clusters).

One of the techniques used to determine k, the number of clusters, is called the elbow method. It is a visual method. The idea is that it starts with k = 2, and increases the k step by step by 1, calculating the clusters and training-related cost. At some k value, the cost drops dramatically, and so it reaches a plateau when k is raised again, and hence the desired k value is obtained [10].

It is noteworthy that this type of combination of techniques has already been the object of research in the area of process mining, but from a different approach [11], which iteratively divided the traces into clusters until the log was partitioned into clusters that allow the generation of more accurate process models.

Other example of this preprocessing approach, Greco et al [12] have devised a novel framework that substantially differs from previous approaches for it performs a hierarchical clustering in which each trace is seen as a point of a properly identified space of features.

Hinkka et al. [7] concluded that the most consistent feature selection algorithm was the cluster algorithm developed in their paper, which first used the k-means algorithm to group the characteristic in the desired number of clusters.

Recently, Fani Sani et al. [13] analyzed several methods of selecting subsets and demonstrated that it is possible to considerably accelerate the discovery using strategies of subset selection of features. In addition, the results show that selection with some bias of process instances compared to random selection results in higher quality process models.

III. Materials and methods

A. Experiment

The database used here has been provided by the Government of the State of Pernambuco (Brazil) and consists of all public data recorded in the Electronic Information System (SEI) that occurred until 27/03/2019.

As an *ad hoc* workflow, it runs business processes with no predetermined pattern of information movement between users. Moreover, the use of techniques such as process discovery allows application on new challenges of process management.

For this, a CSV file has been created using the relevant attributes for the research and data referring to the following processes of a public agency of Pernambuco Government: (1) passive transparency; (2) Equity Movement; (3) Holiday Alteration; (4) Electoral License; (5) Contract Monitoring.

For α-Algorithm execution, by default, the attributes that represent the process event log have been renamed and the data were converted to the eXtensible Event Stream (XES) format developed by Verbeek et al. [14] to meet the α-Algorithm assumption.

Finally, the experiment has been confirmed using a sample of dataset made available by BPI Challenge 2019 collected [15] from a large multinational company operating from the Netherlands in the area of coatings and paints. Specifically, one type of cases in the data “3-way matching, invoice before goods receipt” has been used.

B. Modeling

In order to reduce the complexity of the model produced by the α-Algorithm, it has been realized an unsupervised search for scenario blocks that produced similar conformance variables and produced models of these blocks, as shown in Fig. 3.

Initially, the α-Algorithm is executed on the entire event log to generate an initial model, to be used at the end of the experiment to compare performance gains.

In summary, the experiment performed the following steps:

1) the event log is randomly divided into *n blocks* with the same number of cases;
2) α-Algorithm is applied to generate the model on *n-1* blocks *remaining* for each block;
3) *token-based replay* (TBR) is performed on the selected block in comparison with the generated model. The procedure is repeated with all the blocks, so that each one of them is used once, guaranteeing the complete approach of all data;
4) before using k-means for clustering traces, the elbow method was applied to determine the number of clusters created (k) (Tbr measures);
5) k-means algorithm is used to create groups of traces in each scenario using the variables produced (p), consumed (c), remaining (r) and missing (m);
6) the best group of traces is chosen by calculating the fitness average of the most representative token-based replay method.

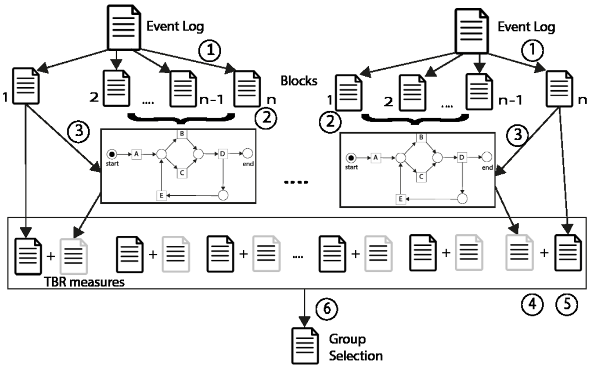


Fig. 3. Method proposed

IV. Analysis

Considering the previously established criteria and after data pre-processing, the α-Algorithm has been used for the generation of the initial Petri Net of these process types.

In order to verify the quality of the initially generated network, a conformance checking has been calculated using the token-based replay method of the generated model in comparison with all the traces related to the process in question (column “Initial” of Table 1).

Subsequently, the dataset has been divided into five blocks, and the α-Algorithm has been used to generate new Petri Nets. Then, the statistical analysis of the generated processes considering all the blocks has been performed. In this context, the graph shown in Fig. 4 demonstrates that the number of tokens produced per trace has greater dispersion, as well as the number of remaining tokens. This feature may imply that the traces are diverse and have relevant cases of unfinished flows, demonstrating their heterogeneity. Therefore, this suggests that the use of clustering techniques improves the model.

Before using k-means for clustering traces, the elbow method had been applied to determine the number of clusters created (k). In the five types of process in question, the mode was k equal 3, thus being the most suitable for clustering.

Finally, after selecting the traces group that has a higher average fitness value from each block, a new process model has been generated, and conformance checking calculated, considering all traces, as presented in the “Final” column of Table 1.

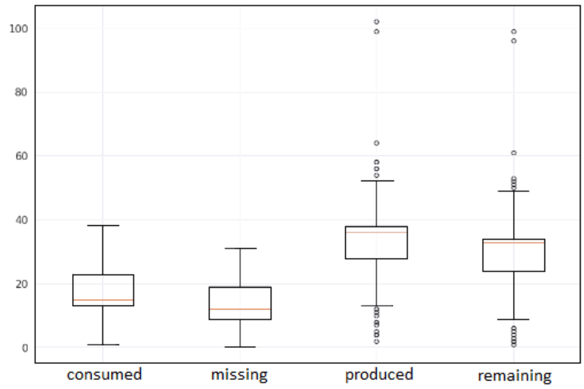


Fig. 4. Boxplot - Conformance Checking Chart of All Blocks - Process Type: Passive Transparency

From the results presented in Table 1, the fitness function of the initial model presented gains in the conformance checking calculation, using the *token-based replay* method, ranging from 6% to 30%, depending on the type of process.

TABLE I
FITNESS GAIN AFTER APPLICATION OF THE CLUSTER APPROACH

Process Type	Initial	Final	Gain
Passive Transparency	0.1476912	0.1966167	33.13%
Equity Movement	0.4651573	0.5134469	10.38%
Holiday Alteration	0.4111576	0.4502492	9.51%
Electoral License	0.2129815	0.2678744	25.77%
Contract Monitoring	0.2114739	0.2262380	6.98%

V. Conclusion

Given the results obtained, the use of clustering techniques (k-means) coupled with process discovery (via the α-Algorithm) showed substantial gains in the generation of new process models. In this sense, less complicated process models can be generated from the considered events, with more adequacy. This is different from the standard approach, where it is only possible to evaluate scenario by scenario, separately.

As there might be a clear distinction of performance between the different categories of processes, it is important to evaluate the process features in order to use the technique best suited to process discovery problems. This is due to the fact that nonconforming flows and divergent trends end up producing insufficient results for this type of approach. Thus, the condition of process log heterogeneity may recommend the use of clustering techniques for effective gains in process model generation.

It is noticeable that the application of the proposed model in all processes tested had positive results. However, depending on the complexity of its configuration, there will be an increase in computational effort. The experiments reveal that there were significant impacts on the solutions even though there was not substantial loss in performance.

Differently from Medeiros et al. [11] and according to Fani Sani et al., this method evidence, which was clustered iteratively such that each of the resulting clusters corresponds to a coherent set of cases, can consider some bias, that in the case of this research are compliance measures, to allow the generation of a more accurate process.

Finally, the application of techniques for feature selection has been evaluated as an opportunity for future work, especially approaches that use artificial intelligence techniques.

References

[1] ABPMP, BPM CBOK VERSION 4.0 - A Guide to Business Process Management - Common Body of Knowledge. 2019.

[2] W. M. P. van der Aalst, ‘Process mining in the large: A tutorial’, Lect. Notes Bus. Inf. Process., vol. 172 LNBIP, pp. 33–76, 2014,https://doi.org/10.1007/978-3-319-05461-2_2.

[3] W. van der Aalst, Process Mining: Data Science in Action, 2nd ed. Berlin Heidelberg: Springer-Verlag, 2016.

[4] P. Markowski and M. R. Przybyłek, ‘Process mining methods for post-delivery validation’, in 2017 Federated Conference on Computer Science and Information Systems (FedCSIS), Sep. 2017, pp. 1199–1202,https://doi.org/10.15439/2017F372.

[5] W. van der Aalst, T. Weijters, and L. Maruster, ‘Workflow Mining: Discovering Process Models from Event Logs’, IEEE Trans. Knowl. Data Eng., vol. 16, no. 9, pp. 1128–1142, Sep. 2004,https://doi.org/10.1109/TKDE.2004.47.

[6] R. P. J. C. Bose, R. S. Mans, and V. D. W. M.P. Aalst, Wanna improve process mining results?: it’s high time we consider data quality issues seriously’, 2013 IEEE Symp. Comput. Intell. Data Min. CIDM13 Singap. April 16-19 2013, pp. 127–134, 2013,https://doi.org/10.1109/CIDM.2013.6597227.

[7] M. Hinkka, T. Lehto, K. Heljanko, and A. Jung, ‘Structural Feature Selection for Event Logs’, ArXiv171002823 Cs Stat, vol. 308, pp. 20–35, 2018,https://doi.org/10.1007/978-3-319-74030-0_2.

[8] A. Rozinat, ‘Process mining: conformance and extension’, 2010,https://doi.org/10.6100/IR690060.

[9] A. Rozinat and W. M. P. van der Aalst, ‘Conformance Testing: Measuring the Fit and Appropriateness of Event Logs and Process Models’, in Business Process Management Workshops, Berlin, Heidelberg, 2006, pp. 163–176,https://doi.org/10.1007/11678564_15.

[10] T. M. Kodinariya and P. R. Makwana, ‘Review on determining number of Cluster in K-Means Clustering’, 2013.

[11] A. K. A. de Medeiros et al., ‘Process Mining Based on Clustering: A Quest for Precision’, in Business Process Management Workshops, Berlin, Heidelberg, 2008, pp. 17–29,https://doi.org/10.1007/978-3-540-78238-4_4.

[12] G. Greco, A. Guzzo, L. Pontieri, and D. Sacca, ‘Discovering expressive process models by clustering log traces’, IEEE Trans. Knowl. Data Eng., vol. 18, no. 8, pp. 1010–1027, Aug. 2006,https://doi.org/10.1109/TKDE.2006.123.

[13] M. Fani Sani, S. J. van Zelst, and W. M. P. van der Aalst, ‘The Impact of Event Log Subset Selection on the Performance of Process Discovery Algorithms’, in New Trends in Databases and Information Systems, Cham, 2019, pp. 391–404,https://doi.org/10.1007/978-3-030-30278-8_39.

[14] H. M. W. Verbeek, J. C. A. M. Buijs, B. F. van Dongen, and W. M. P. van der Aalst, ‘XES, XESame, and ProM 6’, in Information Systems Evolution, Berlin, Heidelberg, 2011, pp. 60–75, https://doi.org/10.1007/978-3-642-17722-4_5.

[15] van Dongen, B.F., ‘Dataset BPI Challenge 2019’. 4TU.Centre for Research Data., 2019, https://doi.org/10.4121/uuid:d06aff4b-79f0-45e6-8ec8-e19730c248f1.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 591–594 DOI: 10.15439/2020F94 ISSN 2300-5963 ACSIS, Vol. 21

Digital assets for project-based studies and data-driven project management

Gloria J. Miller
Managing Consultant
maxmetrics
Heidelberg, Germany
https://orcid.org/0000-0003-2603-0980

Abstract— **Projects offer learning opportunities and digital data that can be analyzed through a multitude of theoretical lenses. They are key vehicles for economic and social action, and they are also a primary source of innovation, research, and organizational change. This research involves a survey of digital assets available through a project; specifically, it identifies sources of data that can be used for practicing data-driven, context-specific project management, or for project-based academic research. It identified four categories of data sources – communications, reports/records, model representations, and computer systems -- and 51 digital assets. The list of digital assets can be inputs in the creation of project artifacts and sources for monitoring and controlling project activities and for sense-making in retrospectives or lessons learned. Moreover, this categorization is useful for decision support and artificial intelligence systems model development that requires real-world data.**

I. Introduction

PROJECTS offer rich environments for conducting research and learning [1, 2] and for practicing data-driven, context-specific project management [3]. They are a key vehicle for economic and social action, and a primary source of innovation, research, and organizational change [4, 5, 6]. They can involve budgets larger than the gross domestic product of a small nation and resources greater than the organizations participating in them [1].

The scale, complexity, uncertainty, and geographic distributions of projects are some of the factors that make projects interesting for analysis through a multitude of theoretical lenses [6]. Projects can be explained and studied using philosophical underpinnings such as the Newtonian understanding of time, space and activity, through the project archetypes such as project-based organizations, project-supported organizations, or project networks, or through the investigation of the changes in project processes or actors [4, 5, 6].

The variety and richness that make projects interesting to study, however, can make them a challenge to efficiently manage. First, there are more than 108 well-known project-specific tools and techniques available to manage project. Besner and Hobbs [7] determined engineering and construction projects are typically large and well-defined for external customers whereas software development projects are relatively small and simple. However, since cost overruns occur in all types of projects, a project that is well-defined is not necessarily efficiently managed [8]. Second, the broad selection of tools demonstrates the multitude of factors managers must consider to plan, monitor, and control projects. Third, although collecting lessons-learned and implementing improvement processes are central concepts in project management standards [9, 10, 11], the learnings rarely happen or do not deliver the intended results [12]. Finally, the administration of projects is moving away from documents to managing task infrastructure through digital information [13].

Researchers have begun to argue that real-time project data should be used in stakeholder engagement [14], performance management [3, 15, 16], monitoring and controlling [17], and policy setting. These approaches support project management, moving from individual human-based decisions to expert decisions to utilizing artificial intelligence. For example, Snider, Gopsill, Jones, Emanuel and Hicks [3] argue that project performance should be evaluated based on an analysis of the data artifacts produced from everyday project activities rather than relying on managerial understanding. Nemati, Todd and Brown [18] explain that project estimation is suitable for an artificial neural network given the numerous potential project configurations. Willems and Vanhoucke [17] found artificial intelligence was used at the front-end of projects but suggested its use has been less investigated during projects. The transitions to these data-driven methods are supported by the growing importance of digital workflows and analytics in project delivery [13].

Thus, even though projects are rich grounds for research and the push towards data-driven project management, the topic of digital data – structured and non-structured – in projects is not sufficiently covered in project management literature. This research involves a literature review to compile a list of digital assets available through a project context. A digital asset classification would be valuable to project researchers and to project managers moving towards data-driven project management. Thus, the study provides a conceptual model for incorporating expert systems and artificial intelligence into the project management process. While there are individual studies that provide some insights into the sources of project management data and the project management standards provide document lists, there is no comprehensive list of project-specific digital assets available in the literature. Furthermore, this study supports the call for new research approaches that investigate the actual or lived experience [2].

The following sections provide a description of the research methodology and a discussion of the results. The final sections of the study present conclusions and implications.

II. Research Methodology

A literature review was performed to identify and classify digital assets in a project context. The "ISO 21500:2012, Guidance on Project Management" [11], *APM Body of Knowledge 6th Edition* [10], and *A Guide to the Project Management Body of Knowledge* (*PMBOK guide*) [9] project management standards were reviewed to identify the project artifacts that could be digital assets. Although criticized by some researchers, the "standards have come to represent an institutionalized collective identity of project managers" [19, p. 37]. Therefore, they offer guidelines for identifying project data sources. From the list of project artifacts, the keywords for the file content, and knowledge areas were compiled into a list.

Journals that focus on project management (i.e., Project Management Journal (PMJ), International Journal of Project Management (IJPM), and IEEE Transactions on Engineering Management (IEEE) [2] and International Journal of Managing Projects in Business (IJMPB)) for the years 2000 – 2020 were selected for the keyword search. The bibliographic data for these journals were downloaded from the Emerald, ScienceDirect, IEEE *Xplore*, or Sage databases into the Endnote reference system. A set of keyword search queries were created for each project management knowledge area. Each query included the selection criteria for any keyword in the data type from the file abstract, any keyword in the file content from the abstract, and any keyword for that knowledge area from the title. An additional query set included any article with digital in the abstract, title, or keyword. The cumulated search queries produced a list of 360 unique articles.

The abstracts for the 360 articles were reviewed to determine if the article described the content or production of a digital project artifact. Based on the abstract review, 97 articles were identified as potentially relevant to the research topic. The full-text review of the 97 articles produced 48 articles that described digital assets in sufficient detail to support the classification. Table I summarizes the number of journal article reviewed for the study.

The coding strategy used to identify and classify digital assets was customized from the classification categories provided by [3] and [20]. In Snider, Gopsill, Jones, Emanuel and Hicks [3], digital assets were classified as digital communication between actors, virtual representations and models of project objects, or textual or numerical documents. That study created decision support monitoring processes based upon the physical attribute (e.g., size or dates), content, or context (e.g., origin, project stage) of the digital asset. Those attributes were not considered in this study.

Quinton and Reynolds [20] specified the dimensions of the data, including data type (attitudinal or behavioral), distances from the data source (primary or secondary), data generation (mythically manufactured or naturally occurring), and data visuality (public or private). They also specified the characteristics of the dataset (big data, open data), the information (encoding format, provider), usage, and ethical challenges. In this study we grouped digital assets using the data dimensions from [20] as well as attributes from our conceptual model.

After collecting and reviewing all articles within the defined scope, we compiled a list of digital assets that met the criteria and developed the classification framework for the articles and for the digital assets. Fig 1. includes the digital assets. For space reasons the digital classification details are excluded from the paper. The digital classification tables are available upon request to the author.

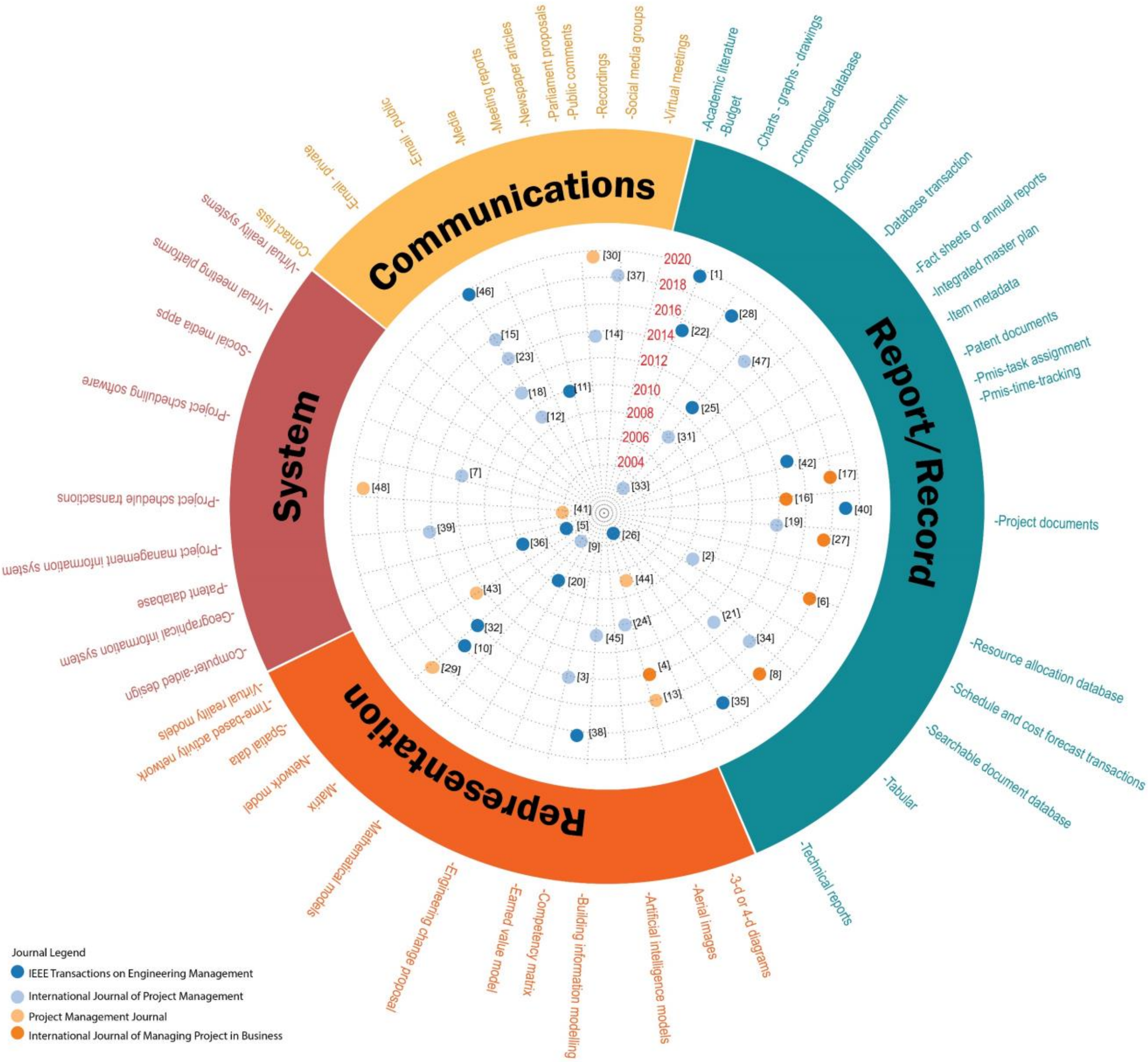
III. Digital Asset classification

Digital assets that were described in the literature were defined and classified. The digital assets were grouped into the communication between actors, virtual representation or models, and records and reports. The research found that the digital assets are embedded in computer systems such as Computer-aided design (CAD), Geographical Information System (GIS), project management information system (PMIS), project scheduling, social media applications, telecommunication or internet meeting platforms, or

Table I.
Journal articles

Journal	Total Articles	Screened Articles	Full-Text Review	with Digital Assets
IEEE	1401	124	34	16
IJMPB	644	54	14	5
IJPM	1918	139	37	19
PMJ	1080	43	12	8
Total	**4414**	**360**	**97**	**48**

Fig. 1 Digital Asset Classification



The figure is an overview of the digital asset classifications with the asset names in the outer circle grouped by category. In the inner circle is a reference number for the journal paper and a color code for the publication. The articles are aligned by the year along the radius and the digital asset along the circumference. The article references are not included in the paper due to space limitation, but the article list is available upon request.

virtual reality technologies. The data that can be extracted as exports, database transactions, or tabular records from such systems are classified in the study. The digital assets were classified using the following characteristics.

The *digital asset* is a descriptive name for the type of data artifact. The description identifies the main purpose of the asset. *Data type* identifies the data as attitudinal or behavioral. Attitudinal data describes what people say, and behavioral data describes what people do. The *data source* identifies the variable as a primary source where raw data can be collected with a specific question in mind (e.g., an email) or a secondary source where the data has already been filtered or interpreted by someone such as the project manager (e.g., a status) or a model. *Visibility* identifies the location and ownership of the data. The options include public-public, private-public, private-private, or open. Public-public data are public data that are accessible from a public location; private-public is private data that are located in a public place usually with access controls; private-private is confidential to a specific individual or group, and open are public data from a public source such as local or government projects. The *encoding* identifies the format of the data. The data can be text, numeric, images, recordings, videos. The *project artifact* can be inputs or outputs of project activities or analytical transformations. Since the papers used a variety of names to describe similar content, names from the project management standards were used. *Reference(s)* (Ref) identifies the articles in which the asset was discussed.

IV. Conclusion

In this study, we compiled a list of digital assets that could be used for project studies or for creating data-driven project management processes. This digital asset classification provides a source for primary and secondary data.

The practical implications are a list of digital assets that can be inputs in the creation of project artifacts and sources for monitoring and controlling project activities and for sense-making in retrospectives or lessons learned. Moreover, this categorization is useful for decision support and artificial intelligence systems model development that requires raw data.

Projects offer a rich environment for where the time and actors are usually fixed at the start of the project. Thus, they are ripe for applying multiple research methods such as action research, case study, and experiments. Digital assets support tracing individual, group, and organizational behaviors. Furthermore, digital data are especially relevant as organizations transition to digital and remote working environments. This categorization offers academic researchers a catalog of data sources and analysis methods for studying complex project phenomena. However, there may be some challenges gaining permission and clearance to utilize the data in the desired method. In addition, ethical use is a concern when dealing with data related to individuals.

This study is limited by the sources used for its investigation. This research was based on a literature review at a single point in time and focused on a small selection of publications. Further research with project and organizational actors is needed to expand on the types of digital assets and further classify the data. An interesting extension would be to add the attributes relevant to each digital asset to the classification model.

References

[1] R. A. Lundin and A. Söderholm, "A theory of temporary organization," *Scandinavian Journal of Management,* vol. 11, no. pp. 437-455, 1995.

[2] N. Drouin, R. Müller, and S. Sankaran, *Novel Approaches to Organizational Project Management Research: Translational and Transformational*. Denmark: Copenhagen Business School Press, 2013.

[3] C. Snider, J. A. Gopsill, S. L. Jones, L. Emanuel, and B. J. Hicks, "Engineering Project Health Management: A Computational Approach for Project Management Support Through Analytics of Digital Engineering Activity," *IEEE Transactions on Engineering Management,* vol. 66, no. 3, pp. 325-336, 2019, https://doi.org/10.1109/TEM.2018.2846400.

[4] A. Jensen, C. Thuesen, and J. Geraldi, "The Projectification of Everything: Projects as a Human Condition," *Project Management Journal,* vol. 47, no. 3, pp. 21-34, Jun 2016, http://dx.doi.org/10.1177/875697281604700303.

[5] R. A. Lundin, "Project Society: Paths and Challenges," *Project Management Journal,* vol. 47, no. 4, pp. 7-15, Aug 2016, http://dx.doi.org/10.1177/875697281604700402.

[6] J. Geraldi and J. Söderlund, "Project studies and engaged scholarship: Directions towards contextualized and reflexive research on projects," *International Journal of Managing Projects in Business,* vol. 9, no. 4, pp. 767-797, 2016, http://dx.doi.org/10.1108/IJMPB-02-2016-0016.

[7] C. Besner and B. Hobbs, "An Empirical Identification of Project Management Toolsets and a Comparison Among Project Types," *Project Management Journal,* vol. 43, no. 5, pp. 24--46, 2012, https://doi.org/10.1002/pmj.21292.

[8] H. K. Doloi, "Understanding stakeholders' perspective of cost estimation in project management," *International Journal of Project Management,* vol. 29, no. 5, pp. 622-636, Jul 2011, https://doi.org/10.1016/j.ijproman.2010.06.001.

[9] PMI, "A Guide to the Project Management Body of Knowledge (PMBOK Guide) --Sixth Edition," Sixth Edition ed. Newtown Square, Pennsylvania, United States: Project Management Institute, Inc., 2017.

[10] APM, "APM Body of Knowledge 6th Edition," ed. Buckinghamshire, United Kingdom: Association for Project Management, 2012.

[11] ISO, "ISO 21500: 2012 Guidance on project management," vol. International Standards Organization, ed. Geneva, Switzerland, 2012.

[12] S. Duffield and S. J. Whitty, "Developing a systemic lessons learned knowledge model for organisational learning through projects," *International Journal of Project Management,* vol. 33, no. 2, pp. 311-324, Feb 2015, https://doi.org/10.1016/j.ijproman.2014.07.004.

[13] J. Whyte, "How Digital Information Transforms Project Delivery Models," *Project Management Journal,* vol. 50, no. 2, pp. 177-194, Apr 2019, http://dx.doi.org/10.1177/8756972818823304.

[14] K. S. K. Chung and L. Crawford, "The Role of Social Networks Theory and Methodology for Project Stakeholder Management," *Procedia - Social and Behavioral Sciences,* vol. 226, no. pp. 372-380, Jul 2016, https://doi.org/10.1016/j.sbspro.2016.06.201.

[15] L. Hossain and A. Wu, "Communications network centrality correlates to organisational coordination," *International Journal of Project Management,* vol. 27, no. 8, pp. 795-811, Nov 2009, https://doi.org/10.1016/j.ijproman.2009.02.003.

[16] M. Takahashi, M. Indulska, and J. Steen, "Collaborative Research Project Networks," *Project Management Journal,* vol. 49, no. 4, pp. 36-52, Aug 2018, http://dx.doi.org/10.1177/8756972818781630.

[17] L. L. Willems and M. Vanhoucke, "Classification of articles and journals on project control and earned value management," *International Journal of Project Management,* vol. 33, no. 7, pp. 1610-1634, Oct 2015, https://doi.org/10.1016/j.ijproman.2015.06.003.

[18] H. R. Nemati, D. W. Todd, and P. D. Brown, "A hybrid intelligent system to facilitate information system project management activities," *Project Management Journal,* vol. 33, no. 3, pp. 42-52, Sep 2002, https://doi.org/10.1177%2F875697280203300306.

[19] P. Eskerod and M. Huemann, "Sustainable development and project stakeholder management: what standards say," *International Journal of Managing Projects in Business,* vol. 6, no. 1, pp. 36-50, 2013, https://doi.org/10.1108/17538371311291017.

[20] S. Quinton and N. Reynolds, *Understanding research in the digital age*: Sage, 2018.

DOI: 10.15439/2020F25
ISSN 2300-5963 ACSIS, Vol. 21

Data Quality Model-based Testing of Information Systems

Janis Bicevskis
Faculty of Computing
University of Latvia, Latvia
Email: Janis.Bicevskis@lu.lv
ORCID: 0000-0001- 5298-9859

Zane Bicevska
DIVI Grupa Ltd, Latvia
Email: Zane.Bicevska@di.lv
ORCID: 0000-0002-5252-7336

Anastasija Nikiforova
Faculty of Computing
University of Latvia, Latvia
Email: Anastasija.Nikiforova@lu.lv
ORCID: 0000-0002- 0532-3488

Ivo Oditis
DIVI Grupa Ltd, Latvia
Email: Ivo.Oditis@di.lv
ORCID: 0000-0003-2354-3780

***Abstract*—This paper proposes a model-based testing approach by offering to use the data quality model (DQ-model) instead of the program's control flow graph as a testing model. The DQ-model contains definitions and conditions for data objects to consider the data object as correct. The study proposes to automatically generate a complete test set (CTS) using a DQ-model that allows all data quality conditions to be tested, resulting in a full coverage of DQ-model. In addition, the possibility to check the conformity of the data to be entered and already stored in the database is ensured. The proposed alternative approach changes the testing process: (1) CTS can be generated prior to software development; (2) CTS contains not only input data, but also database content required for complete testing of the system; (3) CTS generation from DQ-model provides values against which the system can be further tested. If the test results correspond to the values obtained during CTS generation, the system under test shall be considered to have been tested according to DQ-model. Otherwise, the user can verify the cause of the differences that may occur due incorrect software, as well as an inaccurate specification.**

***Index Terms*—complete test set, data quality model, information system, model-based testing, symbolic execution.**

I. Introduction

SOFTWARE testing attracts the attention of researchers and practitioners since software development starts. Their main aim is to develop reliable software that can be used in the real-life circumstances. Unfortunately, this challenge has not yet been resolved and is far from being resolved. The proposed testing strategies and techniques are not able to ensure the reliability of software. Errors and bugs still cause system failures, despite millennial resources devoted to testing. According to Utting [1], software testing is a vital part of software development that requires between 30 and 60 percent of spent resources.

Model-based testing (MBT) is one of widely used solutions to improve the quality of the software. In scope of MBT, a model of information system (IS) is created according to which the system is tested. If IS works correctly on tests that cover all elements of the model, it is assumed that full/ complete system testing has been performed according to the selected model. For instance, if a program control graph is used as a test model, full/ complete testing is considered to have been performed if all the paths of the graph are executed. The advantages of MBT are also reflected in model-based testing user survey [2], according to which, respondents report on the average a 59% reduction in escaped bugs, 17% reduction in testing costs, and 25% reduction in testing duration.

The aim of this study is to propose an alternative model-based testing approach that uses a data quality model as a test model. As a result, a data quality (DQ) model-based testing approach called DQMBT is proposed. The DQ-model contains data objects and data quality conditions concepts where a data object describes real-world objects on which the information system accumulates data, while data quality conditions are aimed to describe the requirements that must meet the values of the attributes of data objects to be recognised as qualitative.

This paper is a continuation of [3], which addressed the basic concepts and introduced the overall structure of the proposed solution. According to [3], the main idea of the solution is as follows: as one of the main and primary tasks of the information systems is to collect and process data objects, the data to be entered must be tested first by verifying their correctness described by the conditions of the values of the data objects. The correct data objects can be stored in the database, while the information about the incorrect data objects must be provided to the data owner, allowing them to be edited and re-entered to the system. The verification of data objects must be carried out at two levels – syntactic and semantic/ contextual (in line with [4]). While syntactic control checks the relevance of the values of data objects attributes to the value syntax, semantic control checks the relevance of attribute values to the values of other data objects that have been already entered and stored in the database. The first use of the proposed solution is to compare the relevance of the data objects to be entered to the data objects already stored in the database, i.e. whether the data entered are correctly retained in the database. These checks must be described in the DQ-model and are not

The research leading to these results has received funding from the research project "Competence Centre of Information and Communication Technologies" of EU Structural funds, contract No. 1.2.1.1/18/A/003 signed between IT Competence Centre and Central Finance and Contracting Agency, Research No. 1.7 “The use of business process models for full functional testing of information systems"

related to implementation in the particular environment. Obviously, information systems are intended not only for collecting data but also for processing them, including, for calculating derived values, transformations etc. However, the primary task is to collect data, followed by many different tasks, so, this solution covers only one but nevertheless one of the main tasks of the information systems (in line with [5]). The second use of the proposed solution is to provide the complete testing capability of software that accumulates and stores data in the database. The values conditions for the attributes of data objects are proposed to be used to prepare test cases that will process all correct and incorrect cases. Using the DQ-model as a test model allows to prepare test cases constructively for the verification of all conditions. Testing software with these test cases, will check the accuracy of entering and storing data in both syntactic and contextual terms. The study therefore proposes a new complete testing criterion - verifying the correctness of all input data and its allocation in the database with tests that check all possible input values conditions.

This paper proposes not only the next set comprehensive set of concepts that are used to achieve the objective of the study being launched, but also provides an example demonstrating this idea, which has been promised in [3]. To sum up, the DQ-model based testing (DQMBT) approach for IS testing is proposed.

The paper deals with following issues: basic concepts and ideas addressed through related works (Section 2), the proposed solution (Section 3), analysis of the proposed solution (Section 4), conclusions (Section 5).

II. Related Works

This section briefly deals with the key concepts underpinning the proposed solution that are addressed through related works.

A. Testing basics

In software engineering, a test case is a specification of the inputs, execution conditions, testing procedure, and expected results that define a single test to be executed to achieve a particular software testing objective, such as to exercise a particular program path or to verify compliance with a specific requirement [6].

The modern definitions of testing underline that testing is a process aimed at verifying software compliance to requirements. An example of this is the definition provided by [7] in 2018, according to which, "*software testing is a way to assess the quality of the software and to reduce the risk of software failure in operation. Software that does not work correctly can lead to many problems, including loss of money, time, or business reputation, and even injury or death*".

Many authors propose different and sometimes conflicting definitions of the concept of testing, in which, in some cases, the meaning of finding error and bug is exaggerated. As part of this study, the term "testing" should be understood in accordance with [8]: "*software testing is an investigation conducted to provide stakeholders with information about the quality of the software product or service under test*".

The viewpoint that testing aims to find bugs, errors and defects in software is outdated and no longer considered comprehensive and completely correct (in line with [7]). Methods that can find software bugs cannot be used to demonstrate that the software is working properly. In addition, despite numerous resources spent on testing, software almost always has bugs and errors.

To sum up, testing is a complex process, since tests are developed and accumulated throughout the whole software development process, starting with the development of a test for each individual function, ending with integration tests aimed at verifying the compatibility and integration of all components of the system.

B. Complete Test Set

Model-based testing opens up new horizons for software testing as it allows the creation of a test set for the selected and previously developed testing model that checks all the requirements for this model. Thus, the test model supports producing tests that fully cover aspects of the selected model. For instance, if a program control graph is used as a model, tests that execute all the graph arcs, are prepared. Such test set is called the complete test set (CTS). If the programme on this test set is working correctly, it shall be assumed that it has been sufficiently tested. Unfortunately, such a test criterion does not ensure the correct operation of the programme in all cases, but it is widely used as it allows for significant improvements in the overall quality of the software.

It is not a secret that theoretical studies on the possibility of automatically generating a complete test set according to a certain program code were carried out even in the 70s, when the first results on automatic generation of CTS were published in a cycle of articles on testing theory, including [9], followed by practical implementation [10]. During these studies, it was found that in cases where programming features are limited to processing a series of files, there is an algorithm capable of creating a complete test system for each such program. Thus, it can be assumed that for simple programmes that do not use complex language structures should also be possible to generate CTS.

This could complement unit testing with the possibility of testing with automatically generated test sets. Further studies demonstrate that if the program allows two two-way counters, the problem of constructing CTS is algorithmically unsolvable. This means that, depending on the programming languages, the impossibility of automatic generation of CTS soon occurs. In addition, despite the impressive age of this topic, it is still popular and widely used, as demonstrated by literature analysis [11]-[15]. In this study, the CTS corresponding to program control graph will be replaced by a DQ-complete test set system corresponding to the DQ-model addressed in the next Section.

C. Test Set Generation

According to the above section, if a program control graph coverage is used as a test model, testing according to this model means checking all possible control flow branching testing. As an example, this solution is provided by the Visual Studio 2015 Enterprise IntelliTest tool.

IntelliTest allows generation of unit test set for a particular class method or for all class methods simultaneously. For each condition, a test will be generated in the program code that will meet the specific condition when the method is operated. This tool analyses each condition branch in the C# code. The *if* branching conditions, statements and all operations that may constitute exceptions are analysed. As a result of the analysis, IntelliTest is designed to achieve the highest code coverage. In the generated test set, tests which have been performed, entry data and error message can be seen. The user may save them for further regression testing. The tool works with programs written in C# programming language. IntelliTest is based on a symbolic execution of a program that operates with symbolic notion of variable values instead of traditional command execution. This allows to establish the realizable conditions for paths, which, when resolved, result in input data for the execution of the corresponding path.

The concept of symbolic execution was introduced by Goodenough and Gerhart in 1975 [16], however, despite this, symbolic execution of programmes and specifications remain popular and become even more popular in recent years (see [17] – [21]). This study is not an exception, and symbolic execution is at the core of the proposed idea.

D. Data Quality Model

The study uses the previously proposed data object-driven data quality model (DQ-model) [4], consisting of 3 key components: (1) a data object defining the data to be analysed, (2) a specification of data quality, which defines the conditions to be met for the recognition of data as qualitative, and (3) a quality assessment process that determines the procedure that must be followed to assess the quality of data.

Each DQ-model component is represented by flowchart-like diagrams defined in a graphical domain specific language (DSL).

As in [22] the proposed solution will be demonstrated by some concrete almost classic example of university, more precisely, student and his achievements. Fig. 1 demonstrates the definition of the data object. Three data objects are defined: (1) *Students*, (2) its sub object *Course*, and (3) *inputMessage*, which contain data on specific courses passed, including course code, assessment, and date to be entered in the corresponding student list of grades (sub object *Course*) and stored in the database (data objects *Students* and *Courses*). Dashed lines represent contextual dependencies between the *inputMessage* attribute *studName* and *Students* instances.

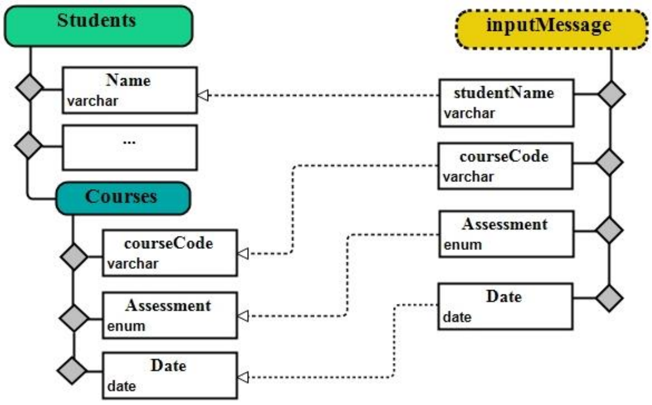


Fig.1 Data objects definition

Similarly, there is a contextual dependency between *inputMessage* attributes and *Courses* stored values. These dependencies are precisely defined in the quality conditions shown in Fig. 2, containing 4 checks:

- the *Students* data object has an instance, where ***Students.Name=inputMessage.studentName;***
- a new instance has been added to the *Students* sub-object *Courses*, where ***Courses.courseCode* = *inputMessage.courseCode***;
- a new instance with the corresponding course assessment has been added to the *Students* sub-object *Courses* data item, where ***Courses.Assessment* = *inputMessage.Assessment;***
- a new instance with the corresponding exam date has been added to the *Students* data object *Courses* sub-object, where ***Courses.Date* = *inputMessage.Date***.

Thus, a DQ-model used to generate tests is obtained from Fig. 1 and 2.

As stated in [4] and [23], the data object is defined according to the data to be analysed, so that parameters that are not relevant to specific users and use-cases are ignored (further denoted with "---" symbols). Data objects of the same structure form data object class. Similarly, the data quality specification shall also be determined by the user/ tester, depending on the use-case. The data quality specification can be defined informally or formally, but at the last stage all requirements are replaced by executable artefacts, such as SQL statements or program code, that further are executed.

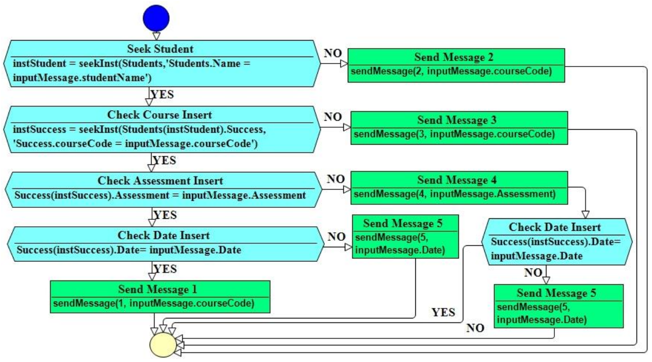

Seek Student
instStudent = seekInst(Students,'Students.Name = inputMessage.studentName')
NO
Send Message 2
sendMessage(2, inputMessage.courseCode)
YES
Check Course Insert
instSuccess = seekInst(Students(instStudent).Success, 'Success.courseCode = inputMessage.courseCode')
NO
Send Message 3
sendMessage(3, inputMessage.courseCode)
YES
Check Assessment Insert
Success(instSuccess).Assessment = inputMessage.Assessment
NO
Send Message 4
sendMessage(4, inputMessage.Assessment)
YES
Check Date Insert
Success(instSuccess).Date= inputMessage.Date
NO
Send Message 5
sendMessage(5, inputMessage.Date)
Check Date Insert
Success(instSuccess).Date= inputMessage.Date
NO
Send Message 5
sendMessage(5, inputMessage.Date)
YES
Send Message 1
sendMessage(1, inputMessage.courseCode)
YES
NO

Fig.2 Requirements definition

The DQ-model is therefore executable. As proposed in [22] the DQ-model uses following methods to identify context of data objects:

- reviewing all class instances by changing the address ***<dataObjectName(instID).attributeName>***, calculated first by selecting the first instance using the ***instID = getFirst(dataObjectName)*** method, followed by the transition to the next instance using the method ***<instID = getNext(dataObjectName)>***. This option shall be used if the quality of the data is to be analysed for all instances of a particular data object;
- using a dynamically calculated address ***<instID = seekInst(dataObject, expression)>***, where an *expression* is a logical expression where operands are attribute names. If an instance of a data object is found as a result of an execution, (1) a reference to the data object is inserted into the variable *instID*, (2) the value *TRUE* is returned to the environment; otherwise, a *NULL* value is inserted in the variable that returns *FALSE*. This option is used if the quality is to be analysed for only one instance of a data object.

The effectiveness of the proposed approach has already been demonstrated by applying it to real data sets and presenting result in the series of articles.

III. The Proposed Solution

This Section demonstrates how the proposed DQ-model can be used as a test model.

A. Data Quality Model as Testing Model

According to MBT principles, the test model is first selected. It serves to generate a set of tests that will test the correctness of the tested program or the system under test (SUT). The test set can be created either manually, partially or fully automatically. If the SUT on this test set works according to the specification, the SUT is considered to have been tested according to the selected model. As for criterion when SUT can be considered to tested sufficiently, a DQ-model coverage of all data quality requirements is selected. Although the proposed solution complies with the principles of the MBT, some important differences have to be mentioned: the proposed solution carries out a verification of the syntactical and contextual/ semantical control of input data and their correct allocation in data objects of the database. As it was mentioned in [3], it covers one of the most important tasks of the information systems, which is followed by other tasks such as calculations, reporting etc.

The proposed DQ-model-based test scheme or general architecture of DQMBT is shown in Fig. 3. The main actions are carried out by a "*Test generator*" using DQ-model to generate test input data, data object content (database) and two protocols – "*Input data test protocol (expected)*" and "*Database content (expected)*". The SUT is executed with generated test input data after the database content generated by the "*Test generator*" has been entered in the database.

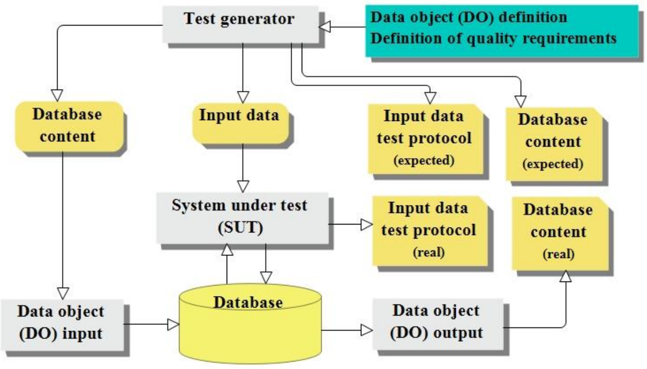

Test generator
Data object (DO) definition
Definition of quality requirements
Database content
Input data
Input data test protocol (expected)
Database content (expected)
System under test (SUT)
Input data test protocol (real)
Database content (real)
Data object (DO) input
Database
Data object (DO) output

Fig. 3 Software verification procedure

The results of the SUT execution are recorded in the "*Input data test protocol (real)*" and the content of the data objects (database) are read after testing the SUT with generated test input data. The "*Input data test protocol (real)*" must coincide with the "*Input data test protocol (expected)*" generated by the "*Test generator*", although there are possible differences in formatting and texts. If these two protocols in general coincide with each other, it is assumed that the SUT is operating in accordance with the DQ-model, otherwise both protocols are sent to IS developers for further investigation of reasons of differences. Differences in protocols may indicate errors in the SUT or differences in the DQ-model from programmers' programs.

The proposed testing ensures complete testing according to the DQ-model, since all quality conditions are tested with generated test inputs and data object content, reaching their full coverage in both fulfilling and rejecting the conditions. In other words, complete/ full testing is performed according to the DQ-model. In addition, a specific test criterion is proposed, more precisely whether data to be entered is correctly allocated in data objects (database) without contradicting the data previously stored. It is clear, the proposed criterion does not guarantee the detection of all errors in the operation of SUT. For instance, SUT operation that record data in non-compliant locations in the database are not controlled, moreover, database integrity may be broken down. These types of errors cannot be detected even in the case of well-developed testing support tools.

The proposed approach is consistent with "black box" testing model because information on the internal design or implementation of the system is not used. Only the DQ-model is used to generate tests. However, it should be acknowledged, that the SUT may contain activities that are not covered by the DQ-model and the tests generated therefore cover the operation of SUT only partially (this is a common challenge for MBT).

This means that either traditional testing methods should be used, or the testing model should be enriched with new features.

The next section addresses the test generation algorithm.

B. Algorithm of Test Generation

In the proposed algorithm, the first phase requires the deployment of a requirements/ condition model (Fig.2) into a tree-like chart given in Fig. 4. The last branch vertices contain numbers, which in the scope of the provided example range from 1 to 6. The tree contains only one node with a number 1 that represents correct data processing from syntactic and semantic/ context checks to correct data allocation and storage in the database. The branch with number 2 represents a violation of the input data context since database does not have data for the specific student. Branches with numbers from 3 to 6 indicate incorrect data allocation in database.

In the second phase of the algorithm, the conditions for the realisation of the corresponding branches are established using the symbolic execution of the DQ-model conditions. For instance, the conditions for the first branch under this example are:

- **exist Students(instStudent)** where ***inputMessage.studName=Students(instStudent).name***
- **exist Course(instCourse)** where ***inputMessage.courseCode=Course(instCourse).name***
- **valid Assessment** where ***inputMessage.Assessment = Course(instCourse).Assessment***
- **valid Date** where ***inputMessage.Date = Course(instCourse).Date.***

When resolving the conditions for the branch realisation in all 6 cases, the test input data is obtained shown in Table I and the content of the data objects (database) in Table II and III. Each instance of a data objects serves as test input data to complete one of the 6 branches. The 6 rows of the Table I correspond to 6 branches that when executed fulfil all the data quality conditions transitions/ paths of the DQ-model.

In other words, the generated test set is a full/ complete DQ-test set. Execution of the SUT with all 6 tests will achieve the complete testing of the SUT according to the DQ-model criterion.

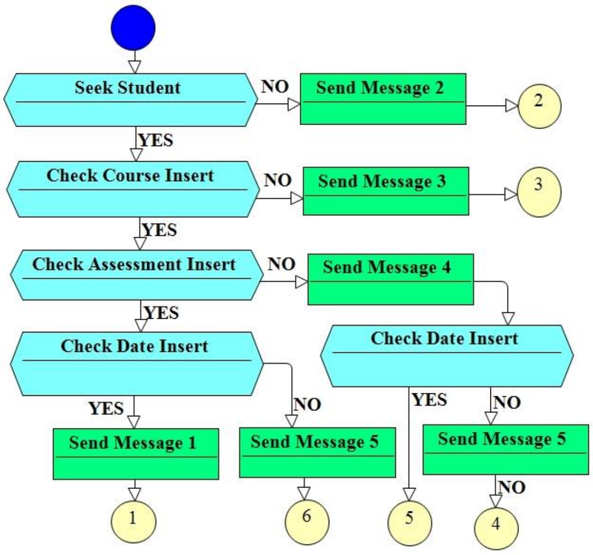


Fig.4 Requirements tree

TABLE I.
DATA OBJECT INPUTMESSAGE

studName	courseCode	Assessment	Date
stud-1	course-1	assess-1	date-1
stud-2	---	---	---
stud-3	course-3	---	---
stud-4	course-4	assess-4	date-4
stud-5	course-5	assess-5	date-5
stud-6	course-6	assess-6	date-6

TABLE III.
DATA OBJECT STUDENTS

Name	Address
stud-1	---
stud-3	---
stud-4	---
stud-5	---
stud-6	---

TABLE IIIII.
DATA OBJECT COURSES

courseName	Assessment	Date
course-1	assess-1	date-1
course-4	assess-4	date-4
course-5	assess-5	date-5
course-6	assess-6	date-6

The test theory generally understands the concept of "test" as the analysis of the value of the data to be entered and the expected results of the SUT execution obtained by the SUT with the entered data. It is known that the result of execution depends not only on the data to be entered but also on the content of the related data objects (database). Thus, the proposed approach generates not only the data to be entered but also the database content that ensures the execution/ completion of the chosen path. This can be achieved by symbolically executing the contextual conditions/ requirements between the interrelated data object of the DQ-model (as shown in Fig. 2).

The next stage of the algorithm supposes the execution of the conditions with the DQ-complete test set (input entered and the generated content of the data objects) that results in obtaining the expected test results, so called benchmark.

When testing the SUT with the previously generated DQ-complete test set, the results of execution must [by its nature] coincide with the results of execution of the DQ-model or benchmarks. Thus, it can be argued that the test objective has been achieved since the tested programme is tested with input data that ensures verification of all data quality conditions, as well as checking the compliance of input data with their retention on the database.

C. Testing Process

At the next stage, the SUT is tested with automatically generated tests. The values of generated data objects are sent to the database that is done by separate procedure (individual for a particular system). This is followed by SUT testing with the DQ tests given in Table I – "*inputMessage*".

TABLE IV.
PROTOCOL

branch	message#	text
1	1	**Message-1**: input successful: *<stud-1, course-1, assess-1, date-1>*
2	2	**Message-2**: **input error: invalid StudName** *<stud-2, course-2, assess-2, date-2>*
3	3	**Message-3**: **database error: invalid courseCode** *<stud-3, course-3, assess-3, date-3>*
4	4, 5	**Message-4**: **database error: invalid Assessment** *<stud-5, course-5, assess-5, date-5>* **Message-5**: **database error: invalid Date** *<stud-5, course-5, assess -5, date-5>*
5	4	**Message-4**: **database error: invalid Assessment** *<stud-5, course-5, assess-5, date-5>*
6	5	**Message-5**: **database error: invalid Date** *<stud-5, course-5, assess-5, date-5>*

The results of the SUT execution must be consistent with the previously obtained protocols. In the case of differences, the inconsistencies between the operation of the SUT and the DQ-model are identified (Table IV).

This can be caused by both errors in the SUT or errors in the specification – the DQ-model. To automate the testing process, most test support tools support the SUT execution with a user-selected set of tests [24] – [27]. These tests usually are accumulated gradually using the same test support tools. As a result, in most cases the tests records formats are internal formats of these tools which are not related to the tested programs. In the proposed case, the situation is more complicated because all generated tests must be able to be performed automatically in one session. This can be achieved by preparing test drivers that establish database content, calls the cyclic execution of the SUT with all generated tests, and read the database content after completion of the tests (see Fig. 5).

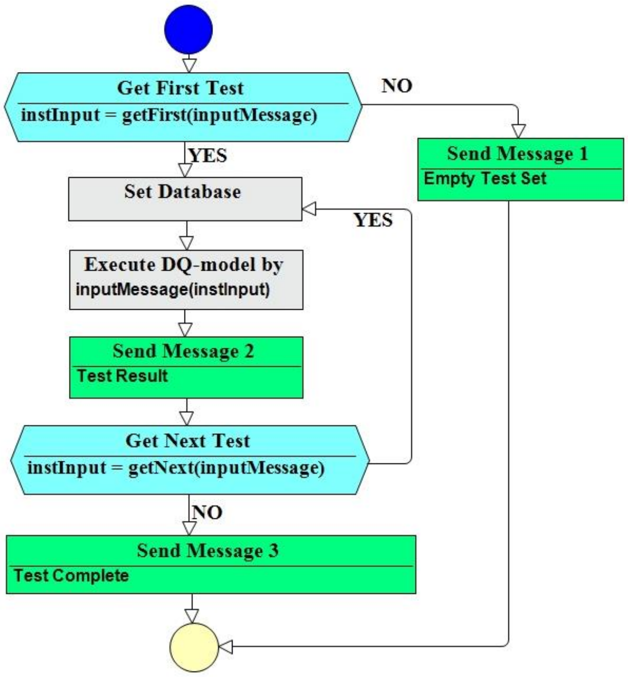


Fig.5. Testing process

IV. ANALYSIS OF THE PROPOSED SOLUTION

The traditional and common approach to software testing is to define and plan test cases prior to their execution and then compare their results with documented expected results [28]. The proposed approach differs from such test process when the tester prepares test cases based on an informal specification, his own experience or intuition without an exact and precise specification of the operation of the SUT.

The proposed approach uses a formal and at the same time executable specification, generates the DQ-complete test set and the expected results of its execution or benchmarks. After automated testing of the SUT, the tester should only compare the results obtained with the expected benchmarks.

The tester does not have to prepare the test by himself and perform the execution of the SUT with them. The quality of testing therefore does not depend on the qualification of the tester, but on the quality of the DQ-model in the way of its accuracy and completeness, i.e. whether the testing model meets the requirements of the system.

From the beginning of testing, it is well known that systematic testing reveals most errors, and after each iteration, the number of errors at the users' end is lower. Even at the end of the 20th century, it was already known that, when tests are selected intuitively, the end user receive 8-10 times more errors compared to when tests are selected based on the formalized model [29].

This time, systematic testing is understood as testing according to the MBT principles. The main advantages of using a complete test set (CTS) in the testing process are:

- a complete test set (CTS) can be generated prior to the development of the programme, thus, it can serve as an additional interpretative example of the specification;
- SUT testing may be initiated immediately after the development of the programme;
- the CTS shall completely verify the syntactic and semantic/ context requirements specified in the specification;
- the tester shall be released from the development of tests and their execution.

However, together with the advantages, some of the challenges and limitations of the proposed solution should also be mentioned:

- the proposed solution supposes the development of a DQ-model that requires resources and specific expertise and knowledge in the development of DQ-models although their development is not too complicated, especially for people with IT background;
- additional tools such as test generator, database content input, output of results, CTS execution driver, ensuring cyclic execution of all CTS tests without the involvement of the tester, are required to support testing;
- the SUT must be prepared for its automated testing with CTS.

V. Conclusion

The solutions proposed by the test theory are only partly capable of meeting the practice needs, since the proposed testing strategies and techniques do not guarantee the development of qualitative programmes. The previous testing paradigm as a search for errors and bugs is now switching to the new one, according to which, testing is tasked with achieving reliable software. This can be achieved through systematic testing, for instance, using model-based testing.

The study therefore proposes an alternative, model-based testing approach called DQMBT, based on a data quality model proposed in previous studies. It defines the data objects and conditions that must meet the parameter values of the data objects to consider the data object to be correct and qualitative. The proposed test algorithm provides such two main features as:

- the generation of a DQ-complete test set to check the correctness of the operation of the programs to be tested, covering all possible quality conditions for the input data;
- the comparison of the relevance of the data objects to be entered and stored in the database to one another, verifying whether the data entered is correctly stored in the database.

The program testing with an automatically generated complete test set (CTS) changes the testing process significantly as the preparation and execution of individual test cases is replaced by complete testing that systematically checks all conditions in a single session.

Using a data quality model as a test model does not solve all program testing problems. The proposed approach covers only a part, however, a very important part of the functional testing of the information systems, more precisely complete testing of the input data and testing of the relevance of data stored in the database with input data. This would lead to a significant improvement in the overall quality of information systems, which today is one of the most important challenges we need to solve [30].

In addition to an in-depth study on the concepts, which are not thoroughly covered in this paper mentioned in Section 4, further studies on the topic include the application of the proposed approach to the real system we are dealing with. This will not only allow a test of the proposed approach, but also lead to a quantitative and qualitative results that could be compared with other strategies currently in use. Then, the question on how to handle system events, when one event takes place more quickly than others, but affects the result of previous events, will be addressed.

References

[1] M. Utting, B. Legeard. Practical model-based testing: a tools approach. *Elsevier*, 2010.

[2] R.V. Binder. 2011 Model-based Testing User Survey: Results and Analysis. *System Verification Associates. System Verification Associates*, 2012.

[3] A. Nikiforova, J. Bicevskis. Towards a Business Process Model-based Testing of Information Systems Functionality. In *Proceedings of the 22nd International Conference on Enterprise Information Systems* - Volume 2: *ICEIS*, ISBN 978-989-758-423-7, pp. 322-329, 2020. DOI: 10.5220/0009459703220329.

[4] A. Nikiforova, J. Bicevskis, Z. Bicevska, I. Oditis. User-Oriented Approach to Data Quality Evaluation. *Journal of Universal Computer Science, 26(1)*, pp.107-126, 2020.

[5] R. Perez-Castillo, A.G. Carretero, M. Rodriguez, I. Caballero, M. Piattini, A. Mate et al. Data quality best practices in IoT environments. In *2018 11th International Conference on the Quality of Information and Communications Technology (QUATIC)*, pp. 272-275. IEEE, 2018, DOI: 10.1109/QUATIC.2018.00048.

[6] 24765-2010 - ISO/IEC/IEEE International Standard - Systems and software engineering – Vocabulary, doi:10.1109/IEEESTD.2010.5733835. ISBN 978-0-7381-6205-8.

[7] K. Olsen, T. Parveen, R. Black, D. Friedenberg, E. Zakaria, M. Hamburg, J. McKay, M. Walsh, M. Posthuma, M. Smith, R. Smilgin, S. Ulrich, S. Toms. Certified tester foundation level syllabus. *Journal of International Software Testing Qualifications Board*, 2018.

[8] P. Saini. Revisiting Mutation Testing in Cloud Environment (Prospects and Problems), *A Journal of Composition Theory*, Volume 12, Issue 9, pp. 2007-2011, 2019, DOI:19.18001.AJCT.2019.V12I9.19.10519.

[9] J. Bārzdiņš, J. Bičevskis, A. Kalniņš. Automatic construction of complete sample systems for correctness testing. *Math. Found. of Computer Science. Springer Verlag, Berlin*, 1975.

[10] J. Bičevskis, J. Borzovs, U. Straujums, A. Zariņš, E.F.jr. Miller. SMOTL - A System to Construct Samples for Data Processing Program Debugging. *IEEE Transactions on Software Engineering*, Vol. SE-5, No.1, pp. 60-66, 1979.

[11] Y. Y. Lin, N. Tzevelekos. Symbolic Execution Game Semantics. *arXiv preprint arXiv:2002.09115*, 2020.

[12] G. P. Farina, S. Chong, M. Gaboardi, M. Relational symbolic execution. In *Proceedings of the 21st International Symposium on Principles and Practice of Programming Languages 2019*, pp. 1-14, ACM, https://doi.org/10.1145/3354166.3354175.

[13] M. Aggarwal, S. Sabharwal. Combinatorial Test Set Prioritization Using Data Flow Techniques. *Arabian Journal for Science and Engineering, 43(2),* pp. 483-497, 2018, https://doi.org/10.1007/s13369-017-2631-y.

[14] M. Handique, J. K. Deka, S. Biswas, K. Dutta. Minimal test set generation for input stuck-at and bridging faults in reversible circuits. In *TENCON 2017 IEEE Region 10 Conference*, pp. 234-239, DOI: 10.1109/TENCON.2017.8227868.

[15] G. Eleftherakis, P. Kefalas, E. Kehris. A methodology for developing component-based agent systems focusing on component quality. In *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 561-568. IEEE, 2011.

[16] J. Goodenough, S. Gerhart, S. Toward a Theory of Test Data Selection. *IEEE Transactions on Software Engineering*, Vol. 1 (2), pp. 156-173, 1975.

[17] A. Ibing. Efficient data-race detection with dynamic symbolic execution. In *2016 Federated Conference on Computer Science and Information Systems (FedCSIS),* pp. 1719-1726. IEEE, 2016, DOI: 10.15439/2016F117.

[18] J. Bicevskis, G. Karnitis. Testing of Execution of Concurrent Processes. *Proceedings of DB&IS'2020* (to be published).

[19] R. Baldoni, E. Coppa, D. D'elia, C. Demetrescu, I. Finocchi. A survey of symbolic execution techniques. *ACM Computing Surveys (CSUR)*, 51(3), pp. 1-39, 2018, https://doi.org/10.1145/3182657.

[20] D. Trabish, A. Mattavelli, N. Rinetzky, C. Cadar. Chopped symbolic execution. *In Proceedings of the 40th International Conference on Software Engineering*, pp. 350-360, 2018, https://doi.org/10.1145/3180155.3180251.

[21] R. Stoenescu, M. Popovici, L. Negreanu, C. Raiciu. Symnet: Scalable symbolic execution for modern networks. In *Proceedings of the 2016 ACM SIGCOMM Conference*, pp. 314-327, 2016, https://doi.org/10.1145/2934872.2934881.

[22] J. Bicevskis, A. Nikiforova, Z. Bicevska, I. Oditis, G. Karnitis. A Step Towards a Data Quality Theory. In *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pp. 303-308. IEEE, 2019, 10.1109/SNAMS.2019.8931867.

[23] J. Bicevskis, Z. Bicevska, A. Nikiforova, I. Oditis. Towards Data Quality Runtime Verification. In *2019 Federated Conference on Computer Science and Information Systems (FedCSIS),* pp. 639-643. IEEE, 2019, DOI: 10.15439/2019F168.

[24] V. Garousi, F. Elberzhager. Test automation: not just for test execution. *IEEE Software,* 34(2), pp. 90-96, 2017, DOI:10.1109/MS.2017.34.

[25] D. M. Rafi, K. R. K. Moses, K. Petersen, M. V. Mäntylä. Benefits and limitations of automated software testing: Systematic literature review and practitioner survey. In *2012 7th International Workshop on Automation of Software Test (AST),* pp. 36-42. IEEE, 2012.

[26] P. Loyola, M. Staats, I.Y. Ko, G. Rothermel. Dodona: automated oracle data set selection. In *Proceedings of the 2014 International Symposium on Software Testing and Analysis,* pp. 193-203, 2014, http://dx.doi.org/10.1145/2610384.2610408.

[27] H. Kaur, G. Gupta. Comparative study of automated testing tools: selenium, quick test professional and testcomplete. *Int. Journal of Engineering Research and Applications,* 3(5), pp. 1739-1743, 2013.

[28] W. Afzal, A. N. Ghazi, J. Itkonen, R. Torkar, A. Andrews, K. Bhatti, K. An experiment on the effectiveness and efficiency of exploratory testing. *Empirical Software Engineering,* 20(3), pp. 844-878, 2015, https://doi.org/10.1007/s10664-014-9301-4.

[29] J. Bicevskis. The Effectiveness of Testing Models. In Proc. of 3rd Intern. *Baltic Workshop "Databases and Information Systems",* 1998.

[30] E. Ziemba, T. Papaj, D. Descours, Assessing the quality of e-government portals-the Polish experience. In *2014 Federated Conference on Computer Science and Information Systems,* IEEE, 2014, pp. 1259-1267, http://dx.doi.org/10.15439/2014F121.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 603–606 DOI: 10.15439/2020F98 ISSN 2300-5963 ACSIS, Vol. 21

A Simulation Study on the Impact of Activity Crashing on the Project Duration and Cost under Different Budget Release Scenarios

Jie Song*, Tom Servranckx*, Annelies Martens* and Mario Vanhoucke*†‡
*Faculty of Economics and Business Administration
Ghent University, Tweekerkenstraat 2, 9000 Gent, Belgium
Email: jieson.song@ugent.be
†Technology and Operations Management Area, Vlerick Business School, Reep 1, 9000 Gent, Belgium
‡UCL School of Management, University College London, 1 Canada Square, London E14 5AA, UK

***Abstract*—The main goal of project control is to identify project opportunities or problems during project execution, such that corrective actions can be taken to bring the project in danger back on track when necessary. In this study, we define different scenarios to allocate the limited budget used for the cost of activity execution, delays, and corrective actions, according to the timing and amount of the budget release. A large computational experiment is conducted on real-life project data to evaluate the performance of each scenario. The results show that both the timing and amount of the budget release have an effect on project performance.**

I. Introduction

PROJECT control is a key part of project management (PM), together with baseline scheduling and schedule risk analysis. Where baseline scheduling focuses on the construction of a timetable for the activities considering the technological and resource constraints in the project, risk analysis identifies high risk activities in the project. Both aspects belong to the static PM phase, i.e. prior to project execution, and are supported by state-of-the-art optimisation and simulation techniques to create optimal and robust project schedules. In contrast, project control is the process of monitoring the project during execution to detect potential problems and taking corrective actions when necessary, and belongs to the dynamic PM phase [1]. Since project success can only be achieved when the static PM phase is combined with an effective project control process, advanced modelling and simulation techniques are needed to support the project control process.

Earned Value Management (EVM) is a project control method to measure the project performance in terms of time and cost [2]. Using EVM, the project progress can be periodically measured and compared to a control limit [3]. When the progress is below this limit, the project is expected to exceed its deadline and a warning signal is generated to initiate corrective action. Since generating efficient and reliable warning signals is important to take effective corrective actions, this research topic has been investigated intensively in recent years [4], [5], [6]. Other recent research studies focused on the corrective action taking process [7], [8], [9]. Since many of these research efforts do not consider the fact that the project budget is limited in most real-life projects, [10] have investigated the performance of four different approaches to allocate a limited budget dedicated to corrective actions over different project phases. The authors developed an extensive simulation experiment to evaluate the four approaches using a large set of artificial projects and showed that the best allocation model considers the planned progression of work in the project and provides a control budget that increases in later project stages.

In this study, we simulate the impact of time-cost trade-offs given a limited project budget on the time and cost performance of a set of real-life projects. The timing and quantity of the release of the budget throughout the project life cycle is modelled using different scenarios: immediate or time-phased budget releases, proportional to the time or cost profile of the project and dynamically increasing or decreasing as the project progresses. We only consider activity crashing as a potential corrective action, which implies investing more budget in an activity to reduce the activity duration (i.e. time-cost trade-offs). This problem is related to the Project Scheduling Game (PSG, [11]), a project control game in which the project execution of a relatively complex project is simulated. The objective is to minimise the final project cost by controlling the project at six decision moments using activity crashing. While an unlimited budget is available in the PSG, we evaluate different strategies for using a limited budget during project execution to complete a project within this budget. Further, we extend the existing research by [10] in three ways. First, we do not consider a control budget that can only be used for activity crashing, but we determine a total project budget that should cover the cost of activity execution, delays and activity crashing. This is a more realistic situation since the cost of activity delays and penalty costs for project delays are explicitly considered. Second, our focus is on cost minimisation rather than timely project completion. Therefore, we consider projects that exceed the predefined project budget to be failed projects and we evaluate the performance of the different scenarios based on the actual project duration and cost and the portion of failed projects. Finally, we test the proposed approaches on real-life project data in order to incorporate realistic cost profiles.

II. Methodology

The methodology of this study consists of four phases. In the *data collection phase*, planning and risk data from real-life projects are collected. In the *scenario analysis phase*, different scenarios to allocate the project budget over the project life cycle are defined. During the *simulation phase*, progress data for each of the real-life projects is simulated to review the impact of activity crashing on the cost profile of the projects. The *evaluation phase* consists of an analysis of the duration and cost performance of the projects. Table I summarises the relevant terminology and parameters used in this study.

A. Data collection

In the data collection phase, real-life planning, progress and risk data from projects in various industries has been collected and documented. After a first meeting, the project owner or manager decides whether they are willing to collaborate. If this is the case, a project that is planned to start in the near future is selected for real-time periodical follow-up. This ensures that the documented data is correct and complete. At each period, it is reviewed whether the information is available at the activity level. If activity level data is available, the activity risk profiles are discussed with the project owner or manager. Otherwise, the follow-up process is terminated. When all collected data is clear, it is registered in a structured manner. This process is repeated periodically until the project is finished.

Table II gives an overview of the collected data and lists the industry, planned duration (PD), Budget at Completion (BAC) and number of activities (nract) of each project. These projects are included in the database of [12] and are online available at https://www.projectmanagement.ugent.be/research/data/realdata.

B. Scenario analysis

Before the project start, the *total project budge*t is defined as the BAC (section II-A) increased with a management reserve to take corrective actions and to deal with activity delays during execution. In the computational experiment, the size of the management reserve is varied and different scenarios considering the timing and quantity of the budget release are considered for the release of the project budget during execution.

a) Size of management reserve: In this study, the size of the management reserve (MR) is defined as the difference between the average actual project cost (APC) without activity crashing and the BAC, multiplied with a factor m_B (equation (1)). The APC without activity crashing is calculated by adding uncertainty profiles to the activity durations and using Monte Carlo simulations to imitate the actual project progress (section II-C).

$$\mathrm{MR} = (\mathrm{APC} - \mathrm{BAC}) \times m_B \tag{1}$$

b) Timing of budget release: Two different approaches are considered regarding the timing of the budget release. First, the *immediate approach* assumes that the entire project budget is made available from the start of the project. Second, the *time phased approach* assumes that the project is divided in different phases, and a portion of the total project budget is released at the start of each phase.

c) Quantity of budget release: The quantity of the budget release depends on the applied timing approach. For the immediate timing approach, the entire budget (BAC + management reserve) is released at the start of the project. For the time phased approach, the planned budget at the end of each phase is increased with a portion of the management reserve and released at the start of each phase. To determine the portion of the management reserve to be released at each phase, two viewpoints are used. First, using the *time focus* viewpoint (equation (2)), the management reserve is allocated to each phase proportional with the relative duration of each phase ($\frac{\mathrm{PD}_{\mathrm{phase}}}{\mathrm{PD}_{\mathrm{project}}}$). Second, using the *cost focus* viewpoint (equation (3)), the management is allocated to each phase proportional with the relative budgeted cost of each phase ($\frac{\mathrm{BAC}_{\mathrm{phase}}}{\mathrm{BAC}_{\mathrm{project}}}$).

$$\text{Assigned budget}_{\text{phase, time}} = \mathrm{BAC}_{\mathrm{phase}} + \frac{\mathrm{PD}_{\mathrm{phase}}}{\mathrm{PD}_{\mathrm{project}}} \times \mathrm{MR} \tag{2}$$

$$\text{Assigned budget}_{\text{phase, cost}} = \mathrm{BAC}_{\mathrm{phase}} + \frac{\mathrm{BAC}_{\mathrm{phase}}}{\mathrm{BAC}_{\mathrm{project}}} \times \mathrm{MR} \tag{3}$$

While equations (2) and (3) allocate the management reserve over the different phases proportionally with the time or cost (i.e. the *standard version*), the management reserve can be allocated in a dynamically increasing or decreasing way as well [10]. The *increasing version* ensures that the allocated management reserve is relatively low at the start of the project and systematically increases along the project progress by allocating the management reserve based on the square of the relative duration or cost of each phase (Equation (4)). The *decreasing version* uses the square root to start with a relatively high amount of the budget which increases degressively along the project progress (Equation 5). In table III, an overview of the scenarios reviewed in the simulation experiment is given.

$$\text{Assigned budget}_{\text{phase, time, increasing}} = \mathrm{BAC}_{\mathrm{phase}} + \left(\frac{\mathrm{PD}_{\mathrm{phase}}}{\mathrm{PD}_{\mathrm{project}}}\right)^2 \times \mathrm{MR} \tag{4}$$

$$\text{Assigned budget}_{\text{phase, time, decreasing}} = \mathrm{BAC}_{\mathrm{phase}} + \sqrt{\frac{\mathrm{PD}_{\mathrm{phase}}}{\mathrm{PD}_{\mathrm{project}}}} \times \mathrm{MR} \tag{5}$$

C. Simulation

In the simulation phase, 1,000 simulated project executions are generated for each project collected in the data collection phase. Before the start of each project, the tolerance limits for the project progress are set as introduced by [4]. During each simulated execution, the project progress is measured and compared to these limits periodically. When the incurred

TABLE I
OVERVIEW OF TERMINOLOGY

Concepts		**Parameters**		**Performance measures**	
Project		*Project*			
nract	number of activities	m_B	multiplier total budget	AFP	Actual Failed Projects
BAC	Budget at Completion	m_D	multiplier delay cost	APD	Actual Project Duration
PD	Planned Duration	C_D	Cost of unit delay	APC	Actual Project Cost
AD	Actual Duration		$= \frac{\text{BAC}}{\text{PD}} \times m_D$		
Activities		*Activities*			
AD$_i$	Actual duration act i	$C_{C,i}$	Crash cost of act i		
$C_{F,i}$	Fixed cost of act i		$= 2 \times C_{F,i}$		
$C_{V,i}$	Variable cost of act i				
UC$_i$	Crashed units of act i				

TABLE II
OVERVIEW OF REAL-LIFE PROJECTS

Project ID	Industry	PD (days)	BAC (€)	nract
C2011_03	Event	97	31,675	24
C2011_04	Construction	125	59,831	20
C2011_05	Telecom	43	180,485	23
C2011_08	Construction	72	254,564	28
C2011_11	Event	299	37,760	26
C2012_01	Manufacturing	45	61,699	31
C2012_11	Manufacturing	13	1,535,854	24
C2013_17	Construction	161	244,205	25
C2014_07	Construction	353	1,102,537	27
C2014_08	Construction	233	1,992,222	41

costs at the period exceed the released budget until that time, the project is interrupted until an additional part of the project budget is released and the project can be resumed. Further, when the total project budget is exceeded, the project is terminated and classified as a failed project. When the project is not interrupted or terminated, the project progress is reviewed. If the progress is below the tolerance limit, the activities eligible for activity crashing are determined by comparing the activity crash cost of the ongoing critical activities (i.e. activities on the critical path) to the expected delay cost reduction. If there are eligible activities and the required budget for activity crashing is available at the period, the actions are taken and the project is continued.

D. Performance evaluation

After completion of the simulation phase, the performance of the simulated executions is reviewed. For each simulation experiment, the number of failed projects (AFP), i.e. the number of projects that exceeded their budget, is observed. Further, the time and cost performance is evaluated using the actual project duration (APD) and actual project cost (APC). The APD and APC are expressed relatively to the planned duration and the total project budget of the projects, respectively.

III. RESULTS AND DISCUSSION

In this section, the performance of the scenarios depicted in table III is evaluated using the project performance measures listed in table I (Experiment 1). In Experiment 2, the impact on the project performance of a change in the total project budget is examined by varying the value of m_B (equation (1)). Finally, Experiment 3 reviews the impact of changes in the cost per unit delay ($C_D = \frac{\text{BAC}}{\text{PD}} \times m_D$) by varying the value of m_D.

a) Experiment 1: Comparison of scenarios: The results of Experiment 1 are summarised in table IV. The results show that the immediate budget release approach (S1) outperforms the time phased budget release approaches in terms of AFP, APD and APC. All project runs are finished within the assigned project budget, with an average duration of 101.7% of the PD. Further, for the standard time phased assignment versions, the time focus (S2) performs better than the cost focus (S5) for all performance measures. This can be explained by the fact that the cost focus assigns higher portions of the management reserve to more costly project phases. Since the activities planned in these phases are typically more expensive to crash, less corrective actions can be taken with the same budget. Finally, the decreasing version of the time focus approach (S3) uses the available project budget most effectively, since it results in the lowest AFP, APD and APC of all time phased approaches.

b) Experiment 2: Impact of changes in total project budget size: Since experiment 1 showed that the time phased scenarios using a time focus outperform the scenarios using a cost focus, the remaining discussion focuses on scenarios S1-S4. In general, table V shows that reducing the size of the management reserve increases the APD, APC and especially the AFP. For an immediate budget release (S1), a reduction of 10% ($m_B = 0.9$) has a limited impact, while a reduction of 20% ($m_B = 0.8$) results in considerably more failed projects. For the time phased scenarios, the impact of reducing the budget is more substantial. For a reduction of 10%, S3 still outperforms S2 and S4. When the management reserve is reduced with 20%, however, the performance of these scenarios become comparable.

c) Experiment 3: Impact of changes in the cost of delays: Table VI shows that reducing the unit cost of delay (m_D) has a limited impact. Both the AFD, APD and APC increase slightly for lower unit costs of delay. The increase in APC can be explained by the fact that activities are only crashed when the expected reduction in delay costs is higher than the increased costs due to the crashing action.

To conclude, this simulation experiment indicates that a management reserve should be considered to control projects during execution. Experiment 1 shows that both the timing

TABLE III
OVERVIEW OF CONSIDERED SCENARIO SETTINGS

Scenario	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**
Timing	Immediate	Time phased	Time phased	Time phased	Time phased	Time phased	Time phased
Quantity	-	Time	Time	Time	Cost	Cost	Cost
Version	-	Standard	Decreasing	Increasing	Standard	Decreasing	Increasing

TABLE VI
IMPACT OF CHANGES IN COST OF DELAYS ($m_B = 1$)

Scenario	m_D	**AFP (%)**	**APD (%)**	**APC (%)**
S1	**1.00**	0	101.7	89.6
	0.75	0	101.7	91.1
	0.50	0	101.7	92.6
S2	**1.00**	16	105.4	90.8
	0.75	20	106.4	92.5
	0.50	22	107.1	92.6
S3	**1.00**	6	102.9	89.7
	0.75	8	103.3	91.2
	0.50	11	103.9	92.8
S4	**1.00**	28	108.5	92.7
	0.75	32	109.1	93.7
	0.50	38	109.7	94.6

TABLE IV
COMPARISON OF SCENARIOS ($m_B = 1, m_D = 1$)

Scenario	**AFP (%)**	**APD (%)**	**APC (%)**
S1	0.0	101.7	89.6
S2	15.8	105.4	90.8
S3	5.8	102.9	89.7
S4	27.7	108.5	92.7
S5	19.8	105.8	91.2
S6	6.1	103.1	89.7
S7	33.7	111.3	94.4

TABLE V
IMPACT OF CHANGES IN TOTAL PROJECT BUDGET SIZE ($m_D = 1$)

Scenario	m_B	**AFP (%)**	**APD (%)**	**APC (%)**
S1	**1.0**	0	101.7	89.6
	0.9	3	101.8	91.5
	0.8	19	102.7	94.0
S2	**1.0**	16	105.4	90.8
	0.9	41	108.7	94.6
	0.8	72	113.3	99.7
S3	**1.0**	6	102.9	89.7
	0.9	28	105.8	92.7
	0.8	70	111.8	98.6
S4	**1.0**	28	108.5	92.7
	0.9	50	110.7	95.9
	0.8	75	114.0	100.2

and amount of the budget release have an effect on the actual project duration and cost. Further, experiment 2 shows that the total size of the management reserve is of importance as well. If the management reserve is too low, the performance of different strategies for the amount of budget release perform equally low. Finally, when the cost of delays is decreased, this has a more substantial impact on the actual project cost than on the actual project duration.

REFERENCES

[1] M. Vanhoucke, *Project Management with Dynamic Scheduling: Baseline Scheduling, Risk Analysis and Project Control.* Springer, 2012, vol. XVIII.

[2] Q. Fleming and J. Koppelman, *Earned Value Project Management*, 3rd ed. Newton Square, Pennsylvania: Project Management Institute, 2010. [Online]. Available: http://books.google.be/books?id=ZMRVngEACAAJ

[3] J. Colin and M. Vanhoucke, “Setting tolerance limits for statistical project control using earned value management,” *Omega The International Journal of Management Science*, vol. 49, pp. 107–122, 2014.

[4] A. Martens and M. Vanhoucke, “A buffer control method for top-down project control,” *European Journal Of Operational Research*, vol. 262, pp. 274–286, 2017.

[5] J. Zhang, S. Jia, and E. Diaz, “Dynamic monitoring and control of a critical chain project based on phase buffer allocation,” *Journal of the Operational Research Society*, vol. 69, pp. 1–12, 2018.

[6] H. Hadian and A. Rahimifard, “Multivariate statistical control chart and process capability indices for simultaneous monitoring of project duration and cost,” *Computers & Industrial Engineering*, pp. 788–797, 2019.

[7] M. Madadi and H. Iranmanesh, “A management oriented approach to reduce a project duration and its risk (variability),” *European Journal of Operational Research*, vol. 219, no. 3, pp. 751–761, 2012.

[8] P. Ballesteros-Pérez, K. Elamrousy, and M. Gonz’ales-Cruz, “Non-linear time-cost trade-off models of activity crashing: Application to construction scheduling and project compression with fast-tracking,” *Automation in Construction*, vol. 97, pp. 229–240, 2019.

[9] A. Martens and M. Vanhoucke, “The impact of applying effort to reduce activity uncertainty on the project time and cost performance,” *European Journal of Operational Research*, vol. 277, no. 2, pp. 442–453, 2019.

[10] J. Song, A. Martens, and M. Vanhoucke, “The impact of a limited budget on the corrective action taking process,” *European Journal Of Operational Research*, vol. 286, no. 3, pp. 1070–1086, 2020.

[11] M. Vanhoucke, A. Vereecke, and P. Gemmel, “The project scheduling game (PSG): Simulating time/cost trade-offs in projects,” *Project Management Journal*, vol. 51, pp. 51–59, 2005.

[12] J. Batselier and M. Vanhoucke, “Construction and evaluation framework for a real-life project database,” *International Journal of Project Management*, vol. 33, pp. 697–710, 2015.

26th Conference on Knowledge Acquisition and Management

KNOWLEDGE management is a large multidisciplinary field having its roots in Management and Artificial Intelligence. Activity of an extended organization should be supported by an organized and optimized flow of knowledge to effectively help all participants in their work.

We have the pleasure to invite you to contribute to and to participate in the conference "Knowledge Acquisition and Management". The predecessor of the KAM conference has been organized for the first time in 1992, as a venue for scientists and practitioners to address different aspects of usage of advanced information technologies in management, with focus on intelligent techniques and knowledge management. In 2003 the conference changed somewhat its focus and was organized for the first under its current name. Furthermore, the KAM conference became an international event, with participants from around the world. In 2012 we've joined to Federated Conference on Computer Science and Systems becoming one of the oldest event.

The aim of this event is to create possibility of presenting and discussing approaches, techniques and tools in the knowledge acquisition and other knowledge management areas with focus on contribution of artificial intelligence for improvement of human-machine intelligence and face the challenges of this century. We expect that the conference&workshop will enable exchange of information and experiences, and delve into current trends of methodological, technological and implementation aspects of knowledge management processes.

Topics

- Knowledge discovery from databases and data warehouses
- Methods and tools for knowledge acquisition
- New emerging technologies for management
- Organizing the knowledge centers and knowledge distribution
- Knowledge creation and validation
- Knowledge dynamics and machine learning
- Distance learning and knowledge sharing
- Knowledge representation models
- Management of enterprise knowledge versus personal knowledge
- Knowledge managers and workers
- Knowledge coaching and diffusion
- Knowledge engineering and software engineering
- Managerial knowledge evolution with focus on managing of best practice and cooperative activities
- Knowledge grid and social networks
- Knowledge management for design, innovation and eco-innovation process
- Business Intelligence environment for supporting knowledge management
- Knowledge management in virtual advisors and training
- Management of the innovation and eco-innovation process
- Human-machine interfaces and knowledge visualization

Technical Session Chairs

- **Hauke, Krzysztof,** Wroclaw University of Economics, Poland
- **Nycz, Malgorzata,** Wroclaw University of Economics, Poland
- **Owoc, Mieczyslaw,** Wroclaw University of Economics, Poland
- **Pondel, Maciej,** Wroclaw University of Economics, Poland

Program Committee

- **Abramowicz, Witold,** Poznan University of Economics, Poland
- **Andres, Frederic,** National Institute of Informatics, Tokyo, Japan
- **Bodyanskiy, Yevgeniy,** Kharkiv National University of Radio Electronics, Ukraine
- **Chmielarz, Witold,** Warsaw University, Poland
- **Christozov, Dimitar,** American University in Bulgaria, Bulgaria
- **Jan, Vanthienen,** Katholike Universiteit Leuven, Belgium
- **Mercier-Laurent, Eunika,** University Jean Moulin Lyon3, France
- **Sobińska, Małgorzata,** Wroclaw University of Economics, Poland
- **Surma, Jerzy,** Warsaw School of Economics, Poland and University of Massachusetts Lowell, United States
- **Vasiliev, Julian,** University of Economics in Varna, Bulgaria
- **Zhu, Yungang,** College of Computer Science and Technology, Jilin University, China

Organizing Committee

- **Hołowińska, Katarzyna**
- **Przysucha, Łukasz,** Wroclaw University of Economics

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 609–618 DOI: 10.15439/2020F44 ISSN 2300-5963 ACSIS, Vol. 21

Comprehension analysis considering programming thinking ability using code puzzle

Hiroki Ito
Ritsumeikan University
Graduate School of Information
Science and Engineering
Email: hirokiito6900@de.is.ritsumei.ac.jp

Hiromitsu Shimakawa
Ritsumeikan University
College of Information
Science and Engineering
Email: simakawa@cs.ritsumei.ac.jp

Fumiko Harada
Connect Dot Ltd.
Email: harada@de.is.ritsumei.ac.jp

***Abstract*—In programming education, the instructor tries to find out the learners who needs help by grasping the learners' development of understanding using tests that require knowledge. However, in reality, not many learners will acquire the skill of writing source codes. This kind of current situation implies that programming ability of learners cannot be measured by tests that require knowledge. This paper focuses on not only the knowledge items required for programming but also the programming thinking (computational thinking), which is the ability to combine the constituent elements of the program. In this paper, we propose a method to estimate the learner's understanding from the learner's process to solve the code puzzles that require programming thinking as well as knowledge. We developed the interface to realize the proposed method. The experimental result with the interface showed that the proposed method could estimate with the accuracy of 80% or more.**

I. Introduction

IN programming education for beginners, there are many learners who cannot create correct sources in spite of passing written tests that ask their knowledge about grammar and how to express algorithms. The programming ability cannot be measured only by verifying only learner's knowledge. This is because programming skill also requires the ability to construct program elements logically with a perspective.[1][2] Programming thinking is the ability to assemble the components of a program with a perspective. The method of measuring programming thinking ability from the requirements to be satisfied has not been established yet. In the actual situation, the only way to verify the true programming ability of a learner, which consists of both knowledge and programming thinking ability, is for the instructor to stand next to the learner and watch the answer. However, in a large-class lecture, it takes too much time to check the understanding of all students in this way. Most of the current educational settings use measurement methods that are biased in terms of knowledge because it is easy to grasp the understanding situation. As the result, many learners will not be able to understand the intention of the task and to acquire the ability to realize it. From such the situation, there is a demand for a method that can easily estimate the learners' understanding situation by considering programming thinking ability.

II. Programming education support

A. Current programming education support

This paper discusses an understanding analysis focusing on programming thinking ability. Programming thinking ability means "what kind of combination of movements is necessary to realize a series of activities intended by oneself, and how to combine symbols corresponding to each movement. And the ability to logically consider how to improve the combination of symbols to get closer to the intended activity. "[3]

Although programming education for beginners is conducted in educational institutions such as universities and newcomer education at companies, many people cannot actually program even if they can pass a written test that asks knowledge. This indicates that programming cannot be achieved by knowledge alone, and is thought to be due to the lack of programming thinking as mentioned above as pointed out by the Ministry of Education, Culture, Sports, Science and Technology in Japan[3].

This paper defines the learning item achievement level as the degree of acquirement of the knowledge given in a lecture, materials, and so on. It is a contrasting skill of the programming thinking ability. Most current programming education support focuses on the learning item achievement level. Therefore, there is an urgent need to establish a learning support method that considers programming thinking.

B. Programming learning by a code puzzle

This research uses a code puzzle as a learning interface. A code puzzle is a rearrangement problem where the learner rearranges code fragments such as source code and pseudo code and assembles them to perform appropriate processing. The code puzzle is inspired by Parson's Programming Puzzle proposed by Parson *et al.*[4] Parson *et al.* said that, for beginners, code puzzles are more effective and better at nurturing logical thinking than full-coding . Moreover, code puzzles present the logic flow unlike the blank filling problem. Code puzzles simplifies to acquire a feature of the learner's learning behavior used for estimation of comprehension from the actions of selecting, moving, and rearranging blocks.

C. *Schema for programming tasks*

Schema is a term in cognitive psychology. Schema in cognitive psychology refers to the relationship between thought patterns and knowledge for problem solving in human long-term memory.[5]

When a person solves a problem of programming, he or she reads the problem and sets a path for problem solving. At this time, if he/she has no necessary knowledge and thought pattern, it is not possible for him/her to set the path for problem solving. In other words, we can consider that humans combine knowledge to solve problems. In the process of combining knowledge, humans form patterns of knowledge and thinking as a schema. The pattern of knowledge or thinking here is exactly the programming thinking. In this research, we consider to estimate the degree of schema construction from the combination of knowledge and logic based on the schema theory.

D. *Related works*

The method proposed by Jadud *et al.*[6] is an early study that identifies learners who need guidance using compilation errors. However, this focuses only on the error, and it is not possible to measure why the error occurred and how much the learner understood.

Mysore et al[7] proposed a Web system Porta that can identify the part where the learner is struggling. The comprehension factor is inherently intricately intertwined. However, Porta estimates the comprehension level only by focusing on the fact that the learner takes time. Therefore, the ability and growth of learners are not taken into consideration.

Guo *et al.*[8] proposed an interface that supports one-to-many programming learning in real time and its implementation Codeopticon. However, Codeopticon depends on the quality of the instructor and forces a heavy burden on the instructor.

Asai *et al.*[9] identified the cognitive load on learners and the factors that caused the cognitive load by the blank filling problem. However, the blank-filling problem prevents the flow of logic. It cannot be said that it considers programming thinking.

Many of these existing researches focus only on the aspect of knowledge and do not estimate the degree of understanding based on logic(programming thinking).

As a study focusing on behavior, Ihantola *et al.*[10] estimated the difficulty level of a task using a decision tree from the answering process such as answering time and keystroke of a programming task. They suggested that the answer process brings significant difference in learner's comprehension. However, they did not mention programming thinking ability and cannot estimate factors of misunderstanding.

As one of the traditional programming education formats, there is Parson's programming puzzle proposed by Parsons *et al.* This is introduced as a tool that is easy for beginners to work on. Parson *et al.* asserted that code puzzles can identify specific points and errors that the learner has stumbled. They argued that, since the code puzzle answer is a well model answer, learners can relive good programming practice. However, although this tool focuses on the acquisition of programming thinking ability, it does not estimate the degree of comprehension.

III. Educational support using behavior when answering code puzzles

A. *Educational support focusing on programming thinking*

The purpose of this study is to provide novel education support method focusing on not only the learning item achievement level as shown in ordinary education support but also the ability to assemble those learning items and to realize the intended program (called Programming thinking). The learning item achievement level in this paper is defined as the level of knowledge necessary to solve programming problems. On the other hand, programming thinking ability is defined as the ability to combine the knowledge to design deliverables that match the programming task. The learning item achievement can be measured easily by paper-based or Web-based tests. However, programming thinking ability cannot be measured without observing the programming process of the learner, which is the learner's behavior. This is because programming thinking occurs in the process of creating a program. A learner who can perform programming thinking will answer a programming problem with prospecting and constructing the logic flow toward the answer. Therefore, it is not possible to judge whether the learner has acquired programming thinking or not unless the learner's behavior in the answering process is investigated.

It has been difficult to measure programming thinking ability unless the instructor watches the learner's answering process to the programming task. Code puzzles are a better tool for learners to focus on combining knowledge. The method proposed in this research estimates such programming thinking ability by analyzing the process in which a learner solves a code puzzle. The outline of the educational support method aimed at in this research is shown in Figure 1. At first, the learners solve the code puzzle. The interface used for the code puzzle is the original application that runs on the WEB. This application collects the operation histories of the learners. These operation histories will differ depending on the learner. Based on this hypothesis, the understanding levels of the learner are classified by machine learning based on the collected operation histories. A machine learning model that can interpret the reason for classification is adopted. The reason feedbacks to the learner and instructor.

B. *Collecting learner behavior using code puzzles*

The method proposed in this study uses the tools shown in Figure 2 and 3 to collect learner's characteristic behaviors to measure the understanding. The test subjects are Japanese, so the content is displayed in Japanese. In this research, the notation of the program is based on PAD proposed by Futamura *et al.*[11]

In the proposed method, an exemplary program or source code that satisfies all the requirements given in the task is

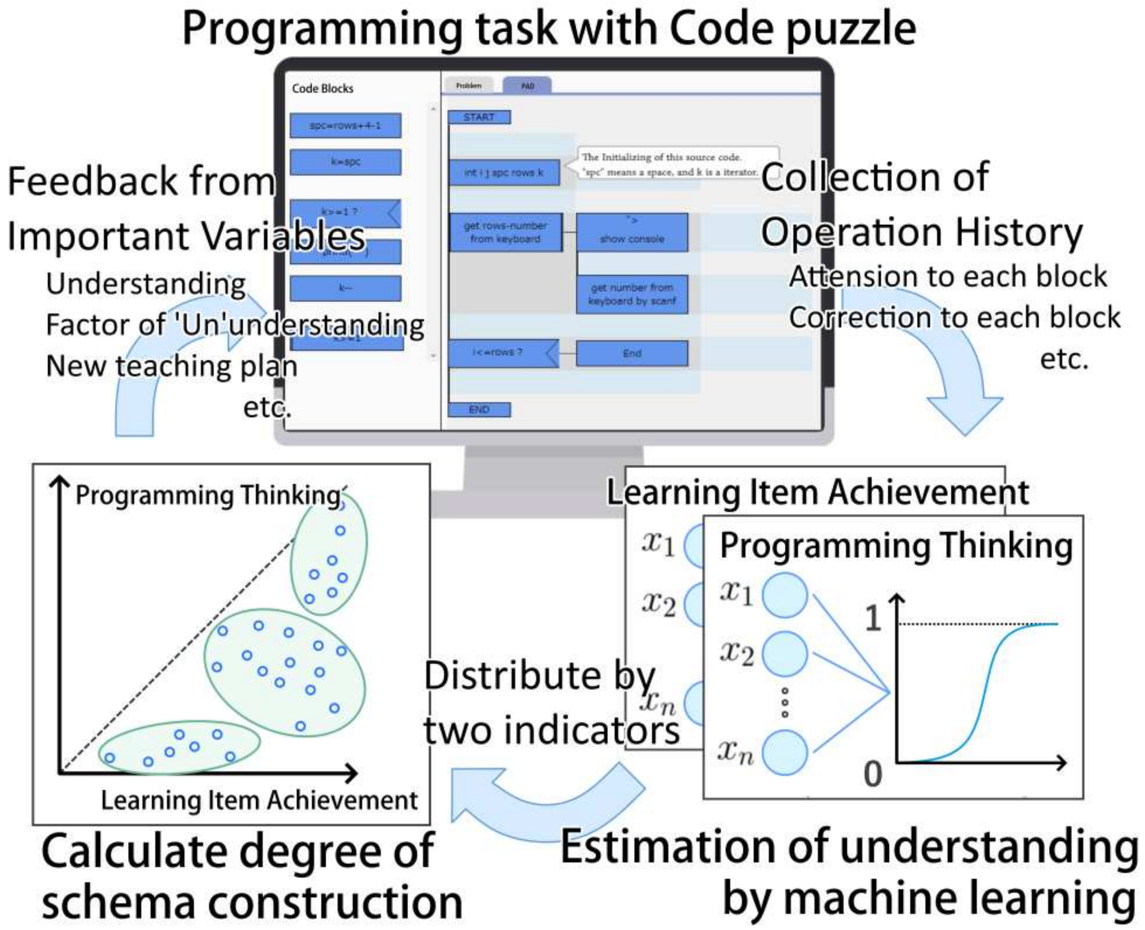

Programming task with Code puzzle
Feedback from Important Variables
Understanding
Factor of 'Un'understanding
New teaching plan
etc.
Collection of Operation History
Attension to each block
Correction to each block
etc.
Code Blocks
spc=rows+4-1
k=spc
k>=1 ?
k--
Problem
PAD
START
int i j spc rows k
The Initializing of this source code. "spc" means a space, and k is a iterator.
get rows-number from keyboard
show console
get number from keyboard by scanf
i<=rows ?
End
END
Programming Thinking
Learning Item Achievement
Learning Item Achievement
Programming Thinking
x_1
x_2
x_n
1
0
Distribute by two indicators
Calculate degree of schema construction
Estimation of understanding by machine learning

Fig. 1. Schematic diagram of the proposed method

divided into code fragments or pseudo code with functional cohesion. A code fragment generated by the division is defined as a block in this paper.

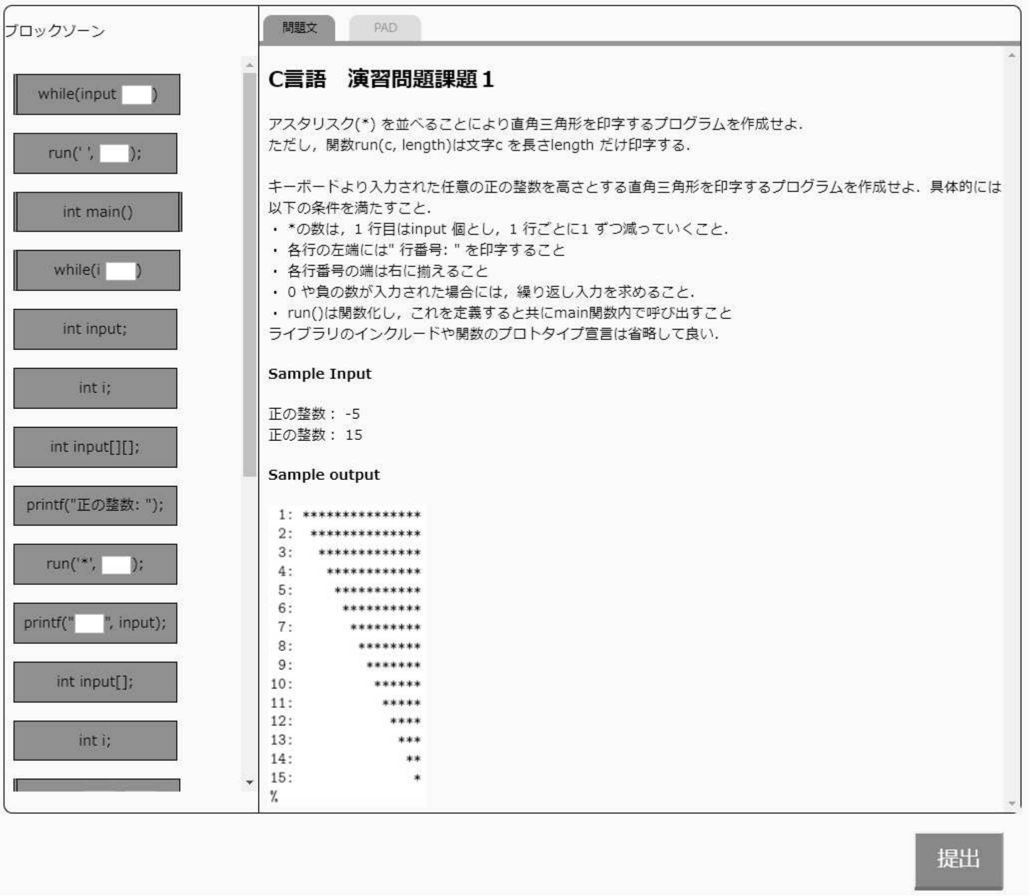
As shown in Figure 2 and 3, blocks for assembling the program are displayed for a task. The learner assembles the program using the blocks so that all the given requirements are satisfied. This can be regarded as a code puzzle that considers the arrangement of blocks so that the constraints are satisfied.

The proposed method examines the learners' thinking processes during solving the code puzzle in order to judge whether the learners have acquired programming thinking or not. When the learners answer with a perspective of how to combine the blocks toward satisfying the given requirements, they will place the blocks in the correct position without hesitation. The learners without such a perspective will wonder where to place the blocks. The learners who incorrectly interpreted the meaning of the problem sentence and/or given requirements will quickly move the blocks to wrong positions. In this way, it is assumed that the learners' thinking processes during answering appears in the behavior of moving blocks. From this idea, the proposed method analyzes the behavior of each learner collected by the tool during the learner is placing blocks.

The learner can switch between the task sentence screen(2) and the drawing screen(3) from the tab at the top of the tool. The learner interprets the problem from the task sentence as shown in Figure 2, selects suitable blocks from the blocks displayed on the left side of the tool as shown in Figure 3, and assembles them by drag and drop. Some of the blocks contain blanks. Blocks with blanks increase the degree of freedom in expressing procedures and are intended to cause the learner to get lost. Additionally, if the learner hovers the mouse over a block, a description of that block is displayed. The learner finishes the answer by pressing the submit button when they think they has answered it correctly.

C. Explanatory variables collected by the tool

The tool analyzes the relationship between the learners' behaviors during answering and their programming ability through machine learning models. In this research, the behavior of the learner is the explanatory variables and the understanding of the learner is the response variables. Before applying this method, we consider what the explanatory variables correspond to the learner's thinking so that the learner's



```
while(input    )
run(' ',    );
int main()
while(i    )
int input;
int i;
int input[][];
printf("正の整数: ");
run('*',    );
printf("    ", input);
int input[];
int i;
```

C言語　演習問題課題１

アスタリスク(*) を並べることにより直角三角形を印字するプログラムを作成せよ.
ただし，関数run(c, length)は文字c を長さlength だけ印字する.

キーボードより入力された任意の正の整数を高さとする直角三角形を印字するプログラムを作成せよ．具体的には以下の条件を満たすこと.
・ *の数は，1 行目はinput 個とし，1 行ごとに1 ずつ減っていくこと.
・ 各行の左端には" 行番号: " を印字すること
・ 各行番号の端は右に揃えること
・ 0 や負の数が入力された場合には，繰り返し入力を求めること.
・ run()は関数化し，これを定義すると共にmain関数内で呼び出すこと
ライブラリのインクルードや関数のプロトタイプ宣言は省略して良い.

Sample Input

正の整数： -5
正の整数： 15

Sample output

```
 1: ***************
 2:  **************
 3:   *************
 4:    ************
 5:     ***********
 6:      **********
 7:       *********
 8:        ********
 9:         *******
10:          ******
11:           *****
12:            ****
13:             ***
14:              **
15:               *
%
```

Fig. 2. Question sentence screen

incomprehensible factors can be investigated. The explanatory variables used in the proposed method are roughly divided into three.

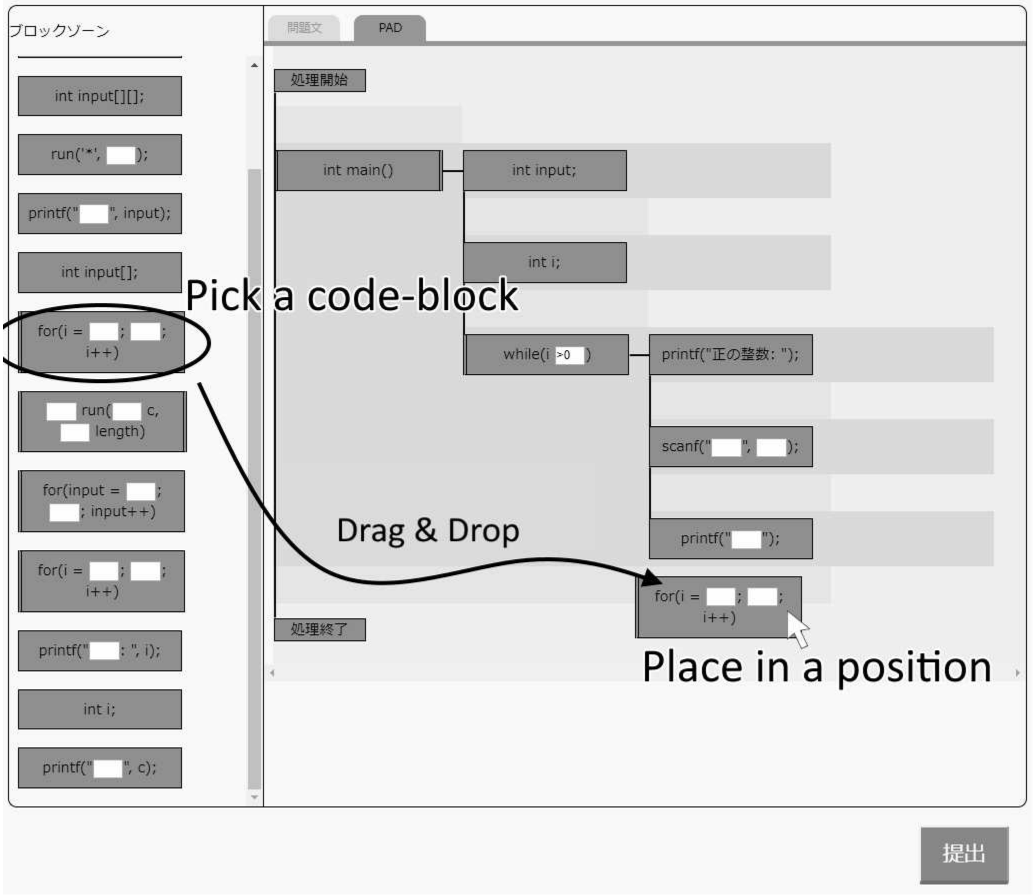
- Attention to each block
  - First time of touching each block
  - Time spent watching at each block description by hover
  - The number of times each block has been dragged and dropped
  - Drag and drop frequency
  - Frequency of drag and drop in each quarter of the answering time
  - Dispersion of time watching the description
  - Dispersion of drag and drop number
- Correctness in placing blocks
  - Correctness or incorrectness of input to each blank
  - Number of times each blank correction failed
  - How close the whole is to the model answer (editing distance)
  - Number of times overall blank correction failed
- Time spent
  - Total task time
  - Ratio of drawing screen display time for task sentence screen display time
  - Number of times to switch between the task sentence screen and the drawing screen

The attention level and correctness of each block reveal the block that confuses the learner. The overall correctness reveals how accurately the blocks are combined. In addition, the time and ratio spent on drawing using PAD indicate how much the learner got confused during drawing.

D. Estimating the understanding for each learner

In this method, the understanding is estimated based on the behavior of the learner collected by the tool. It is estimated using the random forest method and logistic regression as machine learning methods. Since the two methods can calculate the importance of variables, it is possible to consider classification factors. Although the random forest method is more accurate, the probability of classification can be known by logistic regression, and this classification probability can be grasped as an understanding level of 0.0 to 1.0.

In the proposed method, the understanding is estimated based on the behavior of the learner collected by the tool. It is estimated using the random forest method and logistic



```
int input[][];
run('*',   );
printf("   ", input);
int input[];
for(i =   ;   ; i++)
   run(   c,    length)
for(input =   ;   ; input++)
for(i =   ;   ; i++)
printf("   : ", i);
int i;
printf("   ", c);
```

```
int main()
int input;
int i;
while(i >0 )
printf("正の整数: ");
scanf("   ",   );
printf("   ");
for(i =   ;   ; i++)
```

Pick a code-block

Drag & Drop

Place in a position

Fig. 3. Drawing screen

regression as machine learning methods. Since the two methods can calculate the importance of variables, it is possible to consider classification factors. Although the random forest method is more accurate, the probability of classification can be known by logistic regression and this classification probability can be grasped as an understanding level of 0.0 to 1.0. The understanding is judged by two measures: the learning item achievement level and programming thinking ability. Therefore, the proposed method uses machine learning to create two models of learning item achievement and programming thinking. This is based on the knowledge that the real ability of programming cannot be realized unless the learning items achievement and the programming thinking are compatible.[12]

According to the schema theory[5], a person combines knowledge to solve tasks. Therefore, the programming thinking, which is the power to combine knowledge, is considered to be based on knowledge.

Therefore, the degree of schema construction is defined as follows by using the learning item achievement k and the programming thinking ability l as an index that integrates the learning item achievement and the programming thinking ability. k and l are defined by values between 0 and 1.

$$\begin{aligned} d &= \begin{Bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{Bmatrix} \bullet \begin{Bmatrix} k \\ l \end{Bmatrix} \\ e &= \left(f\left(k\right) - l\right)^2 = \left(k - l\right)^2 \\ s &= d - e \end{aligned}$$

d is the dot product with the unit vector $(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and vector $(k.l)$. In other words, d takes the maximum value if both of k and l are 1.0. d become larger as k approaches to l. From the geometrical perspective, as d becomes larger the direction vector $(k.l)$ approaches to that of the line with the slope 45 degrees, whose direction vector is $(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$. e is the squared error from the line $f(x) = x$. When calculating s, which represents the balance between learning item achievement and programming thinking ability, e indicates imbalance and functions as a penalty.

E. Factors of understanding for each learner

In order to estimate the understanding of a learner, it is possible to obtain the classification result of whether or not the learner understands the task from the machine learning model. When using logistic regression and random forest, it is possible to see which explanatory variables and how much influenced the classification by referring to the regression coefficient and variable importance, respectively. From these indexes, the factors that a learner classified as not being understood are found. The position on the code to which the factor corresponds, that is, the misunderstanding part, becomes clear. In this study, this factor is called the misunderstanding factor. In addition, these indicators provide clues as to where learners tend to stumble and which parts a learner does not know.

F. Feedback to learners and teachers

The proposed method provides two types of feedback:

- Degree of Schema construction including programming thinking
- Misunderstanding factors and Misunderstanding points that take programming thinking into consideration

The schema construction can visualize understanding of a large number of learners with unified numerical values. Based on the schema construction, the learner can grasp how much he/she understand with compared to others. Furthermore, the instructor can detect learners who need guidance at early stage.

In addition, when analysis is performed over a period of time, it is possible to calculate the degree of growth and to find stumbling based on changes in the degree of schema construction.

The misunderstanding factors and misunderstanding points can be considered from the coefficient of logistic regression and the importance variables of the random forest. The misunderstanding factors can be grasp by comparing the learner's behavior based on the variables of high importance. This analysis tells the learner what part they do not understand or how they tends to answer. In addition, the instructor can be informed of what part learners confused and what learners do not understand. Therefore, it is possible to create a new teaching plan for individuals and the whole.

IV. Experiment

A. Objective and method

The purpose of this experiment is to clarify the understanding considering the learner's programming thinking ability from the learner's operation history using code puzzles. The subjects are 17 university students, including first-year undergraduate students who started learning programming and first-year graduate students who are accustomed to programming. Each of the subjects solved a given programming task. The time to solve the task took the minimum of 7 minutes and the maximum of 40 minutes. The degrees of attainment were various between the subjects. The experiment was conducted with the time constraint of 1 hour per person. In the first 10 minutes, we gave a brief tutorial on how to use the tool. In the next 40 minutes, each subject worked on a programming task using the code puzzle. In the last 10 minutes, for the labeling described later, we conducted a questionnaire asking about the cognitive load of this task and usual experience. While solving the task, the instructor did not give any hints and just watched to measure programming thinking ability.

B. Actual understanding and cognitive load measurement

Before performing analysis by machine learning, the instructor labeled the objective variables used in supervised learning. The label used for training is a binary value (0/1) indicating whether or not the subject understood. For subject, a 0 or 1 label was assigned to each subject for each of the learning item achievement and programming thinking ability.

1) The problem content was very complicated.
2) The knowledge used for filling in the blanks and functions was very complicated.
3) The concepts and ideas in the task were very complicated.
4) It was very unclear how to use tools and PAD notation.
5) It was difficult to understand how to use tools and PAD notation.
6) Tools and PAD notation are very inefficient from a learning perspective.
7) The task has improved my understanding of programming.
8) The task improved the understanding of programming process
9) The task has increased my understanding of programming concepts and definitions.
10) The task has improved my knowledge of programming.
11) It took a lot of mental effort because the task was complicated.
12) Due to the explanation in the task and the usage of the tool, I took a lot of mental effort.
13) It took a lot of effort to improve knowledge and understanding in the task.
14) Relative programming score
15) Relative programming experience

Fig. 4. Questionnaire for measuring cognitive load

The questionnaire answer in the experiment was used for labeling. Figure 4 shows the list of questionnaires used in this method. The questionnaire includes the items to investigate the cognitive load [13] felt by the subject while solving the task. This questionnaire is a question group based on a 10-point Likert scale. The subject answers the question sentence subjectively. Generally, if the learner does not misunderstand a task, a high cognitive load means a low understanding of the task. However, in reality, there were many contradictory in the answer to the questionnaire and the performance of the task. In other words, there were many subjects who did not perform the task well despite the small cognitive load. This suggests that some subjects misunderstood the content of task. From this, it can be said that it is difficult to evaluate understanding, especially programming thinking ability, unless the instructor watches the learner's behavior of the process of solving the task. Therefore, in the labeling, with referring to the questionnaire, the instructor give the labeling from the viewpoint of both the learning item achievement and the programming thinking based on the behavior during task.

C. Estimation of understanding and distribution of learners

Based on the operation history data collected from the 17 subjects and the labels given by the instructor, the learning item achievement and programming thinking ability were classified. Both the random forest and logistic regression are used for classification. With the random forest, high classification accuracy and variable importance can be obtained. On the other hand, with the logistic regression, variable importance based on regression coefficient and understanding by values from 0.0 to 1.0 can be obtained. For the analysis, we used Scikit-learn in Python to find the optimum parameters by the grid search and ensured the generalization performance by using the Leave-one-out cross-validation.

TABLE I
PERFORMANCE OF LEARNING ITEM ACHIEVEMENT CLASSIFICATION MODELS

Logistic regression	
Accuracy	0.82
Precision	0.78
Recall	0.88
F-score	0.82

Random forest	
Accuracy	0.82
Precision	0.86
Recall	0.75
F-score	0.80

TABLE II
PERFORMANCE OF PROGRAMMING THINKING CLASSIFICATION MODEL

Logistic regression	
Accuracy	0.82
Precision	0.79
Recall	1.00
F-score	0.88

Random forest	
Accuracy	0.94
Precision	0.92
Recall	1.00
F-score	0.96

Tables I and II show the accuracy rates of the model created by this experiment. The accuracy rates of both models exceeds 80% and it can be said that the models have a high degree of accuracy.

Additionally, the distribution of the subjects' understanding is displayed on the two-dimensional coordinates using the two axes of the classification probabilities of learning item achievement and programming thinking ability predicted by the logistic regression. The distribution map is shown in Fig. 5. The label of the distribution map is the degree of schema construction of the corresponding subject. In the distribution map, the subjects close to the diagonal line of 45 ° show that the degree of learning items achievement and the programming thinking ability are similar and that the two abilities are well balanced. Among them, the subjects located in the upper right shows that both abilities are high. On the other hand, learners distant from the 45 ° line to the lower side have a high learning item achievement but have a low programming thinking ability, which indicates a poor balance. If the schema construction degree is used, those who have high knowledge are evaluated only to some extent and those who have a balance of both values are highly evaluated.

V. Consideration

A. Correlation between schema build and grade

The validity of the schema construction level is confirmed by comparing the calculated schema construction level with the results of the actual class performance.. Table III shows the calculated schema construction level of the subjects and the total scores of the 32 fill-in-the-blank tasks that the subjects have took in the actual classes. The correlation coefficient among them is also shown. The subjects were those who were able to obtain actual class performance data among the 17 subjects. In addition, we excluded those that could not be tested due to poor physical condition and excluded weeks of tasks that were not directly related to programming.

TABLE III
CORRELATION WITH THE CALCULATED SCHEMA CONSTRUCTION LEVEL AND A LIST OF GRADES

Subject	Schema construction level	Score
A	1.922	754
B	1.887	760
C	1.885	736
D	1.510	713
E	1.502	769
F	1.016	723
G	1.001	775
H	0.430	542
I	0.215	674
	Correlation	0.67

The correlation coefficient of 0.67 cannot be said to be extremely high. However, it correlates to some extent with the credible data of scores. This shows the validity of the degree of schema construction. The reason why a high correlation does not appear is that the data of the performance of the fill-in-the-blank tasks does not consider programming thinking ability. On the other hand, the proposed method easily quantifies the real programming ability considering programming thinking ability.

B. Consideration of Understanding Factor and Learner Behavior by Important Variables

We compare the classification models of learning item achievement and programming thinking ability and consider the differences. Table IV shows the important variables of the models created in this experiment.

The subjects with low learning item achievements moved many variable blocks and loop blocks by focusing on the variables related to "touch" and "hover". From this, it is said that they did not understand how to use variables and could not answer the basic parts such as loop statements. It is considered that this is because the subject with low learning item achievement could not think the meaning of the variable name. In addition, the learning item achievement is lower when the iterator variable “i” is declared earlier and the touch frequency up to 1/4 hour is higher. This suggests that the learners who worked on the task sooner after provision of the task got confused. It is considered that they have repeatedly switched between the drawing screen and the task display

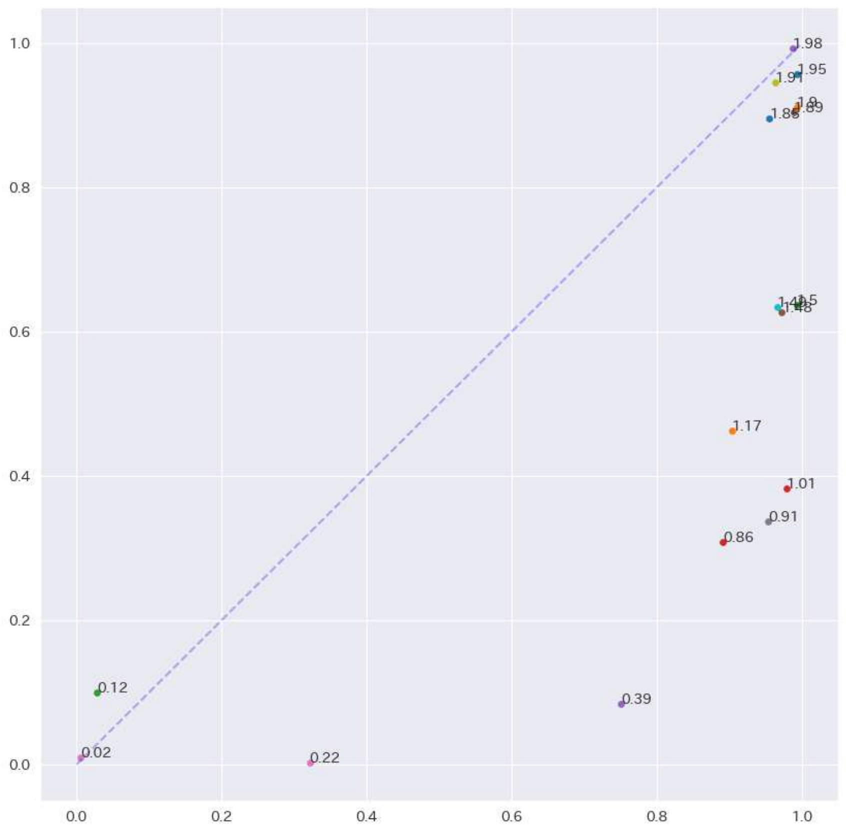


Fig. 5. Distribution chart based on learning item achievement and programming thinking ability

screen. On the other hand, the subjects with high programming thinking ability took less time to adopt the "input block". From this, it is considered they immediately grasped the meaning from the name of the variable and adopt it. Moreover, since they read the explanation of the variable especially "loop block" carefully, it can be said that they tend to think carefully how many times to loop.

In the learning item achievement level, the "one-character printing block", which is the deepest part of the loop, is important. From this, it can be seen that the subjects with low learning item achievement could not reach to think this module. On the other hand, in programming thinking, the time spent for watching the explanation of single-character printing is important. This suggests that the subjects with high programming thinking have reached the deepest part of the loop and considered it.

In addition, the variables related to the touch frequency is important in the learning item achievement. This indicates that the subjects with low learning item achievement tended to assemble modules without thinking. In programming thinking ability, the variable of touch frequency is important. In other words, it was found that the subjects without programming thinking touched many modules that were not necessary for the task. They have been at a loss. In addition, there are many important variables that indicate "looking at the module description". This shows that the subjects with high programming thinking ability firmly identified the modules to be used at first and established the course before tackling the task.

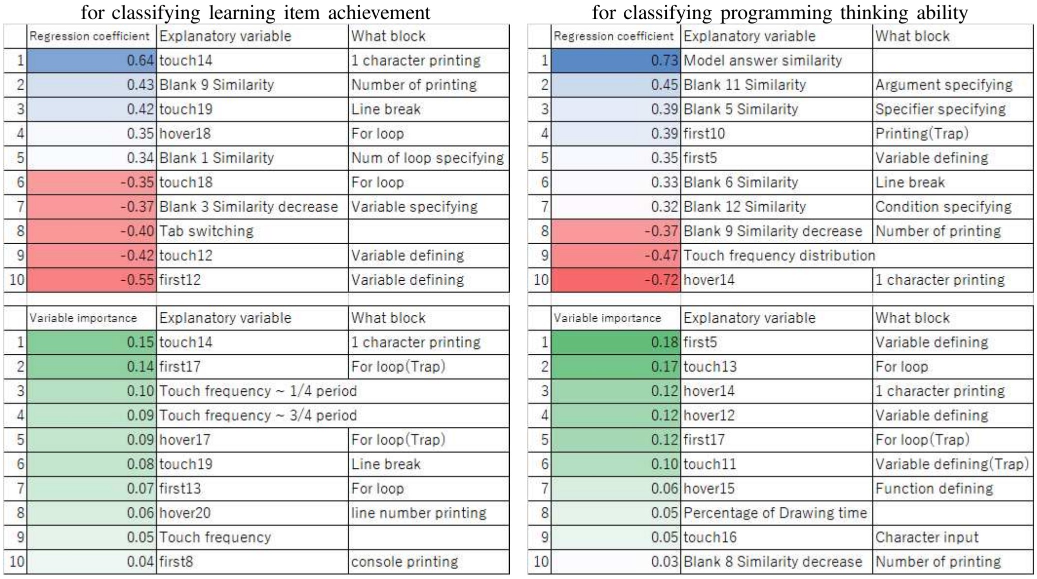
The important variables for the programming thinking ability can indicate similarity between the model and the subject's answers. This shows that an ideal answer cannot be achieved only with the learning item attainment level. It is because programming thinking is indispensable for solving code puzzles.[4]. In addition, the variable of "the time of looking at the explanation of 1-character printing" is important. This results implies that the subjects who were confused about the matters such as "1-character printing" may be greatly evaluated negatively. Moreover, since "designation of arguments and

TABLE IV
TOP 10 IMPORTANT VARIABLES BY LOGISTIC REGRESSION AND RANDOM FOREST

touch: Number of drag and drop
hover: Time spent looking at the code block description
first: How fast the code block was first adopted

[1]List of important variables
for classifying learning item achievement

[2]List of important variables
for classifying programming thinking ability



[1]

	Regression coefficient	Explanatory variable	What block
1	0.64	touch14	1 character printing
2	0.43	Blank 9 Similarity	Number of printing
3	0.42	touch19	Line break
4	0.35	hover18	For loop
5	0.34	Blank 1 Similarity	Num of loop specifying
6	-0.35	touch18	For loop
7	-0.37	Blank 3 Similarity decrease	Variable specifying
8	-0.40	Tab switching	
9	-0.42	touch12	Variable defining
10	-0.55	first12	Variable defining

	Variable importance	Explanatory variable	What block
1	0.15	touch14	1 character printing
2	0.14	first17	For loop(Trap)
3	0.10	Touch frequency ~ 1/4 period	
4	0.09	Touch frequency ~ 3/4 period	
5	0.09	hover17	For loop(Trap)
6	0.08	touch19	Line break
7	0.07	first13	For loop
8	0.06	hover20	line number printing
9	0.05	Touch frequency	
10	0.04	first8	console printing

[2]

	Regression coefficient	Explanatory variable	What block
1	0.73	Model answer similarity	
2	0.45	Blank 11 Similarity	Argument specifying
3	0.39	Blank 5 Similarity	Specifier specifying
4	0.39	first10	Printing(Trap)
5	0.35	first5	Variable defining
6	0.33	Blank 6 Similarity	Line break
7	0.32	Blank 12 Similarity	Condition specifying
8	-0.37	Blank 9 Similarity decrease	Number of printing
9	-0.47	Touch frequency distribution	
10	-0.72	hover14	1 character printing

	Variable importance	Explanatory variable	What block
1	0.18	first5	Variable defining
2	0.17	touch13	For loop
3	0.12	hover14	1 character printing
4	0.12	hover12	Variable defining
5	0.12	first17	For loop(Trap)
6	0.10	touch11	Variable defining(Trap)
7	0.06	hover15	Function defining
8	0.05	Percentage of Drawing time	
9	0.05	touch16	Character input
10	0.03	Blank 8 Similarity decrease	Number of printing

conditions" is important, it can be said that a learner with high programming thinking ability grasps the flow of data and chooses appropriate variables.

The findings about the subjects with low understanding obtained in this experiment are shown below.

Regarding the low learning item achievement, it is difficult to interpret the usage of variables and loops block and there is a tendency to get lost in basic matters. , They also tend to tackle the task immediately without interpreting the task deeply.

Regarding the low programming thinking ability, it is greatly evaluated negatively when the subjects cannot approach the model answer and eventually stumbles on a basic matter. They do not look at the explanations deeply, touch many blocks, and do not set the course for answers. In other words, they cannot think deeply. Furthermore, they cannot assemble the structure firmly. They do not understand the data flow and cannot specify arguments or conditions. From these facts, it is considered that the learning item achievement is the basic ability and the programming thinking ability is the comprehensive ability including the learning item achievement.

C. Usefulness in educational settings

The results of this experiment show that it is possible to estimate the comprehension level considering the programming thinking ability from the learner's operation history when answering the code puzzle. At present, there are many one-to-many forms in educational settings such as universities or companies. Though they sometimes hire multiple assistants to support learners, the current situation is that the number of assistant is insufficient. Therefore, it is difficult for the instructor to grasp the understanding level of all learners.

The schema construction estimated by the proposed method can measure the understanding of many learners at once if a model is learned by labeling dozens of samples. In particular, the classification probability output by the logistic regression is a value from 0.0 to 1.0. Therefore, by setting a threshold appropriately, we can discriminate between learners who can be able to solve the task and learners who need guidance.

In the field of education, it is common to learn various elements with multiple tasks. The schema construction calculated for multiple tasks over a period can be visualized by studying the transitions of learning progress and growth. Moreover, by referring to the important variables, the instructor can consider where the learner stumbled. For example, in the task of this experiment, it can be seen that many learners did not understand what the meaning of the variable has from its name because the importance of the block for the variable was high. Since the importance of variables on blocks regarding the number of loops was high, it can be seen that many learners could not fully consider the processing flow.

VI. ISSUES AND FUTURE

The problem of the proposed method is the ambiguity of blocks used in code puzzles and the cost of labeling. At first, creating blocks used in code puzzle is a large cost for creating a task in an actual educational setting. As a countermeasure, it is possible to implement an algorithm for adding similar blocks or randomly select blocks from other tasks' blocks and them.

In addition, labeling is not a small cost in the educational setting. However, this is unavoidable as long as supervised learning is used. Blikstein *et al.* [14] suggested that the understanding of learners can be measured by using the clustering method with programming structure. Therefore, in our research, there is a possibility that the understanding and the factor of misunderstanding can be estimated by labeling using such the clustering method. The model created in one year can be used in the following years as long as the tasks are the same and the level of learners is the same. In general, at one university, the levels of students are almost the same for several years. Furthermore, the distribution of the programming ability of students as described in this paper can be visualized based on the learning item achievement and programming thinking ability calculated by the model. Furthermore, there is a program that visualizes the students' behavior. With visualization tools, it is not difficult to label for multiple people. This also allows refinement of the model.

VII. Conclusion

This paper proposed a method to estimate the learner's programming thinking ability as well as learning item achievement by analyzing his/her answering process of the code puzzles.

In order to measure programming ability precisely, it is necessary to provide the learners an environment like a code puzzles where he/she can focus on combining programming elements and to extract his/her answering process and behavior. The estimation is performed by learning the random forest and logistic regression models, where the objective variables are programming thinking or learning item achievement levels, respectively, and the explanatory variables are learner's actions in solving code puzzles.

As the result of the experiment, it was found that the proposed method was able to estimate the understanding with the accuracies of more than 80In addition, considering the difference between the learning item achievement and the programing thinking ability from the difference of the variable importance, it was confirmed that the programming thinking ability is based on the learning item achievement and that the programming ability cannot be measured only by the learning item achievement. Furthermore, based on the schema theory[5], we defined the schema construction level from two labels, learning item achievement and programming thinking ability. We compared it with the performance of the fill-in-the-blank problems in actual classes. These results suggest that just the performance of the fill-in-the-blank problems does not indicate the programming ability.

The results of this experiment show that the learner's programming ability can be measured more accurately by considering the learner's logical constructive ability in the code puzzle rearrangement problem. The accurate measurement of the learner's programming ability contributes to developing the learner's true programming ability, which cannot measured by only the score of written tests. In addition, the importance of each variable in the behavior analysis leads to the identification of learner's misunderstanding factors and the improvement of class contents.

In the future, we will develop a new method to deal with more complicated programming problems and to simplify making tasks to be applied to the proposed method.

References

[1] J. T. S. P. o. C. S. B.-S. C. B. Kenneth L. Whipkey, "Identifying predictors of programming skill," *ACM SIGCSE Bulletin*, vol. 16, no. 4, pp. 36–42, 1984.

[2] L. J. M. U. of Cincinnati, "Identifying potential to acquire programming skill," *Communications of the ACM*, vol. 23, no. 1, pp. 14–17, 1980.

[3] M. of education, "Elementary programming education guide (second edition)," https://www.mext.go.jp/component/a_menu/education/micro_detail/__icsFiles/afieldfile/2018/11/06/1403162_02_1.pdf.

[4] D. Parsons and P. Haden, "Parson's programming puzzles: a fun and effective learning tool for first programming courses," in *Proceedings of the 8th Australasian Conference on Computing Education-Volume 52*, 2006, pp. 157–163.

[5] W. Schnotz and C. Kürschner, "A reconsideration of cognitive load theory," *Educational psychology review*, vol. 19, no. 4, pp. 469–508, 2007.

[6] M. C. Jadud, "Methods and tools for exploring novice compilation behaviour," in *Proceedings of the second international workshop on Computing education research*, 2006, pp. 73–84.

[7] A. Mysore and P. J. Guo, "Porta: Profiling software tutorials using operating-system-wide activity tracing," in *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, 2018, pp. 201–212.

[8] P. J. Guo, "Codeopticon: Real-time, one-to-many human tutoring for computer programming," in *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, 2015, pp. 599–608.

[9] H. S. So Asai, Dinh Thi Dong Phuong, "Identification of factors affecting cognitive load in programming learning with decision tree," vol. 14, no. 11, 2019, pp. 624–633.

[10] P. Ihantola, J. Sorva, and A. Vihavainen, "Automatically detectable indicators of programming assignment difficulty," in *Proceedings of the 15th Annual Conference on Information technology education*, 2014, pp. 33–38.

[11] Y. Futamura, T. Kawai, Y. Horikoshi, M. Tsutsumi *et al.*, "Program design and creation with pad (problem analysis diagram)," *IPSJ Transactions*, vol. 21, no. 4, pp. 259–267, 1980.

[12] J. W. Coffey, "Relationship between design and programming skills in an advanced computer programming class," *Journal of Computing Sciences in Colleges*, vol. 30, no. 5, pp. 39–45, 2015.

[13] B. B. Morrison, B. Dorn, and M. Guzdial, "Measuring cognitive load in introductory cs: adaptation of an instrument," in *Proceedings of the tenth annual conference on International computing education research*, 2014, pp. 131–138.

[14] P. Blikstein, M. Worsley, C. Piech, M. Sahami, S. Cooper, and D. Koller, "Programming pluralism: Using learning analytics to detect patterns in the learning of computer programming," *Journal of the Learning Sciences*, vol. 23, no. 4, pp. 561–599, 2014.

DOI: 10.15439/2020F184
ISSN 2300-5963 ACSIS, Vol. 21

Cluster-based approach for successful solving real-world vehicle routing problems

Emir Žunić
Info Studio d.o.o. Sarajevo and Faculty of Electrical Engineering, University of Sarajevo, B&H
emir.zunic@infostudio.ba

Dženana Đonko
Faculty of Electrical Engineering, University of Sarajevo, Bosnia and Herzegovina
ddonko@etf.unsa.ba

Haris Šupić
Faculty of Electrical Engineering, University of Sarajevo, Bosnia and Herzegovina
hsupic@etf.unsa.ba

Sead Delalić
Faculty of Science, University of Sarajevo and Info Studio d.o.o. Sarajevo, B&H
delalic.sead@pmf.unsa.ba

Abstract—**Vehicle routing problem as the generalization of the Travelling Salesman Problem (TSP) is one of the most studied optimization problems. Industry itself pays special attention to this problem, since transportation is one of the most crucial segments in supplying goods. This paper presents an innovative cluster-based approach for the successful solving of real-world vehicle routing problems that can involve extremely complex VRP problems with many customers needing to be served. The validation of the entire approach was based on the real data of a distribution company, with transport savings being in a range of 10-20 %. At the same time, the transportation routes are completely feasible, satisfying all the realistic constraints and conditions.**

Keywords—Clustering, Transport, Vehicle Routing Problem, Feasible solutions, Multi-phase approach

I. Introduction

VEHICLE routing problem (VRP) is the generalization of the problem of the commercial traveler, and is basically the process of selecting the set of the most appropriate roads and routes for the vehicles from the available fleet of vehicles during the serving of the set of customers (delivery points), resulting in the lowest total cost of delivery of goods and properties. The vehicle routing problem is one of the most studied problems in the scientific and academic community, with constant attempts of implementing more powerful and advanced approaches for solving this problem. In addition, the industry itself pays a lot of attention to this problem, and the reason for this is the potential reduction of the delivery costs and the possibility of the significant financial savings in the process. The central problem in the logistics of one company is the optimization of transport and the management of the transport fleet of vehicles in an optimal way. If the solution of this problem is left to the computer, there are multiple benefits: the obtained solution is optimal, more efficient and financially favorable, and at the same time spends less time and resources on planning and organizing the transport of goods. However, nowadays, many companies use experimental methods more often during the optimization of the transport of goods compared to some modern approaches and algorithms, and the main reason is that there are very few practically applicable solutions that are able to successfully solve the complex VRP problems, which can often be non-standard, while satisfying all the realistic constraints.

On the other hand, with the progress of logistics processes in the early 1950s [1], there has been a lot of research focused on their various applications. With the globalization of this process, the importance of logistics management has significantly grown in the last few years. Logistics tries to optimize existing distribution processes. One of the most important elements in the logistics chain is the transportation system. According to numerous studies, transportation presents one third of the total logistics cost, and transportation systems significantly affect the performances of the complete logistics system. Transportation is necessary in the complete process of production of goods, from the very production to delivery of goods to the final customers. It is possible to get the maximum benefit for distributors and manufacturers only with good coordination among all the components. Logistics planning cannot reach its full potential without a well developed transportation system. Therefore, good transportation systems can increase the efficiency, reduce the operating costs and increase the service quality. The success of solving VRP problems can significantly improve processes in the transportation part of business of each company. Variants of VRP problems differ according to the number of depots (one or more of them), maximum allowed duration or the length of the vehicle route, different vehicle capacities, customers' requests for delivery or collection of certain amounts of goods during service and time windows within which it is necessary to start and finish customer service, as well as time windows (working hours) of vehicles, illustrated in Fig. 1. In a realistic environment, it is important to take into consideration a great number of additional constraints, usually being the result of the specific loading locations and/ or unloading, specific business processes of the distribution company or legal decisions and obligations (e.g. compulsory rest breaks for drivers). These problems are called Rich Vehicle Routing Problems – RVRP.

This work was supported by Info Studio d.o.o. Sarajevo

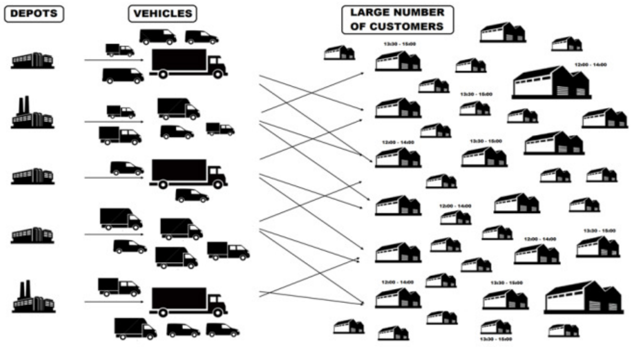


Fig. 1 Vehicle Routing Problem

Transport managers with long-term experience consider an idea to create the transport routes in complex realistic situations primarily from independent clusters joined by the vehicles from the available fleet. Transport routes are obtained by most distribution companies in this way. Most of the available software solutions trying to successfully solve this problem with distribution companies work on the same principle of grouping customers into the regions (clusters), and it is possible to connect several neighboring regions into the same route. This approach usually gives feasible transport routes which is the most important fact from the aspect of the practical application of such an algorithm. For that reason, the obtained routes are very logical and understandable, easily feasible and meet the needs of any of the companies that offer the transportation of goods in their activities, with the special emphasis on logistics. The motivation for the development of the cluster-based algorithm was precisely this: being able to solve the VRP problem while accepting the given real constraints and conditions.

The literature used for this work is presented in Section 2, with the emphasis on the most powerful algorithms for successfully solving real VRP problems. Section 3 presents the algorithm for successfully solving VRP problems based on the clusters (regions), while the obtained results on the realistic set of instances used for the validation can be seen in the Section 4. Conclusions of the work, as well as the guidelines for future research in this area are mentioned in Section 5.

II. Related Work

During the analysis of the available literature, systems made for application in real environments were considered very interesting. These problems were mainly solved with heterogenic fleets of vehicles, also belong into the HF problems. OR/MS Today, published in June 2006, contains an overview of 17 advanced commercial software solutions capable of solving the instances for more than 1000 locations (customers), and creating 50 and more routes in less than two hours [2]. For the practical application, the vehicle routing is one of the most difficult problems of operational research. For example, every day more than 100.000 UPS (United Parcel Service, global mail delivery company) drivers follow the computer - generated routes and deliver on average 15.6 million packages.

VRP problems with the included restrictions are of a great importance. The most outstanding constraint in the vehicle routing process are time windows. The time window represents the time interval during the day when it is allowed to visit the customer and deliver him the articles. In Work [3], the authors present the mathematical model for the vehicle routing problem with time windows (VRPTW), a version of the VRP problem suitable for delivery planning in towns with delivery time restrictions. However, many towns in the world limit the delivery time for central areas in the morning hours. A model with the exact solution for a smaller number of instances is used in this work and it compares the performances with the larger instances on a modified genetic algorithm, as well as the process of Taboo search. The results do not show the advantage of any of the algorithms, but confirm that the delivery cost has been significantly increased by introducing these conditions. The modified memetic algorithm for finding the minimal distance in vehicle routing problem with the time windows is presented in work [4]. Various metaheuristic approaches for solving VRPTW problems have been described in work [5], such as the Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) algorithm and Artificial Bee Colony (ABC) algorithm. A hybrid algorithm combining ACO and the Firefly Algorithm (FA) algorithm was presented in work [6]. The VRPTW problem was observed in detail as a part of this paper. An advanced simulated quenching algorithm was described and applied to VRPTW problem in work [7].

Nowadays, the vehicle routing problem is very important in the scientific community, as it is stated in work [8]. This work describes the research status of this scientific field, including different variants of vehicle routing problem. This paper focuses on the application of nature - inspired algorithms and the most popular classical approaches. Implementations of many algorithms have been described and compared: Genetic algorithm (GA), Taboo search, Simulated quenching (SA), Particle Swarm Optimization (PSO), Bee Colony Optimization (ABC), ant Colony optimization (ACO), Ant Colony Optimization (ACO), Cuckoo Search, Imperialist Competitive Algorithm (ICA), Bat Algorithm (BA) and Firefly Algorithm (FA). Each of these algorithms includes several parameters with their settings determining the result and the quality of the obtained routes.

Several papers from the available literature dealing with the practical realization of VRP problems basically use an algorithm and a principle described in detail and suggested in work [1]. In this work, the authors include the mathematical formulation of the problem, which is easy to understand and can be adapted to the practical problem, and the very way of finding the optimal routes is quite intuitive. The authors propose the clustering (grouping customers into the clusters) to be included in the first place. Clustering is based on the geographical characteristics of the customer, time constraints, and the constraints in the form of quantity of goods to be delivered. This work was the starting point for the approach presented later in this paper, with significant improvements in both performance and algorithm terms. It was also used as the integral part of the algorithms proposed in works [10]-[16]. All these facts indicate that the cluster-based approach is exceptionally suitable for solving such problems, both as an individual algorithm, as well as the starting point of the more complex multiphase algorithms.

III. Proposed Cluster-Based Approach

The key element in the supply chain is the transportation system which unites different activities, spatially and temporally divided. Transport includes one third of the logistic costs, and significantly affects the performances of the logistic system. Distribution companies are usually not able to optimize their transportation activities in the most efficient way, and thus lose significant financial resources. Complex VRP problems are divided into mega-clusters (regions) in their initial phase being geographically remote from the depot. In this way, division into logical geographic regions is primarily done, with the best solution being found there; in fact, the vehicles are assigned for the purpose of serving each of the regions, respecting all the constraints.

The first phase in the entire approach is to make mega-clusters for all the customers needed to be served. Mega clusters are formed depending on the distance of their centre from the depot. The initial creation of the mega clusters can be attained by using the simplest clustering methods, such as *k-Means*, *k-Medoids* or hierarchical techniques of clustering. In this way, the set of several hundred (or even thousands) customers can be divided into several remote clusters, which significantly reduce the complexity of the problem in the next phases of the proposed approach.

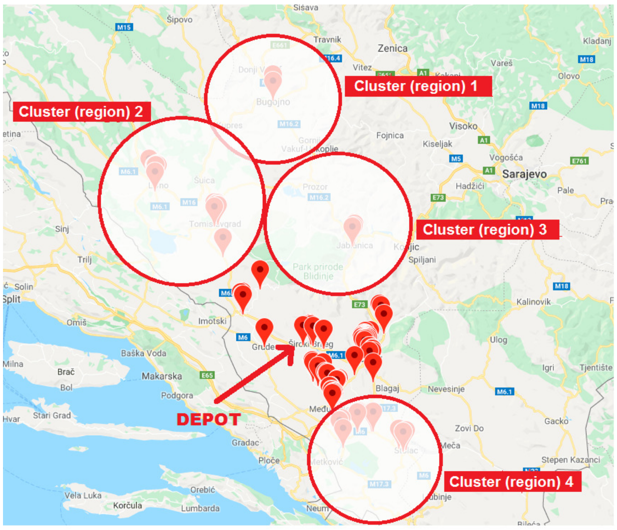


Fig. 2 Customer division in geographically divided regions (clusters)

In the example shown in Fig. 2., special vehicles from the available vehicle fleet during the creation of the transport routes are used for each of the individual clusters and they cannot be combined with each other. Customers who do not belong to any of the clusters are optimized by using the remaining vehicles in the next phase of the approach, and can be combined with each other into one or more transport routes. Usually, the customers around the depot are without any clusters, while the remote customers belong to specific clusters, which is usually the approach being the most acceptable in practice. If there are not enough customers to fulfil the most efficient vehicle in the specific cluster, customers who do not belong to any of the clusters and are in the right direction, can be joined, too.

The number of clusters is determined in the way that within each cluster there has to be minimum: (*Number of customers of the cluster* / *Total number of customers*) ≥ [(1 / *Number of remaining vehicles in the fleet*)] · 100%. The number of clusters described in this way should not exceed the number of available vehicles in the fleet. If this is the case, the smaller, usually neighboring clusters are merged into bigger ones (regions). If there is not any possibility of cluster merging due to the ordered capacity of goods compared to the dimensions and limitations of the available vehicles, then additional (fictive) vehicles are introduced, with a significantly higher delivery cost. Every company can rent additional vehicles from outside companies dealing the delivery of goods as their primary activity.

In order to clarify the proposed approach in mathematical terms, the routing network is being considered and presented by the directed graph $G\{I, P, A\}$ which connects the nodes presenting the customers $I=\{i_1, i_2, \ldots, i_n\}$ and the nodes presenting the depot through the set of directed branches $A=\{(i, j)|(i, j)\in(I\cup P)\}$ It is assumed that the branch $(i,j)\in A$ is the route with the lowest cost which merges the nodes i and j. At the location of each customer $i\in I$, a fixed load should be delivered w_i, vol_i within the time window $[a_i, b_i]$, where a_i is the earliest time serving the customers should start, and b_i is the latest time serving should finish.

A fleet of heterogeneous vehicles $V=\{v_1, v_2, \ldots, v_m\}$ with different cargo capacities (q_v, q_v^V), located in several depots, $p\in P$ is used for delivery. Each vehicle v has to leave the assigned depot $p\in P$, deliver the goods to several destinations and come back into the same depot p. Thus, vehicle route v is the route consisting of knots $r=(p, \ldots, i, i+1, \ldots, p)$ connected by directional branches belonging to set A, and each route starts and finishes in depot p assigned to the vehicle v. A pair of matrices depending on the vehicles $C=\{c_{ij}^{\ v}\}$ and $\Gamma=\{t_{ij}^{\ v}\}$ is assigned to the set of branches $a_{ij}\in A$. They indicate the cost and the time of the travel from the node i into the node j using the vehicle v, respectively. It is assumed that the inequality of the triangle is satisfied by the elements c_{ij} and t_{ij}, actually it is: $c_{ik}+c_{kj} \geq c_{ij}$ and $t_{ik}+t_{kj} \geq t_{ij}$. The total required mass and the volume of the goods

(w_i,vol_i), as well as the serving time by the vehicle v (st_i^v) are given for the node i.

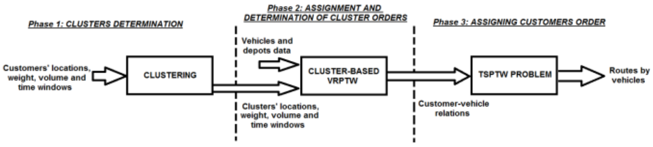

Phase 1: CLUSTERS DETERMINATION
Phase 2: ASSIGNMENT AND DETERMINATION OF CLUSTER ORDERS
Phase 3: ASSIGNING CUSTOMERS ORDER
Customers' locations, weight, volume and time windows
CLUSTERING
Vehicles and depots data
CLUSTER-BASED VRPTW
Clusters' locations, weight, volume and time windows
TSPTW PROBLEM
Customer-vehicle relations
Routes by vehicles

Fig. 3 Phases and the components of the hierarchical hybrid approach

The solution of this problem has to fulfill the following constraints: (i) each route must start and end in the same depot; (ii) each node must be serviced by one and only one vehicle; (iii) the node must not be served by a vehicle with the constraint that it cannot visit that node; (iv) the total load assigned to vehicle v must never exceed its capacity q_v and q_v^V; (v) the length of time vehicle v is active should be less than the maximum allowed length of working time tv_v^{max}; (vi) the customer i service must be completed within the interval [a_i, b_i], otherwise penalties will have to be paid. The main aim of the problem is to minimize the total cost of servicing for all the customers. There are four types of costs being considered to fulfill the aim: fixed costs for used vehicles, distance-related costs, costs related to the travelling time along the selected routes, costs related to the waiting time, penalty costs for disrupting time windows and total working hours.

The previously mentioned algorithm [9], which was originally improved, modified and tested over the input data set, combines the clustering methods into the optimization framework, illustrated in Fig. 3.

The algorithm is based on the traditional *first cluster then route* philosophy. Node clusters are firstly defined; then, such clusters are assigned to the vehicles and put in order on the assigned routes, while the routing and the scheduling are made particularly for each of the individual route in terms of the original nodes. A three-phase VRPTW hierarchical hybrid approach is defined in this phase of the total proposed approach.

Finding a good cluster set, with each of the clusters consisting of several customers, without having any information about the routes is a very difficult assignment. Therefore, a heuristic algorithm based on time windows is used for efficiently arranging the customers into a small number of the possible clusters. This clustering procedure leads to the compact version of VRPTW formulation presented below, by replacing the nodes with the clusters. The clustering procedure and the compact VRPTW model are basic building blocks of the proposed hybrid approach. After grouping the customers into the clusters during the Phase 1, the VRPTW problem of a small dimension is being solved and formulated in terms of clusters rather than nodes. This solution enables clusters to be assigned to the vehicles and to construct the routes by collecting the clusters into the same route (Phase 2). The detailed routing for each of the routes found in Phase 2 is made in the last phase. In Phase 3, the number of routing problems are solved for the number of visitors that have to visit the routes. The aim is to solve the more compact form of the basic VRPTW formulation. The model takes into consideration only the nodes within the cluster of the route being analyzed. As the order of visiting the clusters has already been determined in Phase 2, the relative order among the nodes belonging to different clusters is already familiar, as well as the values of numerous variables S_{ij}. This causes a significant reduction of the number of binary variables and constraints in the model solved in Phase 3.

i) Heuristic clustering algorithm (Phase 1)

The aim of Phase 1 is to significantly decrease the calculations for the later phases. This aim is accomplished by defining the small set of feasible clusters or “hyper nodes”, where each of them include several customers, and then by establishing the approximate distances and travel time between any of the two clusters. As the mathematical model contains several clusters, and not a large number of customers, dimensionality of the VRPTW problem can be significantly decreased. In order to find a good set of clusters for big problems, the following heuristic procedure is used:

1) a) Open the list of nodes L, and sort them according to the ascending order of time a_i. If some of the nodes have the same a_i, sort them in ascending order of time b_i.

b) Open the list of the available vehicles V, and sort them according to the descending order (q_v/cf_v).

c) Select the maximum allowable distance between any of the two nodes within the same cluster (d^{max}) and the maximum allowable waiting time Δ.

2) (n-th main iteration) Open the empty list K_n related to the next cluster to be created C_n. Add the first term from the list V to the cluster C_n and delete it from V.

3) (a) take the first node i from the list L that can be serviced by the vehicle assigned to the cluster C_n and put it on the bottom of the list K_n. Initialize the cluster parameters C_n:

$$aC_n \leftarrow a_i \qquad bC_n \leftarrow b_i$$
$$wC_n \leftarrow w_i \quad volC_n \leftarrow vol_i$$
$$\mathrm{st}C_n \leftarrow \mathrm{st}_i$$

(b) Delete node i from the L list and make a copy of the current list L, and name it L'.

4) Take the first node j from the list L', and verify: (i) current cargo mass that should be delivered to the cluster C_n plus w_j does not exceed the mass cargo capacity q_v of the assigned vehicle v; (ii) the current cargo volume that should be delivered to the cluster C_n plus vol_j does not exceed the volume cargo capacity of the assigned vehicle v; (iii) the node j can be serviced by the vehicle v. If any of these three items is not completed, delete node j from the list L' and repeat the procedure (4). Otherwise, proceed to step (5).

5) (a) Calculate the distance d_{ji} between node j and its closest node i in the K_n list.

(b) Verify d_{ji} to be less than the maximum allowed distance d^{max}. If it is not, delete node j from the current list L' and turn back to the step (4). Otherwise, proceed to step (6).

6) Verify to satisfy the following constraint:

$$aC_n + \mathrm{st}C_n + t_{ij}^v + \mathrm{st}_j \le \max\left(bC_n, b_j\right)$$

If it is not, delete node j from the current list L' and turn back to the step (4). Otherwise, proceed to step (7).

7) Verify that the following constraint is satisfied:

$$aC_n + \mathrm{st}C_n + t_{ij}^v + \Delta \ge a_j$$

If it is not, close the cluster C_n deleting the current list L and recording the list K_n which defines the cluster C_n, and turn back to the step (4). Otherwise, proceed to step (8).

8) (a) Put node j at the bottom of the K_n list and update the parameters of the cluster C_n as follows:

$$wC_n \leftarrow wC_n + w_j \qquad volC_n \leftarrow volC_n + vol_j$$

$$\mathrm{st}C_n \leftarrow \max\left(\mathrm{st}C_n + t_{ij}^v + \mathrm{st}_j, a_j + \mathrm{st}_j - a_i\right)$$

(b) If $bC_n < b_j$, then: $bC_n \leftarrow b_j$.

Otherwise, the latest finishing time of the service bC_n is unchanged. Delete node j from the lists L and L' and proceed to the following step (9).

9) If the L' list is empty, keep the K_n list that defines the cluster C_n and proceed to the step (10). Otherwise, turn back to the step (4).

10) Repeat steps 2-9 until list L is empty.

11) Calculate the time and road distance between any of two clusters defined by the algorithm. The distance between the two clusters can be calculated as the shortest distance between the two nodes belonging to those clusters, or as the average distance among all of their nodes, or like Max-Min distance.

The input into the procedure are the set of nodes (customers) I, the set of vehicles V, road and time distances among the nodes, serving time, the quantity of goods to be served and time windows. The aim of the procedure is to identify the set of feasible clusters that include the customers, where "the feasible cluster" means: (a) The load of the cluster can be assigned to a single vehicle; b) there is a route that connects the clusters and at the same time meets all the constraints of the time windows. The set of clusters that will be formed should be effective in the sense that (c) the vehicle waiting time due to early arrival has to be as short as possible and (d) an average distance per node inside the cluster for the assigned vehicle should remain low. In order to achieve these goals, the list of L nodes is appropriately arranged in step (1) to enable the generation of the feasible low-cost clusters. Before adding the next node into the cluster being generated, its proximity with other nodes in the cluster is tested, as well as the time and capacity constraints (steps 4-7). The maximum resting time of Δ is allowed for the early arrival at a customer's location. If it is exceeded, integration of the node into the cluster is rejected (step 7).

In order to define the mathematical formulation using the term "cluster" and not the node, the terms "cluster time window" and "cluster service time" are introduced. The earliest time cluster C_n servicing can start is given with $\min_{i \in Cn}(a_i)$, while the latest time the servicing of $C_n(bC_n)$ has to be finished is updated by taking the maximum of (bC_n, b_i), where bC_n denotes the current time the cluster C_n service has to be completed, and b_i represents the appropriate value for the new input into the cluster C_n. Whenever a new node is added, the parameters of the cluster time window $[aC_n, bC_n]$ are updated. On the other hand, cluster stC_n service time is a good approximation of the total time the assigned vehicle spends visiting the cluster C_n. This time does not only include the service time of the assigned nodes, but also the travelling and resting time along the cluster.

As the fleet size is the variable in this problem, the procedure must choose the most efficient vehicles in order to complete a delivery process. This can be attained by sorting the vehicles, as is done in step (2) according to ascending value of cost-effectiveness. Finally, the time and road distance between any of the two clusters is determined in step (11). It should be noted that an arbitrary number of clusters may occur as a result of Phase 1.

One vehicle is assigned for each of the clusters. As the number of clusters can be higher than the number of available vehicles, the vehicle list is expanded with an arbitrary number of virtual vehicles. These vehicles are only used for completing Phase 1. After the clusters are formed, virtual vehicles are removed from the list, while the available ones are left in it. Such reduced list is used as an input into Phase 2.

ii) Cluster-based multi-depot HFTW problem (Phase 2)

The aim of Phase 2 is to assign the clusters to the vehicles and to determine the order of those belonging to the same route by solving the compact version of MILP model, presented at the beginning of this section.

After completing Phase 2: (i) customers are assigned to the vehicles via the clusters; (ii) used vehicles are assigned to the depots; (iii) an almost optimal set of cluster-based routes is revealed; (iv) the order of cluster visiting belonging to the same route is obtained, which indirectly gives a partial arrangement of the order of customer visits serviced by the same vehicle.

iii) Scheduling of one route (Phase 3)

Scheduling the nodes inside the cluster and determining the starting time of the customer service for each of the routes is the aim of Phase 3. In order to achieve this, the VRPTW problem has to be formulated in the previous chapter as many times as there are routes in Phase 2. Including only the nodes from the clusters related to the same route, TSP formulation can be done for each of the routes. The relative order among clusters found in Phase 2 allows the further reduction of the number of variables of the order S_{ij}.

IV. Results discussion

The proposed three-phase algorithm was tested on real data from one of the largest distribution companies in Bosnia and Herzegovina. The tested data that will be listed below are available at 4TU Research Data, research data centre [17], in order to be available to other scientists for their research and possible comparisons of the results. Because of the clustering mechanism and the division of the initial set of customers into the smaller regions, the proposed approach proved to be very efficient for bigger problems.

Results are listed below, presenting one instance of the problem for which it was necessary to find the optimal visits of 237 customers from one central depot, with an very heterogeneous fleet made of 16 vehicles. Vehicles and the depot had precisely defined working hours; customers had their appropriate time windows, including the additional constraints such as SDVRP (Site Dependent Vehicle Routing Problem), constraints where the customers, according to the ordered mass of goods, are serviced before because of a real cost reduction in this way.

In the initial phase, the proposed approach divided the input set of customers into 6 clusters, presented in different colours (except the red one) in Fig. 4. Geographically, they can be observed as separate units.

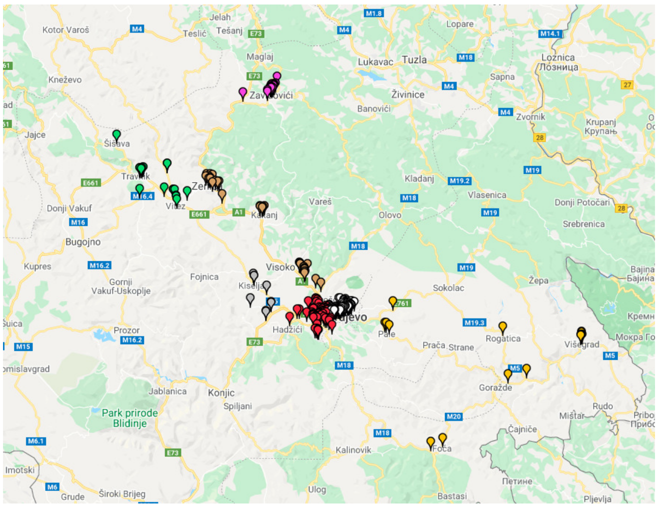


Fig. 4 Graphical presentation of cluster forming for the realistic example

Customers marked with red do not belong to any of the clusters; they are located near the given depot, and they are potential candidates for completing the routes related to the previously defined 6 clusters. It is performed in Phase 2 of the proposed approach, while in Phase 3 the optimal order within each of the created clusters is determined.

Figure 5 shows the obtained transport route for one vehicle combining the cluster presented in yellow in Fig. 4 (customers from 1 to 15 in Fig. 5), with the customers gathered around the depot, not initially belonging to any of the clusters (customers from 16 to 24 in Fig. 5).

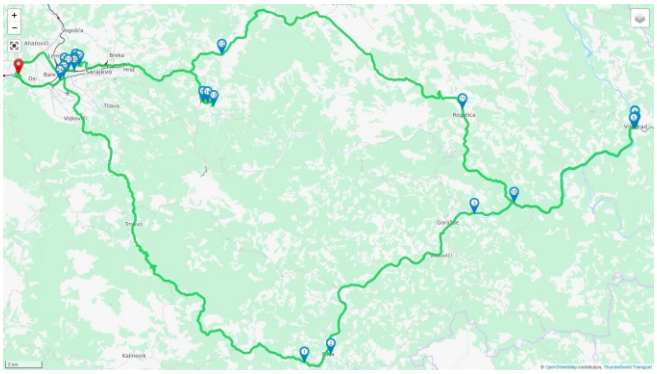


Fig. 5 A transportation route for the vehicle: combination of more clusters

Order of customer visits for the mentioned testing example presented in Table I.

Table I - Testing results on realistic company data: order of customer visits

Vehicle	Customers visits order (customers are marked with their corresponding codes) * The code 1000 indicates a depot, which means that each route begins and ends at the depot
A08-J-523	1000, 140513, 923425, 5231, 4763, 921850, 5992, 5383, 923487, 920355, 5274, 142963, 923509, 923482, 923414, 5382, 941071, 144566, 139746, 141055, 142571, 5172, 5934, 920263, 923781, 923687, 5029, 143352, 137557, 923002, 1000
A09-J-051	1000, 131821, 940895, 4302, 5971, 139518, 941079, 922465, 923485, 941068, 133081, 139095, 15297, 135665, 923504, 921685, 135952, 134436, 134757, 941076, 133857, 922629, 923474, 142730, 922545, 5281, 5711, 133301, 4421, 15194, 1000
E03-O-502	1000, 133584, 131380, 923857, 921044, 140232, 921804, 923460, 140231, 924891, 137608, 923453, 132979, 131423, 5599, 4334, 15266, 7624, 131422, 1000
E57-A-655	1000, 923508, 5380, 922377, 135790, 133202, 5270, 5716, 15077, 139770, 5387, 923491, 144832, 5272, 5384, 137825, 921026, 1000
E81-M-660	1000, 923704, 139623, 920779, 5922, 141457, 15319, 5379, 135146, 140851, 140594, 941290, 1000
E85-T-014	1000, 138152, 138320, 924238, 133238, 132293, 924186, 134190, 137739, 144596, 136433, 138508, 139038, 923370, 15082, 143746, 143101, 5285, 923502, 15074, 144879, 136655, 133672, 921045, 920279, 1000
E90-J-092	1000, 924298, 7699, 5984, 924542, 144765, 923479, 923364, 924442, 923498, 923490, 5824, 921334, 5712, 1000
J37-M-937	1000, 15146, 144419, 5710, 139499, 5999, 140671, 143456, 134865, 142725, 133927, 941289, 921902, 921438, 923287, 920952, 138999, 143622, 138468, 140937, 133984, 143422, 920264, 1000
J67-T-730	1000, 4431, 5908, 134972, 15043, 143689, 133561, 4981, 921039, 924531, 136232, 142935, 136638, 923488, 922775, 4344, 923511, 941096, 923493, 941092, 140079, 1000
M04-A-376	1000, 137617, 131450, 920108, 15155, 921541, 920152, 139039, 137177, 5113, 5512, 922425, 139564, 133867, 10220, 15154, 143513, 924012, 1000
M80-T-354	1000, 920604, 923286, 923514, 923771, 923521, 1000
O40-K-191	1000, 924594, 133725, 1000
O40-K-192	1000, 920286, 1000
O78-K-073	1000, 941319, 922313, 5436, 923483, 5277, 941069, 923471, 138126, 5264, 923884, 15287, 138138, 139378, 921261, 1000
T34-E-701	1000, 15133, 5164, 922607, 940840, 921579, 143442, 131671, 923495, 143680, 5434, 5822, 923500, 923480, 138219, 131636, 141110, 1000

Customers' arrival times for the routes presented in Table I are respectively presented in Table II. Time windows of each of the customers are shown in brackets. First and last times in Table II show the departure time of the vehicles from the depot, as well as the return period of the vehicle to the depot.

TABLE II - TESTING RESULTS ON REALISTIC COMPANY DATA: ORDER OF CUSTOMER VISITS

Vehicle (distance [km])	**Customers visits times * Each route starts and ends at the depot - the first and last time for each vehicle**
A08-J-523 (114.802 km)	07:44, 08:01-08:15 (- 19:01), 08:19-08:34 (- 19:01), 08:37-08:52 (- 19:01), 08:55-09:16 (- 19:01), 09:17-09:33 (- 19:01), 09:34-10:00 (08:00 - 17:00), 10:01-10:21 (08:00 - 17:00), 10:21-10:41 (- 19:01), 10:43-11:01 (- 19:01), 11:02-11:18 (08:00 - 17:00), 11:18-11:35 (- 19:01), 11:36-11:54 (- 19:01), 11:56-12:13 (- 19:01), 12:13-12:34 (- 19:01), 12:35-12:50 (08:00 - 17:00), 12:56-13:12 (- 19:01), 13:13-13:29 (- 19:01), 13:33-13:53 (- 19:01), 14:03-14:18 (- 19:01), 14:20-14:36 (- 19:01), 14:50-15:06 (- 19:01), 15:13-15:32 (08:00 - 17:00), 15:33-15:50 (08:00 - 18:00), 15:50-16:07 (- 19:01), 16:17-16:33 (- 19:01), 16:41-16:56 (- 19:01), 16:56-17:11 (- 19:01), 17:15-17:30 (- 19:01), 17:37-17:52 (- 19:01), 18:19
A09-J-051 (53.249 km)	07:41, 08:00-08:36 (08:00 - 17:00), 08:37-08:52 (- 19:01), 08:56-09:14 (- 19:01), 09:16-09:38 (08:00 - 17:00), 09:39-09:55 (08:00 - 19:00), 09:56-10:11 (- 19:01), 10:11-10:26 (- 19:01), 10:26-10:41 (- 19:01), 10:42-11:01 (- 19:01), 11:05-11:21 (08:00 - 19:00), 11:21-11:36 (- 19:01), 11:40-11:55 (08:00 - 17:00), 11:58-12:14 (- 19:01), 12:14-12:30 (- 19:01), 12:30-12:50 (08:00 - 18:00), 12:50-13:05 (- 19:01), 13:06-13:22 (- 19:01), 13:22-13:38 (08:00 - 17:00), 13:40-13:55 (- 19:01), 13:57-14:13 (- 19:01), 14:14-14:42 (- 19:01), 14:42-15:03 (- 19:01), 15:04-15:20 (08:00 - 17:00), 15:24-15:38 (- 19:01), 15:39-15:57 (08:00 - 17:00), 16:00-16:15 (08:00 - 17:00), 16:16-16:41 (08:00 - 18:00), 16:43-16:59 (- 19:01), 17:00-17:14 (08:00 - 19:00), 17:32
E03-O-502 (265.176 km)	06:55, 08:01-08:18 (- 19:01), 08:20-08:35 (08:00 - 17:00), 08:36-08:51 (- 19:01), 09:11-09:25 (- 19:01), 09:40-09:55 (- 19:01), 09:56-10:10 (- 19:01), 10:13-10:28 (- 19:01), 10:29-10:44 (- 19:01), 11:12-11:35 (- 19:01), 12:04-12:19 (- 19:01), 12:19-12:34 (- 19:01), 12:50-13:06 (- 19:01), 13:06-13:23 (08:00 - 17:00), 13:33-13:50 (- 19:01), 13:54-14:12 (- 19:01), 14:12-14:29 (08:00 - 17:00), 14:30-14:49 (- 19:01), 14:55-15:10 (08:00 - 17:00), 16:15
E57-A-655 (44.618 km)	07:53, 08:00-08:17 (- 19:01), 08:29-08:45 (08:00 - 17:00), 08:45-09:01 (08:00 - 17:00), 09:03-09:20 (- 19:01), 09:28-09:59 (08:00 - 18:00), 10:03-10:19 (08:00 - 17:00), 10:20-10:35 (08:00 - 17:00), 10:38-11:04 (08:00 - 17:00), 11:08-11:32 (- 19:01), 11:47-12:04 (08:00 - 17:00), 12:05-12:23 (- 19:01), 12:23-12:44 (08:00 - 18:00), 12:46-13:04 (08:00 - 17:00), 13:04-13:22 (08:00 - 17:00), 13:23-13:50 (- 19:01), 13:55-14:15 (08:00 - 18:00), 14:27
E81-M-660 (31.868 km)	07:49, 08:00-08:19 (- 19:01), 08:27-08:44 (- 19:01), 08:44-09:07 (08:00 - 18:00), 09:07-09:37 (08:00 - 17:00), 09:40-10:15 (08:00 - 18:00), 10:16-10:40 (08:00 - 17:00), 10:41-11:02 (07:00 - 18:00), 11:04-11:21 (- 19:01), 11:22-11:38 (08:00 - 17:00), 11:41-11:58 (08:00 - 18:00), 12:00-12:17 (- 19:01), 12:29
E85-T-014 (301.048 km)	06:38, 08:01-08:16 (- 19:01), 08:21-08:36 (- 19:01), 09:17-09:32 (- 19:01), 10:06-10:21 (- 19:01), 10:21-10:36 (- 19:01), 10:38-10:59 (- 19:01), 11:00-11:20 (- 19:01), 11:20-11:39 (08:00 - 18:00), 11:39-12:08 (08:00 - 18:00), 12:37-12:52 (- 19:01), 13:11-13:26 (- 19:01), 14:15-14:32 (- 19:01), 14:47-15:02 (- 19:01), 15:06-15:27 (08:00 - 17:00), 15:29-15:45 (- 19:01), 16:11-16:26 (- 19:01), 16:28-16:47 (08:00 - 17:00), 16:48-17:04 (- 19:01), 17:04-17:24 (- 19:01), 17:24-17:44 (08:00 - 18:00), 17:46-18:01 (- 19:01), 18:02-18:18 (- 19:01), 18:20-18:36 (- 19:01), 18:37-18:52 (- 19:01), 19:01
E90-J-092 (163.846 km)	07:03, 08:01-08:51 (- 19:01), 08:52-09:13 (- 19:01), 09:20-10:10 (08:00 - 18:00), 10:15-10:31 (- 19:01), 10:33-10:50 (- 19:01), 10:52-11:07 (- 19:01), 11:10-12:00 (- 19:01), 12:02-12:18 (- 19:01), 12:20-12:35 (- 19:01), 12:39-12:54 (- 19:01), 12:55-13:34 (08:00 - 17:00), 13:38-13:53 (08:00 - 17:00), 14:03-14:22 (08:00 - 17:00), 15:16
J37-M-937 (41.36 km)	07:45, 08:00-08:15 (- 19:01), 08:16-08:38 (08:00 - 17:00), 08:39-08:55 (08:00 - 17:00), 08:55-09:11 (- 19:01), 09:12-09:31 (08:00 - 17:00), 09:32-09:47 (- 19:01), 09:47-10:03 (08:00 - 18:00), 10:05-10:21 (- 19:01), 10:21-10:37 (- 19:01), 10:37-10:55 (- 19:01), 11:00-11:15 (- 19:01), 11:16-11:32 (- 19:01), 11:32-11:47 (- 19:01), 11:48-12:04 (- 19:01), 12:06-12:27 (08:00 - 17:00), 12:28-12:45 (- 19:01), 12:46-13:22 (08:00 - 17:00), 13:23-13:41 (08:00 - 17:00), 13:42-13:57 (- 19:01), 14:00-14:15 (- 19:01), 14:16-14:32 (08:00 - 18:00), 14:35-15:25 (08:00 - 18:00), 15:40
J67-T-730 (113.851 km)	07:24, 08:01-08:16 (- 19:01), 08:17-08:41 (- 19:01), 08:42-09:42 (08:00 - 17:00), 09:43-10:00 (08:00 - 18:00), 10:01-10:16 (08:00 - 17:00), 10:34-10:50 (- 19:01), 10:50-11:05 (- 19:01), 11:06-11:20 (08:00 - 17:00), 11:21-11:37 (- 19:01), 11:42-11:57 (- 19:01), 11:59-12:27 (08:00 - 18:00), 12:27-12:44 (- 19:01), 12:45-13:07 (- 19:01), 13:07-13:27 (- 19:01), 13:29-13:45 (- 19:01), 13:46-14:05 (- 19:01), 14:05-14:24 (- 19:01), 14:25-14:44 (- 19:01), 14:49-15:09 (- 19:01), 15:16-15:31 (- 19:01), 15:51
M04-A-376 (254.545 km)	06:30, 08:15-08:31 (- 19:01), 08:31-08:53 (- 19:01), 08:55-09:11 (- 19:01), 09:11-09:28 (- 19:01), 09:28-09:44 (- 19:01), 09:44-10:05 (- 19:01), 10:05-10:20 (- 19:01), 10:20-10:36 (- 19:01), 10:37-11:00 (- 19:01), 11:02-11:16 (- 19:01), 11:18-11:34 (- 19:01), 11:35-12:02 (08:00 - 17:00), 12:09-12:24 (- 19:01), 12:33-12:50 (- 19:01), 12:53-13:10 (- 19:01), 13:14-13:36 (- 19:01), 13:38-13:55 (- 19:01), 15:41
M80-T-354 (33.444 km)	07:40, 08:00-08:17 (- 19:01), 08:18-08:34 (- 19:01), 08:35-08:50 (- 19:01), 08:50-09:20 (- 19:01), 09:23-10:23 (- 19:01), 10:39
O40-K-191 (243.667 km)	06:30, 08:18-08:41 (- 19:01), 08:56-09:13 (- 19:01), 10:46
O40-K-192 (19.283 km)	07:47, 08:00-08:50 (08:00 - 18:00), 09:01
O78-K-073 (42.028 km)	07:39, 08:00-08:18 (- 19:01), 08:19-08:34 (- 19:01), 08:35-09:07 (08:00 - 19:00), 09:08-09:26 (- 19:01), 09:26-09:41 (08:00 - 17:00), 09:41-10:00 (- 19:01), 10:00-10:23 (- 19:01), 10:25-10:47 (- 19:01), 10:48-11:11 (08:00 - 17:00), 11:12-11:31 (- 19:01), 11:32-11:51 (08:00 - 17:00), 11:51-12:08 (- 19:01), 12:18-12:32 (- 19:01), 12:34-13:15 (08:00 - 18:00), 13:31
T34-E-701 (260.631 km)	07:51, 08:00-08:31 (08:00 - 18:00), 10:22-10:38 (- 19:01), 12:15-12:30 (- 19:01), 12:31-12:49 (- 19:01), 12:50-13:07 (- 19:01), 13:08-13:24 (- 19:01), 13:25-13:42 (08:00 - 17:00), 13:43-14:04 (- 19:01), 14:05-14:34 (08:00 - 18:00), 14:37-14:58 (07:00 - 18:00), 15:01-15:16 (08:00 - 17:00), 15:19-15:35 (- 19:01), 15:37-15:58 (- 19:01), 16:00-16:16 (08:00 - 17:00), 16:18-16:39 (08:00 - 17:00), 16:39-16:55 (08:00 - 17:00), 17:03

Table II shows the total number of kilometres for the used testing instance of the realistic routing for each of the vehicles from the fleet.

The proposed approach managed to find the optimal solution by using 15 vehicles from a fleet of 16 available, thus satisfying all the set constraints. During a testing period of more than a month, using the proposed approach, compared to the previous routing based on long term experience of the transport managers, produced savings from the possibility to use one or two vehicles less during the daily routing. From the financial aspect of the company, those savings range from 10 to 20 %, which is really more than satisfactory in real-time environments.

From the previous results, it can be concluded that the algorithm was always able to find solutions that completely fit the constraints. Therefore, costs were cut due to the more efficient use of the fleet, particularly the use of smaller vehicles. Larger vehicles, which have greater costs, were only used when the smaller vehicles could not serve all the customers; meaning that the use of larger vehicles was absolutely necessary. In these scenarios, the algorithm routed the larger vehicles to serve as few customers as possible and those closer to the depot, consequently minimizing the cost of these vehicles.

Looking at the algorithm's run-time, it is the number of customers and their constraints that have the greatest impact. Aside from that, the peculiarities of a fleet of vehicles, which include its size and type vehicles available, also have an effect on the run-time.

V. Conclusion

The optimization of the transport system, from the perspective of the logistical approach of transport management, is primarily achieved by the optimization of the transport flows of goods and using the transport means. When it comes to the use of transport means, the optimization process solves the problems based on the basic characteristics of the routing problem. The problem of determining the optimal route (path) of the transport means performing the service on the transport network, in terms of minimizing the distance, travel time or service costs (transport).

In theory and practice, there are numerous models for solving these and similar problems. Most of the efficient models for solving complex VPR problems are based on a heuristic approach founded on clustering techniques. An innovative multiphase, cluster-based model was presented in this work. The model presented is able to successfully solve VRP problems of large dimensions, satisfying the realistic constraints such as the heterogenic vehicle fleet, time windows of the customers, constraints of what customer can be served by a specific vehicle, etc. From the aspect of practical application, this approach generated significant results on the examples of the transport route optimization of one of the biggest distribution companies in Bosnia and Herzegovina.

The presented approach can be additionally improved by extending it for the multi-depot transport option, as well as including additional constraints such as multiple trips, possibility of returning goods during the one transport route, etc. Also, in the initial phase of the cluster creation, heuristic approaches such as ant colonies, swarm of bees or heuristic neighbors with a changing environment can be applied.

References

[1] Dantzig, G. B., and Ramser, J. H. (1959). The truck dispatching problem. Management science, Vol. 6(1), 80-91. doi: 10.1287/mnsc.6.1.80

[2] Hall, R. (2006). On the road to integration. OR/MS Today 33(3), 50–57.

[3] Grosso, R., Munuzuri, J., Escudero-Santana, A., and Barbadilla-Martín, E. (2018). Mathematical Formulation and Comparison of Solution Approaches for the Vehicle Routing Problem with Access Time Windows. Complexity. doi:10.1155/2018/4621694

[4] Nalepa, J., and Blocho, M. (2016). Adaptive memetic algorithm for minimizing distance in the vehicle routing problem with time windows. Soft Computing, Vol. 20 (6), 2309–2327. doi: 10.1007/s00500-015-1642-4

[5] Dixit, A., Mishra, A., and Shukla, A. (2019). Vehicle Routing Problem with Time Windows Using Meta-Heuristic Algorithms: A Survey. In: Yadav N., Yadav A., Bansal J., Deep K., Kim J. (eds) Harmony Search and Nature Inspired Optimization Algorithms. Advances in Intelligent Systems and Computing, Vol. 741, 539-546. doi: 10.1007/978-981-13-0761-4_52

[6] Goel, R., and Maini, R. (2018). A hybrid of ant colony and firefly algorithms (HAFA) for solving vehicle routing problems. Journal of Computational Science, Vol. 25, 28-37. doi: 10.1016/j.jocs.2017.12.012

[7] Mahmudy, W. F. (2014). Improved Simulated Annealing for Optimization of Vehicle Routing Problem With Time Windows (VRPTW). Kursor, Vol. 7(3). doi: 10.21107/KURSOR.V7I3.1092

[8] Osaba, E., Yang, X. S., and Del Ser, J. (2020). Is the Vehicle Routing Problem Dead? An Overview Through Bioinspired Perspective and a Prospect of Opportunities. In: Yang XS., Zhao YX. (eds) Nature-Inspired Computation in Navigation and Routing Problems. Springer Tracts in Nature-Inspired Computing. Springer, Singapore, doi: 10.1007/978-981-15-1842-3_3

[9] Dondo, R., and Cerdá, J. (2007). A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows. European Journal of Operational Research, Vol. 176(3), 1478-1507. doi: 10.1016/j.ejor.2004.07.077

[10] Žunić, E., Đonko, D., and Buza., E. (2020). An adaptive data-driven approach to solve real-world vehicle routing problems in logistics. Complexity. doi: 10.1155/2020/7386701

[11] Žunić, E., Hindija, H., Beširević, A., Hodžić, K., and Delalić, S. (2018). Improving Performance of Vehicle Routing Algorithms using GPS Data. 14th Symposium on Neural Networks and Applications (NEUREL), Belgrade, Serbia, 2018, 1-4. doi: 10.1109/NEUREL.2018.8586982

[12] Žunić, E., Djedović, A., and Đonko, D. (2017). Cluster-based analysis and time-series prediction model for reducing the number of traffic accidents. International Symposium ELMAR, 25-29. doi: 10.23919/ELMAR.2017.8124427

[13] Žunić, E., and Đonko, D. (2019). Parameter setting problem in the case of practical vehicle routing problems with realistic constraints. 2019 Federated Conference on Computer Science and Information Systems (FedCSIS), 755-759. doi: 10.15439/2019F194

[14] Žunić, E., Delalić, S., Hodžić, K., Beširević, A., and Hindija, H. (2018). Smart warehouse management system concept with implementation. 14th Symposium on Neural Networks and Applications (NEUREL), 1-5. doi: 10.1109/NEUREL.2018.8587004

[15] Zunic, E., Besirevic, A., Delalic, S., Hodzic, K., and Hasic, H. (2018). A generic approach for order picking optimization process in different warehouse layouts. in 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). doi: 10.23919/MIPRO.2018.8400183

[16] Žunić, E., Djedović, A., and Đonko, D. (2016). Application of Big Data and text mining methods and technologies in modern business analyzing social networks data about traffic tracking. XI International Symposium on Telecommunications (BIHTEL), 1-6. doi: 10.1109/BIHTEL.2016.7775717

[17] Žunić, E. (Emir). (2018). Real-world VRP benchmark data with realistic non-standard constraints - input data and results. 4TU.Centre for Research Data. Dataset. https://doi.org/10.4121/uuid:598b19d1-df64-493e-991a-d8d655dac3ea

Advances in Software and System Engineering

ADVANCES in Software and Systems Engineering (ASSE) is a workshop organized in the scope of the FedCSIS Track 5. It is aimed at covering all research aspects related to the development and application of various methodologies, techniques, and technologies in Software and System Engineering (SSE). We particularly emphasize here a strong synergy of Software and System Engineering, in various ways.

One viewpoint of such synergy is in the fact that many Software Engineering methods and techniques were evolved to the level of their practical application in System Engineering, and by this they were also recognized latter on as very successful and common System Engineering methods and techniques. Another important viewpoint is in that services and products engineered by various Software Engineering methods and techniques become components of often very complex systems in various problem domains. By this, engineers are mostly forced to synchronously apply both Software and System Engineering methods and techniques to specify and develop complex systems in the target problem domains. The third and not less important viewpoint is that software systems are to be also observed as systems in general. In this way, software engineers need to apply not only specific software engineering methods and techniques in the software development process, but also common system engineering methods and techniques.

The main goal of ASSE is to address open questions and real potentials for various applications of modern methodologies, techniques, and technologies in Software and System Engineering so as to develop and implement effective software services in the support of information management and system engineering. Also, ASSE targets all research and development aspects, which bring to the societies new or improved approaches, processes, methodologies, or techniques of SSE.

Topics

Submissions to ASSE are expected from, but not limited to the following topics:

- Advanced methodology approaches in SSE – new research and development issues
- Advanced SSE Process Models
- Applications of SSE in various problem domains – problems and lessons learned
- Applications of SSE in Lean Production and Lean Software Development
- Total Quality Management and Standardization for SSE
- Artificial Intelligence and Machine Learning methods in advancing SSE approaches
- SSE for Information and Business Intelligence Systems
- SSE for Embedded, Agent, Intelligent, Autonomous, and Cyber-Physical Systems
- SSE for Design of Multimedia and Interaction Systems
- SSE with User Experience and Interaction Design Methods
- SSE with Big Data and Data Science methods
- SSE with Blockchain and IoT Systems
- SSE for Cloud and Service-Oriented Systems

Technical Session Chairs

- **Luković, Ivan,** University of Novi Sad, Serbia
- **Geylani, Kardas,** Ege University International Computer Institute, Turkey
- **Mazzara, Manuel,** Innopolis University, Russia

Program Committee

- **Ahmad, Muhammad Ovais,** Karlstad University, Sweden
- **Ben Ayed Elleuch, Nourchène,** Higher Colleges of Technology, ADW, United Arab Emirates
- **Blech, Jan Olaf,** Aalto University, Finland
- **Dejanović, Igor,** University of Novi Sad, Serbia
- **Derezinska, Anna,** Warsaw University of Technology, Institute of Computer Science, Poland
- **DUTTA, ARPITA,** IIT Kharagpur, India
- **Escalona, Maria Jose,** Universidad de Sevilla, Spain
- **Essebaa, Imane,** Hassan II University of Casablanca, Morocco
- **García-Mireles, Gabriel Alberto,** Universidad de Sonora, Mexico
- **Göknil, Arda,** Turkey
- **Hanslo, Ridewaan,** Council for Scientific and Industrial Research, South Africa
- **Heil, Sebastian,** Chemnitz University of Technology, Germany
- **Jarzębowicz, Aleksander,** Gdansk University of Technology, Poland
- **Kaloyanova, Kalinka,** Sofia University, Bulgaria
- **Karolyi, Matěj,** Institute of Biostatistics and Analyses, Faculty of Medicine, Masaryk University, Brno, Czech Republic
- **Katić, Marija,** Birkbeck, University of London, United Kingdom
- **Khlif, Wiem,** University of Sfax, Tunisia
- **Krdzavac, Nenad,** University of Cambridge, Cambridge Centre for Advanced Research and Education, Singapore
- **Marcinkowski, Bartosz,** Department of Business Informatics, University of Gdansk, Poland
- **Milosavljević, Gordana,** University of Novi Sad, Faculty of Tecnical Sciences, Serbia
- **Misra, Sanjay,** Covenant University, Nigeria
- **Morales Trujillo, Miguel Ehecatl,** University of Canterbury, New Zealand

- **Özkan, Necmettin,** Kuveyt Turk Participation Bank, Turkey
- **Özkaya, Mert,** Turkey
- **Ristić, Sonja,** University of Novi Sad, Faculty of Technical Sciences, Serbia
- **Rossi, Bruno,** Masaryk University, Czech Republic
- **Sierra Rodríguez, José Luis,** Universidad Complutense de Madrid, Spain
- **Tarhan, Ayca,** Hacettepe University Computer Engineering Department, Turkey
- **Torrecilla Salinas, Carlos,** University of Seville, Spain
- **Varanda Pereira, Maria João,** Instituto Politecnico de Braganca, Portugal

DOI: 10.15439/2020F73
ISSN 2300-5963 ACSIS, Vol. 21

An incremental malware detection model for meta-feature API and system call sequence

Pushkar Kishore
Dept. of C.S.E.
NIT Rourkela
Odisha, India
518CS1002@nitrkl.ac.in

Swadhin Kumar Barisal
Dept. of C.S.E.
NIT Rourkela
Odisha, India
swadhinbarisal@gmail.com

Durga Prasad Mohapatra
Dept. of C.S.E.
NIT Rourkela
Odisha, India
durga@nitrkl.ac.in

***Abstract*—In this technical world, the detection of malware variants is getting cumbersome day by day. Newer variants of malware make it even tougher to detect them. The enormous amount of diversified malware enforced us to stumble on new techniques like machine learning. In this work, we propose an incremental malware detection model for meta-feature API and system call sequence. We represent the host behaviour using a sequence of API calls and system calls. For the creation of sequential system calls, we use NITRSCT (NITR System call Tracer) and for sequential API calls, we generate a list of anomaly scores for each API call sequence using Numenta Hierarchical Temporal Memory (N-HTM). We have converted the API call sequence into six meta-features that narrates its influence. We do the feature selection using a correlation matrix with a heatmap to select the best meta-features. An incremental malware detection model is proposed to decide the label of the binary executable under study. We classify malware samples into their respective types and demonstrated via a case study that, our proposed model can reduce the effort required in STS-Tool(Socio-Technical Security Tool) approach and Abuse case.**

Theoretical analysis and real-life experiments show that our model is efficient and achieves 95.2% accuracy. The detection speed of our proposed model is 0.03s. We resolve the issue of limited precision and recall while detecting malware. User's requirement is also met by fixing the trade-off between accuracy and speed.

***Index Terms*—meta-feature, API call, system call, incremental malware detection, Abuse Case, STS-Tool**

I. Introduction

TODAY, we are facing one of the toughest security threats, malware. Whenever an unknown application is installed by a user on their systems, the malware detector uploads the application's executable on the cloud to verify whether an application is malicious or benign. After the executable is received, the detection system unpacks it using tools like PEiD[1], PolyUnpack [1], etc. Then, the detection system disassembles the binary to extract API or system calls and trains a machine-learning based model for classification.

Sequential series is a critical class of data, which can be applied in anomaly detection [2], trend analysis [3], periodic pattern detection [4], short-term prediction [5], etc. API call profile has API call sequence, e.g. <WriteFile; VirtualQueryEx; UnmapViewOfFile; Sleep; ...>. Anomaly score describes the sophisticated aggregation of the anomaly records. Numenta Hierarchical Temporal Memory (N-HTM) [6], an anomaly score generator, can be used to generate anomaly score for each API call in an API call sequence. We will treat the set of anomaly score of every instance in an API call sequence as a newer *API anomaly score sequence*, e.g. *API relative frequency call sequence* can be: <1, 1, 1, 1, 2, 2, ...>. For the case of the system call sequence, we use the dataset generated by NITRSCT [7]. Embedding various features in a single malware detector becomes non-functional when adversarial attack occurs. So, we design an incremental malware detector which accomplishes the task of malware detection if one of its layers fails. The accurate classification of malware families is still a tough problem and is also significant in malware analysis. Whenever software is used, security needs to be assured thoroughly among the users and software. During the software development life cycle (SDLC), the Abuse case and the STS-Tool approach can produce secured software. To specify security requirements for the software, Abuse case is used. Abuse case [8] is a model for specifying security requirements. The term Abuse case is an alteration of the use case. STS (Socio-Technical Security) [9] models security requirements considering actors as various stakeholders and their goals as main objectives. It tackles security-related issues during the early phase of the socio-technical system design.

Despite having modern malware detection systems, researchers are still facing many challenges. First, a single-layer malware detection system is prone to adversarial or evasion attacks and the detector will fail. Besides, accuracy is acutely limited during run-time [10]. Secondly, user's expectation is not met while fixing trade-off between accuracy and speed. Thirdly, a lot of effort is wasted in the wrong direction in STS-Tool approach and Abuse case. Lastly, it is very tough to provide labels to malicious samples according to their class. To address the above challenges, we propose a novel incremental malware detection model for meta-feature API and system call sequence. API calls can be extracted using tools like IDA[2], W32dasm[3], etc. This will help in quick preparation

[1] https://www.softpedia.com/get/Programming/Packers-Crypters-Protectors/PEiD-updated.shtml

[2] https://www.hex-rays.com/products/ida/

[3] https://www.softpedia.com/get/Programming/Debuggers-Decompilers-Dissasemblers/WDASM.shtml

of the collection of API call sequences. We use NITRSCT [7] generated datasets for the system calls. We use N-HTM for generating an anomaly score for each API call in an API call sequence. For an API call sequence, <WriteFile; VirtualQueryEx; UnmapViewOfFile; Sleep; ...>, <0.2, 0.3, 0.7, 0.1, ...> may be the anomaly score sequence. We predict the final label of the executable using an incremental model. At first, we apply malware detection on the system call dataset using one-class SVM. Then, we send only benign samples for testing to malware detector based on meta-feature API calls dataset using one-class SVM. The reason behind sending only benign samples is to ensure that none of the malicious executables gets executed due to wrong labelling by the first detector. So, we cross-check it with the second detector.
We select six meta-features, which represent the characteristics of the API anomaly score sequence. We assume that an API anomaly score sequence **X=** $\{x_1, x_2, x_3, ..., x_Z\}$, where Z is the length of the sequence. **X** is divided into m sub-sequences. For each sub-sequence, we calculate the values of meta-features: Kurtosis, Coefficient of Variation, Oscillation, Regularity, Square wave and Variation of trend. By combining various sub-sequences, we get the final dataset having the six meta-features. A correlation matrix with a heatmap is used to select the best meta-features. The incremental model helps detect malware whenever the dataset is ready. We classify the malicious binary executable into their respective classes. A case study is included to demonstrate that effort required in Abuse case and STS-Tool can be reduced by our proposed model.

The main objectives of this paper are:

1) To represent API call sequence, we propose to use N-HTM. In order to improve the convergence speed of the malware detector, we use meta-features derived from the API anomaly score sequence. We use the incremental model trained using one-class SVM to detect the malware.
2) To reduce the number of meta-features using correlation matrix displayed in heatmap. This reduction accelerates the convergence speed of our model.
3) To implement the incremental model and assess it using an extensive scale of real-world data set. We use anomaly score to assign malicious binary executable its proper class.
4) To demonstrate that effort required in Abuse case and the STS-Tool approach can be reduced using our proposed model.

Paper Organizations The remaining part of this paper is organized as follows: Section II briefly describes the related work. Section III introduces the methodology of our proposed model. Section IV presents the experimental results. Section V discusses the comparison with related work. Section VI shows the threats to the validity and Section VII presents the conclusions and future work.

II. Related Work

A. API Call Based Method

Many researchers used API calls to represent binary executable. Patnaik, Barbhuiya and Nandi [11] checked the target process's API call similarity with the API call signature of the malware. Huang, Zhang and Tan [12] detected stealthy behaviour by analyzing the user interface components of top-level function. Fan et al. [13] proposed constructing sub-graphs of API calls to represent the similar behaviour of malware of the same family.

B. System Call Based Method and Malware Detection Model

Canzanese, Mancoridis and Kam [14] used system call n-gram method for representing binary executable and support vector machine (SVM) for malware detection. The performance is quite good as system calls precisely represent the binary executable's behaviour. This method fails if any malware hides in a computer and conceals its malicious behaviour. Zhang, Qin, Zhang, Yin and Zou [15] proposed a lightweight framework for malware detection based on the graph and information theory. But, whenever a malware attack occurs, its detection will be complex as there will be numerous interactions between files, processes, etc, leading to the nexus of graph. Raff, Barker, Sylvester, Brandon, Catanzaro and Nicholas [16] used convolutional neural networks (CNN) and bytecode n-grams for malware detection. Bytecodes are noisy compared to opcodes, thus the accuracy is limited. Kang, Yerima, McLaughlin and Sezer [17] used the Naive Bayes (NB) method for detecting 2-opcode vectors represented malware. This method's accuracy is very small as NB assumes that the features are independent. Puerta, Sanz, Santos and Bringas [18] used opcode frequencies to represent binary executable and detected malware using SVM. Lack of simplicity of features jeopardizes the accuracy.

C. Sequential Series

Numerous approaches are available to find anomalies in univariate/multivariate sequences. We group these methods into four categories: (1) Statistics-based methods [19] (2) Intelligent- computing methods [20] (3) Bayesian networks [21] and (4) Model-based approaches [22]. Statistical based methods come from techniques that detect abnormal changes. A variety of intelligent computing methods are available for detecting anomalies, such as deep learning [23], SVM [5], fuzzy theory and rough sets theory.

D. Anomaly score generation

For generating anomaly scores of a sequential data, Ahmad, Lavin, Purdy and Agha [6] proposed a technique named N-HTM. It is suitable for real-time applications and robustly detects anomalies for any data stream. They have also shown that their system is efficient, produces accurate results even in the presence of noisy data, adaptable to statistical change in the data, detects subtle temporal anomalies and minimizes false positives.

E. *Security Approaches*

For the STS (Socio-Technical Security) approach, STS-ml [9] is used which includes actors and is a goal-oriented modelling language. This approach relates security requirements to interaction. Paja, Dalpiaz and Giogini [9] proposed a technique to handle security requirement conflicts in socio-technical systems. The STS modelling language allows stakeholders to impose security concerns over the interactions. For example, if buyers send their personal data to a seller, the seller must not disclose the data to third parties, only the buyer should be able to access them. There is a commitment between the actors which ensures that they will consider security requirements while delivering service. One example of the security requirement is that the seller commits that they will not reveal buyer's personal data to anyone.

III. Proposed Methodology

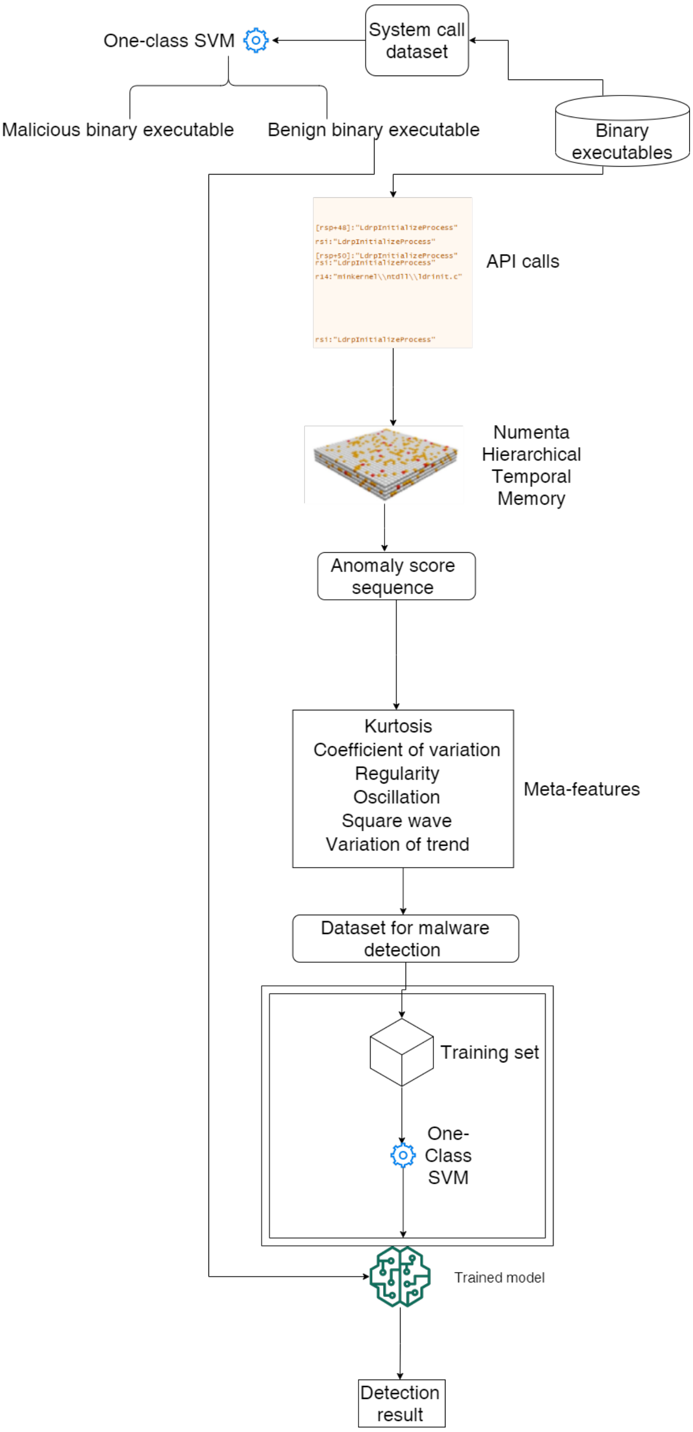
In this section, we propose an incremental malware detection model for meta-feature API and system call sequence. We present the architecture of our malware detection model in Figure 1. It comprises of seven steps: Creating system call dataset, classification using one-class SVM, unpacking and disassembly process, generating anomaly score sequence from API call sequence using Numenta Hierarchical Temporal Memory, defining meta-features for the anomaly score sequence, creating meta-feature API call dataset and classification using one-class SVM for benign binary executables. By performing the above seven steps, we can detect malware. We present the algorithm in Algorithm 1. The time complexity is O(n^2) and space complexity is O(mn), where m is the number of features and n is the number of instances. The description of the above seven steps are discussed below.

A. *Creating system call dataset*

We use NITRSCT [7] generated system call dataset[4]. This dataset contains system calls gathered from Windows OS based benign and malicious binary executables. The features are represented in the form of a vector having three consecutive ordered system calls.

B. *Classification using one-class SVM*

We train the system call dataset using the one-class SVM model. Upon testing, we send the benign results to the next malware detector, i.e, meta-feature API detector. Benign binary executable is sent for re-verification since we do not want any malicious binary executable to damage the host. If any evasion or adversarial attack occurs, attackers make sure that the label of the real malware is misrepresented as benign. In this case, our incremental model comes to rescue, we will recheck that false labelled malware using another detector which blocks execution of the malicious executables. Meta-feature API detector is based on N-HTM technique and is least prone to attacks compared to deep learning techniques. After analysing the robustness of the models, we decide to keep machine-learning based detector in front and N-HTM at last so that our incremental model will still work if an attack occurs on the machine-learning based detector.



```
[rsp+48]:"LdrpInitializeProcess"
rsi:"LdrpInitializeProcess"
[rsp+50]:"LdrpInitializeProcess"
rsi:"LdrpInitializeProcess"
r14:"minkernel\\ntdll\\ldrinit.c"
rsi:"LdrpInitializeProcess"
```

Fig. 1. Proposed architecture of our approach

C. *Unpacking and disassembly process*

Unpacking and disassembly processes are used to unpack and disassemble executables for getting their API calls. At-

[4] https://github.com/pushkarkishore/NITRSCT

tackers may have packed some binary executables using some packing tools which are harder to disassemble. We unpack them first, if they are packed with ASPack[5], UPX[6], etc. Then, we disassemble the unpacked executable to get the API calls using Ollydbg[7]. We use limited disassembly tools to avoid distortion of the results. After completion of disassembling, we build an API call sequence having a list of API calls.

```
Algorithm 1: Meta-feature API and system call based
malware detection
  Input: A set of API calls, APIs = {$API_1$, $API_2$, . . ., $API_j$}, where $API_j$ represents the jth call;
         System call dataset.
  Output: A final label informing whether the binary executable is malicious or benign.
  Function MalwareDetector($APIs$):
      for training instances in the system call dataset do
          Apply one-class SVM technique to train the malware detection system;
      for each test instance in the system call dataset do
          Apply one-class SVM classifier to predict the final label, i.e. anomalous or benign;
          Send the benign samples for testing to meta-feature API detector;
      for $API_j$ in APIs do
          Generate a vector of API relative frequencies V($api_i$) according to a set of API calls;
          Apply N-HTM model to create API anomaly score sequence of each API call sequence;
          Generate dataset having best meta-features;
      for training instances in the meta-feature API call dataset do
          Apply one-class SVM technique to train the malware detection system;
      for each test instance obtained from system call detector do
          Apply one-class SVM classifier to predict the final label, i.e. malicious or benign;
      return;
  end
```

D. Generating API anomaly score sequence using Numenta Hierarchical Temporal Memory (N-HTM)

The *API anomaly score sequence* is generated from the API call sequence using relative frequency, e.g. for an API call sequence, <WriteFile; VirtualQueryEx; UnmapViewOfFile; Sleep; WriteFile; Sleep ...>, the *API relative frequency call sequence* will be: <1, 1, 1, 1, 2, 2, ...>. N-HTM [6] calculates an anomaly score for an API call upon receiving new patterns from the API call sequence. If the received pattern is predicted, then anomaly score is zero, while for the completely non-predictable pattern, it is one. Partial prediction of pattern has an anomaly score between zero and one. The similarity between actual and received patterns is calculated using sparse distributed representation. The anomaly score is dependent on the difference of overlap between actual and predicted bits.

TABLE I
ANOMALY SCORES OF SAMPLE API CALL SEQUENCE

Timestamp	Value	Anomaly Score
1-3-20 0:00	1	0.03010299967
1-3-20 0:00	1	0.03010299967
1-3-20 0:01	2	0.03010299967
1-3-20 0:01	2	0.03010299967
1-3-20 0:02	3	1
1-3-20 0:02	4	1
1-3-20 0:03	5	1

The anomaly score of a sample API call sequence is depicted in Table I. In Table I, 'Value' represents the API call sequence, where 1, 2, 3, ... represents the relative frequency of the API call. Anomaly score is associated with all the entries of the API call sequence.

E. Defining meta-features for the anomaly score sequence

We define six meta-features which is statistical representation of the API anomaly score sequence. The approach used in the different meta-features is discussed below:

1) **Kurtosis**: It measures whether a sequence is heavily tailed or lightly tailed related to the normal distribution [24]. For ECG data [25], kurtosis is effective in detecting the abrupt peaks from a sequence. It reflects variability of a sequence. It actually measures the number of outliers present in the distribution. Sequence with high kurtosis has generally heavy tails; but, low kurtosis shows light tails. We use Equation 1 for the sub-sequence created from API anomaly score sequence for calculating kurtosis.

$$K_z = \frac{1}{n}\sum_{i=1}^{n} D_i^4 - 3 \tag{1}$$

where, n is the length of the z-th sub-sequence derived from an API anomaly sequence; D_i values are the standardized data values defined using standard deviation with n as the denominator.

2) **Coefficient of variation**: It calculates the local variability relative to the complete sequence [26]. Local variability of a sub-sequence significantly rises if the abrupt peak occurs within the sub-sequence's interval. This meta-feature indicates the sub-sequence's sharp curve changes. It is used mainly for checking the consistency of sequence. We use Equation 2 for the sub-sequence created from the API anomaly score sequence for calculating coefficient of variation.

$$C_z = \frac{\sigma_z}{\mu} \tag{2}$$

[5] http://www.aspack.com

[6] https://upx.github.io.

[7] http://www.ollydbg.de/

where, σ_z denotes the standard deviation of the z-th sub-sequence; and μ is the mean value of all sub-sequences of API call sequence.

3) **Oscillation**: It is a periodic fluctuation between two consecutive anomaly scores in a sequence. In this work, we calculated the oscillation of a sub-sequence.
4) **Regularity**: Sample entropy [27] is used to calculate the regularity of series. It is also widely used for diagnosing the presence or absence of a disease [28]. Regularity will be higher, if there are less number of abrupt peaks in the sequence.
5) **Square wave**: These waves are generated by binary logic devices and encountered in digital switching circuits. A sequence can start and maintain the signal with high values in the first half, and sharply reduces for the second half. We have assumed that the curve of a variable is consistent if the square wave is represented and consistent with expectation. In the case of API anomaly score sequence, z-th sub-sequence is $X_z = \{ x_{z,1}, ..., x_{z,i}, x_{z,i+1}, ..., x_{z,N} \}$ and i = $\lfloor 0.5N \rfloor$. The binarized sub-sequence of X_z, represented as TX_z is calculated using Equation 3.

$$TX_z = X_z > 0.5 * max(X_z) \tag{3}$$

Confirmation of the z-th sub-sequence being square wave is done using Equation 4 .

$$S_z = 0.5 - rs * TX_z / LEN(TX_z) \tag{4}$$

where, LEN denotes the length of TX_z, vector rs = {1, 1, ..., 1, 0, 0, ..., 0} filters the signal with high values in TX_z. In vector rs, the value of “1” is i. According to Equation 4, the sub-sequence corresponds to a lower value, if it represents a square wave.
6) **Variation of trend**: Trend analysis provides a way to differentiate between two series. We smoothen the original API anomaly score sequence and calculate the variation on it. The variation of trend of z-th sub-sequence is defined using Equation 5 .

$$T_z = std(smooth(X_z)) \tag{5}$$

where, smooth is used for smoothing the original API sequence and std is used for finding its standard deviation. For a sequence having random trend, T_z will be small, if abrupt peaks are absent.

We have used the correlation matrix with heatmap to select the best meta-features. A heatmap is a graphical representation of data where values are represented as colours. So, viewer can refer to the colour for getting the value of data.

F. Creating meta-feature API call dataset

Two datasets are considered in performance analysis of our malware detection model: a malware dataset and a benign dataset. We have collected benign binary executables from 10 hosts in offices, computer laboratories, and isolated testbed to test within real-life environment. The malware which we use in our experiments are collected from VirusTotal[8]. In order to make sure that, the instances are unpacked, we detect the packers. Using the unpacking tools, we have unpacked the executables. Our final dataset contains 5114 malware binary executables and 4800 benign binary executables as shown in Table II. We split the malware and benign dataset into training dataset and a test dataset. The volume of the malware training dataset and benign dataset is fixed to avoid training biases. We randomly choose 2000 malware and benign samples for training purpose, while remaining samples are used for testing purpose. The final label is predicted using incremental model.

TABLE II
MALWARE DATASET

Sl. No.	Malware family	Number of samples
1	Backdoor	1352
2	Worm	559
3	Trojan	2394
4	Virus	809
Total number of samples		5114

G. Classification using one-class SVM for benign binary executables

We use a one-class support vector machine, which classifies by separating hyperplane. Linear SVM [29] uses a hyperplane $w^T\mathbf{x}$, which separates data points belonging to two classes, optimally. Here, w defines the hyperplane that learns from the training data points using stochastic gradient descent (SGD) method. We express the objective function used in SGD as a feature vector $\mathbf{x}_i$ belonging to X and their respective labels y_i having a value between 0 to 1. A regularisation constant α penalises the model having a higher complexity and the loss function L determines the objectives of SGD. We have used the soft-margin SVMs, where L denotes the hinge loss, which is represented by Equation 6.

$$L(t, y) = max(0, 1 - t_y) \tag{6}$$

The objective function used is as follows:

$$E(\mathbf{w}) = \frac{1}{p}\sum_{i=1}^{p} L(y_i, \mathbf{w}^T\mathbf{x}_i) + a \left\|\mathbf{w}\right\|_2 \tag{7}$$

where, p is the number of training points and $\mathbf{w}$ represents the weight. During detection, an instance is labelled 'malicious' if

$$\mathbf{w}^T\mathbf{x} > \Lambda \tag{8}$$

Testing instances consist of those binary executables which are marked benign by the system call detector.

IV. Experimental Results

In this section, we present the details of our experiment to show the performance of the proposed approach. First, we present the experimental setup, and then we discuss the performance of our model. We show that our approach can

[8] https://www.virustotal.com/

perform better by comparing with the state-of-art methods. We also assign proper class to the malicious samples after finding that it is malicious. Lastly, we demonstrate via a case study that effort required in Abuse case and STS-Tool approach can be reduced using our proposed model.

A. Setup

We implement all the experiments on one computer. The version of the CPU is Intel i5-3470 @ 3.20 GHz, the RAM is 16.0 GB and the operating system is Windows 10. We implement our approach using Python programming language in which the matrix computations are dependent on numpy.

B. API calls

To extract API calls, we use Ollydbg to disassemble binary executables and then obtain API calls. We collect API calls from 9914 binary executables having 100 to 10,000 number of API calls in each binary executable. In a few binary executables, we are not able to extract API calls; thus, we use a vector of all zero values to represent them.

C. Selection of Meta-features

To accelerate the convergence speed of our proposed model, we use a correlation matrix with heatmap to select the best meta-features. The heatmaps obtained from both malicious and benign binaries available in the dataset having six meta-features are shown in Figure 2 and 3 respectively. In the case of heatmap of malware binaries, coefficient of variation is highly correlated with variation of trend, oscillation is highly correlated with square wave. We have removed the regularity feature, since it is neutrally correlated with other features. And, kurtosis is not also considered since it's not strongly correlated with others as (coefficient of variation, variation of trend) and (oscillation, square wave) does. So, we select 4 features from malware binaries namely coefficient of variation, variation of trend, oscillation and square wave. In the case of heatmap of benign binaries, coefficient of variation is highly correlated with variation of trend and oscillation is highly correlated with square wave. So, we select the same 4 features available in malware binary executable. So, the final meta-features used for the final dataset creation are coefficient of variation, variation of trend, oscillation and square wave.

D. Performance analysis of malware detection

The parameters and metrics which we use for performance analysis of our proposed model are classification accuracy, detection false positive rate, detection true negative rate, detection false negative rate, detection precision, detection recall, F1-score, training time cost and detection time cost. The classification accuracy is calculated using Equation 9. Recall of malware detection model is the true positive rate evaluated using Equation 10, where True Positive (TP) is the number of malware instances correctly classified and False Negative (FN) is the number of malware cases misclassified as benign one. TNR is the true negative rate, which is evaluated using Equation 11 , where False Positive (FP) is the number of benign instances which are misclassified as malware binaries and True Negative (TN) is the number of benign instances which are correctly classified. FPR represents the false positive rate, FNR represents the false negative rate, Precision represents malware detection precision, and F1-score is computed using Precision and Recall, which are shown in Equations 12-15.

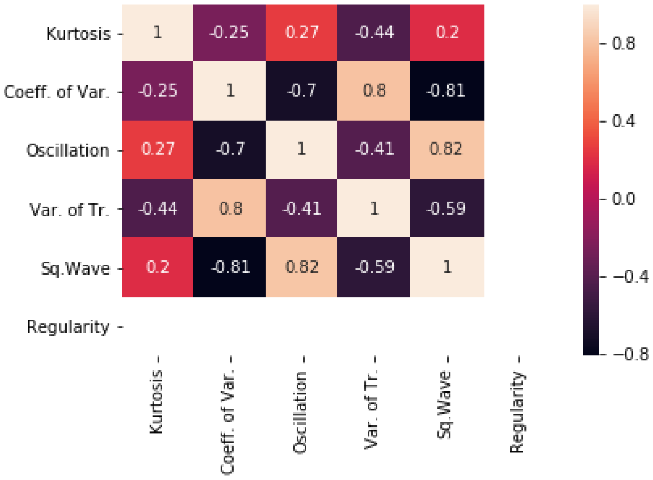


Fig. 2. Heatmap of malware binaries

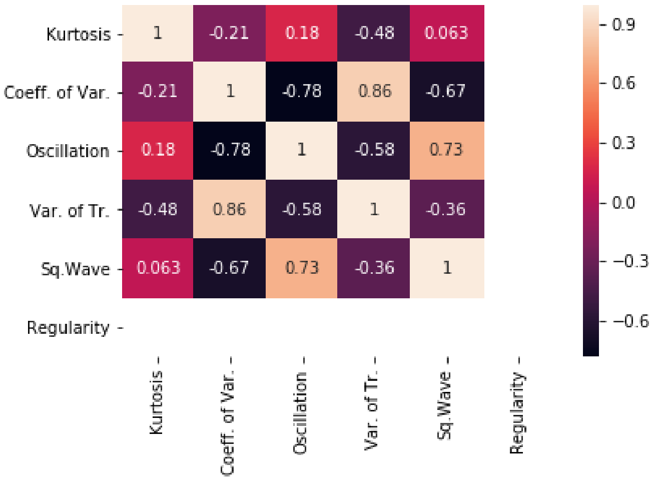


Fig. 3. Heatmap of benign binaries

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{9}$$

$$TPR(Recall) = \frac{TP}{TP + FN} \tag{10}$$

$$TNR = \frac{TN}{FP + TN} \tag{11}$$

$$FPR = \frac{FP}{FP + TN} \tag{12}$$

TABLE III
PERFORMANCE EVALUATION OF OUR MODEL

Sl. No.	Performance Parameters	Value
1	Accuracy (%)	95.2
2	Recall (%)	93
3	TNR (%)	88
4	FPR (%)	12
5	FNR (%)	7
6	Precision (%)	95.4
7	F1-score (%)	94.1
8	Detection Time (s)	0.03
9	Training Time (s)	160

$$FNR = \frac{FN}{FN + TP} \tag{13}$$

$$Precision = \frac{TP}{FP + TP} \tag{14}$$

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{15}$$

The performance evaluation of our model is shown in Table III.

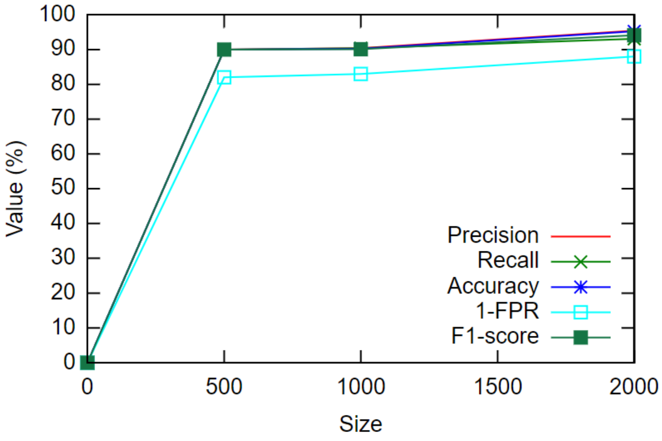


Fig. 4. Stability evaluation of accuracy

E. Stability evaluation of malware detection

Since malware variants are swiftly growing in numbers, we always face issue that training samples are always smaller than the volume of the test dataset. When the detection set contains numerous binary executables and the training set is smaller, then the ratio of training/(training+detection) is small and will lead to limited accuracy. So, here we evaluate the stability of our proposed malware detection model by testing with different volumes of training sets. The various sizes of our training sets is 500, 1000, and 2000. Figure 4 represents the precision, recall, 1-FPR, and the F1-score of our proposed detection model for different sizes. From Figure 4, we can infer that our model is stable when the ratio of training/(training+detection) is less than 0.2.

Figure 5 shows the training time of our approach for different sizes of data set. We observe that the training time is moving in a steep way upto size 1000, and then smoothens after size, 1200.

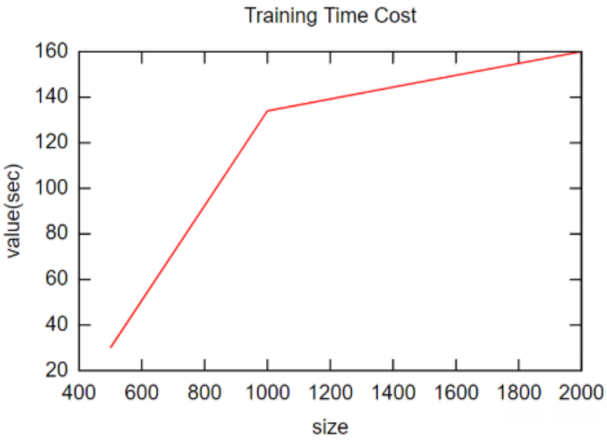


Fig. 5. Training time of our approach for different volume of data sets

F. Classification accuracy of malware families

We evaluate the classification accuracy of malware families evaluated using one-versus-all strategy SVM[9]. To make the evaluation quick, we use the average anomaly score of each sample only, reducing it to single feature. Results are represented in Table VIII. We observe that detecting virus is not feasible, but have better accuracy for backdoor and Trojan. Worms can be easily labelled by our model.

G. Case Study

We have considered a case of travel planning scenario to demonstrate how our proposed model optimizes the effort required in Abuse case and STS-Tool approaches. Abuse case is determined by those interactions between an actor and the system which can harm the resources associated with actors, stakeholders or systems. STS model comprises of three complementary views: social, information and authorisation. These three views together help plan a model for system-at-hand. Now, we discuss below the effort optimization process for the above case study.

We identify the agents and roles in the model, for example, Tourist, Travel Agency Service (TAS) and Hotel are roles, while Hotel Service, Bob, Payment service and Amadeus Service (AS) are agents. Then, we identify the goals of agents and roles, as described in Table V. We design a goal model of an actor that ties together the goals and documents, For example, an actor possesses documents; an actor needs documents to fulfil a goal; an actor produces documents during goal fulfilment; an actor modifies a document while fulfilling a goal. The goal model is described in Table VI. Using the technique of goal delegation, we can transfer the fulfilment responsibility of the goal from one actor to another. Also, the

[9] https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/

TABLE IV
SECURITY REQUIREMENTS FOR TRAVEL PLANNING SCENARIO

Roles	Security Requirements	Requester	Why security requirement is handled by our model
TAS	non-repudiation-of-acceptance (Tourist, TAS, Tickets booked)	Tourist	Trojans and Worms are detectable
Tourist	non-repudiation-of-delegation (Tourist, TAS, Tickets booked)	TAS	Trojans and Worms are detectable
TAS	true-redundancy-multiple-actor (Tickets booked)	Tourist	Unhandled
Hotel	no-redelegation (hotel booked)	Tourist	Unhandled
AS	integrity-of-transmission (provided(TAS, AS, Itinerary details))	TAS	Unhandled
All agents	not-achieve-both (eticket generated, credit card verified)	Org	Unhandled
AS	availability (flight ticket booked, 85%)	TAS	Backdoor is detectable
Tourist	delegated To(trustworthy(Hotel))	Tourist	Backdoors and Trojans are detectable
TAS	need-to-know (Personal data and Itinerary, Tickets booked)	Tourist	Unhandled
TAS	non-modification (Personal data and Itinerary)	Tourist	Trojans and Viruses are detectable
TAS	non-production (Personal data and Itinerary)	Tourist	Trojans and Backdoors are detectable
TAS	non-disclosure (Personal data and Itinerary)	Tourist	Trojans and Backdoors are detectable

TABLE V
GOAL OF AGENTS

Name of Agent or Role	Goals
Amadeus Service	Eticket generated and credit card verified
TAS	Flight ticket booked and Train ticket booked
Payment Service	Prepayment made
Hotel Service	Room selected and Prepayment made
Tourist	Tickets booked and Hotel booked
Hotel	Hotel booked and Room selected

TABLE VI
GOAL MODEL

Agent or Role	Goal	Asset rules
AS	Eticket generated	Produce flight tickets
AS	Eticket generated	Need itinerary details
TAS	Flight ticket booked	Need itinerary details
TAS	Train ticket booked	Produce tickets
TAS	Tickets booked	Need travelling order
Tourist	Trip planned	Produce travelling order
Tourist	Trip planned	Need travelling order
Tourist	Trip planned	Modify travelling order
Tourist	Hotel booked	Need IdDoc copy
Hotel	Hotel booked	Need and modify IdDoc copy

information is exchanged between actors, named document provision. Goal delegations and document provisions for every roles and agent are shown in Table VII. We eliminate some security issues as they are handled by our proposed malware detection model, such as non-repudiation of delegation or acceptance, trustworthiness, and availability. Only we consider no-redelegation, integrity of transmission, confidentiality of transmission, separation of duties, combination of duties and redundancy concerns. Security requirements for this example is described in Table IV.

TABLE VII
GOAL DELEGATIONS AND DOCUMENT PROVISIONS

Agent or Role	Delegations or Provisions
TAS	Delegates flight ticket booked to AS
TAS	Provisions itinerary details to AS
Hotel Service	Delegates prepayment made to Payment Service
Hotel	Delegates hotel booked to Hotel Service
Tourist	Delegates hotel booked to Hotel
Tourist	Provisions IdDoc copy to Hotel
Tourist	Delegates tickets booked to TAS
Tourist	Provisions traveling order to Hotel

TABLE VIII
CLASSIFICATION ACCURACY OF MALWARE FAMILIES

Sl. No.	Actual class	Accuracy (%)
1	Backdoor	66
2	Worm	99
3	Trojan	70
4	Virus	1

There may be malware which can cause TAS to non-repudiate acceptance from Tourist. This type of malware is easily discovered by our proposed model. Similarly, the malware causing non-repudiation of delegation by Tourist, when requested by TAS, is also detected by our model. TAS tries to book tickets using either railways or airways, so true-redundancy-multiple-actor security requirement is there, which is uncoverable by our model. Hotel cannot redelegate the request done by tourist, and it is undetected by our model. Amadeus service's integrity maintenance can be done by applying intrusion detection in network channels, which cannot be done on hosts. Agent's plan of action is previously defined and is undetectable by our model. Non-Availability of any service for specific duration is easily detected by our model. Trustworthiness is easily insured as our proposed model will detect the malware which results in suggestions without considering ratings of the desired results. Since the data is stored in database or files, the modification, production and disclosure are easily recognized by our model. However, the information which is pre-required is undetected by our model.

For the Abuse case, we see that its assets'safety condition is embedded within the STS-Tool approach. Hence, Abuse case approach is not needed if we follow the STS-Tool approach. After analysing the case study, we have seen that our proposed model can tackle 7 out of 12 security requirements as shown in Table IV. Henceforth, the effort required for designing security model is reduced to approx 50%.

TABLE IX
COMPARISON OF PERFORMANCE OF OUR APPROACH WITH EXISTING STATE-OF-ART APPROACHES

Method	Dataset	Accuracy(%)	Precision(%)	Recall(%)	1-FPR(%)	F1-score(%)	Detection time(s)	Training time(s)
SVM [18]	Android Genome	83.5	86.5	80.6	87.4	83.4	0.001	31
NB [17]	Android Genome	79.7	78.3	82.2	77.2	80.2	0.005	134
CNN [16]	VirusShare	83.8	82.5	85.8	81.8	84.1	0.053	213467
SVM [14]	VirusTotal	86.6	92.4	79.8	93.4	85.6	0.094	179
Graph theory [15]	MobileSandbox	93.6	92.3	94	92.1	91.1	0.001	NA
Our approach	VirusTotal	95.2	95.4	93	88	94.1	0.03	160

V. Comparison with related work

We have compared the performance of our model with that of several state-of-art methods and shown in Table IX. By comparing with the other state-of-art methods, we observe that our approach significantly improves the classification accuracy, the detection precision and F1-score while retaining the detection speed. Accuracy is better than other models, thus it can be used for industrial malware detection. Precision is also higher, which means that 95.4% of the results are relevant results. Recall is 93%, which is lower than the recall evaluated by using Graph theory method. It means that 93% of total relevant results are correctly classified by our model. In most problems, we either give maximum priority to higher precision or recall, which depends upon the problem under consideration. In general, we use a simple metric which will use precision and recall to maximize the number to improve the model. That metric is known as F1-score, which is the harmonic mean of precision and recall. Our model is imperative in terms of F1-score, which is 94.1%. Specificity is equivalent to "1-FPR", which means that instances which are benign and being labelled as benign is 88%. It is a subsidiary parameter, as it's lower value can only block the benign process. However, our main objective of not executing malware executable on the host will not be affected by lower specificity. In terms of detection time, our model lies behind models using SVM [18], NB (Naive Bayes) [17] and Graph theory [15] based approach. But, those models, i.e. SVM model [18], NB model [17] and Graph theory model [15] have lower detection accuracy, which implies that there is poor trade-off between detection time and accuracy. Only the models using SVM [18], NB [17] and Graph theory [15] have lower training time than our proposed model. But, models using SVM [18], NB [17] and Graph theory [15] have lower detection accuracy than our proposed model. So, we draw inference that two-phase approach of detection, representing API calls in the form of meta-features and using average anomaly score of instances for classification are effective in designing an industrial-applicable malware detector.

VI. Threats to Validity

In this section, we identify some possible threats to the validity of our approach. Malware is generally packed using packers, but sometimes they are not detected by malware detectors. Majority of the packers can be unpacked using numerous techniques and tools such as PolyUnpack [1], which recover the original source file. For API calls based detection technique to work, unpacking techniques should always provide the original code. Obfuscation in the software will make it tough to de-obfuscate it. So, our discussion in this section is concerned about the limitation caused by obfuscation.

Obfuscation is a semantic preserving transformation which results in obfuscated programs. When we collect the obfuscated programs from the same source, then it is similar. Our model can detect obfuscation to some degree. Obfuscation can be of several types such as identifier renaming, junk code injection, control flow based obfuscation, etc. Identifier renaming obfuscation renames variables but it cannot impact the representations of binary executable. The junk code injection can change the distributions but can be easily detected and denoised. Control flow based obfuscation changes the control flow graph leading to error in control flow based detection. Some noisy instructions are added, but still, it is similar to the original instructions. In general, our approach can resist limited number of mistakes caused by obfuscation. Adversarial attack, which is a limitation of machine -learning based application is partially handled using our incremental model. But, we need to focus on many issues to obtain an appropriate technique for keeping our model safe from adversarial attacks.

We consider the malware targetting windows OS and NITRSCT is also limited to Windows OS. We assess the performance of our model on windows based dataset only. Android OS and Linux targetting malware may remain undetected.

VII. Conclusions and Future work

In this paper, we proposed an incremental malware detection model for meta-feature API and system call sequence, which effectively identified malware. We use the system call dataset generated using NITRSCT. Then, the API calls are collected by disassembling malicious executables and legitimate ones. Then anomaly score sequence of API calls is generated using N-HTM. A dataset is created using anomaly score of API calls, which contains six meta-features. Number of meta-features are reduced using correlation matrix with heatmap. After the final meta-features are selected, then the proposed model is used to detect malware. We also provided the labels to malware according to its class which can analyse the feature selection process of the malicious samples. Through a case study, we have demonstrated that our proposed model eliminates the need for having Abuse case approach and reduces the effort required, to around 50% for STS-Tool. Our model smoothly used meta-feature API calls (high-level features) and system calls to cover characteristics of malware and improved detection precision and F1-score by more than 3%. Real-life experimental results have shown that our approach achieved 95.2% accuracy and have detection speed lower than 0.1s. In addition, training time is also lower which doesn't increase the time complexity of the model.

As future work, we will ensemble static analysis based features with dynamic analysis to reduce dependency on unpacking tools. We will try to apply this model to detect the malware present in Android OS, anti-fraud systems, other domains, etc. We will try to improve the classification accuracy of each malware family. Exploration of a few more meta-features will be done by us to find a minimal number of features which will provide higher accuracy.

References

[1] P. Royal, M. Halpin, D. Dagon, R. Edmonds, and W. Lee, "Polyunpack: Automating the hidden-code extraction of unpack-executing malware," in *2006 22nd Annual Computer Security Applications Conference (ACSAC'06)*. IEEE, 2006, pp. 289–300. [Online]. Available: https://doi.org/10.1109/acsac.2006.38

[2] K. Yan, Z. Ji, and W. Shen, "Online fault detection methods for chillers combining extended kalman filter and recursive one-class svm," *Neurocomputing*, vol. 228, pp. 205–212, 2017. [Online]. Available: https://doi.org/10.1016/j.neucom.2016.09.076

[3] C. S. Sharma, S. N. Panda, R. P. Pradhan, A. Singh, and A. Kawamura, "Precipitation and temperature changes in eastern india by multiple trend detection methods," *Atmospheric research*, vol. 180, pp. 211–225, 2016. [Online]. Available: https://doi.org/10.1016/j.atmosres.2016.04.019

[4] A. K. Chanda, C. F. Ahmed, M. Samiullah, and C. K. Leung, "A new framework for mining weighted periodic patterns in time series databases," *Expert Systems with Applications*, vol. 79, pp. 207–224, 2017. [Online]. Available: https://doi.org/10.1016/j.eswa.2017.02.028

[5] Z. Ji, B. Wang, S. Deng, and Z. You, "Predicting dynamic deformation of retaining structure by lssvr-based time series method," *Neurocomputing*, vol. 137, pp. 165–172, 2014. [Online]. Available: https://doi.org/10.1016/j.neucom.2013.03.073

[6] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, "Unsupervised real-time anomaly detection for streaming data," *Neurocomputing*, vol. 262, pp. 134–147, 2017. [Online]. Available: https://doi.org/10.1016/j.neucom.2017.04.070

[7] P. Kishore, S. K. Barisal, and S. Vaish, "Nitrsct: A software security tool for collection and analysis of kernel calls," in *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*. IEEE, 2019, pp. 510–515. [Online]. Available: https://doi.org/10.1109/tencon.2019.8929513

[8] G. McGraw, "Software security," *IEEE Security & Privacy*, vol. 2, no. 2, pp. 80–83, 2004. [Online]. Available: https://doi.org/10.1109/msecp.2004.1281254

[9] E. Paja, F. Dalpiaz, M. Poggianella, P. Roberti, and P. Giorgini, "Sts-tool: socio-technical security requirements through social commitments," in *2012 20th IEEE International Requirements Engineering Conference (RE)*. IEEE, 2012, pp. 331–332. [Online]. Available: https://doi.org/10.1109/re.2012.6345830

[10] W. Wang, Z. Gao, M. Zhao, Y. Li, J. Liu, and X. Zhang, "Droidensemble: Detecting android malicious applications with ensemble of string and structural static features," *IEEE Access*, vol. 6, pp. 31 798–31 807, 2018. [Online]. Available: https://doi.org/10.1109/access.2018.2835654

[11] C. K. Patanaik, F. A. Barbhuiya, and S. Nandi, "Obfuscated malware detection using api call dependency," in *Proceedings of the First International Conference on Security of Internet of Things*. ACM, 2012, pp. 185–193. [Online]. Available: https://doi.org/10.1145/2490428.2490454

[12] J. Huang, X. Zhang, L. Tan, P. Wang, and B. Liang, "Asdroid: Detecting stealthy behaviors in android applications by user interface and program behavior contradiction," in *Proceedings of the 36th International Conference on Software Engineering*. ACM, 2014, pp. 1036–1046. [Online]. Available: https://doi.org/10.1145/2568225.2568301

[13] M. Fan, J. Liu, X. Luo, K. Chen, Z. Tian, Q. Zheng, and T. Liu, "Android malware familial classification and representative sample selection via frequent subgraph analysis," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 8, pp. 1890–1905, 2018. [Online]. Available: https://doi.org/10.1109/tifs.2018.2806891

[14] R. Canzanese, S. Mancoridis, and M. Kam, "System call-based detection of malicious processes," in *2015 IEEE International Conference on Software Quality, Reliability and Security*. IEEE, 2015, pp. 119–124. [Online]. Available: https://doi.org/10.1109/qrs.2015.26

[15] J. Zhang, Z. Qin, K. Zhang, H. Yin, and J. Zou, "Dalvik opcode graph based android malware variants detection using global topology features," *IEEE Access*, vol. 6, pp. 51 964–51 974, 2018. [Online]. Available: https://doi.org/10.1109/access.2018.2870534

[16] E. Raff, J. Barker, J. Sylvester, R. Brandon, B. Catanzaro, and C. K. Nicholas, "Malware detection by eating a whole exe," in *Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[17] B. Kang, S. Y. Yerima, K. McLaughlin, and S. Sezer, "N-opcode analysis for android malware classification and categorization," in *2016 International Conference On Cyber Security And Protection Of Digital Services (Cyber Security)*. IEEE, 2016, pp. 1–7. [Online]. Available: https://doi.org/10.1109/cybersecpods.2016.7502343

[18] J. G. de la Puerta, B. Sanz, I. Santos, and P. G. Bringas, "Using dalvik opcodes for malware detection on android," in *International Conference on Hybrid Artificial Intelligence Systems*. Springer, 2015, pp. 416–426. [Online]. Available: https://doi.org/10.1093/jigpal/jzx031

[19] E. Garoudja, F. Harrou, Y. Sun, K. Kara, A. Chouder, and S. Silvestre, "Statistical fault detection in photovoltaic systems," *Solar Energy*, vol. 150, pp. 485–499, 2017. [Online]. Available: https://doi.org/10.1016/j.solener.2017.04.043

[20] L. Dong, L. Shulin, and H. Zhang, "A method of anomaly detection and fault diagnosis with online adaptive learning under small training samples," *Pattern Recognition*, vol. 64, pp. 374–385, 2017. [Online]. Available: https://doi.org/10.1016/j.patcog.2016.11.026

[21] J. C. M. Oliveira, K. V. Pontes, I. Sartori, and M. Embiruçu, "Fault detection and diagnosis in dynamic systems using weightless neural networks," *Expert Systems with Applications*, vol. 84, pp. 200–219, 2017. [Online]. Available: https://doi.org/10.1016/j.eswa.2017.05.020

[22] M. Gan, C. P. Chen, H.-X. Li, and L. Chen, "Gradient radial basis function based varying-coefficient autoregressive model for nonlinear and nonstationary time series," *IEEE Signal Processing Letters*, vol. 22, no. 7, pp. 809–812, 2014. [Online]. Available: https://doi.org/10.1109/lsp.2014.2369415

[23] S. Kanarachos, S.-R. G. Christopoulos, A. Chroneos, and M. E. Fitzpatrick, "Detecting anomalies in time series data via a deep learning algorithm combining wavelets, neural networks and hilbert transform," *Expert Systems with Applications*, vol. 85, pp. 292–304, 2017. [Online]. Available: https://doi.org/10.1016/j.eswa.2017.04.028

[24] M. K. Cain, Z. Zhang, and K.-H. Yuan, "Univariate and multivariate skewness and kurtosis for measuring nonnormality: Prevalence, influence and estimation," *Behavior Research Methods*, vol. 49, no. 5, pp. 1716–1735, 2017. [Online]. Available: https://doi.org/10.3758/s13428-016-0814-1

[25] G. R. Iannotti, F. Pittau, C. M. Michel, S. Vulliemoz, and F. Grouiller, "Pulse artifact detection in simultaneous eeg–fmri recording based on eeg map topography," *Brain topography*, vol. 28, no. 1, pp. 21–32, 2015. [Online]. Available: https://doi.org/10.1007/s10548-014-0409-z

[26] K. N. Rajesh and R. Dhuli, "Classification of ecg heartbeats using nonlinear decomposition methods and support vector machine," *Computers in biology and medicine*, vol. 87, pp. 271–284, 2017. [Online]. Available: https://doi.org/10.1016/j.compbiomed.2017.06.006

[27] R. K. Tripathy, S. Deb, and S. Dandapat, "Analysis of physiological signals using state space correlation entropy," *Healthcare technology letters*, vol. 4, no. 1, pp. 30–33, 2017. [Online]. Available: https://doi.org/10.1049/htl.2016.0065

[28] P. Marwaha and R. K. Sunkaria, "Complexity quantification of cardiac variability time series using improved sample entropy (i-sampen)," *Australasian physical & engineering sciences in medicine*, vol. 39, no. 3, pp. 755–763, 2016. [Online]. Available: https://doi.org/10.1007/s13246-016-0457-7

[29] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004. [Online]. Available: https://doi.org/10.1023/b:stco.0000035301.49549.88

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 639–648 DOI: 10.15439/2020F199 ISSN 2300-5963 ACSIS, Vol. 21

Towards Extending UML's Activity Diagram for the Architectural Modeling, Analysis, and Implementation

Mehmet Alp Kose,
Altinbas University, Institute of Graduate Studies,
Istanbul, Turkey
Email: alp.kose@ogr.altinbas.edu.tr

Mert Ozkaya,
Yeditepe University, Department of Computer Engineering, Istanbul,Turkey
Email: mozkaya@cse.yeditepe.edu.tr

Abstract— **SAWUML is a general-purpose software modeling language that extends UML by unifying component and sequence diagrams for the specifications of software architectures. While component diagram is used for modeling the system structures, sequence diagram is extended with the Design-by-Contract approach for the modeling of system behaviors. In this paper, we aim at enhancing the language usability by providing alternative modeling choices for practitioners. To this end, we extended SAWUML's notation set with UML's activity diagram for the behavior modeling. So, practitioners may now use either sequence or activity diagrams, while the system structures are still modeled with component diagrams. We also extended SAWUML's modeling editor for creating software architecture models together with component and activity diagrams and the code generators for automatically obtaining (i) formal models in SPIN's ProMeLa for formal verification and (ii) Java-based implementation. We illustrate our language extension with the gas station case-study.**

I. Introduction

SOFTWARE architecture is the structure of a system that comprises components, their behavioral specifications, and interactions with each other [1]-[3]. The software architecture is concerned with which components a system consists of and whether these components are integrated and working together, as well as what kind of interfaces the components will have, what will be the inter-component communication and dependencies. For modelling software architectures an Architecture Description Language (ADL) plays an important role [4]-[6].

An ADL is a formal specification language for describing the structures and behaviors of components and connectors at an abstraction level for the software architecture of a system. ADLs are designed for different domains (e.g., embedded, automotive, multi-agent, and distributed) and purposes such as modeling software structures, modeling software architectures from different viewpoints (e.g., structure, behavior, concurrency), non-functional property specifications and analysis, formally verifying system behaviors, and code generation.

UML [7] is an ADL that is one of the most widely used modeling languages in industry [8]-[9]. UML is a general-purpose software modeling language that can be used to visually specify the structural and behavioral aspects of any software systems at various levels of abstractions. The structural aspects of software systems can be specified using UML's class diagram, component diagram, or package diagram. The behavioral aspects of software systems can be specified using UML's state diagram, activity diagram, or sequence diagram.

We proposed SAWUML in our previous research [10], which is a UML-based ADL and enables practitioners to use UML's component and sequence diagrams together for the architectural modeling. SAWUML enables to specify the structural aspects of software architectures in terms of component diagrams. SAWUML also enables the behaviors of components to be specified with an extended form of sequence diagrams with Design-by-Contract [11]. SAWUML is supported with a toolset, which consists of a visual modeling editor and a set of code generators. The architectural models in SAWUML can be automatically transformed in SPIN's ProMeLa formal verification language for formally verifying the architectural models against pre-defined (i.e., deadlock and incompleteness) and user-defined linear temporal logic (LTL) [12] properties. Also, SAWUML models can be transformed in Java for facilitating the implementation of software architectures.

In this paper, we aim to improve SAWUML's notation set so as to enhance the language usability. To this end, we extend SAWUML with the activity diagram notation set and intend to offer practitioners two alternative choices for the behavioral modeling. Practitioners may now either use the sequence diagram or the activity diagram depending on what looks more usable and familiar to them. Indeed, while activity diagram is inspired from flowchart and promotes the behavioral modeling in terms of the component activities and their transitions, sequence diagram focuses more on the collaborations of components and promotes the specifications of the order in which the components operate their activities. It should also be noted that we extended with the activity diagram as we believe that the activity diagram is already familiar to many practitioners with different profiles (including those with very limited technical knowledge) [13]-[15]. We also extend SAWUML's existing toolset. The modeling editor now also supports our new behavior notation set that extends the UML activity diagram. Also, we extended the existing code generators for ProMeLa and Java properly. So, while practitioners who feel more comfortable with the UML sequence diagram (i.e., its notation and syntax) may use SAWUML's sequence diagram extension, those who feel comfortable with UML's activity diagram may use the activity diagram extension introduced in this paper so as to model, analyze, and implement their software architectures.

A. Paper Structure

In the rest of the paper, we firstly provide an overview of SAWUML. Next, we discuss the structure and behavior specifications of software architectures. Then, we introduce the extended SAWUML with activity diagrams and their specifications. After that, SAWUML's generators for translation in SPIN's ProMeLa formal verification language and Java code are introduced. We illustrated the extended SAWUML and its toolset via the gas station system. Lastly, we evaluated the extended toolset of SAWUML for the formal verification and software implementation and then conclude the paper.

II. Overview of SAWUML

SAWUML [10] supports both the structural and behavior modeling of software architectures. While the structural aspects of a system are specified with component diagrams, the behavioral aspects are specified with sequence diagrams.

A. Structural Modeling

There are two types of ports defined in SAWUML, which are required and provided. Fig. 1 shows the types of ports.

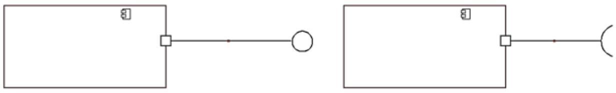
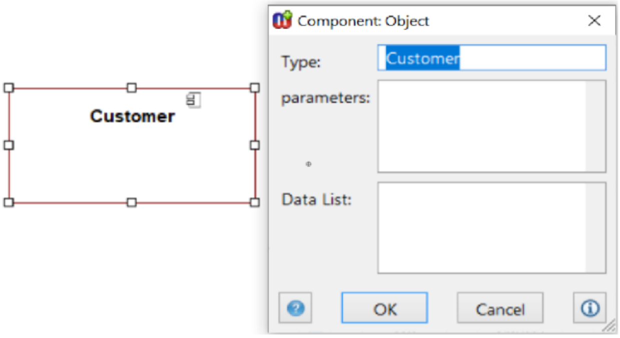


Fig. 1 Components with a provided port and a required port respectively

Every required port is connected to a provided port. Component's required port sends method-call(s) to the provided port of the connected component. The method(s) can take some parameter(s), which may be assigned with arguments upon method-call requests. The required and provided ports are specified in terms of methods that the ports request (if required) or receive the request of (if provided). Note that the required and provided ports of any two components that are connected must be specified with the same set of methods. Indeed, as described in the next section, the required port exhibits the behaviors for sending those method-calls and the provided port exhibits the behaviors for receiving those method-calls.



Component: Object

Type: Customer

parameters:

Data List:

Fig. 2 A component with its specifications

B. Behavioral Modeling

Fig. 2 shows a component with its specifications in SAWUML. When a component box is clicked, a dialog box opens for specifying the component details i.e., component type name, component parameters, and component data list. The type name is unique for every component in a software architecture specifications. Practitioners can pass information to a component through the parameter specification of the component. A component data list represents the state data of the component, which are manipulated by the method-call behaviors operated by the component ports.

For behavioral modeling, a sequence diagram is used. When the relevant port is right clicked, the sequence diagram in a subgraph editor is opened. As seen in Fig. 3 and Fig. 4, there are two life-line objects, the one on the left represents the component with the required port and, the one on the right represents the component with the provided port.

For the sequence diagram of the required port, two arrows are used as depicted in Fig. 3. The solid arrow represents making the method-call to the provided port. The solid arrow herein is supplemented with a contract that consists of pre-condition and promise assignment. The promise herein is used for assigning parameter argument data for the method-call. The dashed arrow represents the method-call response received from the provided port of the connected component. The dashed arrows are supplemented with the pre- and post- condition notations. The pre-condition herein describes the condition on the result data received from the provided port. If pre-condition is satisfied, the post-condition is evaluated, which ensures certain values for the component state data.

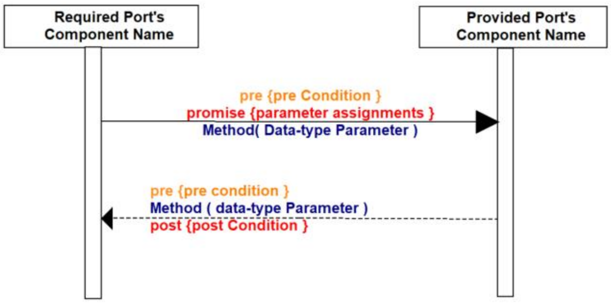

Required Port's Component Name
Provided Port's Component Name
pre {pre Condition }
promise {parameter assignments }
Method(Data-type Parameter)
pre {pre condition }
Method (data-type Parameter)
post {post Condition }

Fig. 3 Required port behavior

There are also two arrows used for the sequence diagram of the provided port as depicted in Fig. 4. The solid arrow represents the receiving method-call from the required port and is supplemented with a contract of pre- and post-conditions. After receiving the method-call from the required port, the pre-condition of the component is checked and if it is satisfied, the data assignments are made in accordance with the post-condition. The dashed arrow indicates a method-call response to the required port back with a return value that is specified via the '\return' notation of the post-condition (*post*).

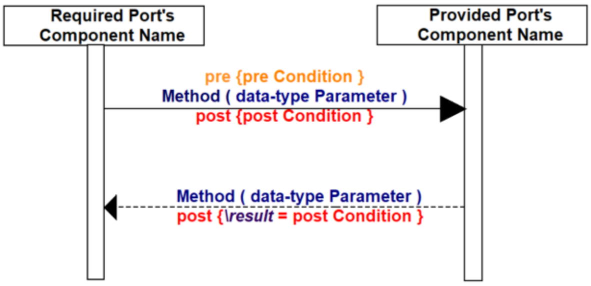

Required Port's Component Name
Provided Port's Component Name
pre {pre Condition }
Method (data-type Parameter)
post {post Condition }
Method (data-type Parameter)
post {\result = post Condition }

Fig. 4 Provided port behavior

III. Extending SAWUML with Activity Diagrams

In this study, we extend SAWUML's behavior modeling with the activity diagram that has been inspired from UML's activity diagram. In this way, practitioners may have the option of selecting either the sequence or activity diagram notation set for the behavioral modeling. It should be noted that the sequence diagram discussed above and the activity diagram extension to be discussed now are both semantically the same but vary in terms of the modeling notations used.

Whenever practitioners double-click on the required port interface icon, a new sub-editor appears as shown in Fig. 5. Using the sub-editor, practitioners can create the activity diagram model to specify the behaviors of the interacting components.

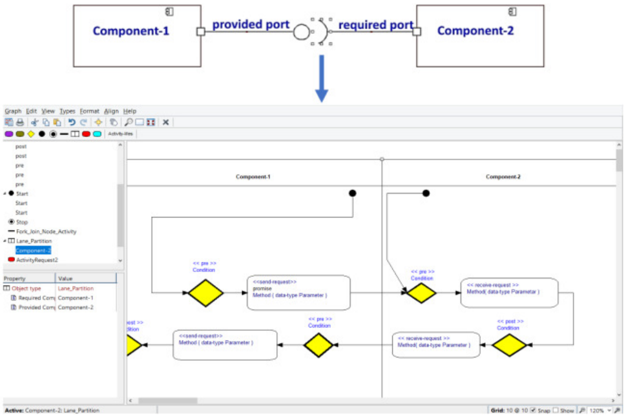

Component-1
provided port
required port
Component-2
Component-1
Component-2
<< pre >> Condition
<<send-request>> promise Method (data-type Parameter)
<< receive-request >> Method(data-type Parameter)
<< post >> Condition
<<send-request>> Method (data-type Parameter)
<< receive-request >> Method(data-type Parameter)

Fig. 5 Accessing to an activity diagram from a component's required port

Fig. 6 shows the notation set for the activity diagram. Although, the activity diagram notation set here is similar to the UML activity diagram, the SAWUML activity diagram has subtle differences, which we discuss in the rest of this section.

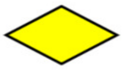
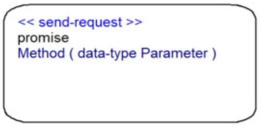
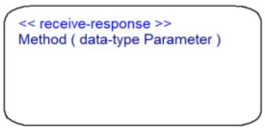
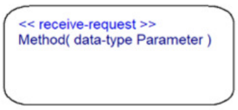
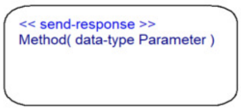
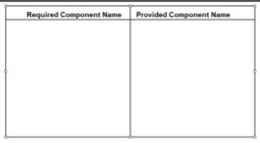
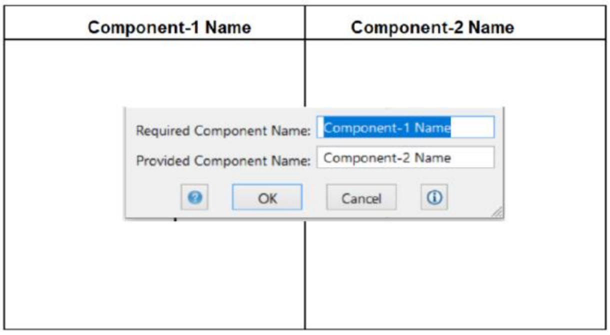
Notation	Visual Symbol
pre/post condition	
an activity of a required component port for sending a request	 << send-request >> promise Method (data-type Parameter)
an activity of a required component port for receiving a response	 << receive-response >> Method (data-type Parameter)
an activity of a provided component port for receiving a request	 << receive-request >> Method(data-type Parameter)
an activity of a provided component port for sending a response	 << send-response >> Method(data-type Parameter)
component lanes	 Required Component Name Provided Component Name
start stop	● ◉

Fig. 6 Design elements of the activity diagram in SAWUML

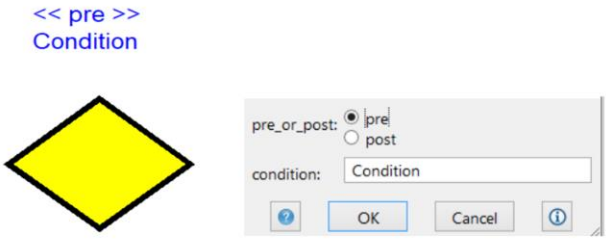
Practitioners need firstly to use the 'component lanes' notation to separate the activities of the interacting component ports. Note that any activities and pre-post-conditions of each component port that represent their method-call behaviors need to be placed in the corresponding lane. While the activities of the required component port are placed in the left lane with the name of the component that is written on the top of the left lane, the activities of the provided component port are placed in the right lane with the name of the component that is written on the top of the right lane (Fig. 7). Start and stop nodes are used for starting and stopping the component behaviors respectively.



Required Component Name: Component-1 Name
Provided Component Name: Component-2 Name

Fig. 7 Component lanes notation with its specifications

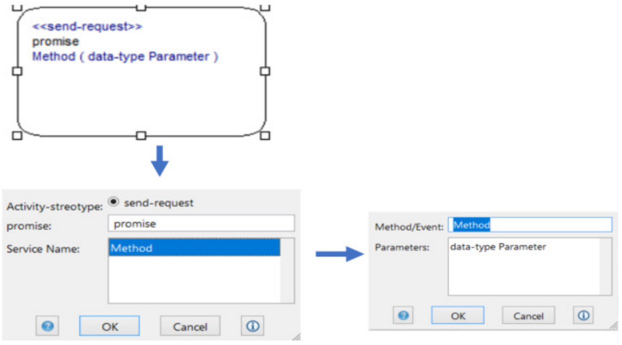
Practitioners may use the pre/post condition notation to specify the pre-post conditions of the method-call behaviors that are operated via the component ports. So, the pre/post condition notation needs to precede/follow the activity notations that represent the method-calls and are explained in the next paragraphs. Fig. 8 shows that whenever the pre/post condition notation is clicked, a new dialog box appears for specifying the type of condition (i.e., pre or post) and the condition statement.



pre_or_post: ● pre ○ post
condition: Condition

Fig. 8 A pre/post notation with specifications

An activity notation of a required component port for sending a request is specified as given in Fig. 6. Whenever the respective notation is clicked, a new dialog box opens as given in Fig. 9. With this dialog box, one can specify the parameter data assignments of the method-call (promise) and the method-call name and parameter list. Note that the pre-condition of the method-call request cannot be specified via the dialog box given in Fig. 9. Practitioners need to use the pre/post-condition notation shown in Fig. 8, which needs to precede the activity notation. So, if the pre-condition is satisfied, the activity specified can be operated.

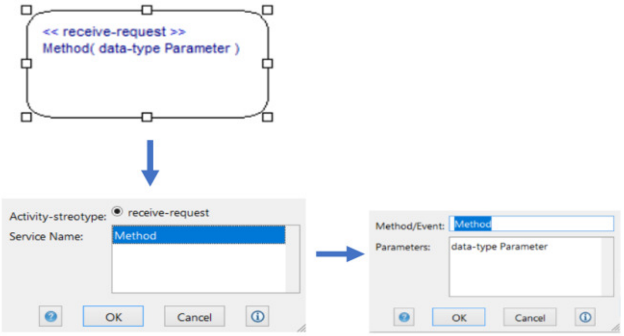


Activity-streotype: ● send-request
promise: promise
Service Name: Method

Method/Event: Method
Parameters: data-type Parameter

Fig. 9 Specifications of an activity of a required component port for sending a request

Whenever an activity of a required component port for sending a request is operated, this may be followed by the activity of a provided component port for receiving that request. So, the activity for receiving a request is specified as shown in Fig. 10. Note again that the activity for receiving a request here may be preceded by the pre/post-condition symbol to specify the pre-condition on the provided port's method-call receipt.

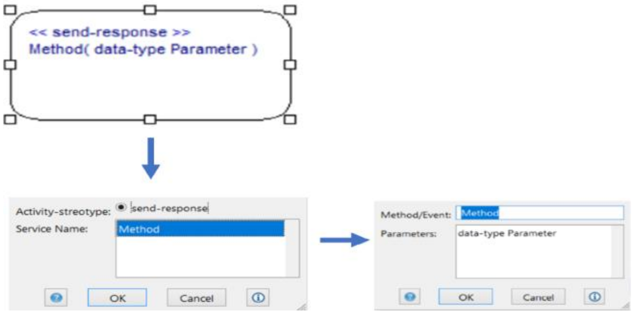


Activity-streotype: ● receive-request
Service Name: Method

Method/Event: Method
Parameters: data-type Parameter

Fig. 10 Specifications of an activity of a provided component port for receiving a request

Whenever an activity of a provided component port for receiving a request is operated, practitioners may use the pre/post-condition notation for the post-condition to ensure that the data will be assigned (if any needed). This is followed by the activity of a provided component port for sending the method-call response. The activity for sending a response is specified as shown in Fig. 11. The pre/post-condition notation may be used after the activity for sending the response so as to the post-condition on the return value of the method-call.

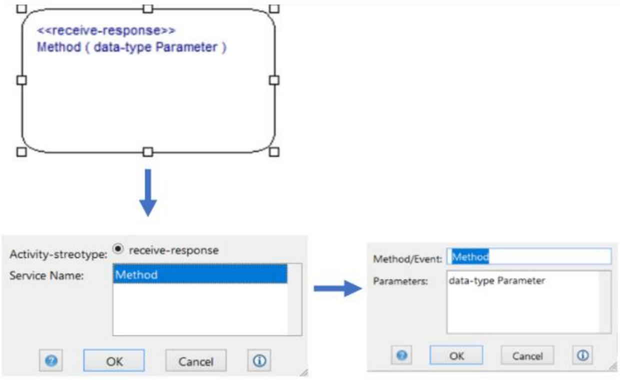


Activity-streotype: ● send-response
Service Name: Method

Method/Event: Method
Parameters: data-type Parameter

Fig. 11 Specifications of an activity of a provided component port for sending a response

Whenever an activity of a provided component port for sending a response is operated, the pre/post-condition notation may need to be used to check if the pre-condition on receiving a method-call response for the required port is satisfied. If so, the activity of a required component port for receiving that response can be operated. The activity for receiving a response is specified as shown in Fig. 12. After the activity for receiving a method-call response is operated, the pre/post-condition notation may be used to specify the post-condition that ensures the post-state of the component.



<<receive-response>>
Method (data-type Parameter)

Activity-streotype: ● receive-response
Service Name: Method

Method/Event: Method
Parameters: data-type Parameter

Fig. 12 Specifications of an activity of a required component port for receiving a response

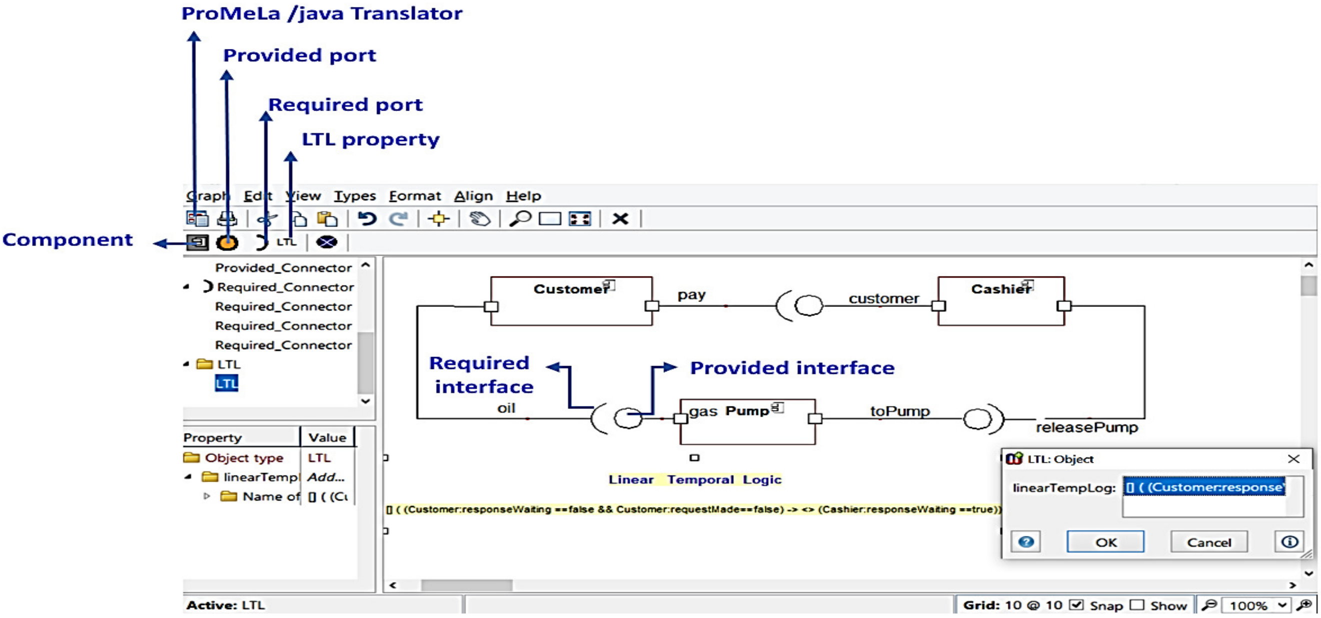
IV. METAEDIT+ BASED TOOL SUPPORT

We used the MetaEdit+ [16] meta-modeling tool to develop the modeling editors and code generators for SAWUML. We defined the language abstract and concrete syntax with MetaEdit+’s GOPPRR meta-modeling framework, which then gave us the supporting modeling editor as depicted in Fig. 13.

In this study, the modeling editor, previously developed with MetaEdit+ has been extended to support the activity diagram notation introduced in the previous section.

We also used MetaEdit+ MERL[1] code generation definition language to extend the ProMeLa and Java code-generators to support the activity diagram notation set.

Selecting the icons (Component, Required port, Provided port) on the tool bar given in the modeling editor (as depicted in Fig. 13) creates the respective of the design element objects in the drawing area of the component diagram editor. Later a component object and a provided/required port object can be connected. When double-clicked, the practitioner is offered two options, to open an activity or a sequence diagram. By pressing the generator icon (ProMeLa/java Translator), a dialog opens, which allows practitioners to select one of the generators (i.e., java generator with activity/sequence diagram or ProMeLa generator with activity/sequence diagram) that are available as in Fig. 14 to run. By pressing the LTL property button, a dialog box opens on the drawing area of the component modeling editor, and the user-defined linear temporal logic (LTL) [12] properties can be entered as shown Fig. 15.



Property	Value
Object type	LTL
linearTempl	Add...
Name of	[] ((Ci

[] ((Customer:responseWaiting ==false && Customer:requestMade==false) -> <> (Cashier:responseWaiting ==true))

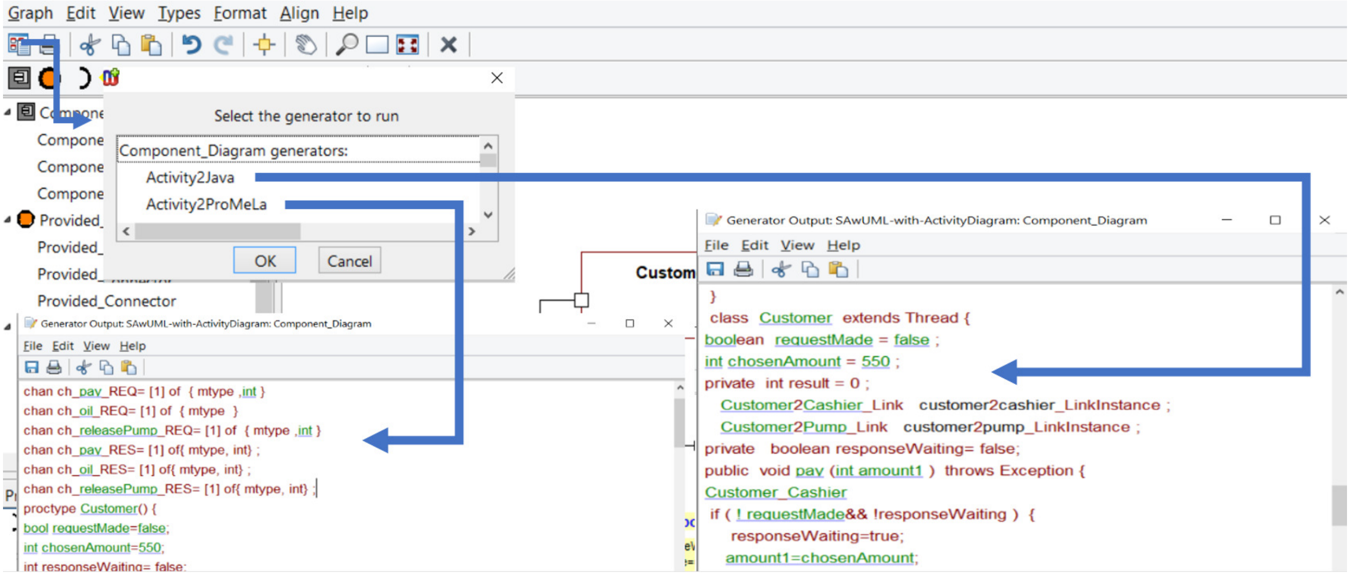
linearTempLog: [] ((Customer:response

Active: LTL

Grid: 10 @ 10 ☑ Snap □ Show 100%

Fig. 13 The modeling editor of a component diagram

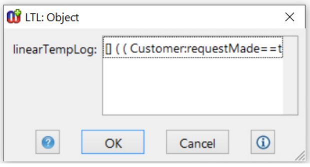
[1] MetaEdit+’s MERL language website: https://www.metacase.com/support/55/manuals/mwb/Mw-5_2_1.html


Graph Edit View Types Format Align Help
Select the generator to run
Component_Diagram generators:
Activity2Java
Activity2ProMeLa
OK
Cancel
Compone
Compone
Compone
Compone
Provided_
Provided_
Provided_
Provided_Connector
Custom
Generator Output: SAwUML-with-ActivityDiagram: Component_Diagram
File Edit View Help
Generator Output: SAwUML-with-ActivityDiagram: Component_Diagram
File Edit View Help

```
chan ch_pay_REQ= [1] of  { mtype ,int }
chan ch_oil_REQ= [1] of  { mtype  }
chan ch_releasePump_REQ= [1] of  { mtype ,int }
chan ch_pay_RES= [1] of{ mtype, int} ;
chan ch_oil_RES= [1] of{ mtype, int} ;
chan ch_releasePump_RES= [1] of{ mtype, int} ;
proctype Customer() {
bool requestMade=false;
int chosenAmount=550;
int responseWaiting= false;
```

```
}
 class  Customer  extends Thread {
boolean  requestMade = false ;
int chosenAmount = 550 ;
private  int result = 0 ;
  Customer2Cashier_Link   customer2cashier_LinkInstance ;
  Customer2Pump_Link   customer2pump_LinkInstance ;
private   boolean responseWaiting= false;
public  void pay (int amount1 )  throws Exception {
Customer_Cashier
 if ( ! requestMade&& !responseWaiting )  {
    responseWaiting=true;
   amount1=chosenAmount;
```

Fig. 14 The results of the ProMeLa generator and the java generator of a model



linearTempLog: [] ((Customer:requestMade==t

Linear Temporal Logic

[] ((Customer:requestMade==true) -> <> (Pump:pumpReleased ==true))

Fig. 15 Example of an LTL property

After running the ProMeLa generator, LTL properties are translated according to the LTL syntax in the ProMeLa language and embedded in the ProMeLA model obtained from the generated SAWUML model. So, using the SPIN model checker [17], the generated ProMeLA model can be formally verified for the LTL-based user-defined properties. Besides the user-defined properties, SPIN also checks a couple of pre-defined properties (i.e., deadlock, incompleteness) that we introduced as part of the ProMeLa translation algorithms. A deadlock error happens when the component processes get stuck executing and none of them will be able to reach their end states. Incompleteness happens if the response behavior specifications for a required port cannot handle all possible cases properly. The code for checking the pre-defined properties have been encoded as part of the ProMeLa code generator. Since there is not enough space in the article, the java generator algorithm[2], ProMeLa generator algorithm[3] and SAWUML toolset[4] can be accessible via the website.

V. Gas Station Case Study

The gas station system [18] is composed of three components that interact with each other. These are the customer, cashier and pump components. The customer component gets gas from the pump component if the customer component pays to the cashier component. Fig. 16 shows the component diagram specification of the gas station system in SAWUML.

Each component's data list specifications are shown in Fig. 17, Fig. 18, Fig. 19. The activity diagrams for the gas station system are shown in Fig. 20, Fig. 21, Fig. 22.

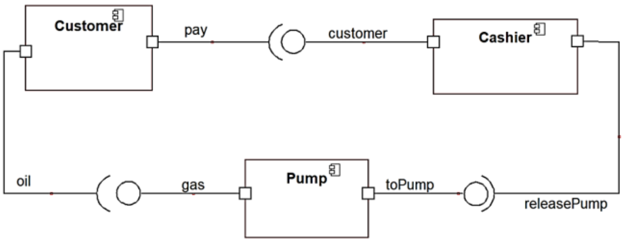
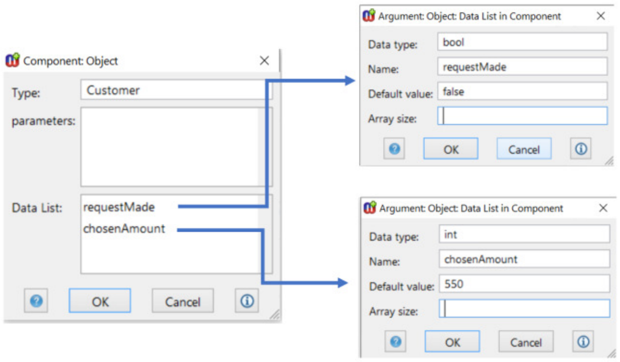


Fig. 16 Gas station in the SAWUML model


Component: Object
Argument: Object: Data List in Component
Argument: Object: Data List in Component
OK
Cancel
OK
Cancel
OK
Cancel

Type: Customer
parameters:
Data List: requestMade
chosenAmount

Data type: bool
Name: requestMade
Default value: false
Array size:

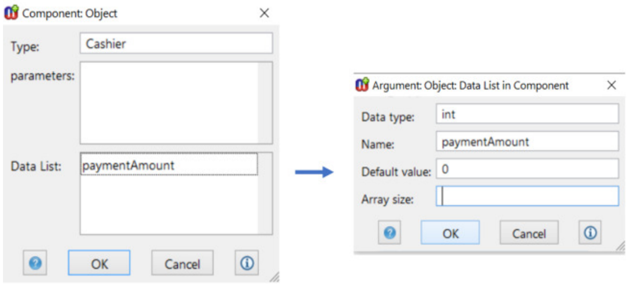
Data type: int
Name: chosenAmount
Default value: 550
Array size:

Fig. 17 The data list of the customer component

[2] SAWUML's java generator algorithm website: https://sites.google.com/view/mkose/javaalgorithm

[3] SAWUML's ProMeLa generator algorithm website: https://sites.google.com/view/mkose/promelaalgortihm

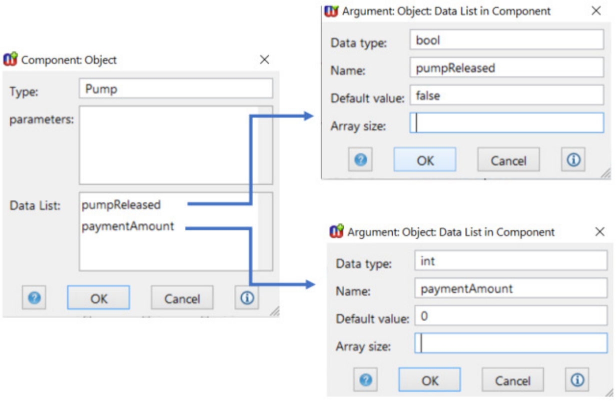
[4] SAWUML's toolset website: https://sites.google.com/view/mkose/sawuml-toolset



Type: Cashier
parameters:
Data List: paymentAmount

Data type: int
Name: paymentAmount
Default value: 0
Array size:

Fig. 18 The data list of the cashier component


Component: Object
Argument: Object: Data List in Component
Argument: Object: Data List in Component
OK
Cancel

Type: Pump
parameters:
Data List: pumpReleased
paymentAmount

Data type: bool
Name: pumpReleased
Default value: false
Array size:

Data type: int
Name: paymentAmount
Default value: 0
Array size:

Fig. 19 The data list of the pump component

In the customer-cashier activity diagram in Fig. 20, before making the *pay* method-call via the customer's required port, a pre-condition (*!requestMade*) is checked via the pre/post-condition symbol. If it is satisfied, the activity for making the *pay* method-call is operated. The activity for the *pay* method-call includes a promise data assignment (*amount=chosenAmount*). After the *pay* method-call is received by the cashier's provided port, firstly the pre-condition is checked via the pre/post-condition symbol (*paymentAmount==0*). If it is satisfied, the cashier's activity for receiving the *pay* method-call request is operated. Then, the post-condition is ensured via the pre-/post symbol (*paymentAmount=amount*). Then, the cashier's send-response activity for the *pay* method-call is operated to send the response to the customer. The customer receives back the *pay* method-call response via its receive-response activity under no pre-condition. After the customer operates the activity for the method-call response, the post-condition is ensured via the pre/post-condition symbol (*requestMade=true*).

In the cashier-pump activity diagram in Fig. 21, before making the *releasedPump* method-call via the cashier's required port, a pre-condition (*paymentAmount!=0*) is checked via the pre/post-condition symbol. If it is satisfied, the cashier's activity for making the *releasedPump* method-call request activity is operated. The activity for the *releasedPump* method-call includes a promise data assignment (*amount2=paymentAmount*). After the *releasedPump* method-call is received by the pump's provided port, the pre-condition is checked via the pre/post-condition symbol *(!pumpReleased*). If satisfied, the activity for receiving the *releasedPump* method-call is operated and the post-condition is ensured via the pre/post-condition symbol (*pumpReleased=true, paymentAmout=amount2*). Then, the pump's send-response activity for the *releasedPump* method-call is operated to send the response to the cashier. The cashier receives back the *releasedPump* method-call response via its receive-response activity under no pre-condition. A post-condition ensures that the data will be assigned (*paymentAmount=0*) via the pre/post-condition symbol.

Fig. 22 gives the activity diagram specification for the customer and pump relationships. Before the customer makes a *pump* method-call via its required port, a pre-condition (*requestMade==true*) is checked via the pre/post-condition symbol. If it is satisfied, the activity for making the *pump* method-call is operated. Whenever the customer sends the method-call request for the pump, firstly, the pre/post-condition symbol of the pump (*pumpReleased==true*) is checked and if it is satisfied, the activity for receiving the pump request is operated. Then, the pre/post-condition symbol for the post-condition of the pump is ensured (*pumpReleased= false*). Afterward, the pump component operates the activity for sending the *pump* method-call response to the customer. The *pump* method-call is sent back with a return value. The pre/post-condition symbol is employed here to state the post-condition on the return value *(\result == paymentAmount*). When the customer receives the *pump* response, the pre-/post-condition symbol is used to operate the pre-condition that compares the return value (*result*) with the *chosenAmount* variable. If both are equal (*result==chosenAmount*) then the activity for the receiving the response for the *pump* method-call can be operated and then the pre/post-condition symbol for the post-condition is ensured (*requestMade = false*).

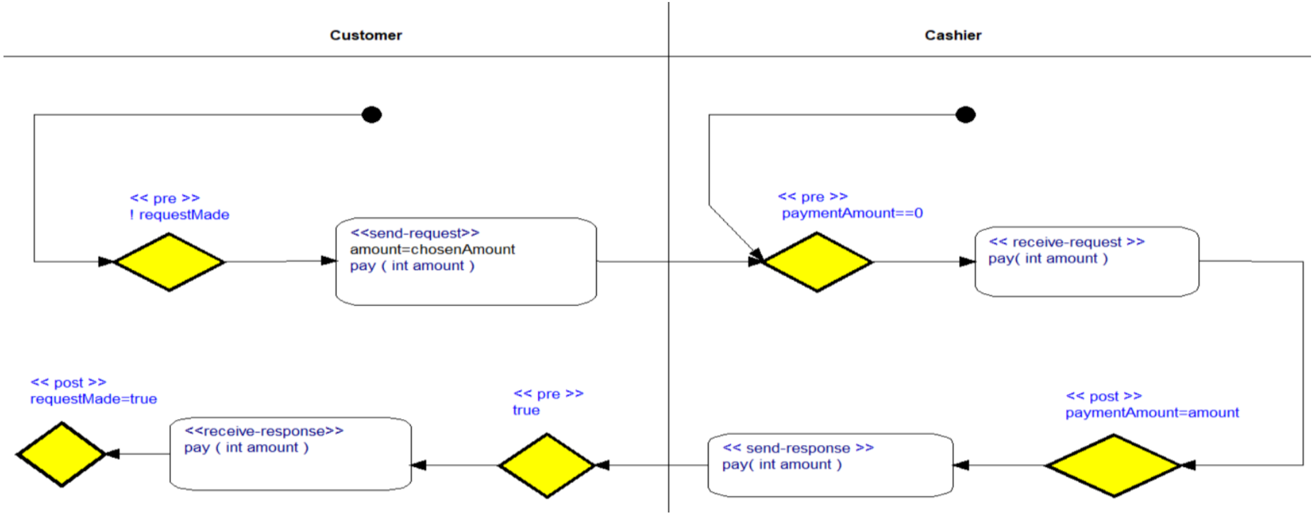


Fig. 20 The activity diagram of customer-cashier components

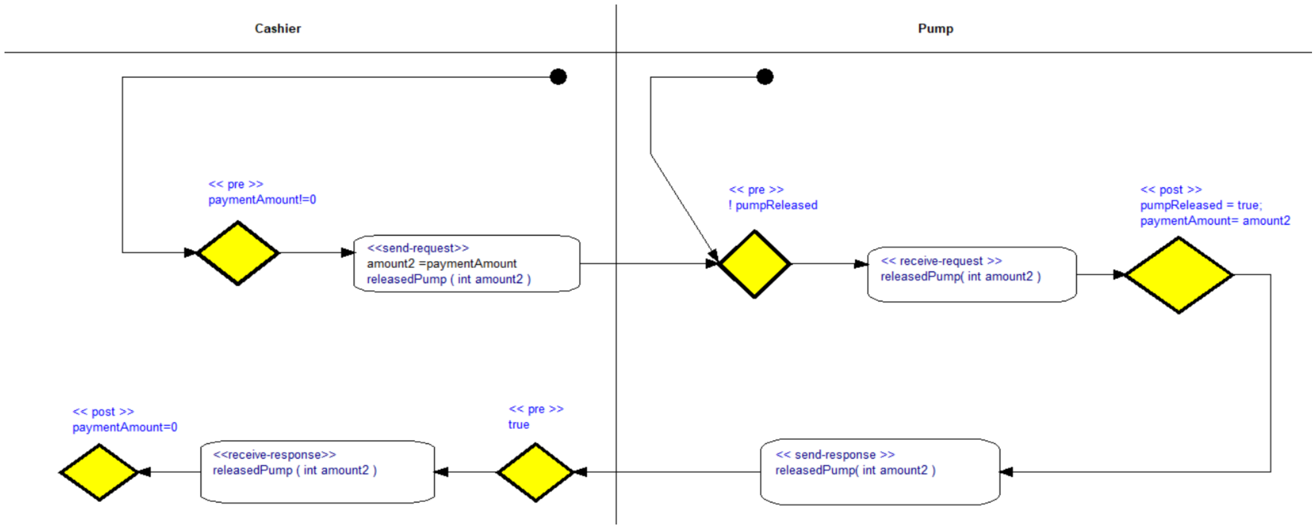


Fig. 21 The activity diagram of cashier-pump components

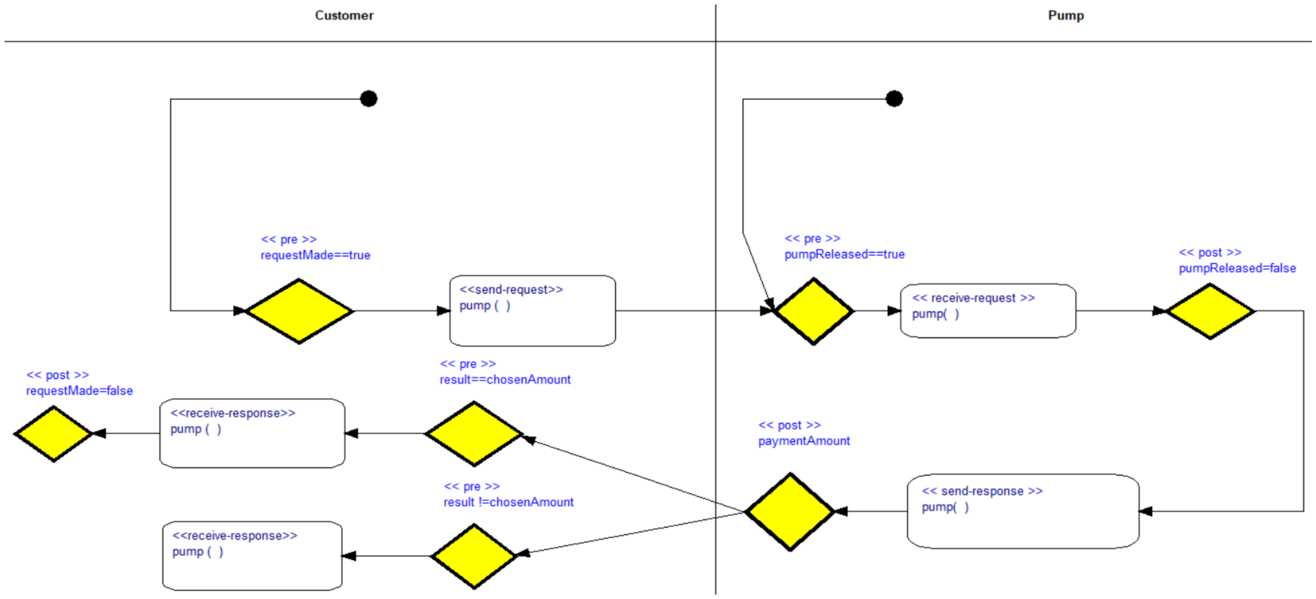


Fig. 22 The activity diagram of customer-pump components

Using the LTL property icon in the editor, the user-defined LTL property has been specified as part of the gas station specification as shown in Fig. 23. The LTL property here states that a particular constraint must always be satisfied. That is, when *requestMade* data of a customer is *true*, then eventually the pump's *pumpReleased* data becomes *true*. This means that whenever the customer sends a gas request to the cashier, the pump will eventually receive a pump-release request from the cashier.

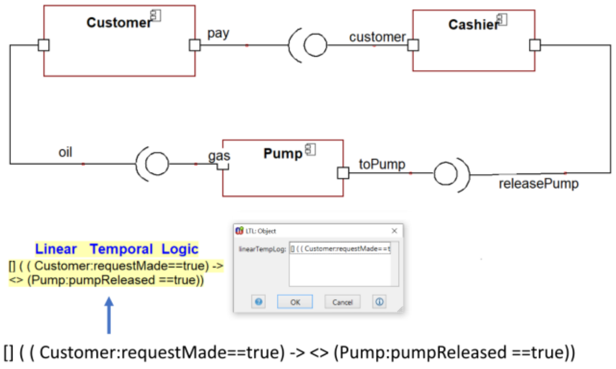

Customer
pay
customer
Cashier
oil
gas
Pump
toPump
releasePump
Linear Temporal Logic
[] ((Customer:requestMade==true) -> <> (Pump:pumpReleased ==true))
LTL Object
linearTempLogic
OK
Cancel
[] ((Customer:requestMade==true) -> <> (Pump:pumpReleased ==true))

Fig. 23 Specifying an LTL property for the gas station model in SAWUML

TABLE I.
FORMAL VERIFICATION RESULTS IN SPIN

Case Studies	Sector-vector (bytes)	States		Memory (Mb)	Time (s)
		Stored	Matched		
Gas station – 1 customer	144	101	59	64.539	0
Gas station – 3 customers	364	121988	148130	106.922	0.413
Gas station – 5 customers	584	3462574	7813281	2016.661	25.4

VI. TOOL EVALUATION

After we specified the gas station model in SAWUML as discussed in Section 5, we used SAWUML's toolset for automatically transforming the gas station model into a formal ProMeLa model that can be accepted by the SPIN model checker and Java code for obtaining the implementation of the gas station model. We considered three different configurations of the gas station model, which vary depending on the number of customers involved (gas station with 1, 3, and 5 customers).

A. Formal Verification

Table I[5] shows the formal verification results that have been produced by the SPIN model checker for each configuration of the gas station model (namely their ProMeLa translations). The SPIN formal verification results are given with (i) the size of the states in the system's state space, (ii) the number of the stored state in the state space, (iii) the number of the matched states that are revisited during the state space search, (iv) the total actual memory usage of the state space, and (v) the elapsed time for the exhaustive analysis of the state space.

Whenever a deadlock occurs, an *invalid end state error* is generated by the SPIN model checker, which indicates that the running component processes cannot reach at the end of their code. To illustrate a deadlock situation, we used the gas station with one customer. We intentionally changed the *requestMade* data of customer's pay port to *true*. So, customer's both pay and oil port pre-conditions are now the same. The end result is that the customer is waiting for making a payment or a pump request, the cashier is waiting for a payment from the customer, the pump is waiting for receiving a release-gas request from the cashier. So, none of the components reach the end of their states and that causes deadlock.

To illustrate an incompleteness error, we again used the gas station with one customer. The response behavior specification for the customer's pump method consists of two cases. The case when the *result* is equal to *chosenAmount*, and the case when the result is not. So, we did not get any incompleteness error. If however the response behavior specification here failed to consider all possible cases (e.g., suppose "the *result* is equal to the *chosenAmount*" case is absent), then we would get an *assertion violation error* via the SPIN model checker.

Lastly, the gas station model has been verified for the user-defined LTL property specified in Fig. 23. The translated ProMeLa model actually includes the LTL translation in ProMeLa. So, whenever we run the ProMeLa model with the SPIN model checker, we successfully performed the formal verification for the LTL property. Note that if the LTL property was violated, we would get an *assertion violation error* in SPIN.

B. Java Implementation

After formally verifying the gas station model, we automatically produced Java code using SAWUML's toolset. Java implementation was created according to the Adapter Design Pattern to enhance the code modularity and understandability. Basically, the adaptee (e.g., the customer) component sends a request to the adapter via the component interface and the adapter transmits this request to another adaptee component (e.g., cashier in customer to cashier relationship).

It should be noted that the produced code includes the structural and behavioral architectural design decisions of the model. Practitioners can develop their systems with other necessary modules (i.e., network, GUI, database connection, etc.) starting from this code.

Since there is not enough space in the article, the generated java file (Configuration.java)[6], and the ProMeLa file (gasStation.pml)[7] can be accessible via the web site.

[5] Spin Version 6.4.5 is used with 2.4 GHz Intel Core i7-8750H, 16GB of RAM, and Windows 10 Home OS. We run the following the SPIN commands which are *spin -a GasStation.pml*, *gcc -o pan pan.c,* and *pan* (Note that for the gas station with 5 customers *pan -m800000* command is used.)

[6] The java file of the gas station system's web site: https://sites.google.com/view/mkose/javafile

[7] The ProMeLa file of the gas station system's web site: https://sites.google.com/view/mkose/promelafile

VII.Discussion & Conclusions

SAWUML is an ADL that uses UML's component and sequential diagrams for the specification of the structural and behavioral design decisions. SAWUML extends the sequential diagram using the Design-by Contract approach to define behavioral specifications of components' methods to send /receive each other. SAWUML is supported with a modeling editor to design architecture modeling and to specify user-defined properties in the form of LTL. The SAWUML models can be automatically transformed in SPIN's ProMeLa formal verification language for checking pre-defined properties (deadlock, incompleteness) and user-defined LTL properties.

In this study, we extended SAWUML by introducing the notation set for the extended activity diagram for modeling the behavioral design decisions. Practitioners may now have the options of selecting either the activity diagrams or sequence diagrams for the behavioral modeling. This is actually intended for enhancing the language usability and providing practitioners different types of notation sets among which they can choose the one that best fit their expertise. We also extended SAWUML's code-generator toolset to enable the architectural models with activity diagrams to be formally verified via the SPIN model checker and transformed into the Java-based implementation.

We evaluated our approach with the gas station system, where we specified the structural and behavioral design decisions with the component and activity diagrams respectively. We then used SAWUML's code generators to transform the models in SPIN's ProMeLa and used SPIN to formally verify the behavioral design decisions. We further automatically generated Java code from the gas station models, which is based on the Adapter design pattern.

SAWUML may actually be considered by any practitioners who use UML to model their software architectures from the structural and behavioral viewpoints. While UML and many tools that support UML do not allow for formally analyzing UML models, SAWUML does so. Moreover, SAWUML integrates the structural modeling with behavioral modeling – i.e., practitioners actually click on the component ports to specify their behaviors with sequence/activity diagrams. Note that this is not possible with UML and practitioners are forced to specify the structural and behavioral models that are cleanly separated. Moreover, in SAWUML we extend the UML sequence and activity diagrams with Design-by-Contract so as to enable practitioners to specify not only the interactions but also the behaviors in terms of pre- and post-conditions on the component state.

As a future work, aim at developing a tool that can reverse engineer the Java model back to the SAWUML model. By doing so, we aim at enabling the existing (i.e., already implemented) projects to be modeled and analyzed automatically and the developers to determine any architecture erosions [19].

VIII.References

[1] Len Bass, Paul Clements, and Rick Kazman, *Software Architecture in Practice*, 2nd ed. Addison-Wesley Proffesional, 2003, pp. 19-26.

[2] N. Medvidovic and R. N. Taylor, "Software architecture: foundations, theory, and practice," *2010 ACM/IEEE 32nd International Conference on Software Engineering*, Cape Town, 2010, pp. 471-472, doi: 10.1145/1810295.1810435.

[3] David Garlan and Mary Shaw, "An introduction to software architecture". *Advances in Software Engineering and Knowledge Engineering,* 1993, pp. 1-39.

https://doi.org/10.1142/9789812798039_0001

[4] Ozkaya M. "The analysis of architectural languages for the needs of practitioners". *Softw Pract Exper*. 2018; 48: 985– 1018.

https://doi.org/10.1002/spe.2561

[5] N. Medvidovic and R. N. Taylor, "A classification and comparison framework for software architecture description languages," in *IEEE Transactions on Software Engineering*, vol. 26, no. 1, pp. 70-93, Jan. 2000, doi: 10.1109/32.825767.

[6] P. C. Clements, "A survey of architecture description languages," *Proceedings of the 8th International Workshop on Software Specification and Design*, Schloss Velen, Germany, 1996, pp. 16-25, doi: 10.1109/IWSSD.1996.501143.

[7] Object Management Group. OMG unified modeling language secification –version 2.5. http://www.omg.org/spec/UML/2.5/; 2015. URL http://www.omg.org/spec/UML/2.5/.

[8] Ozkaya M. "Do the informal & formal software modeling notations satisfy practitioners for software architecture modeling?" *Inf Softw Technol* 2017; 95: 15–33. doi: 10.1016/j.infsof.2017.10.008.

[9] I. Malavolta, P. Lago, H. Muccini, P. Pelliccione and A. Tang, "What industry needs from architectural languages: a survey," in *IEEE Transactions on Software Engineering*, vol. 39, no. 6, pp. 869-891, June 2013, doi: 10.1109/TSE.2012.74.

[10] Ozkaya M. and Kose M. A. "SAwUML – UML-based, contractual software architectures and their formal analysis using SPIN". *Journal of Computer Languages, Systems and Structures*, 2018; 54: 71- 94. https://doi.org/10.1016/j.cl.2018.04.005

[11] B. Meyer, "Applying 'design by contract'," in *Computer*, vol. 25, no. 10, pp. 40-51, Oct. 1992, doi: 10.1109/2.161279.

[12] A. Pnueli, "The temporal logic of programs," *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*, Providence, RI, USA, 1977, pp. 46-57, doi: 10.1109/SFCS.1977.32

[13] Reggio, G., Leotta, M., Ricca, F., Clerissi, D. "What are the used UML diagrams? a preliminary survey". *Proceedings of 3rd International Workshop on Experiences and Empirical Studies in Software Mod*eling (EESSMod 2013), vol. 1078, pp. 3–12. *CEUR Workshop Proceedings,* 2013.

[14] Wrycza, Stanisław & Marcinkowski, Bartosz. "A light version of UML2:survey and outcomes". 2007, doi:10.13140/RG.2.1.3445.1046.

[15] G. Reggio, M. Leotta and F. Ricca, ""Precise is better than light" a document analysis study about quality of business process models," *Workshop on Empirical Requirements Engineering (EmpiRE 2011)*, Trento, 2011, pp. 61-68, doi: 10.1109/EmpiRE.2011.6046257.

[16] Kelly S, Lyytinen K, Rossi M. "Metaedit+ a fully configurable multi-user and multi-tool CASE and CAME environment". In: Bubenko J, Krogstie J, Pastor O, Pernici B, Rolland C, Sølvberg A, editors. *Seminal contributions to information systems engineering, 25 years of CAiSE*. Springer; 2013. p. 109–29. ISBN 978-3-642-36925-4. doi: 10.1007/978- 3- 642- 36926- 1 _ 9 .

[17] Holzmann GJ. "*The SPIN Model Checker - primer and reference manual*". Addison-Wesley Professional, 2003, ISBN 978-0-321-22862-8.

[18] Naumovich G , Avrunin GS , Clarke LA , Osterweil LJ. "Applying static analysis to software architectures". In: Jazayeri M, Schauer H, editors. *Software engineering–ESEC/FSE'97. Lecture Notes in Computer Science,* 1301. Springer; 1997. pp. 77–93. ISBN 3-540-63531-9.

[19] Dewayne E. Perry and Alexander L. Wolf. 1992. "Foundations for the study of software architecture". *SIGSOFT Softw. Eng. Notes* 17, 4 (Oct. 1992), pp. 40–52. DOI:https://doi.org/10.1145/141874.141884

DOI: 10.15439/2020F147
ISSN 2300-5963 ACSIS, Vol. 21

The Use of ARCore Technology for Online Control Simulations

Matúš Pohančenik, Jakub Matišák and Katarína Žáková
Faculty of Electrical Engineering and Information Technology
Slovak University of Technology in Bratislava
Ilkovičova 3, 812 19, Bratislava, Slovakia
xpohancenikm@stuba.sk, jakub.matisak@stuba.sk, katarina.zakova@stuba.sk

***Abstract*—The paper describes an educational mobile application that controls the 3D model of towercopter using augmented reality for smartphones. It is developed using the ARCore technology that allows insertion of 3D objects into a real space via smartphone or tablet. The application serves as a simple guide for a real device which is placed in laboratory and enables to create simulations based on user input data. The application interface is connected with Scilab API simulation module that provides data for 3D model animations. Users can set their own controller parameters into the predefined control structures. Application is a part of virtual laboratory and can help students with understanding of problems connected with education process.**

***Index Terms*—ARCore, towercopter, augmented reality, computer based education, 3D model**

I. Introduction

Augmented reality is a technology that enriches our physical world and adds digital information to it. In simple terms, it is a kind of a system combining physical and virtual worlds with the most accurate connection of virtual and real objects. Widespread adoption of augmented reality has been on the rise in recent years and today its use can be seen in a variety of areas from the entertainment and industry (e.g. [1]) to medicine (e.g. [2]), teaching (e.g. [3]) and to the military.

In the presented paper we focus on the field of interactive education. Many researches proved that interactive AR education can help with students results (e.g. [4]). We think that augmented reality has a huge potential to change the ways and methods of education, and has the opportunity to make it much more engaging. Most students are active users of smartphones that they use to access social networks or play games, but they don't use smartphones for study purposes. AR for smartphones with additional information has a huge potential to provide students better and easier understanding of complex information.

Augmented reality is still a relatively unknown concept in the educational process, even though it provides completely new and more interesting ways of learning. It makes it easier for teachers to get students' attention and motivate them, while students get new tools to visualize their subjects and complex concepts (e.g. [5]), as well as to gain practical skills.

The aim of this work is to create an application for mobile devices using AR, which will allow the rendering of complex objects and transform them into virtual 3D models, thus facilitating the understanding of abstract and complex content. Large numbers of people better absorb new information by direct seeing how they behave and function. The goal of our application is therefore to focus genuinely on visual learning and thus help students to better understand theoretical knowledge on real devices.

II. Application

The created application enables to simulate the behavior of a towercopter, which is a laboratory model of a flying machine with a one degree of freedom ([6], [7]) and should serve as an aid in teaching on subjects dealing with the basics of automatic control. In the next part of the work we will introduce the application in more detail. The paper describes architecture of the application, system requirements, used technologies and some challenges or problems we encountered during development.

A. Development Tools

The rapid growth of AR has required significant investment from leading technology companies. Technology that once seemed futuristic is now a reality thanks to the developments that have made it possible. It all started with technology giants Google and Apple, who introduced development tools for creating AR applications for Android and iOS devices [8].

After studying the issue of augmented reality, all available platforms and its development [9], we decided to develop the application using the Google ARCore framework, that operates on devices with the Android operating system. As a development environment, the Unity platform has been chosen because of its compatibility with the ARCore and previous experience. The application can be run on all Android devices that support the augmented reality and run at least version 7.0 of the Android operating system. Devices must have installed the ARCore, which can be downloaded from the official Google Play store.

This work has been supported by the Slovak Grant Agency, grant VEGA No. 1/0733/16, the grant APVV SK-IL-RD-18-0008, by Young Researchers grant program of the Slovak University of Technology, the grant 3DVinOE and by Tatra Banka Foundation grant program E-talent, the grant No. 2018et016. This support is gratefully acknowledged.

B. Application Requirements

The app should provide a full-fledged augmented reality experience. It should scan the environment and recognize vertical and horizontal planes and allows the user to place virtual 3D models of towercopter on these scanned surfaces. It should also be possible to move 3D models on these surfaces and rotate them.

The used devices must have a camera with a sufficiently high-quality image and sensors to monitor the movement of the device so that it is possible to determine as accurately as possible how the device moves in real time and thus ensure the most accurate mapping of the surrounding environment. Last but not least, the device must have a sufficiently powerful processor be able to make calculations related to mapping and definition of objects quickly and accurately. A complete list of supported devices can be found on the official Google ARCore website [10].

The following list summarizes all requirements to the application. It should allow *adding objects to scene, objects manipulation, models simulation, graph rendering, displaying details of components and data saving and sharing.*

C. System Architecture

The ARCore app essentially works like a game. There is a certain scene on which each frame is drawn, where it updates images. This process is shown in Fig.1 in the ”Image Rendering” node. The whole life cycle of application is also shown in Fig. 1.

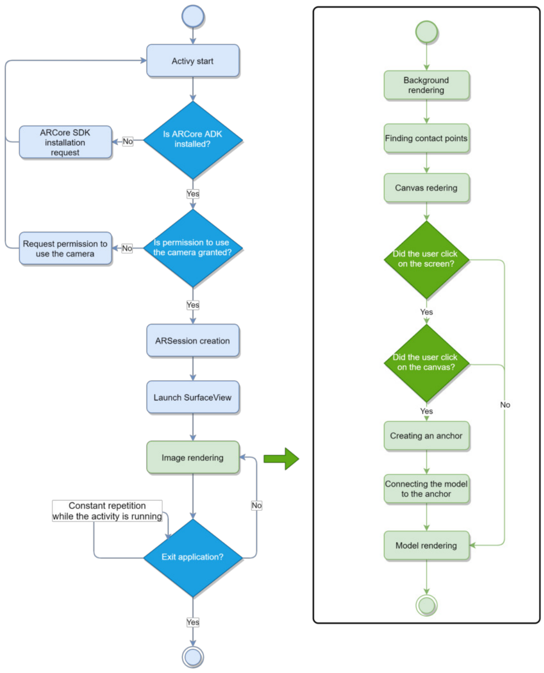

Activy start
Is ARCore ADK installed?
No
ARCore SDK installation request
Yes
Is permission to use the camera granted?
No
Request permission to use the camera
Yes
ARSession creation
Launch SurfaceView
Image rendering
Constant repetition while the activity is running
No
Exit application?
Yes
Background rendering
Finding contact points
Canvas redering
Did the user click on the screen?
Yes
Did the user click on the canvas?
Yes
No
Creating an anchor
Connecting the model to the anchor
Model rendering

Fig. 1. Life cycle of the scene

The main entry point for the ARCore API is the Session class. This class manages the state of the system and handles the life cycle of the entire application. It allows the user to create a relation, configure it, start or stop it, but most importantly it allows access to the camera image and the position of the device. However, before creating a new session, it is necessary to verify whether ARCore is installed. It is also necessary to have to have the permission to access the camera equipment granted. If one of these conditions is not met, it is not possible to create a session and it is not possible to start and use the application [11].

The image rendering action itself consists of several smaller operations that must be performed sequentially: *background rendering, finding points of contact, canvas rendering and model rendering.*

D. 3D Model

In the application we use a virtual model of towercopter [6], which is a faithful copy of the real plant. The basic framework of this model was created based on [7].

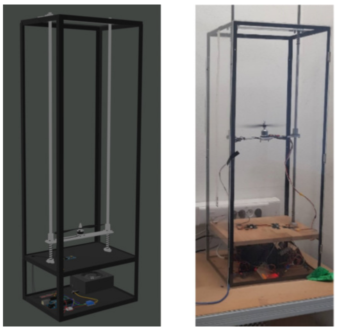


Fig. 2. Virtual and real towercopter model

The communication part consists of *ToF sensor VL53L1X, electronic speed controller EMAX Simon Series 25A, Keyes SR1y relay module, power supply, cables, Arduino UNO R3 (3D model obtained from the [12]).*

The towercopter model was originally exported from Blender in the .dae (Collada) format imported into Unity, but all colors and materials as well as complete cabling were lost. During the second export, we tried to use the .obj format. However, after the import, some materials and colors were lost again. Finally, we chose .fbx format, which was imported into Unity without any loss. For better clarity, we also created a second variant of the model, which has the communication part located next to the main frame. This option is intended to give users a more direct view to the individual components, especially on devices with smaller displays. All components can be seen in Fig. 3.

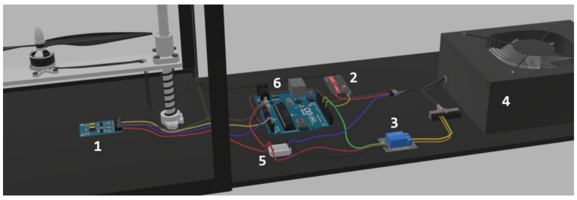


Fig. 3. Components

E. Components Interaction

Unity uses the so-called Game Objects as basic objects that represent models, props, and scenery. They do not perform any functionality on their own, they are just kind of containers for components that implement the actual functionality of the application. The most important objects used in application are as follows: *First Person Camera, Manipulation System, Object Generator, Event System, etc*. Of course, the application has to use more Game objects.

As it is usual during development, we did not miss a few problems that complicated our work when implementing some functions of the application.

The first problem we encountered was related to the interaction with the UI components of the user interface. More precisely, if we clicked on the button that was located above the scanned planes, this click also called the function that placed the 3D models in the virtual environment. Unity has a special EventSystem class implemented to detect such interactions, which is responsible for processing events in the scene. When looking for a solution, we found that this is a known bug that has been present on Android devices for a long time and still cannot be completely removed by developers. With this fact, we were forced to create our own method to control the interaction of UI components.

Another complication was encountered when implementing the screen for a visualization of individual component. The screen is not designed as a separate scene because its change would cause the lost of the entire main AR scene. Therefore the detailed component screen is only displayed as a panel that covers the entire main stage. However, when rendering the panel together with the 3D objects of the individual components, this panel was magnified several times. The source of this behavior was an AR script, which is responsible for rendering the background from the camera's point of view. We were not able to eliminate the problem, due to the fact that this script is responsible for a very important functionality of application and we did not want to make extensive changes to it. As an alternative we chose the deactivation of the main application screen whenever the component detail is displayed. This means that while the detailed screen is displayed, the application does not draw a real image from the device's camera. After closing the detail, the script is run again without any loss of objects in the AR scene.

F. Gestures

The application allows to manipulate virtual objects on scanned planes. This functionality is provided through gestures. All supported gestures are shown in the Fig.4.

Individual gestures are introduced in the following list (from the left side): *selection, movement on the screen, rotation, vertical movement (lifting), resize.*

Fig. 4. Gestures [13]

G. Components Details

Last but not least, the application supports the display of basic information about the individual components of the towercopter connection. After clicking on one of components, a panel containing the rotating 3D model of the component, its name and the basic information that characterizes it is displayed.

H. Simulation Process

The main function of application includes the ability to simulate towercopter behavior based on user-specified input parameters. The application enables this functionality thanks to the connection with the API simulation module, which provides us with real-time data necessary for the most accurate display of the simulation [14]. It uses the simulation software SciLab/Xcos, in which it is possible to create and simulate behavior of the system using block diagrams.

The device must have access to the Internet in order to run the simulation and perform it successfully. It is also necessary that at least one virtual object (in this case a towercopter) is placed on the scanned planes. After pressing the "Simulate" button in the main menu, the form will be displayed on the left side of the screen. First, users are allowed to choose the type of the controller that should be used in the simulation. At present, application supports the behavior of a PID controller, but it allows easy expansion with possible additional block diagrams in the future. After selecting the controller, a form for the change of the simulation parameters is displayed. The form is pre-filled with the predefined values of the controller and additional simulation parameters. Items in this form are generated automatically based on the selected controller. In the presented experiment, the following parameters can be set: *P, I, D, height (cm), sampling period (s), simulation time (s)*.

After submitting the form, the simulation data are obtained through the HTTP GET request to the web interface of the API simulation module. The request is implemented asynchronously so that the application can still be used until a response with data arrives. The output data are returned in JSON format and contains two variables: *time* and *height*. Then the data are processed. The individual values are gradually used to modify the coordinates of individual moving objects (platform, propeller, engine ...) and in this way the process of rendering the movement of the towercopter 3D model can start. Fig.5 shows the sequence diagram that graphically describes the process of obtaining simulation data.

The simulation of the towercopter movement is plotted in real-time on a graph. Since Unity is primarily a game engine, it does not contain any of its own libraries that would support the drawing of graphs (it allows only to use assets), so we had to design own implementation from scratch.

When the simulation starts, the graph is displayed at the bottom of the screen (Fig.6 left). It can be minimized or possibly completely closed. After the simulation, the user is allowed to share the simulation data through all applications that support this feature on Android devices or save them directly to the device memory. The export format is .csv that is

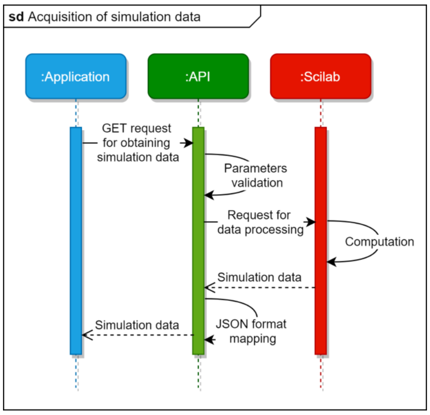


Fig. 5. Obtaining of simulation data

compatible with many often used programs e.g Matlab, Scilab or MS Excel. The graph plots the elevation curve, with the X-axis of the graph representing time in seconds. In Fig. 6 (right) the application from user's point of view can be seen.

One of the challenges was the creation of moving cables during takeoff. Unity could address individual cables as separate objects, but it cannot address individual joints, because model was created in Blender. Therefore we modified the basic model so that the part of the cables that would move on its own was replaced with a component based on the Bézier curve of three points.

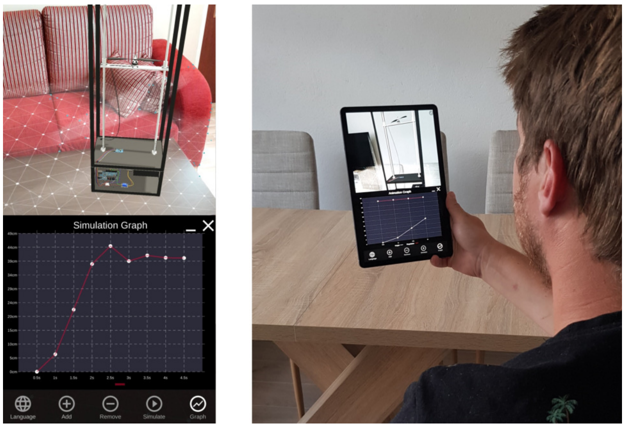


Fig. 6. User point of view

III. Conclusion

The augmented reality still has to overcome some challenges. Over the next few years we are likely to witness a shift in the development of the concept of augmented reality, in terms of software, hardware and a multitude of new applications.

The main functionality of created application is the ability to simulate the behavior of a towercopter based on the selected controller. Currently, the application supports a PID controller, but it can be easily expanded. Before starting the animation, the user is allowed to specify the input parameters, then they are sent to the API simulation module, which returns the complete data based on the simulation of the block diagram. The data are gradually projected into the motion of the towercopter and they are also displayed on a graph. Simulation data can be downloaded or shared.

Among other things, the application should serve as a support tool for teaching subjects dealing with control theory. It aims to help students better understand the behavior of controllers by directly visualizing their behavior. It also provides students with access to a 3D virtual copy of a real towercopter device, allowing a detailed view of how the device is constructed and connected before students come into direct contact with it, which can streamline the teaching process and prevent possible injury or damage to the device.

As a future work, authors would like to extend the application for another devices and use the ARCore Cloud API, that allows to create shared AR applications.

References

[1] D. Nguyen and G. Meixner, "Gamified augmented reality training for an assembly task: A study about user engagement," in *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems, FedCSIS 2019*, pp. 901–904, Institute of Electrical and Electronics Engineers Inc., September 2019. doi: 10.15439/2019F136.

[2] N. S. Rosni, Z. A. Kadir, M. N. M. Mohamed Noor, Z. H. Abdul Rahman, and N. A. Bakar, "Development of mobile markerless augmented reality for cardiovascular system in anatomy and physiology courses in physiotherapy education," in *Proceedings of the 2020 14th International Conference on Ubiquitous Information Management and Communication, IMCOM 2020*, Institute of Electrical and Electronics Engineers Inc., January 2020. doi: 10.1109/IMCOM48794.2020.9001692.

[3] K. Zhang, J. Suo, J. Chen, X. Liu, and L. Gao, "Design and implementation of fire safety education system on campus based on virtual reality technology," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017*, pp. 1297–1300, Institute of Electrical and Electronics Engineers Inc., November 2017. doi: 10.15439/2017F376.

[4] M. B. Ibáñez, Á. Di Serio, D. Villarán, and C. Delgado-Kloos, "Impact of visuospatial abilities on perceived enjoyment of students toward an ar-simulation system in a physics course," in *IEEE Global Engineering Education Conference, EDUCON*, vol. April-2019, pp. 995–998, IEEE Computer Society, April 2019. doi: 10.1109/EDUCON.2019.8725185.

[5] M. T. Abhishek, P. S. Aswin, N. C. Akhil, A. Souban, S. K. Muhammedali, and A. Vial, "Virtual Lab Using Markerless Augmented Reality," in *Proceedings of 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering, TALE 2018*, pp. 1150–1153, IEEE, jan 2018. doi: 10.1109/TALE.2018.8615245.

[6] L. Karcol, "Interaktívny WebGL model "towercoptera" [in slovak]," Master's thesis, Slovak University of Technology in Bratislava, 2017.

[7] P. Ťapák and M. Huba, "One Degree of Freedom Copter," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018. doi: 10.1007/978-3-319-74727-9_11.

[8] P. Duffy, "Augmented reality," *Nextechar Solutions*, October 2019.

[9] P. Nowacki and M. Woda, "Capabilities of ARCore and ARKit Platforms for AR/VR Applications," in *Advances in Intelligent Systems and Computing*, vol. 987, (Brunów, Poland), pp. 358–370, Springer Verlag, 2019. doi: 10.1007/978-3-030-19501-4_36.

[10] Google LLC, "Google arcore supported devices." https://developers.google.com/ar/discover/supported-devices/. 16.06.2020.

[11] Google Developers, "Session — ARCore — Google Developers," 2020.

[12] Evan Boldt, "Blender arduino model." https://robotic-controls.com/learn/arduino/blender-arduino-model.

[13] Tom Loots, "Touch gestures." https://dribbble.com/shots/1383148-Touch-Gestures-freebie, 2020.

[14] P. Milán, "Komunikácia medzi 3D enginom a simulačným prostredím [in slovak]," 2019.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 653–656 DOI: 10.15439/2020F29 ISSN 2300-5963 ACSIS, Vol. 21

A Framework for Developing Proxemic Mobile Applications

Paulo Pérez
E2S / University of Pau (LIUPPA)
ESTIA Institute of Technology
Bidart, France
ppdaza@iutbayonne.univ-pau.fr

Philippe Roose
E2S / University of Pau (LIUPPA)
IUT de Bayonne
Anglet, France
philippe.roose@iutbayonne.univ-pau.fr

Yudith Cardinale
Universidad Simón Bolívar (USB)
Caracas, Venezuela
Universidad Católica San Pablo
Arequipa, Perú
ycardinale@usb.ve

Marc Dalmau
E2S / University of Pau (LIUPPA)
IUT de Bayonne
Anglet, France
dalmau@iutbayonne.univ-pau.fr

Dominique Masson
Dev1-0
Technopole Izarbel
Bidart, France
d.masson@dev1-0.com

Nadine Couture
University of Bordeaux
ESTIA Institute of Technology
Bidart, France
n.couture@estia.fr

***Abstract*—The widespread diffusion of smart and mobile devices continuously connected to the Internet has facilitated the users' contact and interaction with other people, devices, and with their physical surroundings. Proxemic interaction, derived from proxemics theory, focuses on how Human-Computer Interaction (HCI) works with smart devices, using the five proxemic dimensions: Distance, Identity, Location, Movement, and Orientation (DILMO). The current tools for developing proxemic applications require fixed devices that make it difficult to build proxemic mobile apps. In this work, we propose a framework to manage all components in a proxemic environment (i.e., interaction objects and DILMO dimensions that govern the HCI). We demonstrate and evaluate the effectiveness and suitability of our framework, through the development of two proxemic mobile applications, as proof-of-concept.**

***Index Terms*—Proxemic interaction, proxemic zone, mobile devices, wearable technologies**

I. Introduction

The use of mobile technologies in our daily life is increasing in a way without precedent. People can interact with different contexts through electronic devices (e.g., personal mobile phones, tablets, wearable technologies, and smart-watches) to accomplish their daily tasks. Many of these tasks require a specific Human-Computer Interaction (HCI). Researchers are therefore seeking to develop new useful and enjoyable interfaces. Proxemic interaction arises as a novel concept to improve HCI [1], [2]. Proxemic interaction describes how people use interpersonal distances to interact with digital devices [3]–[5], using the five physical proxemic dimensions: Distance, Identity, Location, Movement, and Orientation (DILMO).

Proxemic interaction is derived from the social Proxemics theory proposed in 1966 by the anthropologist Edward T. Hall [6]. Hall describes how individuals perceive their personal space relative to the distance between themselves and others. According to Hall's proxemics theory, interaction zones have been classified into four zones: (i) intimate zone, comprised between 0 and 50 cm of distance; (ii) personal zone, defined by a distance of 0.5 cm to 1 m; (iii) social zone, when the distance is between 1 m and 4 m ; and (iv) public zone, if distance is more than 4 m. He underlines the role of proxemic relationships as a method of communication based on the distance between people.

In this context, solutions such as Toolkit [7] and ProximiThings [8] have been proposed to support the development of proxemic interactions. However, existing solutions present limitations for implementing proxemic interaction in mobile technologies, because they require special hardware devices connected to the system (e.g., a Kinect Depth sensor, which must be installed on a PC for sensing proxemic information).

This work aims to propose a concrete solution, a framework, for developing proxemic environments comprised by entities, whose interactions are defined according to DILMO dimensions in mobile applications. Our proposed framework represents a threefold contribution: (i) it offers functionalities to define and manage all components in a proxemic environment: the interaction objects, the DILMO dimensions that govern the HCI, and the proxemic mobile applications; (ii) an API integrated into the framework, that allows developers to simplify the process of proxemic information sensing (i.e., measure of DILMO dimensions) by mobile phones and wearable sensors; and (iii) the proof-of-concept to demonstrate and evaluate the effectiveness and suitability of our framework by describing the implementation of two proxemic mobile applications; these two mobile apps are based on HCI defined as a function of different DILMO combinations that specify different context-based infrastructures for proxemic environments based on mobile devices.

II. Related work

Proxemic concepts are based on physical, social, and cultural factors that influence and regulate interpersonal interactions [7]. In order to know how the factors should be applied to proxemic interactions for ubiquitous computing applications, Greenberg [4] identified five dimensions: Distance, Identity, Location, Movement, and Orientation (we call them DILMO as an abbreviation), which are associated with people, digital devices, and non digital things.

Proxemic interaction is a remarkable interaction technique that allows the user to control the digital devices in a flexible way [3], [7], [9]. Some previous works have proposed tools to support the development of proxemic applications considering proxemic interactions.

The work presented in [8], illustrates how the proxemic dimensions can support interaction among entities (people and objects), with a proposed context-aware framework. This framework provides capabilities that help developers build a front-end application. However, the framework requires a cloud computing architecture and an active connection to the server for processing proxemic information. In [7], a framework, called Proximity Toolkit, used to discover novel proxemic-aware interaction techniques is proposed. The framework is a guide on how to apply proxemic interaction design for domestic ubiquitous computing environments. This framework allows the rapid building of proxemic-aware systems and it offers a flexible architecture for sensing proxemic data from different types of sensors. However, the implementation of this framework requires a hardware architecture based on fixed devices (e.g., a Kinect Depth sensor and a client-server architecture) for allowing the server to process the proxemic information from appliances.

These studies demonstrate the current interest for researchers to develop tools that support the design and implementation of proxemic applications. However, proxemic interaction on mobile devices has not been implemented in full. In the next section, we describe the framework for developing proxemic mobile apps.

III. Framework to Develop Proxemic Mobile Apps

We propose a framework for supporting the design and development of proxemic mobile apps to control and manage a proxemic environment (denoted as P_E), based on mobile technology and smart wearable technology. In order to P_E, we propose a methodological process comprised by three single steps:

1) Define the distances that delimit the proxemic zones (denoted as P_Z), according to which entities will interact: intimate zone (denoted as $P\_Z_{intimate}$), personal zone (denoted as $P\_Z_{personal}$), social zone (denoted as P_Z_{social}), and public zone (denoted as P_Z_{public}).
2) Define the appropriate DILMO combination to define the HCI for the target mobile app to be developed.
3) Implement the mobile app, considering the technology supported by the entities in the P_E.

To support this development process, the framework is composed by mainly three components aligned with each step (see Figure 1): (i) Proxemic Zones module; (ii) DILMO module; and (iii) an API that supports the instantiation of the two previous mentioned components.

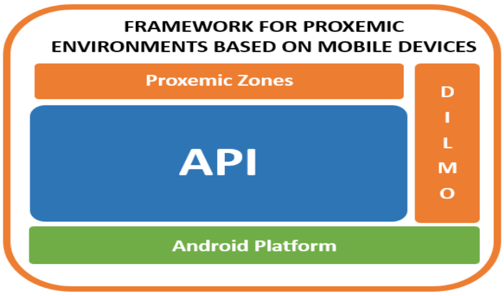
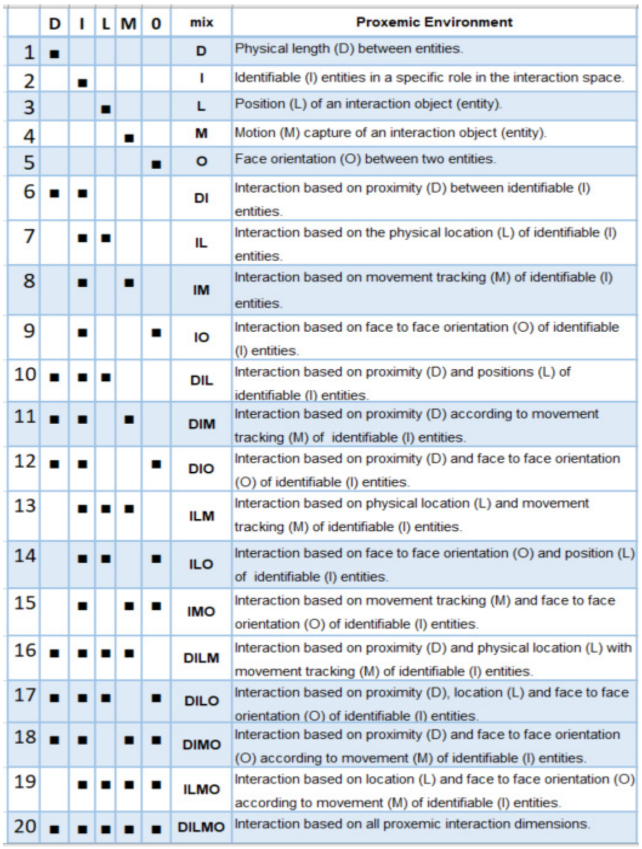


Fig. 1. Framework architecture.

1) The **Proxemic Zones module** allows the definition of the four proxemic zones (i.e., $P\_Z_{intimate}$, $P\_Z_{personal}$, P_Z_{social}, and P_Z_{public}), according to user needs. The interaction between two entities changes in accordance of the P_Z in which they are located. DILMO measurements are considered, based on the P_Z of the entities.
2) The **DILMO module** allows defining different scenarios of HCI on a P_E, based on different combinations of DILMO dimensions. This module provides a nomenclature to describe each combination of such proxemic dimensions (see Figure 2). For each DILMO combination, it is important to identify or create the interaction objects (i.e., entities) that will interact in the defined P_E. In proxemic environments, an entity (denoted as E) is an interaction object that can be detected by another E; if these interaction objects have a unique identification or specific role, they are designated as identities (denoted as I). Figure 2 is a guide that allows the developer to know which methods must be implemented on the API or which object must be created from the API according to DILMO for processing proxemic information. For example, a DIL proxemic environment means that Distance (D) and Location (L) are considered for Identities (I). Combination of proxemic dimensions are also valid, although the entities have not unique identification; e.g., *a person*, *a device beacon*, instead of *the smartphone's owner*, *my device beacon*. Hence, proxemic environments denoted in rows 10 to 20 in Figure 2, can become as `DL`, `DM`, `DO`, `LM`, `LO`, `MO`, `DLM`, `DLO`, `DMO`, `LMO`, and `DLMO`, respectively, when all interaction objects are not identifiable entities.
3) The **API** facilitates developers processing proxemic information and values. The API provides classes and methods to define the P_Z, as well as to manage the different combinations of DILMO dimensions. For example, for a DIL proxemic environment, methods to identify entities (I) and to process D and L are available in DILMO class. Thus, the API behaves as a bridge between the Proxemic Zones and DILMO modules.

In the current version of our framework, the API considers the extraction of DILMO values from smartphones or mobile devices based on the Android operating system by using motion sensors and cameras that the majority of smart devices have in their hardware configuration [10]. For example, through the BLE mechanisms [11], it is possible to know the distance (D) between two mobile devices. Another way to estimate the distance between two entities is to use computer vision. In the next section, we describe the proof-of concept.



	D	I	L	M	0	mix	Proxemic Environment
1	■					D	Physical length (D) between entities.
2		■				I	Identifiable (I) entities in a specific role in the interaction space.
3			■			L	Position (L) of an interaction object (entity).
4				■		M	Motion (M) capture of an interaction object (entity).
5					■	O	Face orientation (O) between two entities.
6	■	■				DI	Interaction based on proximity (D) between identifiable (I) entities.
7		■	■			IL	Interaction based on the physical location (L) of identifiable (I) entities.
8		■		■		IM	Interaction based on movement tracking (M) of identifiable (I) entities.
9		■			■	IO	Interaction based on face to face orientation (O) of identifiable (I) entities.
10	■	■	■			DIL	Interaction based on proximity (D) and positions (L) of identifiable (I) entities.
11	■	■		■		DIM	Interaction based on proximity (D) according to movement tracking (M) of identifiable (I) entities.
12	■	■			■	DIO	Interaction based on proximity (D) and face to face orientation (O) of identifiable (I) entities.
13		■	■	■		ILM	Interaction based on physical location (L) and movement tracking (M) of identifiable (I) entities.
14		■	■		■	ILO	Interaction based on face to face orientation (O) and position (L) of identifiable (I) entities.
15		■		■	■	IMO	Interaction based on movement tracking (M) and face to face orientation (O) of identifiable (I) entities.
16	■	■	■	■		DILM	Interaction based on proximity (D) and physical location (L) with movement tracking (M) of identifiable (I) entities.
17	■	■	■		■	DILO	Interaction based on proximity (D), location (L) and face to face orientation (O) of identifiable (I) entities.
18	■	■		■	■	DIMO	Interaction based on proximity (D) and face to face orientation (O) according to movement (M) of identifiable (I) entities.
19		■	■	■	■	ILMO	Interaction based on location (L) and face to face orientation (O) according to movement (M) of identifiable (I) entities.
20	■	■	■	■	■	DILMO	Interaction based on all proxemic interaction dimensions.

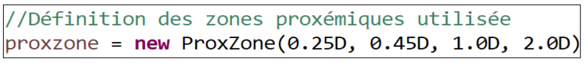
Fig. 2. DILMO proxemic dimensions and nomenclature to describe each combination that is available through the API methods for processing proxemic information.

IV. Proof-Of-concept Of our Framework

Our goal is to create proxemic environments based on mobile devices or wearable technology and demonstrate that our framework allows developers to build proxemic mobile applications effectively. For this purpose, we show the implementation of two mobile applications, called IntelliPlayer and Tonic, based on proxemic interactions. These apps were implemented using Android Studio version 3.3.

Both apps have been developed by undergraduate students, as part of their final project in computer science. The developing team was integrated by four students whose average age was 21 years-old. Two training sessions of two hours each were organised for the students. The training process allows students to understand the development process for building proxemic mobile applications with our framework. They learned: (i) how to define each P_Z; (ii) how to select each proxemic dimension for recreating a P_E; and (iii) how to use methods and classes in the API. The development time of both applications was 64 hours by two developers over a period of 4 weeks. IntelliPlayer took 44 hours of work, while Tonic was finished in 20 hours, in the same four weeks.

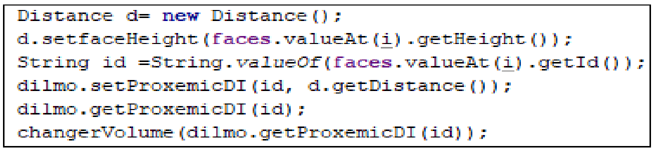
IntelliPlayer is a mobile application that plays a video in a smartphone and reacts according to four proxemic zones and DILO proxemic dimensions. In the first step of the methodological approach the four P_Z were created: P_Z_{intime} (0 mts to 0.25mts), $P\_Z_{personal}$ (0.26 mts to 0.45 mts), P_Z_{social} (0.46mts to 1 mts), and P_Z_{public} (1.1 mts to 2 mts). Then, in the second step, the HCI was designed according to D, I, L, and O, thus a DILO P_E was defined. Figure 5 shows the proxemic zones that have been defined by developers through the API, as shown in Figure 3.



```
//Définition des zones proxémiques utilisée
proxzone = new ProxZone(0.25D, 0.45D, 1.0D, 2.0D)
```

Fig. 3. Example of ProxZone Class Constructor invocation.

With this application, we illustrate a proxemic environment using a mobile player app that reacts to the distance (D) and location (L) of a person (E_1) and his face orientation (O), with respect to the smartphone (I_1) displaying a video. The computer vision technique has been used for this purpose, based on the properties of an Android camera and through the API methods `setFaceHeight(float faceHeight)` and `getDistance()`. Figure 4 shows a block code of this case.



```
Distance d= new Distance();
d.setfaceHeight(faces.valueAt(i).getHeight());
String id =String.valueOf(faces.valueAt(i).getId());
dilmo.setProxemicDI(id, d.getDistance());
dilmo.getProxemicDI(id);
changerVolume(dilmo.getProxemicDI(id));
```

Fig. 4. Block code of IntelliPlayer.

With the distance (D) between the user (E_1) and the smartphone (I_1), IntelliPlayer determines the proxemic zone (P_Z) of E_1 (a user), with respect to the smartphone (I_1). To do so, it invokes the method `getProxemicZoneByDistance()`.

IntelliPlayer automatically adjusts the volume of the video according to the P_Z in which E_1 (the user) is with respect to the smartphone (I_1): when E_1 is in P_Z_{intime}, it decreases to 25% volume of speaker; for $P\_Z_{personal}$, it increases to 50% volume; for P_Z_{social}, it increases to 75% volume; and for P_Z_{public}, it increases to 100% volume) (see Figure 5).

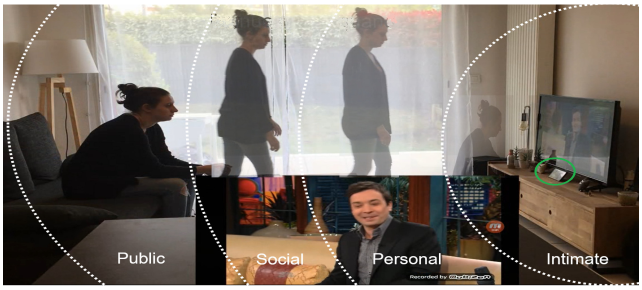


Fig. 5. IntelliPlayer proxemic zones.

Another useful function of IntelliPlayer is to provide a video description that users can read on the screen according to user location (L). When a user (E_1 or E_2) is in the

$P\_Z_{personal}$ and her/his orientation (O) is in front of the screen, the application can obtain the face location (L) (by using `setRelativeLocationScreen(float L)` and `getRelativeLocationOnScreen()` methods from the API) to split the screen, with the video (running) and information about the video on the right or on the left, according the detected L.

Tonic is an educational mobile app for learning musical notes, developed for illustrative purposes. In the first step of the methodological approach, the students have defined four proxemic zones: P_Z_{intime} (0 mts to 0.5mts), $P\_Z_{personal}$ (0.51 mts to 1 mts), P_Z_{social} (1.1 mts to 2 mts), and P_Z_{public} (2.1 mts to 4 mts). In the second step, a DIMO combination was stated for the proxemic environment, P_E.

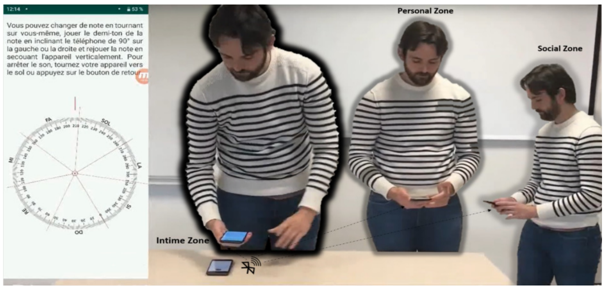

Vous pouvez changer de note en tournant sur vous-même, jouer le demi-ton de la note en inclinant le téléphone de 90° sur la gauche ou la droite et rejouer la note en secouant l'appareil verticalement. Pour arrêter le son, tournez votre appareil vers le sol ou appuyez sur le bouton de retour

Fig. 6. Tonic Proxemic Zones based on Bluetooth Low Energy.

Tonic allows a user to play a note and modify it from his smartphome on another smart mobile. The user's smartphone is identified (I_1) by the mobile device which plays the sound (I_2). Thus, I_2 plays and modifies a sound, by using proxemic interactions based on the P_Z, and on D, I, M, and O dimensions (i.e., a DIMO proxemic environment). In Tonic, the distance (D) between the two devices is obtained by using BLE technology. I_1 broadcasts its identifier to nearby portable electronic devices, thus it is caught by I_2. The volume of the sound is adjusted according to the P_Z in which I_2 is, with respect to I_1 (see Figure 6). The musical notes are changed according to the movement (M) and orientation (O) of I_2, with respect to I_1. According to M a tone is increased/decreased, while according to O a semi-tone is increased/decreased. M and O are calculated based on the capabilities of the smartphone, such as accelerometer, gyroscope, compass, and magnetometer. These sensors provide proxemic information that is mainly used in the API. Movement M was determined by using methods `setAzimuthWithRange(float MAX, float MIN, float value)` and `isAzimuthInRange()`; while O was managed by methods based on the smartphone's technical capacities. To manage P_Z and D, the methods used from the API, in the third step of the approach: `getProxemicDI(String I)`, `setProxemicDIL(String I,double D,float L)`, and `setBluetoothDistance(double rssi, double txtPower)`.

The development of these applications show how the framework provides a development process that allowed students to rapidly build proxemic apps and creating proxemic environments based on the exclusive use of mobile devices. These apps offer a portable implementation that facilitates using the proxemic interaction in comparison to previous works that have used fixed platforms for similar purposes.

V. Conclusion

Proxemic interaction provides different options for HCI and mobile interaction while offering several advantages and better experiences for users. Through this work, we have explored the use of proxemic interaction based on the combination of proxemic dimensions – i.e., Distance, Identity, Location, Movement, and Orientation (DILMO) – to offer a new context-oriented interaction for mobile applications. We have presented a methodological process to guide developers on creating realistic proxemic environments supported by an API and a framework, in which the end-users only need to use mobile or wearable devices. The API [1] and Apps[2] are available for free downloading.

References

[1] F. Brudy, C. Holz, R. Rädle, C.-J. Wu, S. Houben, C. N. Klokmose, and N. Marquardt, "Cross-device taxonomy: Survey, opportunities and challenges of interactions spanning across multiple devices," in *Proceedings of the CHI Conference on Human Factors in Computing Systems*. ACM, 2019, p. 562, https://doi.org/10.1145/3290605.3300792.

[2] J. E. Grønbæk, C. Linding, A. Kromann, T. F. H. Jensen, and M. G. Petersen, "Proxemics play: Exploring the interplay between mobile devices and interiors," in *Proceedings of Companion Publication of the Conference on Designing Interactive Systems*, ser. DIS '19. ACM, 2019, pp. 177–181, https://doi.org/10.1145/3301019.3323886.

[3] T. Ballendat, N. Marquardt, and G. Saul, "Proxemic interaction: designing for a proximity and orientation-aware environment," in *Proceedings of International Conference on Interactive Tabletops and Surfaces*, ser. ITS' 10. ACM, 2010, pp. 121–130, http://doi.org/10.1145/1936652.1936676.

[4] S. Greenberg, N. Marquardt, T. Ballendat, R. Diaz-Marino, and M. Wang, "Proxemic interactions: the new ubicomp?" *Interactions*, vol. 18, no. 1, pp. 42–50, 2011, https://doi.org/10.1145/1897239.1897250.

[5] J. E. Grønbæk, M. S. Knudsen, K. O'Hara, P. G. Krogh, J. Vermeulen, and M. G. Petersen, "Proxemics beyond proximity: Designing for flexible social interaction through cross-device interaction," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020, pp. 1–14, https://doi.org/10.1145/3313831.3376379.

[6] E. T. Hall, *The Hidden Dimension: An anthropologist examines man's use of space in private and public*. New York: Anchor Books; Doubleday & Company, Inc, 1966.

[7] N. Marquardt, R. Diaz-Marino, S. Boring, and S. Greenberg, "The proximity toolkit: prototyping proxemic interactions in ubiquitous computing ecologies," in *Proceedings of the 24th annual ACM symposium on User interface software and technology*, ser. UIST '11. ACM, 2011, pp. 315–326, https://doi.org/10.1145/2047196.2047238.

[8] C. Cardenas and J. A. Garcia-Macias, "Proximithings: Implementing proxemic interactions in the internet of things," *Procedia Computer Science*, vol. 113, pp. 49–56, 2017, https://doi.org/10.1016/j.procs.2017.08.286.

[9] D. Ledo, S. Greenberg, N. Marquardt, and S. Boring, "Proxemic-aware controls: Designing remote controls for ubiquitous computing ecologies," in *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*, ser. MobileHCI'2015. ACM, 2015, pp. 187–198, https://doi.org/10.1145/2785830.2785871.

[10] G. Developers, "Sensors overview, developer guides," 2019, https://developer.android.com/guide/topics/sensors.

[11] AltBeacon, "The open and interoperable proximity beacon specification," 2018, https://altbeacon.org/.

[1] The API is available in https://www.iutbayonne.univ-pau.fr/ ppdaza/

[2] The APPS are available in https://https://github.com/llagar910e/

DOI: 10.15439/2020F163
ISSN 2300-5963 ACSIS, Vol. 21

Community Detection in Model-based Testing to Address Scalability: Study Design

Alper Silistre*, Onur Kilincceker†, Fevzi Belli‡, Moharram Challenger§ and Geylani Kardas*
*International Computer Institute, Ege University, Izmir, Turkey. Email: alpersilistre@gmail.com, geylani.kardas@ege.edu.tr
†University of Paderborn, Paderborn, Germany. Mugla Sitki Kocman University, Mugla, Turkey. Email: okilinc@mail.upb.de
‡University of Paderborn, Paderborn, Germany. Izmir Institute of Technology, Izmir, Turkey. Email: belli@upb.de
§University of Antwerp and Flanders Make, Belgium. Email: moharram.challenger@uantwerpen.be

***Abstract*—Model-based GUI testing has achieved widespread recognition in academy thanks to its advantages compared to code-based testing due to its potentials to automate testing and the ability to cover bigger parts more efficiently. In this study design paper, we address the scalability part of the model-based GUI testing by using community detection algorithms. A case study is presented as an example of possible improvements to make a model-based testing approach more efficient. We demonstrate layered ESG models as an example of our approach to consider the scalability problem. We present rough calculations with expected results, which show 9 times smaller time and space units for 100 events in the ESG model when a community detection algorithm is applied.**

I. Introduction

SOFTWARE testing is of critical importance considering today's technological developments. The solution methods proposed in this area differ according to the software systems to be tested. Graphical user interfaces (GUIs) are an integral part of the web and mobile software systems. Many model-based approaches have been proposed varying based on the model used for testing the GUI functions. These models differ in terms of semantics and syntactic terms. Scalability is shown as one of the disadvantages of all model-based approaches. As the model grows, operations such as test production become more difficult, thus creating the problem of scalability.

Model-based testing is a testing method that executes test cases generated from the abstract view of a system under test (SUT). This abstraction specifies the behavior of the system so we can model it with one of the different models such as Finite State Machine (FSM) [1], Event Sequence Graph (ESG) [4] [5], Event Flow Graph (EFG) [2] [3], and Regular Expression (RE) [6][7]. The major benefit is, code-based testing is a time consuming and error-prone method to cover all cases of the SUT while we can generate and run test sequences efficiently with model-based testing. There are many automation tools to achieve this task and many papers worked on this topic for several decades.

Model-based approaches suggest a hierarchical and layered structure at the model generation stage. No specific approach has been found in the literature regarding the problem that may arise if the tester who created the model skipped this important issue. In this study, the community detection approach, which is frequently used in different areas, is proposed to eliminate this serious problem. Thus, the model expressed as layered will be hierarchical, that is, a layered structure by community detection method. In the scope of the study, an ESG model will be layered, and then automatic test set generation and test execution operations will be performed on this ESG model.

The current work in this paper is a design study. We briefly review the literature and provide proof of concept with a case study to support the proposed approach. Expected results with the scope of the current work are discussed, and rough theoretic calculations are presented.

The rest of the paper is organized as follows: Section 2 gives the related work on models used in GUI testing and community detection problem. Section 3 introduces the proposed approach. A case study to exemplify the proposed approach is given in section 4. Expected results and implications with possible threats to the validity of the proposed approach are presented in section 5. Finally, section 6 concludes the paper.

II. Related Work

This section briefly presents related literature regarding model-based GUI testing and community detection problem.

A. Model based GUI Testing

Shehady and Siewiorek [1], introduce a model to be used in model-based testing called VFSM (Variable Finite State Machine) for the GUI with fewer states than a formal FSM model. The VFSM and FSM are equivalent at its core. So, they show how to convert a VFSM to an FSM to create a test suite. The major benefit here is that the total state count is less which makes it more efficient in terms of end-to-end test generation and execution.

Memon et al. [2] come up with a new technique that helps automated test generation by using ESG models. Essentially, this technique is a planning algorithm that uses AI. The algorithm needs start and final states of the model and several defined operators to work on the model. It creates test sequences between these start and final states by looking into GUI interactions and events between these states.

Memon [3] shows event-flow model generation which he describes as one scalable model for using in the model-based testing area. Event-space exploration (ESES) strategies are used inside the proposed method. The event-flow model is a combination of different models which are assessed in the paper. In order to reduce the cost of model creation, the process is also automated.

Belli [4] presents a new approach called the 'holistic' approach. He discusses that in order to test the GUI of a system properly, we must take incorrect test cases along with correct test cases into account. The GUI of an application should work without failing even the events are illegal. This is an important part of achieving complete system coverage.

Belli et al. [5] examine existing work on models that have been used in model-based GUI testing such as; ESG, EFG, etc., and analyze these model notations and how to create mutation from them. They also present ways to apply test generation from models and additional optimization techniques.

Kilincceker et al. [6] introduce RE as being a model-based testing method and use RE to model hardware design combined with RE-based test generation. RE is represented by an abstract syntax tree and a tree traversal algorithm is used in the test generation. RE-based coverage criteria are used to assess the adequacy of the testing method. Kilincceker and Belli [16] propose novel coverage criteria based on the analysis of a RE model. These coverage criteria are used to generate test sequences for testing GUI systems in [7] by means of random test generation. They also use to test GUI of mobile applications [13], hardware design [15], and web-based systems [16] combined with holistic testing.

B. Community Detection

Harary et al. [8] take into account the result of a Festinger's work [17] which was finding a clique in a group depending on whether certain elements satisfy required conditions and improve this with a new study to find all cliques in a group that has three or fewer cliques with a concept called unicliqual person. They then remove the restrictions with an "inductive reduction method".

Fortunato [9] explains community detection in graphs in very detail from main definitions to different methods to detect these communities, with example algorithms and techniques. He gives details about his ideas on the topic and mentions the given problem yet to be solved properly.

Leskovec et al. [10] examine different algorithms for network community detection to understand them better and compare them with each other in terms of performance and their capabilities. They also take biases in those algorithms into consideration while conducting their study. They show how complicated to detect communities in large network groups.

Sadi et al. [11] study community detection algorithms with a method that is using Ant Colony Optimization to reduce network graphs without losing its ability to solve the given problem. They work especially on the scalability part of the topic because the cost of computation is a lot in any given large network graph. They aim to reduce the number of nodes and then applying algorithms to work on this reduced graph.

To the best of our knowledge, there is no work to address scalability for model-based testing by utilizing the community detection algorithm. To this end, the current work aims to fill the gap by providing a community detection algorithm to address the scalability problem of model-based testing.

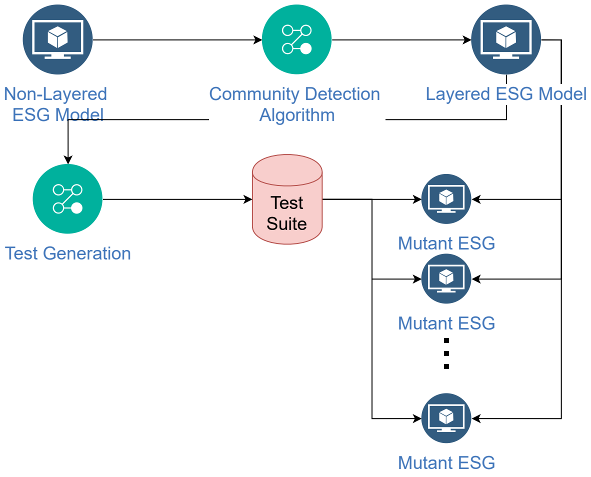


Fig. 1. The proposed approach

III. The Proposed Approach

The proposed approach offers a community detection algorithm on a non-layered ESG model to obtain a layered one. It also includes further necessary steps in conventional model-based testing approaches. These further steps, apply to the layered ESG model, are test generation and test execution as depicted in Fig. 1. By detecting communities in our model, we will treat them as sub-graphs. These sub-graphs will have their own process to generate test sequences with a model-based test generation and execution process. This sub-graph will be treated as a single node in the main graph. Since this sub-graph has nodes that are part of the community, extracting their interaction with other nodes in the main graph will be beneficial in terms of scalability because we do not need to create test sequences that involve nodes in a community to have a relationship with another node in a higher level graph.

Moreover, the test suite resulted from the test generation step becomes more compact and useful from the layered ESG model. Finally, the test suite is executed on mutant layered ESG models to evaluate their effectiveness using mutation score. To do this, the current work utilizes model-based mutation testing by means of appropriate model mutation operators given in [16].

IV. Case Study

The community detection step in the proposed approach is exemplified in a case study that is the internal part of a commercial tourist web page namely ISELTA [1]. The case study is the Special Module of ISELTA web page. Special Module enables agents to offer special advertisements.

Special Module of ISELTA contains arrival and departure dates, accommodation type, number of items, total price, description nation and international, name of the advertisement.

[1] ISELTA, Available at: http://iselta.ivknet.de/

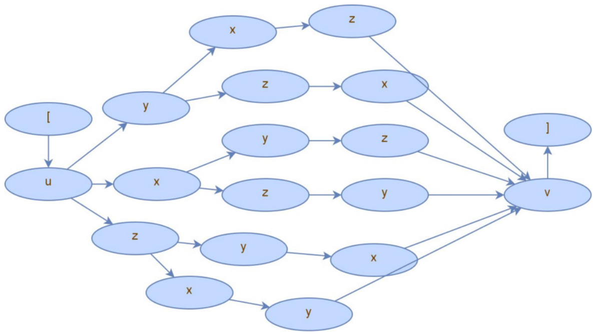


Fig. 2. ESG model of the Add Layer for ISELTA Special Module

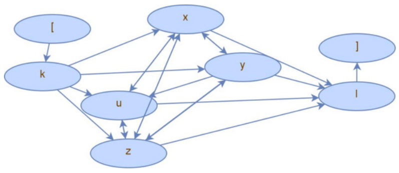


Fig. 3. ESG model of the Edit Layer for ISELTA Special Module

We represent events of “number of items”, “total price”, “description” and “name” input fields of the ISELTA Special Module in ESG models. Other events are neglected.

Special Module is represented as a non-layered ESG model that contains 23 nodes excluding opening and closing events namely pseudo-events representing the starting and finishing states of the ESG model.

Current work uses LEMON (Local Expansion via Minimum One Norm) [18] method for the detection of communities in the graph model of GUI. LEMON uses a local expansion method to find the communities. It detects communities using a sparse vector and is able to achieve the highest detection accuracy compared with state-of-the-art methods such as OSLOM [19], DEMON [20], LC [21]. We skip details of the definitions and algorithms for community detection algorithm due to lack of space in the current paper. The community detection algorithm is applied to the non-layered ESG model and results in two sub-layers and one upper layer. The upper layer contains 2 sub-layers (sESG) namely Add and Edit layers given in Fig. 2 and Fig. 3, respectively. The resulting ESGs include 17 events in the Add layer and 6 events in the Edit layer excluding pseudo-events. Instead of applying test generation on 23 events in the non-layered ESG model, the current work divides this non-layered ESG to sub-layers by using a community detection algorithm. The current work is expected to speed up the test generation to save time and potentially scale on large models. Moreover, the resulting test suites from the test generation become more compact and efficient.

V. Discussion

A. *Expected results and implications*

This section provides the expected results of the proposed approach in terms of scalability and efficiency. To do so, a rough calculation to measure scalability and efficiency will be carried on.

Let’s assume that we have 100 events in a non-layered ESG. In the best case, the community detection algorithm detects 10 sub ESG events containing 10 events. Then, we have 10 events in the upper layer and 10 events in each sub layer resulting in 110 events in total. Assume that the test generation algorithm runs on O(n^2) time complexity and O(n^2) space complexity. The test generation algorithm results in 100^2 time units and 100^2 space units in the full resolution ESG model. However, for the layered ESG models with a community detection algorithm applied, the result is 10^2 time and space units for each sub-layer. Since we have 10 sub-layers, we will have $10^2 * 10$ which will result in 1000 time and space units and another 10^2 from the upper layer. This results in 1000 + 100 equal 1100 time and space units. This provides about 9 times faster test generation and 9 times more compact test suites for 100 events in the best case if we can divide 100 events into 10 equal sub-layers. Additionally, any further increase in the number of layers will provide much smaller time and space units.

In the worst case, the community detection algorithm does not detect any layer and the non-layered and layered ESG models have the same number of events. However, the cost of the community detection algorithm is neglected from the calculation due to no additional cost on the test generation but to overall methodology. The computational complexity of a conventional community detection algorithm is O(m^2n) for a graph with n vertexes and m edges [12].

B. *Threats to validity*

1) Conclusion Validity: The size of our case study may be small to show an example of the defined approach in this paper. This is a potential threat to generalize the approach since multiple examples of bigger ESG models should be assessed to be sure about the robustness of the approach. With this, any unforeseeable problems might pop up, and we can address solutions to these possible problems when the total event size of the ESG model goes beyond the numbers given here. We plan to evaluate the proposed approach on medium and large size of case studies to cope with this threat.

2) Internal Validity: We have described how the model-based testing approach works on models which defined as an abstraction of the system. In theory, working with abstractions makes things efficient since we do not need to use a code-based white-box testing approach. This gives us the ability to test huge models otherwise would be time-consuming if we need to test them manually or with code. This may cause a threat to the internal validity of the approach because testing on the abstraction can never be full as it would run on the actual system. However, it is possible to execute generated test suites from models in actual systems using appropriate test automation tools (such as Selenium [2]) for code-based testing.

Another problem might be the problem of creating wrong or missing models (a model does not cover the whole SUT) from

[2] Selenium, Available at: https://www.selenium.dev/

a system if it is a complex one. This will naturally prevent us to test the whole system. Because of these reasons, we must depend on the correctness of the model for an efficient model-based testing approach.

3) External Validity: Model-based testing aims to detect behavioral and functional faults in a system. Using this method to identify problems in visual aspects and semantics of GUI widgets in a system is a threat to the external validity of the approach because model-based testing is not the first method that comes to mind for this. Code-based testing approaches are more applicable for testing these visual aspects of the system under test. However, the proposed approach is applicable for any testing approach that uses behavioral and sequential models rather than concurrent systems modeled such as by Petri-Nets.

4) Construct Validity: The theoretic advantages, discussed in the previous section, require to be validated on the case studies in terms of time and space complexity. However, the scenarios for average and worst cases result in additional cost of community detection algorithm (O(m^2n) for a graph with n vertexes and m edges [12]). This can be a potential threat to construct validity. On the other hand, the community detection algorithm reduces the total cost of time and space complexity comparing to the full resolution model where community detection is not applied.

VI. Conclusion

The study design given in this paper introduces a model-based testing approach combined with the community detection algorithm to cope with possible scalability problems.

A proof of concept with a small size case study is given to exemplify the use of the community detection algorithm in the current work. The community detection algorithm is applied to the full resolution ESG model to create a layered ESG model. The layered ESG model contains several small layers to improve the efficiency of further steps such as; test generation and execution. Moreover, expected results are introduced with rough theoretic calculations.

A community detection algorithm executes on the full resolution ESG model to obtain a layered ESG model. Then, the test sequences will be generated on this resulting layered ESG model. These test sequences will be executed on mutant models obtained from the original ESG model to evaluate the quality of the generated test sequences. To this end, we expect to provide 9 times faster and 9 times more compact test suites in the best case with respect to layered ESG model rather than full resolution ESG model when we assume that time and space complexity of test generation algorithm equal to O($n2$). Moreover, this also provides saving time in further test execution time. Thanks to 9 times more compact test suites, the time required during the test execution phase can be 9 times less. However, the proposed approach comes with an additional cost of the community detection algorithm. It can be noticed that the case study is trivial and presented to explain the idea. However, we plan to evaluate our approach on medium and large sizes of ESG models to assess these expected advantages by automating procedures.

References

[1] R. K. Shehady & D. P. Siewiorek, "A method to automate user interface testing using variable finite state machines," Proceedings of IEEE 27th International Symposium on Fault Tolerant Computing, Seattle, WA, USA, 1997, pp. 80-88, doi: 10.1109/FTCS.1997.614080.

[2] A. M. Memon, M. E. Pollack & M. L. Soffa, "Hierarchical GUI test case generation using automated planning," in IEEE Transactions on Software Engineering, vol. 27, no. 2, pp. 144-155, Feb. 2001, doi: 10.1109/32.908959.

[3] Memon, A. M. (2007). An event-flow model of GUI-based applications for testing. Software testing, verification and reliability, 17(3), 137-157.

[4] Belli, F. (2001, November). Finite state testing and analysis of graphical user interfaces. In Proceedings 12th international symposium on software reliability engineering (pp. 34-43). IEEE.

[5] Belli, F., Beyazıt, M., Budnik, C. J., & Tuglular, T. (2017). Advances in model-based testing of graphical user interfaces. In Advances in Computers (Vol. 107, pp. 219-280). Elsevier.

[6] Kilinccceker, O., Turk, E., Challenger, M., & Belli, F. (2018, July). Regular expression based test sequence generation for HDL program validation. In 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C) (pp. 585-592). IEEE.

[7] Kilincceker, O., Silistre, A., Challenger, M., & Belli, F. (2019, July). Random test generation from regular expressions for graphical user interface (GUI) testing. In 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C) (pp. 170-176). IEEE.

[8] Harary, F., & Ross, I. C. (1957). A procedure for clique detection using the group matrix. Sociometry, 20(3), 205-215.

[9] Fortunato, S. (2010). Community detection in graphs. Physics reports, 486(3-5), 75-174.

[10] Leskovec, J., Lang, K. J., & Mahoney, M. (2010, April). Empirical comparison of algorithms for network community detection. In Proceedings of the 19th international conference on World wide web (pp. 631-640).

[11] Sadi, S., Öğüdücü, Ş., & Uyar, A. Ş. (2010, July). An efficient community detection method using parallel clique-finding ants. In IEEE Congress on Evolutionary Computation (pp. 1-7). IEEE.

[12] Yang, Z., Algesheimer, R., & Tessone, C. J. (2016). A comparative analysis of community detection algorithms on artificial networks. Scientific reports, 6, 30750.

[13] Mercan, G., Akgündüz, E., Kılınççeker, O., Challenger, M., & Belli, F. (2018). Android uygulaması testi için ideal test ön çalışması. CEUR Workshop Proceedings.

[14] Kılınççeker, O., & Belli, F. (2017). Grafiksel kullanıcı arayüzleri için düzenli ifade bazlı test kapsama kriterleri. CEUR Workshop Proceedings.

[15] Kilincceker, O., Turk, E., Challenger, M., & Belli, F. (2018, April). Applying the Ideal Testing Framework to HDL Programs. In ARCS Workshop 2018; 31th International Conference on Architecture of Computing Systems (pp. 1-6). VDE.

[16] Kilincceker, O., & Belli, F. (2019, November). Towards Uniform Modeling and Holistic Testing of Hardware and Software. In 2019 1st International Informatics and Software Engineering Conference (UBMYK) (pp. 1-6). IEEE.

[17] Festinger, L. (1949). The analysis of sociograms using matrix algebra. Human relations, 2(2), 153-158.

[18] Li, Y., He, K., Bindel, D., & Hopcroft, J. E. (2015, May). Uncovering the small community structure in large networks: A local spectral approach. In Proceedings of the 24th international conference on world wide web (pp. 658-668).

[19] Lancichinetti, A., Radicchi, F., Ramasco, J. J., & Fortunato, S. (2011). Finding statistically significant communities in networks. PloS one, 6(4), e18961.

[20] Coscia, M., Rossetti, G., Giannotti, F., & Pedreschi, D. (2012, August). Demon: a local-first discovery method for overlapping communities. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 615-623).

[21] Ahn, Y. Y., Bagrow, J. P., & Lehmann, S. (2010). Link communities reveal multiscale complexity in networks. nature, 466(7307), 761-764.

Joint 40th IEEE Software Engineering Workshop and 7th International Workshop on Cyber-Physical Systems

THE IEEE Software Engineering Workshop (SEW) is the oldest Software Engineering event in the world, dating back to 1969. The workshop was originally run as the NASA Software Engineering Workshop and focused on software engineering issues relevant to NASA and the space industry. After the 25th edition, it became the NASA/IEEE Software Engineering Workshop and expanded its remit to address many more areas of software engineering with emphasis on practical issues, industrial experience and case studies in addition to traditional technical papers. Since its 31st edition, it has been sponsored by IEEE and has continued to broaden its areas of interest.

One such extremely hot new area are Cyber-physical Systems (CPS), which encompass the investigation of approaches related to the development and use of modern software systems interfacing with real world and controlling their surroundings. CPS are physical and engineering systems closely integrated with their typically networked environment. Modern airplanes, automobiles, or medical devices are practically networks of computers. Sensors, robots, and intelligent devices are abundant. Human life depends on them. CPS systems transform how people interact with the physical world just like the Internet transformed how people interact with one another.

The joint workshop aims to bring together all those researchers with an interest in software engineering, both with CPS and broader focus. Traditionally, these workshops attract industrial and government practitioners and academics pursuing the advancement of software engineering principles, techniques and practices. This joint edition will also provide a forum for reporting on past experiences, for describing new and emerging results and approaches, and for exchanging ideas on best practice and future directions.

Topics

The workshop aims to bring together all those with an interest in software engineering. Traditionally, the workshop attracts industrial and government practitioners and academics pursuing the advancement of software engineering principles, techniques and practice. The workshop provides a forum for reporting on past experiences, for describing new and emerging results and approaches, and for exchanging ideas on best practice and future directions.

Topics of interest include, but are not limited to:

- Experiments and experience reports
- Software quality assurance and metrics
- Formal methods and formal approaches to software development
- Software engineering processes and process improvement
- Agile and lean methods
- Requirements engineering
- Software architectures
- Design methodologies
- Validation and verification
- Software maintenance, reuse, and legacy systems
- Agent-based software systems
- Self-managing systems
- New approaches to software engineering (e.g., search based software engineering)
- Software engineering issues in cyber-physical systems
- Real-time software engineering
- Safety assurance & certification
- Software security
- Embedded control systems and networks
- Software aspects of the Internet of Things
- Software engineering education, laboratories and pedagogy
- Software engineering for social media

Technical Session Chairs

- **Bowen, Jonathan,** Museophile Ltd., United Kingdom
- **Hinchey, Mike** (Lead Chair), Lero-the Irish Software Engineering Research Centre, Ireland
- **Szmuc, Tomasz,** AGH University of Science and Technology, Poland
- **Zalewski, Janusz,** Florida Gulf Coast University, United States

Program Committee

- **Ait Ameur, Yamine,** IRIT/INPT-ENSEEIHT, France
- **Banach, Richard,** University of Manchester, United Kingdom
- **Challenger, Moharram**
- **Cicirelli, Franco,** Universita della Calabria, Italy
- **Ehrenberger, Wolfgang,** Hochschule Fulda, Germany
- **Gomes, Luis,** Universidade Nova de Lisboa, Portugal
- **Gracanin, Denis,** Virginia Tech, United States

- **Havelund, Klaus,** Jet Propulsion Laboratory, California Institute of Technology, United States
- **Hsiao, Michael,** Virginia Tech, United States
- **Nesi, Paolo,** DSI-DISIT, University of Florence, Italy
- **Pullum, Laura,** Oak Ridge National Laboratory, United States
- **Sanden, Bo,** Colorado Technical University, United States
- **Sekerinski, Emil,** McMaster University, Canada
- **Sojka, Michal,** Czech Technical University, Czech Republic
- **Trybus, Leszek,** Rzeszow University of Technology, Poland
- **van Katwijk, Jan,** Delft University of Technology, The Netherlands
- **Vardanega, Tullio,** University of Padova, Italy
- **Velev, Miroslav,** Aries Design Automation, United States
- **Zhu, Huibiao,** Software Engineering Institute - East China Normal University

DOI: 10.15439/2020F145
ISSN 2300-5963 ACSIS, Vol. 21

Evaluation of Open-Source Linear Algebra Libraries targeting ARM and RISC-V Architectures

Christian Fibich, Stefan Tauner, Peter Rössler, Martin Horauer
Dept. of Electronic Engineering, University of Applied Sciences Technikum Wien
Höchstädtpl. 6, 1200 Vienna, Austria
{fibich, tauner, roessler, horauer}@technikum-wien.at

***Abstract*—Basic Linear Algebra Subprograms (BLAS) has emerged as a de-facto standard interface for libraries providing linear algebra functionality. The advent of powerful devices for Internet of Things (IoT) nodes enables the reuse of existing BLAS implementations in these systems. This calls for a discerning evaluation of the properties of these libraries on embedded processors.**

This work benchmarks and discusses the performance and memory consumption of a wide range of unmodified open-source BLAS libraries. In comparison to related (but partly outdated) publications this evaluation covers the largest set of open-source BLAS libraries, considers memory consumption as well and distinctively focuses on Linux-capable embedded platforms (an ARM-based SoC that contains an SIMD accelerator and one of the first commercial embedded systems based on the emerging RISC-V architecture). Results show that especially for matrix operations and larger problem sizes, optimized BLAS implementations allow for significant performance gains when compared to pure C implementations. Furthermore, the ARM platform outperforms the RISC-V incarnation in our selection of tests.

***Index Terms*—Embedded Systems, Basic Linear Algebra Subprograms, BLAS, Benchmarks, ARM, RISC-V**

I. Introduction

EDGE computing – processing sensor data as close to their origin as possible – is a paradigm to de-centralize processing and storage in order to improve the scalability of IoT applications. More potent embedded CPUs allow more complex processing tasks, for example signal and image processing operations as well as inference (and even training) of neural networks. Vector or matrix operations are essential building blocks of the digital algorithms that are at the core of these applications. This raises the question whether it is viable to re-use and adapt proven mathematical software libraries written for server and desktop computers for embedded target platforms.

Netlib, a repository for open-source mathematical software run by AT&T, Bell Labs, University of Tennessee, and Oak Ridge National Laboratory, both maintains the BLAS specification document [1] and provides a reference implementation of this specification[1]. BLAS essentially describes a programming interface for three levels of algorithms: level 1 contains vector-vector operations, level 2 defines vector-matrix operations, and level 3 specifies matrix-matrix operations all for single- and double-precision real numbers, as well as for single- and double-precision complex numbers. The reference implementation itself does not contain optimized code – for example, matrix-multiplication is implemented using a simple iterative algorithm with three nested loops – it is, however, widely used as a performance baseline for implementers of optimized BLAS libraries. For example, implementations taking advantage of Single Instruction Multiple Data (SIMD) hardware extensions provided by modern CPUs as well as GPUs via interfaces such as CUDA and OpenCL are common.

Since many modern embedded applications rely on algorithms that mandate complex computations an evaluation whether some BLAS implementations can be re-used under resource constraints imposed by typical embedded platforms seems in demand. Unlike the usual target systems for BLAS libraries this work evaluates them on comparably small embedded systems that display significantly divergent characteristics due to architectural differences. For example, modern x86 CPUs can retire over 10 instructions per cycle per core under ideal circumstances while running at a multi-GHz clock rate and dozens of MB of caches. This is in stark contrast to embedded CPUs, which are often still non-speculative in-order architectures with sub-GHz frequencies. In particular, in this paper we evaluate different implementations of the BLAS specification targeting the ARM Cortex-A9 and RISC-V RV64GCSU architectures, respectively. Both share some similarities such as having RISC architectures, comparably low clock frequencies and other attributes often found in embedded systems. Both Instruction Set Architectures (ISAs) support extensions leading to a wide variety of implementations. We chose two representative examples for our tests that are described in more detail in Section IV.

Furthermore, it was the intention of the authors to present a more comprehensive overview and evaluation of existing open-source BLAS libraries, when compared to related work that is discussed in Section II. In Section III, the scope of this work – the evaluated libraries as well as the benchmark applications and CPU architectures that constitute the basis of

This work has been supported by the Austrian Federal Ministry for Digital and Economic Affairs (BM:DW) and the National Foundation for Research, Technology and Development as related to the Josef Ressel Center “Innovative Platforms for Electronic-Based Systems” (INES) managed by the Christian Doppler Society (CDG).

We thank Kapsch TrafficCom AG for their support of this research.

[1] https://www.netlib.org/blas/, last visited on 2020-06-27

these evaluations – is described. Section IV provides details on the specific build and run-time environments as well as employed metrics and measurement methodologies. Finally, the results of the evaluations are discussed in Section V before the paper concludes.

II. Related Work

Due to its widespread use, libraries implementing the BLAS specification have been compared in the past. For example, [2] presents a wrapper for multiple BLAS library implementations facilitating their interchangeable use along with an evaluation by way of several benchmarks. Similar evaluations are provided by the *BLIS* framework presented in [3] targeting the performance of several BLAS level 2 and 3 algorithms and comparing them with an optimized ISO C implementation.

The code generator *LGen* generates computational kernels for some BLAS operations with fixed-sized operands that are optimized for different target architectures and thus improve over generic implementations that employ varying operand sizes, see [4]. This approach was refined in [5] where the targets have been shifted to some high-end embedded processors.

Finally, *BLASFEO* (see also Section III-B) is a BLAS implementation that is intended to enable executing optimization algorithms on embedded targets. It is especially optimized for widely used embedded application processors (e.g., ARM Cortex-A7, -A9, -A15) as well as modern Intel and AMD architectures (e.g., Intel Haswell, Sandy Bridge). In [6], the authors compare their implementation to multiple other BLAS libraries. BLASFEO provides its own API that is better suited for small input sizes than standard BLAS as it reduces the overhead of aligning the matrices for internal processing. However, the standard BLAS API has been implemented coequally to the native API and evaluated on Intel and ARM microarchitectures [7].

TABLE I: An overview of BLAS Benchmarks

	FlexiBLAS	BLIS	LGen	BLASFEO		*This work*
	2013	2015	2015	2018	2020	2020
	[2]	[3]	[4], [5]	[6]	[7]	
ATLAS	✓	✓	✓			✓
BLASFEO				✓	✓	✓
BLIS		✓			✓	✓
Eigen			✓	✓		✓
Intel MKL	✓	✓	✓	✓	✓	
Ne10						✓
Netlib	✓					✓
OpenBLAS	✓	✓		✓	✓	✓
Performance	✓	✓	✓	✓	✓	✓
RAM Footprint		✓				✓
x86	✓	✓	✓	✓	✓	
ARM			✓	✓	✓	✓
RISC-V						✓

An overview of related BLAS evaluations is shown in Table I listing evaluated libraries, their scope, and the respective target architectures. The last column depicts the contribution of this work that focuses on embedded platforms, in particular targeting ARM Cortex-A9 and RISC-V. The focus on embedded systems ruled out Intel's Math Kernel Library (MKL) implementation that specifically targets x86 architectures. Instead, the performance of ARM's Ne10 library was included. Although this is not an implementation of the BLAS specification, it has an overlapping scope. The treatment of memory consumption of the libraries and the inclusion of RISC-V sets this work apart from past publications.

In addition to academic literature, several authors of the libraries evaluated in this work have published their own benchmark results. ATLAS provides timing data for the `DGEMM` (double precision generic matrix multiplication) BLAS function (dimensions between 1000 and 2000) in various library versions and CPU architectures (e.g., Intel, MIPS, SPARC)[2]. The results, however, are provided for up to ATLAS version 3.9.5, which is about 8 years old. The Eigen project provides benchmark results of various level-2 and level-3 BLAS functions, as well as more complex functions such as lower-upper factorization of matrices[3]. The results date from 2011 and are compared to GotoBLAS (an OpenBLAS predecessor), Intel MKL, and ATLAS 3.8.3. BLIS provides the most recent results, dating from 2020. Results are provided for server- and workstation-class CPUs (i.e., 10s of cores, ARM ThunderX2, Intel SkylakeX, Intel Haswell, AMD EPYC). BLAS level 3 functions are evaluated against Eigen, OpenBLAS, BLASFEO, and MKL. Different sets of results are available for large[4] and small matrices[5].

Unlike the publications discussed above this work examines BLAS libraries on comparably small embedded systems. Preliminary results of this evaluation were published in [8]. This work widens the scope of the analysis both by inclusion of the RISC-V architecture and the very recent BLASFEO linear algebra library. It is, to the best of our knowledge, the first work that evaluates the performance of general-purpose linear algebra libraries on the RISC-V architecture. Furthermore, it extends previous work with a more in-depth analysis of performance differences between the different libraries, especially in the dot product, matrix multiplication, and neural network benchmarks.

In the past 10+ years the use of linear algebra algorithms on heterogeneous systems with potent GPUs have been investigated for High-Performance Computing (HPC). Due to the limitations of the communication between host CPUs and GPUs these implementations (e.g., ViennaCL, cuBLAS) often use custom APIs.[6] Similar considerations affect the use of custom accelerators on programmable logic, which is often

[2] http://math-atlas.sourceforge.net/timing/, 2020-06-27

[3] https://eigen.tuxfamily.org/index.php?title=Benchmark, 2020-06-27

[4] https://github.com/flame/blis/blob/master/docs/Performance.md, 2020-06-27

[5] https://github.com/flame/blis/blob/master/docs/PerformanceSmall.md, 2020-06-27

[6] ViennaCL's API is derived from the C++ `Boost.uBLAS` library that is object-oriented and uses operator overloading. cuBLAS supports a BLAS-compatible interface but it is deprecated and requires additional manually initiated transfers, cf. https://docs.nvidia.com/cuda/cublas/index.html

facilitated by High-Level Synthesis (HLS).[7] For these reasons directly comparing the performance of BLAS libraries running on other hardware than general-purpose CPUs is out of scope of this paper.

III. Scope of the Evaluation

A. Target Hardware

Unlike in high-performance computing the market for embedded processors is less concentrated due to the wide variety of applications. Nevertheless (or possibly because of that) ARM has been the leading provider for embedded RISC IP cores in the past few decades offering a wide range of processors from tiny cores for microcontrollers to the base for high-end smartphones with multiple gigabytes of memory. Almost all of its recent CPUs (at least optionally) comprise an FPU capable of SIMD operations called *NEON*. We focus our work on the mid-range Cortex-A9 implementation of the well-established and widely used ARMv7 architecture as one of our targets.

In stark contrast to ARM the RISC-V architecture is quite young but has gained substantial interest from academia and industry alike.[8] One of the obvious reasons is that the use of its ISA is royalty-free and thus a wide range of implementation exists. However, as of summer 2020 the standardization of the ISA has only frozen the most significant parts but some specifications have not been finalized relevant to embedded systems in general (e.g., the Bit Manipulation Extension[9]) and this paper in particular (e.g., the Vector Extension[10]).

The ISA allows for a modular design by offering base specifications for unprivileged and privileged execution environments as well as standard (i.e., defined by the foundation) and custom ISA extensions. Designers can thus build cores tailored to the specific use cases. For example, a RISC-V core developed in the PULP project contains a dot-product accelerator as a non-standard vector extension [9]. This accelerator can calculate the dot product of two vectors comprising two 16-bit or four 8-bit integers.

B. Evaluated Libraries

This section provides a brief description of the distinctive features of each library under investigation. The libraries were evaluated in unmodified form, with the respective versions and release dates indicated in Table II.

The aim of the ATLAS (Automatically Tuned Linear Algebra Subprograms) project[11] is to provide implementations that are highly optimized for the particular target platform they will be used on. For this purpose, ATLAS contains many different variants of its kernels that suite best for particular properties of the target (e.g., cache line size, vector accelerators). ATLAS' build process is carried out on the actual target, requiring a C compiler on the target platform. During the build process, the kernel variants delivering the highest performance are determined empirically. Further details on the basic principles of this optimization and the ATLAS project can be found in [10]. In the results section of this work, the version of ATLAS built with default settings is referred to as "ATLAS". The "ATLAS-Neon" version was built with disabled IEEE-754 compatibility options, potentially allowing better tuning to the NEON vector processor in ARM CPUs. While newer *unstable* releases of ATLAS are available, the most recent *stable* release of ATLAS was evaluated in this work.

TABLE II: Summary of evaluated libraries

Library	Version / Commit	Release Date
ATLAS	3.10.3	2016-07-28
BLASFEO	Commit `d404e3471dbb`	2019-10-23
BLIS	Commit `bc16ec7d1e2a`	2019-09-23
Eigen	Commit `8e409c71423f`	2019-09-27
Ne10	Commit `1f059a764d0e`	2018-11-15
Netlib BLAS	Commit `b5dd8d4016f7`	2019-09-12
OpenBLAS	Commit `2beaa82c0508`	2019-10-09

OpenBLAS[12] is widely-used optimized implementation of the BLAS specification. Details on the especially optimized matrix-multiplication kernels part of OpenBLAS and its predecessor GotoBLAS can be found in [11]. OpenBLAS is implemented in C, but provides assembly implementations of performance-critical kernels for several CPU architectures and accelerators. Architectures for which such kernels exist include MIPS, ARMv6/v7/v8, x86, and Power8/9. However, the low-level kernels for ARMv7 relevant to this work only make use of the VFPv3 instruction set, and do not include specific support for the NEON SIMD engine.

BLASFEO [6] is a fairly new linear algebra library[13]. It aims at improving performance at small vector and matrix sizes (between 10s and 100s per dimension). The authors motivate this goal with optimization algorithms calculated on embedded devices themselves. BLASFEO consists of three independent implementations of linear algebra functionality: *Wrapper* which is a custom interface to standard BLAS libraries such as Netlib BLAS, *Reference* which is implemented in ANSI C and serves as a reference that is easily portable to new architectures, and *High-Performance*. The *High-Performance* version provides kernel implementations in assembly optimized for various architectures and common accelerators (e.g., SSE3, AVX, AVX2 available in x86 CPUs and VFPv3/VFPv4, NEON, and NEON2 found in ARM CPUs). To allow for maximum efficiency many assembler routines are inlined, which can be further enforced at compile time. BLASFEO uses an internal, aligned format for storing matrices and vectors, and an API that differs from BLAS. The latter difference is especially relevant to matrix multiplication: While BLAS provides the `xGEMM` functions that allow transposing

[7] For example FBLAS (https://github.com/spcl/FBLAS) or Xilinx' Vitis BLAS Library (https://www.xilinx.com/products/design-tools/vitis/vitis-libraries/vitis-blas.htm

[8] https://riscv.org/members-at-a-glance/, 2020-06-27

[9] https://github.com/riscv/riscv-bitmanip, 2020-06-27

[10] https://github.com/riscv/riscv-v-spec/blob/master/v-spec.adoc, 2020-06-27

[11] http://math-atlas.sourceforge.net, 2020-06-27

[12] https://www.openblas.net, 2020-06-27

[13] https://github.com/giaf/blasfeo, 2020-06-27

either of the two input matrices to be multiplied via function parameters, BLASFEO offers different functions for each of these cases called `blasfeo_sgemm_{nn|nt|tn|tt}()`. However, the authors have also implemented a standard-conforming interface (denoted *CBLAS* hereinafter) that covers most but not all BLAS functions [7]. As a consequence, BLASFEO was evaluated only in the SDOT and SGEMM benchmark by using its native and CBLAS API, respectively.

Another BLAS library investigated in this work is BLIS [3][14]. A main motivation for the originators of BLIS was to provide portability to new architectures but also to accelerators. This is done by using a set of target-specific kernels which implement the BLAS routines. A generic implementation exists by relying on compiler optimizations. Furthermore, ports are available for ARMv7 or ARMv8 architectures, Intel or Power7 as well as others. The ARMv7a port provides kernels for both single-precision as well as double-precision floating-point matrix multiplications utilizing NEON SIMD intrinsics. As a special feature, checks for errors related to, e.g., buffer sizes or matrix and vector dimensions are performed by BLIS. However, according to the description of the test suite[15], these checks may result in performance degradation, and for this reason all the error checks have been disabled for our benchmark evaluations.

The last library that has been considered for our benchmarks is Ne10[16] which includes functions for generic linear algebra but also for signal processing or image processing. Three versions of the Ne10 library are provided: a portable implementation using plain C, another implementation that makes use of NEON intrinsics, and finally, an implementation based on ARM assembly code. However, no BLAS interface is provided by Ne10. Furthermore, operations like generic matrix multiplication or generic dot product are not provided and therefore, the Ne10 vector multiplication kernel has been used to implement this kind of calculations for our work. Since the optimized version of Ne10 (based on ARM assembly implementations) was benchmarked, only results for ARM are available.

C. Benchmark Applications

Our first synthetic benchmark application is the dot product of two vectors which is an example for an often used linear algebra operation. It can be expressed by a BLAS level 1 operation called `[DS]DOT`. Since the dot product of two vectors can be implemented as a Multiply and Accumulate (MAC) operation it can be assumed that this operation is perfectly suited to transformations such as automatic vectorization or utilization of SIMD facilities. Our implementation performs a calculation of the dot product of two randomly selected values for a given number of iterations. The time required for each calculation is measured and at the end the mean value per `SDOT` function call is calculated. The vector length varies from 4 to 1500[17]. Note, that a generic dot product operation does not exist in the ARM Ne10 library and thus, vector multiplication plus iterative addition of the results are used in our Ne10 implementation.

The next synthetic benchmark application is matrix multiplication which is also an example for an often used linear algebra operation. It can be implemented by the BLAS level 3 operation `[DS]GEMM`. Our implementation performs a calculation of the product of two randomly selected square matrices for a given number of iterations and measures an average of the runtime. The dimensions of the square matrix used for our benchmark range from 4x4 to 1500x1500. Since a matrix multiplication operation does not exist in the ARM Ne10 library, vector operations have been used instead, for the Ne10 implementation.

Our final synthetic benchmark application is LINPACK-PC which is a C implementation of benchmarks for linear algebra operations provided by Netlib[18]. A number of BLAS operations such as `[DS]DOT`, `[DS]SCAL` (scaling a vector by a constant) or `I[DS]AMAX` (calculating the maximum element of a vector) are used by the LINPACK-PC benchmarks, operating on matrices or vectors with a maximum dimension of 200 to 201 elements. The performance of a series of operations is measured and are reported in units of MFLOPS.

To verify that our results hold in real-world applications too we employ a benchmark in the field of Artificial Neural Networks (ANNs). In detail, we made use of KANN[19], which is a C-based framework for constructing and training of artificial neural networks. Through the CBLAS interface KANN makes use of the BLAS operations `SAXPY` (vector addition) and `SGEMM`. Two out of the existing sample applications provided with KANN are used for our benchmarks: (1) *RNN* is basically a Recurrent Neural Network (RNN) using 64 neurons trained to add integer numbers, and (2) *MNIST-MLP* implements a MLP (Multi-Layer Perceptron) with 64 neurons to process data from the MNIST (Modified National Institute of Standards and Technology) database of handwritten digits. For both ANN benchmarks the training time as well as the interference time has been evaluated.

IV. Experimental Setup

A. ARM Cortex-A9 on Cyclone V SoC Development Board

An Intel/Altera Cyclone V SoC Development Kit[20] was used as an ARM-based embedded platform. This board contains a Cyclone V SoC FPGA that implements a hard-wired dual-core ARM Cortex-A9 CPU besides the programmable logic fabric. The usage of an FPGA board was motivated by the fact to generate BLAS accelerators by means of HLS as

[14] https://github.com/flame/blis, 2020-06-27

[15] https://github.com/flame/blis/blob/c665eb9b888ec7e41bd0a28c4c8ac4094d0a01b5/docs/Testsuite.md, 2020-06-27

[16] https://github.com/projectNe10/Ne10, 2020-06-27

[17] 200k iterations are performed for vector dimensions 4–64, 100k iterations for 100–500 and 50k for dimensions 1000–1500

[18] https://www.netlib.org/benchmark/linpack-pc.c, 2020-06-27

[19] https://github.com/attractivechaos/kann, 2020-06-27

[20] https://www.intel.com/content/www/us/en/programmable/products/boards_and_kits/dev-kits/altera/kit-cyclone-v-soc.html, 2020-06-27

future research work while for this work the device manufacturer's *Golden Hardware Reference Design*[21] was used. The ARM core implements a 32 KiB L1 data cache, a 32 KiB L1 instruction cache as well as a shared L2 cache (512 KiB). A VFPv3 floating-point unit and a NEON SIMD engine are also integrated on-chip [12] while the board comes with peripherals such as 1 GiB of external DDR3-1600 memory, a GBit Ethernet PHY and an SD card slot to boot a Yocto Linux (using a 3.9.0 kernel).

B. RV64GCSU on SiFive Unleashed Board

The second target is one of the first commercially available 64-bit RISC-V development boards featuring four `RV64GCSU` cores in the SiFive Freedom U540 SoC and has 8 GiB DDR4-2400 RAM on board. The ISA specification `RV64GCSU` is shorthand for `RV64IMAFDCSU` which stands for 64-bit registers, compressed multiplication, atomic, single- and double-precision floating-point instructions with support for supervisor and user mode in the RISC-V ISA standard. The HiFive Unleashed[22] does also feature Ethernet and boots from a (micro) SD card like the Cyclone V board. The cores in the U540 are based on SiFive's U54 with 32 KiB L1 cache for data and instructions, respectively, and share a common 2 MiB L2 cache. The manufacturer-provided buildroot Linux with a 4.15 kernel was used.

C. Build Environment

Our build environment for the libraries under investigation, Section III-B, was as follows (see Section V for additional notes):

- As compilers, `gcc` and `gfortran` from the GNU Compiler Collection (GCC) version 8.3.0 (from Debian 10) for ARM (`armhf` Hard-Float Application Binary Interface (ABI)) and 64-bit RISC-V, were used
- GCC's `-O3` flag was used to specify the desired level of optimization. This flag enables, for example, automatic vectorization and loop peeling.
- Moreover, `-ffast-math` has been used to allow specific non-IEEE754-compliant optimizations in order to fully exploit the ARM Neon engine
- All libraries, including libc, have been linked statically.
- Since some of the libraries described in Section III-B provide multi-code/multi-thread support while others do not (NETLIB or Ne10, for instance) only single-threaded versions were benchmarked to allow a fair comparison between the libraries. Results may be of interest for single-core systems.

During the ATLAS build flow, the build system adapts the kernels used to the target platform. This requires the build flow to be executed on the target platform itself, requiring a full toolchain installation. To enable benchmarking ATLAS on ARM, the GNU C compiler (Version 7.2.0) was built using crosstool-NG[23] with hard-float ABI and support for ARM NEON.

D. Measurement Mechanics

Both, the ARM and RISC-V based CPU were configured to run at 800 MHz to allow for direct comparability. However, the raw memory bandwidth available to the RISC-V SoC is 50% higher in theory. Performance and memory measurements were taken using facilities provided by the target system:

- The POSIX facility `clock_gettime()` was used to measure the time spent in the respective core BLAS functions of the synthetic dot product benchmark and the matrix multiplication benchmark. A timestamp was taken from the system-wide clock (via untampered `CLOCK_REALTIME`) before and after each call of the respective BLAS function. These times were accumulated and divided by the number of invocations in order to determine the average time per call.
- The LINPACK-PC library has a built-in mechanism to determine its performance in terms of Million Floating Point Operations per Second (MFLOPS) utilizing the ISO C `clock()` facility.
- The total execution time of the respective binaries was used as a performance metric in the KANN benchmarks. In the RNN inference and training case, as well as the MLP training case, the runtimes lie in the range of several minutes. Execution time of the executables was measured using GNU `time`.
- The memory footprint of the libraries has been determined over a snapshot of the application's Resident Set Size (RSS). This was done with Linux' `/proc/<PID>/smaps` interface that provides information about all memory mappings of a process. To capture realistic values including potential internal memories within the libraries these snapshots are taken with swapping disabled between the two middle iterations of the whole benchmark.

  Since the `smaps` interface is not provided by the RISC-V Linux kernel, only results for the ARM platform can be shown here (anyway, the CPU architecture should only have minor influence on the memory footprint). In order to avoid interference between the necessary instrumentation with the performance results, this was done using purpose-built executables for every library.

V. Results & Discussion

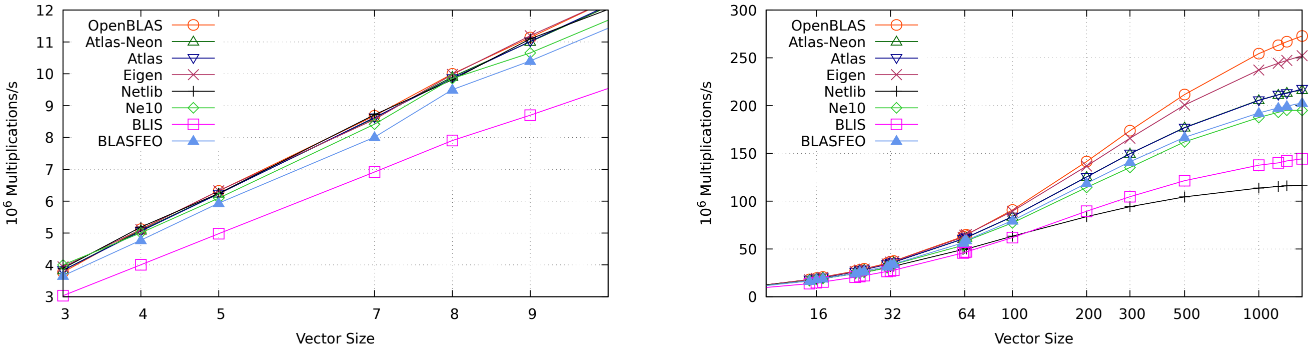
A. Fundamental Metrics: SDOT & SGEMM

In Figures 1 and 2 the performance results obtained from the single-precision dot product are shown. The results are split into smaller vector sizes (3-10, Figures 1a and 2a) and larger vector sizes (10-1500, Figures 1b and 2b) to retain readability at both ends of the evaluated spectrum of vector sizes. Figure 3 depicts the results of the matrix multiplication benchmarks. The y-axes of these figures are scaled to reflect the number

[21] https://rocketboards.org/foswiki/Documentation/GSRDGhrd, 2020-06-27

[22] https://www.sifive.com/boards/hifive-unleashed, 2020-06-27

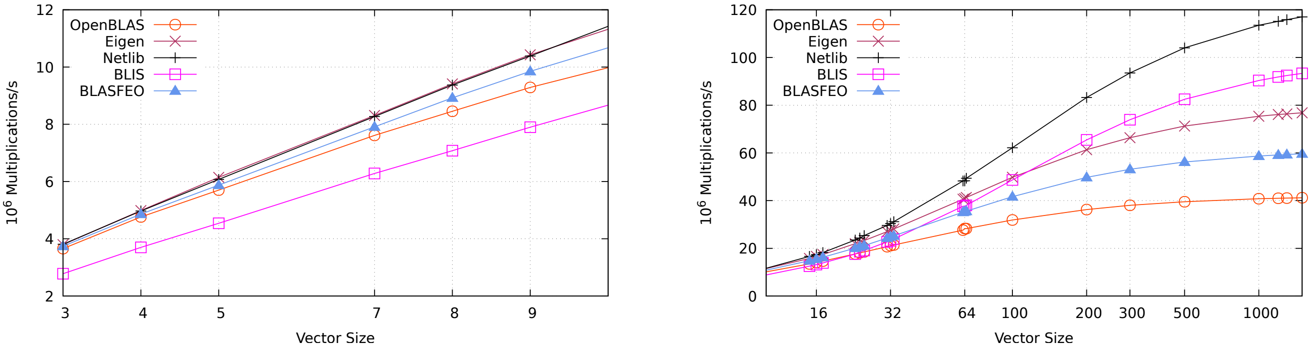
[23] https://crosstool-ng.github.io, 2020-06-27



(a) Vector Sizes 3–10

(b) Vector Sizes 10–1500

Fig. 1: SDOT performance in multiplications calculated per second on ARM platform



(a) Vector Sizes 3–10

(b) Vector Sizes 10–1500

Fig. 2: SDOT performance in multiplications calculated per second on RISC-V platform

of floating-point multiplications processed per second (i.e., a higher number corresponds to better performance). Since the number of multiplications is the determining factor this scaling allows for sharp discrimination between results. This metric was obtained by multiplying the number of SDOT/SGEMM calls per second with the respective multiplications per call (n for SDOT and $n^3 + 2n^2$ for SGEMM where n is the dimension of the input data).

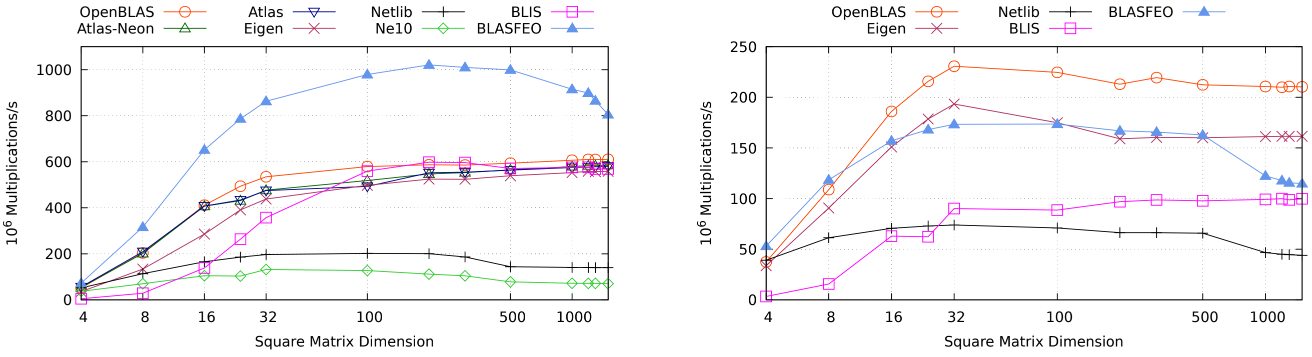
On ARM the widely-used OpenBLAS library performs well in both test cases. In the SDOT benchmark (Figure 1), OpenBLAS and Eigen are the fastest libraries, with effects getting increasingly noticeable with vectors longer than 100 elements. Especially at small vector lengths, BLIS falls far behind the other evaluated libraries. In the SGEMM benchmark (Figure 3a), OpenBLAS and ATLAS are clearly outperformed by BLASFEO, especially by small to medium dimensions but also at bigger ones (by about 25% at dimension 1500). The dip of BLASFEO's performance relative to the other libraries' that is visible for the largest matrices is probably because not all the data fits into the last level of cache and BLASFEO's lack of cache blocking. Similar results have been shown by its authors [6]. The performance of Eigen and BLIS is once again worse at smaller matrix sizes.

On RISC-V the situation for SDOT (Figure 2) is very different – almost inverse to that on ARM. The Netlib implementation shows the best results for all vector sizes, beating OpenBLAS and BLASFEO by almost a factor of 3 and 2 respectively, although the latter are the best-performing libraries in most of our other benchmarks no matter the architecture. BLIS is trailing the field at small vector sizes but performs relatively well for bigger vectors with multiple hundreds of elements where it becomes the second fastest library unlike in other benchmarks.

SGEMM on RISC-V (Figure 3b) looks similar to ARM at a much lower absolute performance level. Even the decreasing performance of BLASFEO starting at about dimension 500 is clearly visible. The most notable difference is that OpenBLAS is faster than BLASFEO for all matrices greater than 16×16. In addition to the absolute performance, different efficiency peaks of the evaluated libraries can be observed in the SGEMM benchmark results. The implementations of Netlib and Ne10 are most efficient at small matrix sizes, while the more optimized libraries OpenBLAS, ATLAS, Eigen, BLIS and BLASFEO are most efficient at sizes between 16 and 32.

B. Memory Usage

Figure 4 shows the memory consumption of the benchmark executables (statically linked with each BLAS library, respec-



(a) ARM platform

(b) RISC-V platform

Fig. 3: SGEMM performance in multiplications per second

tively) on the ARM platform. This data is provided for the smallest input size of the dot product and matrix multiplication benchmarks in order to assess the base memory consumption of each library. It subsumes both memory allocated for the input/output vectors/matrices and the library's own memory consumption (e.g., for buffers and other internal variables). As matrix dimensions grow larger, their size begins to dominate the application's memory consumption. This has been observed in the SGEMM benchmark starting with matrices of dimensions 200 × 200 and larger[24]. It can be seen that the memory consumption of the optimized ATLAS, BLASFEO and OpenBLAS libraries lies close to the NETLIB reference implementation, while Eigen and BLIS consumption is considerably higher.

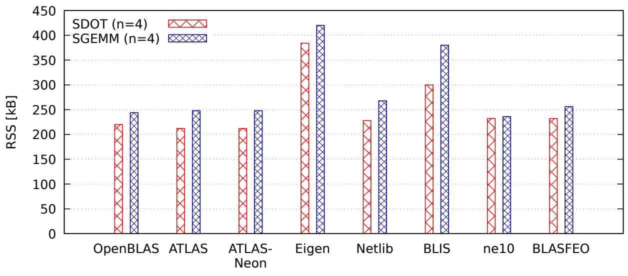


Fig. 4: Total application memory usage (RSS) on ARM

C. LINPACK

The results obtained by integrating the evaluated libraries into the LINPACK-PC benchmark are shown in Figure 5. LINPACK-PC calculates an MFLOPS metric from the number of iterations of its core algorithm completed in a set period of time. As Ne10 and ATLAS could not be benchmarked on the RISC-V platform, no bars for RISC-V are shown here. The results denoted by *LINPACK-** show the performance of the plain-C implementations of the used linear algebra operations that are provided by LINPACK-PC itself. The *"Unroll"* results were obtained from the same C source code that is provided in unrolled form by LINPACK. The former, however, was compiled with GCC's `-O3` optimizations turned on while the latter was compiled using GCC's `-ffast` compiler flag. *LINPACK-ffast* denotes a variant of the C code that has not been unrolled, but compiled with `-ffast` optimizations turned on.

These optimizations improve the performance of the standard C implementation even beyond the evaluated BLAS libraries on the ARM platform. In these cases, the entire program is available to the compiler as C source, and does not contain calls to an external library, which might enable more extensive optimization. On the RISC-V platform in contrast, there seems to be no benefit attached to these optimizations. Furthermore, the BLAS libraries cannot even improve on the performance of the basic LINPACK implementation, reaching only about 72–97% of its throughput.

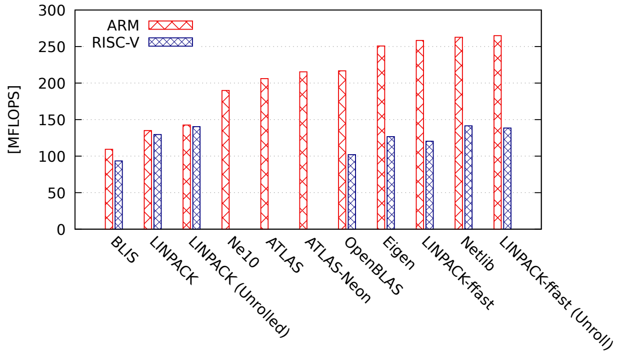


Fig. 5: Averaged LINPACK performance

D. Neural Networks

Figure 6 shows the speedup of the inference and training of two types of neural networks (*MLP* and *RNN*) relative to the C implementation provided by the KANN project itself (denoted as *"KANN"* in the graphs). For the training cases on the ARM platform, all evaluated libraries lead to a performance gain,

[24] Three single-precision 200 × 200 matrices (A, B, and C for SGEMM) consume 3 × 200 × 200 × 4 Byte = 480 kB, more than each libraries' base memory consumption.

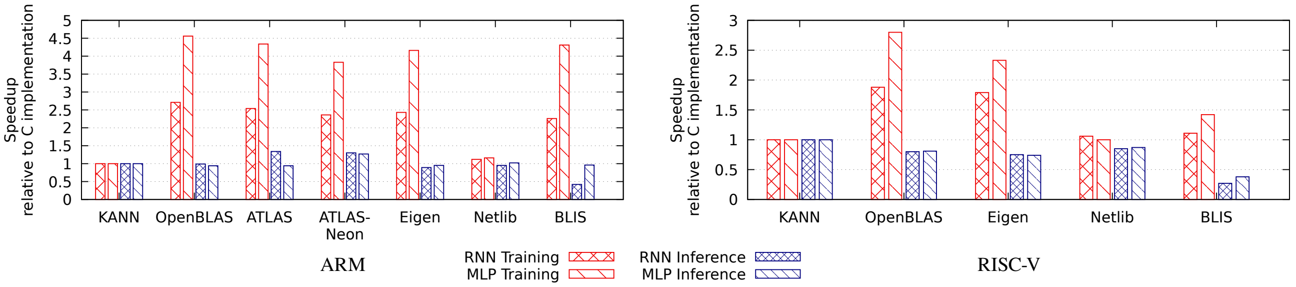
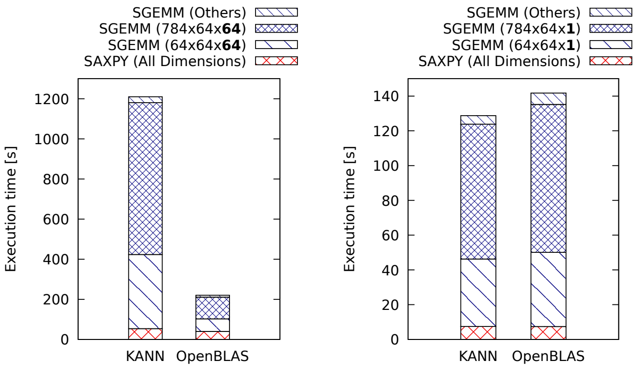


Fig. 6: Speedup of KANN benchmark relative to plain C implementation



(a) Training

(b) Inference

Fig. 7: Distribution of execution times of BLAS functions in KANN benchmark on ARM platform

even Netlib's reference implementation. To a lesser extent this is even true when training the neural networks on RISC-V although all libraries show significantly worse relative results.

The inference case behaves quite differently to the training as some libraries even lead to a slowdown of the application. Figure 7 depicts a plausible explanation. It compares the plain C implementation (*"KANN"*) to the fastest BLAS library implementation (OpenBLAS) regarding their cumulative execution time spent in BLAS functions when running on the ARM platform. For this purpose, the BLAS wrapper used by KANN was instrumented. The execution times of the MLP and RNN benchmarks for each BLAS function and dimension were summed up. This execution time is subdivided into calls to `SAXPY` of all dimensions, the two most prominent dimensions of `SGEMM`, and all other dimensions of `SGEMM`. In both training and inference, `SGEMM` accounts for the overwhelming majority of execution time spent in BLAS functions. In the inference case (Figure 7b), the two most prominent `SGEMM` operations act upon matrices where at least one dimension is 1. In these cases, OpenBLAS takes about 10% longer than the C implementation (supposably due to the overhead of the API). In contrast, the most prominent `SGEMM` operations in the training case (Figure 7a) act on significantly larger matrices, where OpenBLAS clearly outperforms the C implementation. BLIS is slower than all other libraries in the inference benchmarks, which may correlate with the lower performance at smaller matrix dimensions in the matrix multiplication benchmarks (see Figure 3a), and overall lower vector performance (see Figure 1).

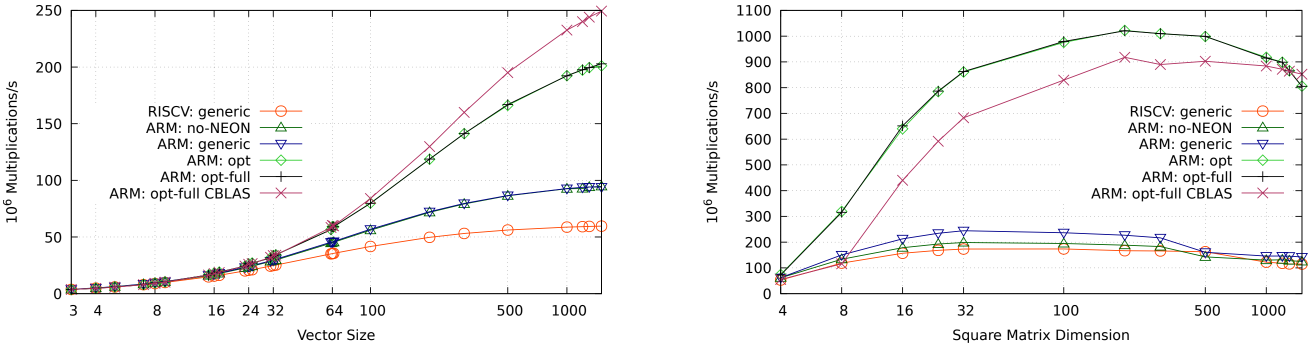
E. BLASFEO

Since BLASFEO is still in active development only the basic `SDOT` and `SGEMM` benchmarks were tested as explained in Section III-B. Instead of a broad application field we evaluated different build and execution options provided by BLASFEO:

- A *generic* C implementation allows using BLASFEO even if no architecture-specific optimizations are available. We used this option to build the library for RISC-V where this is the case. For comparison two versions based on this option were created on ARM: One uses all available compiler optimizations (notably the use of NEON for SIMD instructions) and one uses the VFPU but without exploiting NEON (*no-NEON* in the respective figures).
- Two *optimized* builds exploiting BLASFEO's assembler implementations and native API were generated. While one (*opt*) reflects the default configuration of the library for supported architectures, the other one (*opt-full*) increases the amount of inlined assembly routines thus avoiding some function calls.
- Additionally, we evaluate the use of BLASFEO via its CBLAS-compatible API.

The results depicted in Figure 8 clearly show that the effort of manual optimization despite advancements in compilers still pays off. In the `SDOT` benchmark all versions including the one running on the RISC-V platform are within 10% for very small vector lengths (i.e. ≤ 10). At vector sizes of about 16–32 four groups begin to form. The generic implementation on RISC-V is clearly the slowest contender reaching only about 25% of the speed of the fastest solution for the largest vector size of 1500 elements. The second group consisting of the two compiler-optimized ARM implementations are able to get about twice as fast as the RISC-V library. Their almost equal result shows that GCC was not able to exploit NEON in the DOT application.

The two assembler implementations with the native BLASFEO API that form the third group calculate slightly over



(a) Million multiplications calculated per second by `SDOT`

(b) Million multiplications calculated per second by `SGEMM`

Fig. 8: Comparison of different build options of the BLASFEO library

200 million multiplications per second for very large vectors. These two variants perform almost the same in this benchmark as well as in `SGEMM` (cf. Figure 8b). Most interestingly though is the result of the CBLAS-enabled implementation that beats all others by well over 20%.

The results for `SGEMM` in Figure 8b are more refined. For 4×4 matrices the performance differences are relatively small showing well-working compiler optimizations. For bigger matrices, however, the hand-optimized implementations are by far superior (up to a factor of 6 for very large matrices). The CBLAS curve might be attributable to the API overhead that diminishes with bigger dimensions (and becomes negligible at about dimension 1000). In this benchmark the RISC-V implementation is able to almost match the compiler-optimized ARM versions. With smaller matrices the ARM versions are slightly better, also showing a small drawback when NEON is not used.

VI. Conclusion

In this work, a selection of linear algebra libraries providing the BLAS interface is evaluated. Compared to related work (which is, as explained in Section II, at least partly outdated) our evaluation covers the largest set of open-source BLAS libraries (see Table I), considers (unlike existing evaluations) memory consumption as well focuses on embedded platforms (to the best of our knowledge, it is the first work that evaluates the performance of general-purpose linear algebra libraries on the RISC-V architecture). Based on the results of these evaluations, as presented in Section V, we draw the following conclusions:

It can be seen from the LINPACK and the inference case in the KANN benchmark, that in some applications a plain C implementation may perform better (or at least as good) as an optimized linear algebra library. The compiler may be able to automatically optimize these applications to an extent that no significant additional performance can be gained by a hand-optimized implementation of the core mathematical operations. If, however, the application is more complex or uses core algorithms that cannot yet be automatically vectorized by the compiler, using optimized libraries can increase throughput. This can be seen in the matrix multiplication benchmark and in the training case in the KANN benchmark, both profiting from fast matrix-matrix multiplication offered by optimized BLAS libraries.

While in some cases a simple C implementation in combination with an optimizing compiler might suffice, BLAS offers a standardized interface to linear algebra functionality. This facilitates prototyping and modularization, allowing to replace the performance-critical parts of an application if required. In most of the benchmarks, the optimized BLAS libraries perform either better or comparable to a C implementation, thereby outweighing the slight performance advantage of non-standard C implementations.

From the results on the relatively new RISC-V target one can easily motivate further work on ISA extensions (similar to PULP's [9] and the proposed standard Vector Extension) since absolute performance levels are only a fraction of ARM's. They also show that generic improvements in libraries are possibly nullified by increased overhead or unanticipated working conditions and that architecture-specific optimizations are necessary to consistently improve results over straightforward implementations.

While performance is in most cases the distinctive metric to select computing libraries such as the ones described here, the aspect of power consumption is of similar importance to many embedded systems that often run on battery power and/or have to comply to strict thermal limits. In future work these potential contradictory properties should be commonly evaluated.

References

[1] BLAST Forum, "Basic Linear Algebra Subprograms Technical Forum Standard," https://netlib.org/blas/blast-forum/blas-report.pdf, 2020-06-27, University of Tennessee, Knoxville, Tennessee, Tech. Rep., 2001.

[2] M. Koehler and J. Saak, "FlexiBLAS - a flexible BLAS library with runtime exchangeable backends," https://www.netlib.org/lapack/lawnspdf/lawn284.pdf, 2020-06-27, LAPACK Working Notes, Tech. Rep., 2013.

[3] F. G. Van Zee and R. A. Van de Geijn, "BLIS: A framework for rapidly instantiating BLAS functionality," *ACM Transactions on Mathematical Software*, vol. 41, no. 3, pp. 1–33, 2015. doi: 10.1145/2764454

[4] D. G. Spampinato and M. Püschel, "A basic linear algebra compiler," in *Proceedings of Annual IEEE/ACM International Symposium on Code Generation and Optimization*, ser. CGO '14. New York, NY, USA: Association for Computing Machinery, 2014. doi: 10.1145/2544137.2544155 p. 23–32.

[5] N. Kyrtatas and D. G. Spampinato, "A Basic Linear Algebra Compiler for Embedded Processors," *2015 Design, Automation Test in Europe Conference Exhibition (DATE)*, pp. 1054–1059, 2015. doi: 10.3929/ethz-a-010144458

[6] G. Frison, D. Kouzoupis, T. Sartor, A. Zanelli, and M. Diehl, "BLASFEO: Basic linear algebra subroutines for embedded optimization," *ACM Trans. Math. Softw.*, vol. 44, no. 4, pp. 42:1–42:30, Jul. 2018. doi: 10.1145/3210754

[7] G. Frison, T. Sartor, A. Zanelli, and M. Diehl, "The BLAS API of BLASFEO: Optimizing performance for small matrices," *ACM Transactions on Mathematical Software*, vol. 46, no. 2, May 2020. doi: 10.1145/3378671

[8] C. Fibich, S. Tauner, P. Rössler, M. Horauer, M. Krapfenbauer, M. Linauer, M. Matschnig, and H. Taucher, "Evaluation of open-source linear algebra libraries in embedded applications," in *2019 8th Mediterranean Conference on Embedded Computing (MECO)*, June 2019. doi: 10.1109/MECO.2019.8760041 pp. 1–6.

[9] M. Gautschi, P. D. Schiavone, A. Traber, I. Loi, A. Pullini, D. Rossi, E. Flamand, F. K. Gürkaynak, and L. Benini, "Near-threshold RISC-V core with DSP extensions for scalable IoT endpoint devices," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 10, pp. 2700–2713, Oct. 2017. doi: 10.1109/TVLSI.2017.2654506

[10] R. C. Whaley, A. Petitet, and J. J. Dongarra, "Automated empirical optimizations of software and the ATLAS project," *Parallel Computing*, vol. 27, no. 1, pp. 3–35, 2001. doi: 10.1016/S0167-8191(00)00087-9

[11] K. Goto and R. A. v. d. Geijn, "Anatomy of high-performance matrix multiplication," *ACM Transactions on Mathematical Software*, vol. 34, no. 3, pp. 12:1–12:25, May 2008. doi: 10.1145/1356052.1356053

[12] Altera Corporation, "cv_5v4: Cyclone V Hard Processor System Technical Reference Manual," https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/cyclone-v/cv_5v4.pdf, 2020-06-27, July 2018.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 673–681 DOI: 10.15439/2020F16 ISSN 2300-5963 ACSIS, Vol. 21

Requirement Elicitation Techniques for Software Projects in Ukrainian IT: An Exploratory Study

Denys Gobov
National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”,
37, Prosp. Peremohy, Kyiv, Ukraine
Email: d.gobov@kpi.ua

Inna Huchenko
National Aviation University,
1, Liubomyra Huzara ave., Kyiv, Ukraine
Email: inna.guchenko@hotmail.com

***Abstract*—Elicitation is a core business analysis/requirement engineering activity that provides inputs for another one: analysis, specification, confirmation, management. There is a significant number of specialized techniques that are used for requirement elicitation. The selection of the appropriate techniques considerably influences a project plan and success of a change as a whole. This paper is intended to analyse the industrial standards and experience of business analysts and requirement engineers in part of elicitation activities. We conducted a survey study involving 328 specialists from Ukrainian IT companies and a series of interviews with experts to interpret survey results. Furthermore, this paper provides the guideline in selecting a particular elicitation technique with respect to the type of project and situation.**

I. Introduction

BUSINESS analysis is the practice of providing opportunities for change in the context of an enterprise’s work by identifying needs and recommending solutions that bring value to stakeholders [1]. This discipline extends the scope of requirement engineering activities and area of their application [2], [3]. There are different views on the set of business analysis tasks depending on project methodology and solution type. Overall all business analysis tasks can be grouped into six knowledge areas: Business Analysis Planning and Monitoring, Elicitation and Collaboration, Requirements Life Cycle Management, Strategy Analysis, Requirements Analysis and Design Definition, and Solution Evaluation. If business analysis provides a basis for all future activities of development and testing, requirement elicitation provides a basis for future activities with requirements: specification and modelling, analysis, verification and validation, prioritization, maintaining, etc. So failure in this are lead to significant issues with project outcomes. According to [4] 39% of respondents recognized errors in the stage of requirements gathering one of the most influential factors that led to the failure of software development projects. Other industry studies reach the same conclusion [5], [6]. Elicitation is not an isolated activity. Information is gathering while performing any task that includes interaction with stakeholders and while the business analyst is analysing existent data. Elicitation may trigger additional elicitation for details to fill in gaps or increase understanding. Elicitation activities can be divided into three tasks: prepare for elicitation, conduct elicitation, and confirm elicitation results [7]. During preparing a business analyst should understand the scope of elicitation and select an appropriate set of elicitation techniques. Choosing the right techniques and ensuring each technique is performed correctly is extremely important to the success of the elicitation activity. The best practices and recommendation regarding elicitation techniques are defined in international standards [8], industrial bodies of knowledge [1], [9], [10], International empirical studies [11], [12].

There is a significant number of elicitation techniques that have proven themselves in practice and are recommended in the sources mentioned above. Each of them has its advantages and limitations, requires stakeholders’ involvement or study materials availability. As a part of business analysis approach and business analysis activities plan a business analyst needs to decide which techniques are best suited for a particular project. Usually multiple techniques are used for elicitation. Decision about the set of techniques depends on time and cost constraints, the types of business analysis information sources and their accessibility, the culture of the company, and the desired outcomes [1]. If elicitation is built based on the collaborative approach the needs of the stakeholders, their availability and location have to be taken into account.

This study was conducted to analyse the current preferences of business analysts and requirement engineers regarding elicitation selection approach for software development projects. We also wished to define attributes of project contexts which influence the probability of choosing a particular elicitation technique. We also wanted to take into account the specifics of distributed and collocated teams to determine the applicable such techniques, which is especially important in the context of the widespread use of outsourcing services and work with international companies. Our approach was to study the experience of practicing specialists from Ukrainian and international companies with branches in Ukraine involved in requirement gathering for IT projects. The main research method was a survey and statistical analysis.

The paper is structured as follows. Section II contains a review of the related works on elicitation activities and survey study regarding requirement engineering and business

This work was supported by IIBA Ukraine Chapter

analysis. In Section III we provide background information on requirements elicitation techniques, collected from industrial bodies of knowledge and study materials prepared by leading international organization in business analysis area. The section III is devoted to the structure of the questionnaire along with the selection of characteristics of the IT project. Also, it contains the survey results and dependencies identified based on the statistical analysis of received data. Section IV concludes the paper with discussion of the findings of our study and future work.

II. Related works

The most of the related works is focused on analysis of elicitation activities and elicitation techniques in particular. Dieste and Juristo [13] performed a systematic review on requirements elicitation techniques based on 26 empirical studies published till the year 2005. They aggregated the results in terms of five guidelines for RE practitioners. Wong at. all [14] perform systematic review on software requirement elicitation activities based on 35 articles and defined that most of the contributions were focused on the "Identify Requirements" activity (91%) and other activities are poorly covered: "Acquire knowledge" (17%),"Identify sources" (4%), "Defining technique" (9%), "Document" (9%) and "Refine requirements" (4%). Pacheco and Garcia [15] performed an systematic review on stakeholder identification during requirements elicitation based on 47 primary studies dated from 1984 to 2011. They found that identified approaches are not able to cover all aspects of stakeholder identification during requirements elicitation. In [16] authors noticed that there is need to replicate studies in different contexts wherein existing requirement engineers' interventions were evaluated and implemented in practice. It confirms that most of the case studies involve practitioners as participants, there is a need to work more closely with practitioners. Several studies assess effectiveness of elicitation techniques in the context of particular project. Hafsa at all. [17] performed systematic study on elicitation techniques in mobile application development project. Based on the analysis of 36 selected articles 22 requirement gathering methods and 8 different categories of requirement gathering challenges for mobile application were identified. In [18] systematic literature review was performed for elicitation techniques for Internet of things application requirement. The interview and prototyping were identified as a most elicitation techniques used between 2012 and 2018, data mining was mentioned as a technique that complements traditional elicitation techniques in order to mitigate the effects of insufficient requirements elicitation. In [19] authors defined several factors that can influence elicitation technique selection. This study selected five practitioners as informants from Yemen's companies and government agency. Dieste and Juristo [20] proposed a framework that can help requirements engineers to select the most adequate elicitation techniques. The set of attributes are relevant to the context of the elicitation process and influence the selection of one or other technique were discovered. Two groups of students were involved in experiment, practitioners did not took part in experiment. Author noticed that there results were not generalizable and should checked with other larger samples. Wong and Mauricio [21] defined a set of factors that influenced each activity of the requirements elicitation process and, consequently, the quality: learning capacity, negotiation capacity, permanent staff, perceived utility, confidence, stress, and semi-autonomous. An empirical study was carried out on 182 respondents from software development companies in Peru. The main limitations of the empirical studies mentioned above are limited number of participants and low practitioners' involvement. During last years a practice of dispersed team and outsourcing/outstuffing services model have become rather rule then exception, but influence of these factors the elicitation has not been analyzed. Survey preparation contained three steps:

- Practical guidelines and bodies of knowledge analysis to define a long list of elicitation techniques.
- It industry trend reports analysis to define an attributes that characterize the context of software projects.
- Preliminary interviews with five business analysts from Ukrainian IT companies to check a list of techniques and project characteristics.

The following sources were used for creating elicitation technique long list: "A Guide to the Business Analysis Body of Knowledge" (BABOK) from the International Insitute of Business Analysis (IIBA), "The PMI Guide to Business Analysis" from the Project Managememnt Institue (PMI), a study guide from the International Requirement Engineering Board (IREB) "Requirements engineering fundamentals" and book "Business Analysis" from British Computer Society (BCS). The analysis of the contents of these sources gives us a set of 17 requirements elicitation. In some cases the different names are used for the one technique, variants are included in the result tables per each sources if it is applicable. Short descriptions of these 17 techniques are given below.

Benchmarking and Market Analysis. Benchmark studies are conducted to compare organizational practices against the best-in-class practices. Market analysis involves researching customers in order to determine the products and services that they need or want, the factors that influence their decisions to purchase, and the competitors that exist in the market [1]. Brainstorming. Brainstorming is an elicitation technique that can be used to identify a list of ideas in a short period of time (e.g., a list of risks, stakeholders, or potential solution options). Brainstorming is conducted in a group environment and is led by a facilitator. A topic or issue is presented and the group is asked to generate as many ideas as possible about the topic [1], [9].

Analysis of business rules involves capturing business rules from sources, expressing them clearly, validating them with stakeholders, refining them to best align with business goals, and organizing them so they can be effectively managed and reused. Sources of business rules may be explicit (for example, documented business policies, regulations, or contracts) or tacit (for example, undocumented stakeholder know-how, gen-

TABLE I
REQUIREMENTS ELICITATIONS TECHNIQUES IN MAIN INDUSTRIAL GUIDELINES

Technique Name	**IIBA**	**PMI**	**BCS**	**IREB**
Benchmarking and Market Analysis	+	+		+(Analogy techniques)
Brainstorming	+	+	+ (Brainwriting, Round robin)	+ (Brainstorming paradox)
Business Rules Analysis	+	+(as a part of document analysis)		+ (as a part of document analysis)
Collaborative games	+(Product box, Affinity map, Fishbowl)	+(Product box, Speedboat, Spider web)	+ (Sticky (post-it) note exercises, Smaller 'break-out' or 'syndicate' groups)	+(Change of perspective)
Concept Modelling	+			
Data Mining	+			
Data Modelling	+			
Document Analysis	+	+	+	+ (System archaeology, Perspective-based reading)
Workshops & Focus Groups	+	+	+	+
Interface Analysis	+			
Interview	+	+	+	+
Mind Mapping	+		+	+
Observation	+	+	+ (Activity sampling, Special purpose records)	+(Field observation, Apprenticing)
Process Analysis/ Modelling	+	+	+ (Scenario)	+ (Modelling action sequences)
Prototyping	+	+		+
Survey or Questionnaire	+	+	+	+

erally accepted business practices, or norms of the corporate culture) [1].

Collaborative games are a collection of elicitation techniques that foster collaboration, innovation, and creativity to achieve the goal of the elicitation activity. Collaborative games use game play to encourage team participation and enhance engagement.The games are used to help the participants share their knowledge and experience on a given topic, identify hidden assumptions, and explore that knowledge in ways that may not occur during the course of normal interactions. The shared experience of the collaborative game encourages people with different perspectives on a topic to work together in order to better understand an issue and develop a shared model of the problem or of potential solutions. Many collaborative games can be used to understand the perspectives of various stakeholder groups [1], [9].

A concept model is used to organize the business vocabulary needed to consistently and thoroughly communicate the knowledge of a domain. Concept models put a premium on high-quality, designin dependent definitions that are free of data or implementation biases. Concept models also emphasize rich vocabulary. As an elicitation technique is used to identify key terms and ideas of importance and define the relationships between them [1].

Data mining is an analytic process that examines large amounts of data from different perspectives and summarizes the data in such a way that useful patterns and relationships are discovered. Data mining can be utilized in either supervised or unsupervised investigations. In a supervised investigation, users can pose a question and expect an answer that can drive their decision making. An unsupervised investigation is a pure pattern discovery exercise where patterns are allowed to emerge, and then considered for applicability to business decisions [1].

A data model describes the entities, classes or data objects relevant to a domain, the attributes that are used to describe them, and the relationships among them to provide a common set of semantics for analysis and implementation. Data modelling is used to understand entity relationships during elicitation. Most guidelines [9], [10], [22] do not recognize it as a core elicitation technique.

Document analysis is used to elicit business analysis information, including contextual understanding and requirements, by examining available materials that describe either the business environment or existing organizational assets. Document analysis may be used to gather background information in order to understand the context of a business need, or it may include researching existing solutions to validate how those solutions are currently implemented. Document analysis may also be used to validate findings from other elicitation efforts such as interviews and observations. [PMI, BABOK] Workshops use a structured meeting led by a skilled, neutral facilitator and a carefully selected group of stakeholders to collaborate and work toward a stated objective. Focus groups bring together prequalified stakeholders and subject matter experts (SMEs) to learn about their expectations and attitudes about a proposed solution. Focus groups provide an opportunity to obtain feedback directly from customers and/or end users [1], [9].

Interface analysis is used to identify where, what, why, when, how, and for whom information is exchanged between solution components or across solution boundaries. . Most solutions require one or more interfaces to exchange information with other solution components, organizational units, or business processes [1].

An interview is a formal or informal approach used to elicit information from stakeholders. It is performed by asking prepared and/or spontaneous questions and documenting the responses. Interviews are often conducted on an individual basis between an interviewer and an interviewee, but they may also involve multiple interviewers and/ or multiple interviewees. Questions that arise during the conversation can be discussed immediately, and the requirements engineer may uncover subconscious requirements through clever questions [9], [10].

Mind Mapping is used to generate many ideas from a group of stakeholders in a short period, and to organize and prioritize those ideas [1].

Observation is an elicitation technique that provides a direct way of obtaining information about how a process is performed or a product is used by viewing individuals in their own environment performing their jobs or tasks. This technique is helpful when domain specialists are unable to spend the time needed to share their expertise with the requirements engineer, or are unable to express and denote their knowledge [1], [10].

Process Analysis is used to understand current processes and to identify opportunities for improvement in those processes. Process Modelling is a standardized graphical model used to show how work is carried out and is a foundation for process analysis [1].

Prototyping is used to elicit and validate stakeholder needs through an iterative process that creates a model or design of requirements. It is also used to optimize user experience, to evaluate design options, and as a basis for development of the final business solution. This technique helps the business users to visualise the solution and hence increases understanding about the system requirements [1], [22].

Questionnaires and surveys are written sets of questions designed to quickly accumulate information from a large number of respondents. Survey respondents can represent a diverse population and are often dispersed over a wide geographical area. As a form of elicitation, this technique has the benefit of reaching a large group of people for a relatively small cost [9], [22].

III. Survey Study

A. *Questionnaire design*

The literature review has shown that many researches have been conducted for identifying common patterns and problems in IT business analysis and requirements elicitation in particular. However, after studying of the existing questionnaries developed for international surveys, we realized the necessity of adjusting them to Ukrainian IT companies' specifics. It was decided to take questions' basis from NaPIRE initiative [12] and rework it with respect to mentioned above sources such as [1], [8], [9], [10]. Survey items were carefully written using the business analysis vocabulary, mostly from BABOK. Types of questions used for the questionnaire are open-ended, closed-ended (multiple and single choice) and Likert scale.

Total number of questions is 43. After several rounds of internal peer reviews, the questionnaire was given for validation by business analysis experts from Ukrainian IT industry leaders. Among comments received as the first feedbacks, there were remarks about time needed for answering the questions (it took too long to complete the questionnaire) and complexity of some terms that might cause the clarity problems for young professionals. After recommended improvements were done, cognitive interviews were conducted with 10 potential respondents to determine how they interpret the terms, questions and answer options. After that step questionnaire was ready for distribution.

Our target group of respondents was IT professionals from Ukraine, mainly business analysts but also other roles involved in business analysis or requirements engineering activities. Overall number of survey participants is 328. English and Ukrainian languages were used for questionnaires. The questionnaire itself was created using Google forms and link to it was shared in the local Business Analysis communities, professional and social networks, and via personal contacts in TOP 10 Ukrainian IT companies. Answers were collected during one month. After that, data were merged and coded for further analysis. The following questions' categories were included into the questionnaire:

- Q1: General Information.
- Q2: Requirements Elicitation and Collaboration.
- Q3: Requirements Analysis and Design.
- Q4: Requirements Verification and Validation.
- Q5: Requirements Management.
- Q6: Attitude to the Business Analysis in the project.
- Q7: Problems, Causes and Effects.

In given article we focus on Elicitation and Collaboration topic in the context of general information questions about respondents' background.

Q1: General Information. Questions in this section were intended to give the context such as:

- Project size.
- Main industrial sector of the current project. Set of industrial sectors was taken from [12] and reworked with respect to domain areas within which services are offered by most of the Ukrainian IT Companies.
- Company type: IT or non-IT. For IT companies the separation was made among Outstaff, Outsource and
- Product companies.
- Company size.

- Class of systems or services such as business, embedded, scientific software etc.
- Team distribution (co-located or distributed).
- Role in the Project
- Experience in business analyst (BA)/requirements engineer (RE) role
- Certifications
- Way of working in the project (adaptive vs predictive)
- Project category for most of the participant's projects (e.g. greenfield engineering)
- BA/RE activities which respondent is usually involved in.

Q2: Requirements Elicitation and Collaboration. Within given questions category we were interested in elicitation sources, techniques and project role having primary responsibility for the solution requirements (functional, non-functional requirements) elicitation on the respondent's ongoing project.

The following types of elicitation sources were considered: collaborative (relies on stakeholders' expertise and judgements); experiments, e.g. observational studies, proofs of concept, and prototypes; research, i.e. information from materials or sources that are not directly known by stakeholder. 16 elicitation techniques were proposed as answer options with ability to select as many as needed for reflecting the full range used by respondents. Typical cases were taken as the base for the requirements elicitation responsibility topic and resulted in the following options: Business Analyst/Requirements Engineer, Product Owner/Business Analyst, Product Owner/Product Manager, Project Lead/Project Manager and Solution Architect. Also, we considered the case when in fact nobody has the primary responsibility.

B. Survey results. Participants Background

Figures 1-6 show the typical environment for the Ukrainian business analyst in terms of company, team, project role and type, experience etc. For the initial analysis the Pareto sorted histogram with cumulative curve was used.

The numbers in each figure allow to make the following observations:

- 41% of respondents are working in the project groups up to 15 members. Less than 13% are participating in projects with over 100 people (Fig. 1).
- 49% of the survey participants are employed in IT outsource companies while IT outstaff, product and inhouse development is represented in almost the same amount within left 51% of respondents (Fig. 2).
- About 80% of respondents are specialists with experience from 1 to 5 years, mode value is 1-3 years (Fig. 3).
- Predictive/rather predictive methodologies (e.g. RUP, Waterfall) are used in less than 15% of the projects (Fig. 4).
- Most of the participants have a Business Analyst role on the project, however, quite often this role is combined with a product ownership (Fig. 5).
- The TOP 3 popular industry sectors are Finance/Banking, e-Commerce/Retail and Healthcare/Pharmaceuticals. Variety of domains is represented in Fig. 6.

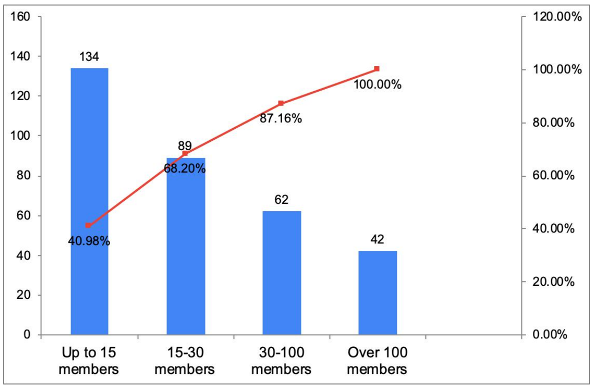


Fig. 1. Project size

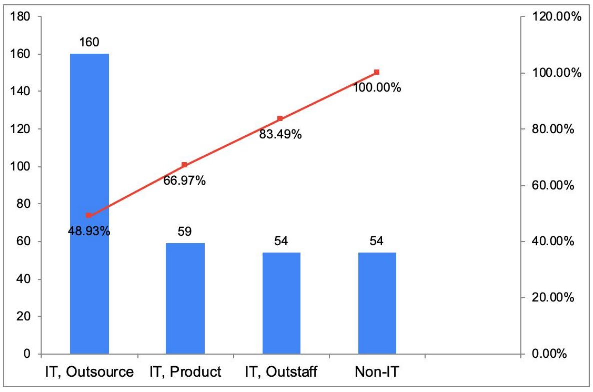


Fig. 2. Company Type

- Only 13% of respondents have certifications and 5% have more than 1 certificate.

Thus, the typical portrait of Ukrainian IT Business Analyst could be described using the observations above.

C. Survey results. Elicitation techniques usage

The most used elicitation techniques are shown in Fig. 7. Participants were allowed to select multiple techniques. Regardless the context in which the Ukrainian Business An-

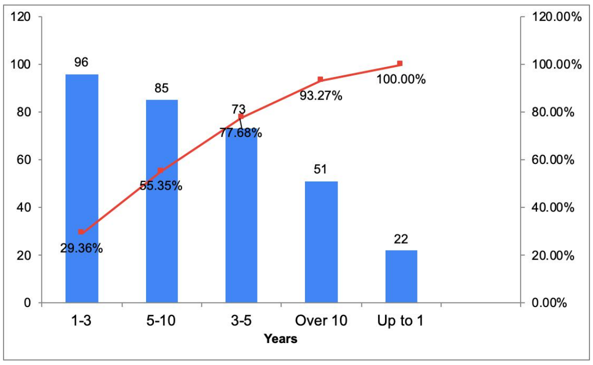


Fig. 3. Experience in BA/RE role

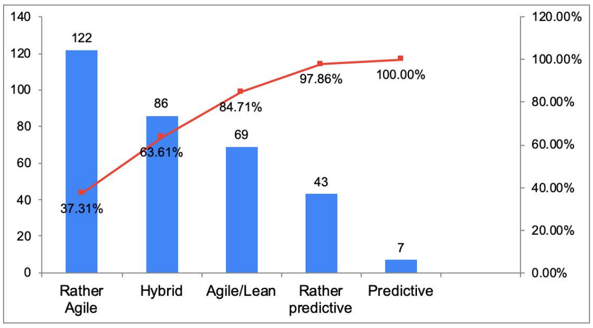


Fig. 4. Ways of working in the project

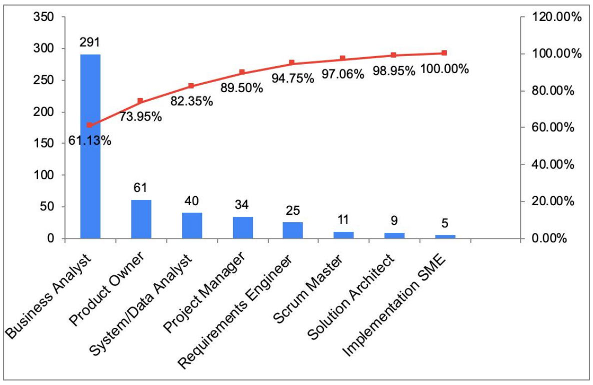


Fig. 5. Role in the project

alyst is working, we may see that the following elicitation techniques are the most popular (Fig. 7):

- Interview
- Document analysis
- Interface analysis
- Brainstorming
- Prototyping
- Process analysis/Process modelling

The rare techniques are Collaborative Games, Design Thinking and Data Mining.

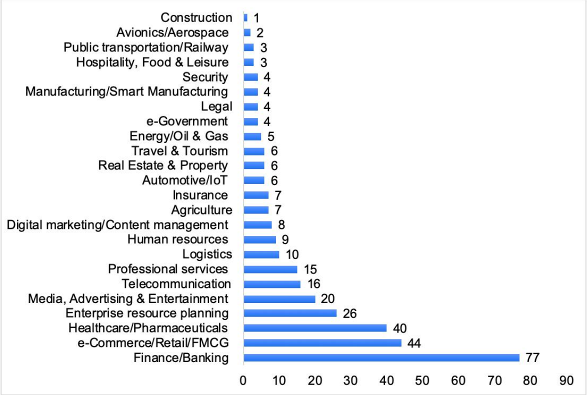


Fig. 6. Industrial sector

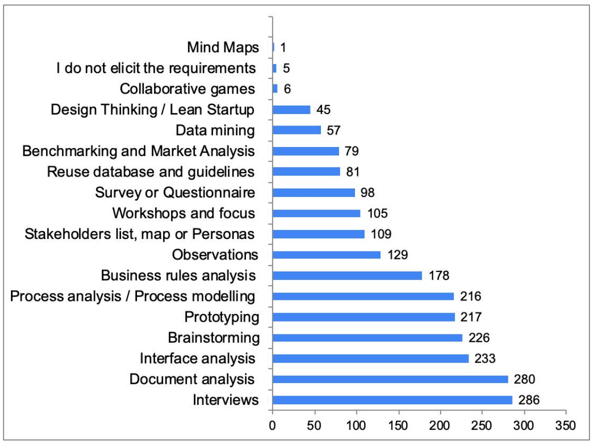


Fig. 7. Elicitation techniques popularity

Most of the respondents use such elicitation sources as collaborative (stakeholders' expertise) and research, 96% and 62% respectively. Only 49% of participants selected experiments as one of the options. The last fact could be explained by general complexity of observational studies and prototypes in the terms of efforts and time.

The elicitation responsibility is taken by Business Analyst/Requirements Engineer in 44% of cases, Business Analyst/Product Owner – 38%, Product Owner/Product Manager – 10%, left 8% are shared between Project Manager and Solution Architect roles.

During the questionnaire results analysis, the significant difference in the usage of several techniques from particular background perspective was noticed. It was decided to check each "background factor-elicitation technique" pair for association. The Chi-Square test of independence, commonly used for testing relationships between categorical variables, was applied to examine the differences within single dependent sample (population). Set of hypotheses about the association between particular factor and technique usage was developed. The example of null and alternative hypothesis is:

`H0`**:** There is **no** association between BA/RE experience and Workshops elicitation technique usage.

`H1`**:** There **is** an association between BA/RE experience and Workshops elicitation technique usage.

Chi-Square test has a number of assumptions critical for results reliability. These assumptions were checked and confirmed on the data preparation stage, namely:

- The data are randomly drawn from a population. This statement is confirmed by the method used for questionnaire distribution.
- The values in the cells are considered adequate when expected counts are not less than 5 and there are no cells with zero count [23], [24].
- The sample size is large enough. The minimum recommended size varies from 20 to 50 in different sources. This statement is also true as respondents' number for the study is 325 (filtered from initial 328).

- The variables under consideration must be mutually exclusive, i.e. no item shall be counted twice. For study purposes, data were transformed in such a way that for every Participant ID usage of the particular elicitation technique was set to “1” if technique was selected and “0” if wasn’t, i.e. the observations were classified into mutually exclusive classes.

After calculation of P-Value, which should be less than 0.05 considering 0,95 confidence level, the conclusion about statistical significance was made for the following factor-elicitation technique pairs:

- *Company Type – Documentation Analysis* (Fig. 8). Given elicitation technique is more frequently used in IT Outsourcing companies. The common problem for outsourcing is lack of the project team familiarity with Customer’s already existing systems and/or business. Thus, studying the existing documentation is the very first step for the BA/RE.
- *Company Type – Process Analysis/Process Modelling* (Fig. 9). As it could be seen from the bar chart, the technique is a must use in non-IT, inhouse development. Also, in IT outsource it is used wider than in outstaff or product development.
- *Methodology – Design Thinking/Lean Startup* (Fig. 10). As survey shows, design thinking is not a popular technique and is used mostly in agile development.
- *Experience – Workshops/Focus Groups* (Fig. 11). The more experience BA/RE has, the more often given technique is applied.
- *Experience – Interviews* (Fig. 12). Less usage of nterviews is observed for the young specialists with experience less than 1 year and for those over 10 years.
- *Experience – Prototyping* (Fig. 13). Respondents with experience over 3 years see the obvious benefit in prototyping and, thus, use this technique frequently.
- *Experience – Stakeholders list, map or Personas* (Fig. 14). The more experience, the less is the gap between use/no use for mentioned set of techniques.

P-Value for each pair having statistically significant relationship is stated under corresponding graphs in Fig. 8 – Fig. 14.

Also, the statistically significant relationships were identified for the Elicitation Responsibility background factor and the techniques below:

- Process Analysis/Process Modelling, p = 0,017
- Prototyping, p = 0,018
- Reuse database and guidelines, p = 0,039
- Design Thinking/Lean Startup, p = 0,049

Corresponding graphs are not included here due to space limitations.

First two techniques are used often if responsibility for the elicitation belongs to Business Analyst/Requirements Engineer and/or Product Owner. As for Design Thinking and Reuse database, the situation is quite opposite, – these techniques are rarely applied by roles mentioned above.

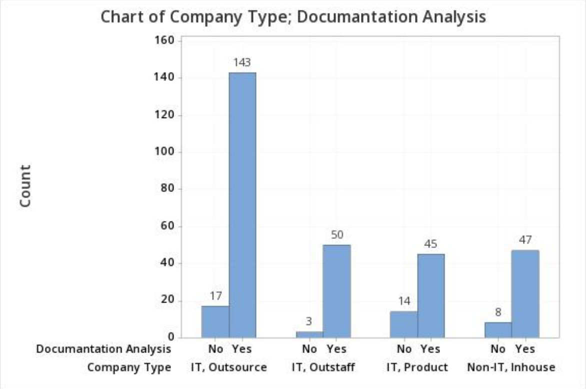


Fig. 8. Relationship between Company type and Documentation Analysis technique usage, **p = 0,028** (Chi-Square Test)

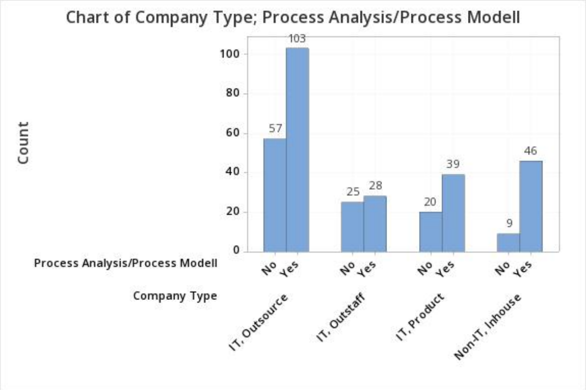


Fig. 9. Relationship between Company type and Process Analysis/ Process Modelling technique usage, **p = 0,005** (Chi-Square Test)

The hypotheses about relation between certificates and experience, company size and responsible for the requirements elicitation, team distribution and particular technique usage weren’t confirmed by Chi-Square Test results.

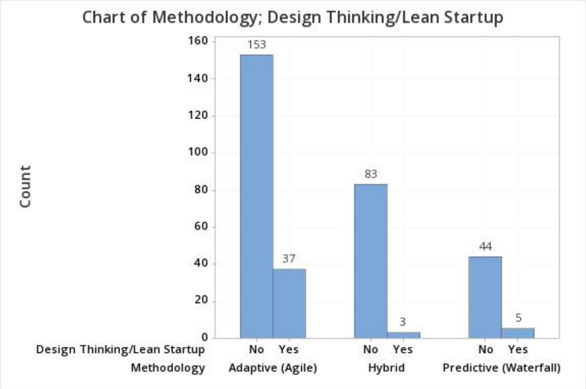


Fig. 10. Relationship between Methodology and Design Thinking/Lean Startup technique usage, **p = 0,001** (Chi-Square Test)

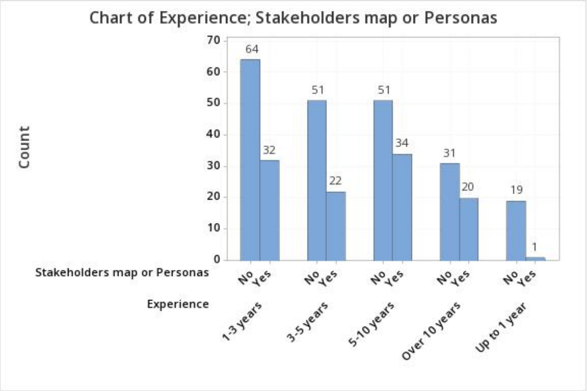


Fig. 14. Relationship between Experience and Stakeholders map or Personas, **p = 0,014** (Chi-Square Test)

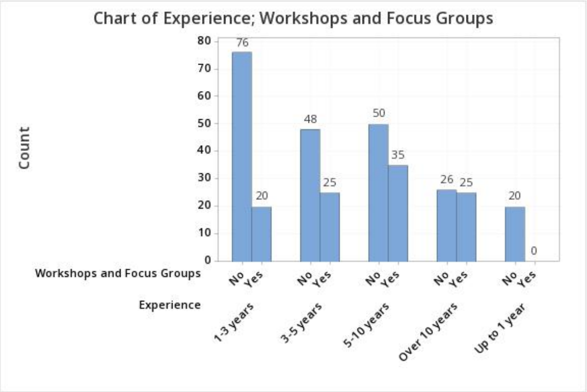


Fig. 11. Relationship between Experience and Workshops/Focus Groups technique usage, **p = 0,001** (Chi-Square Test)

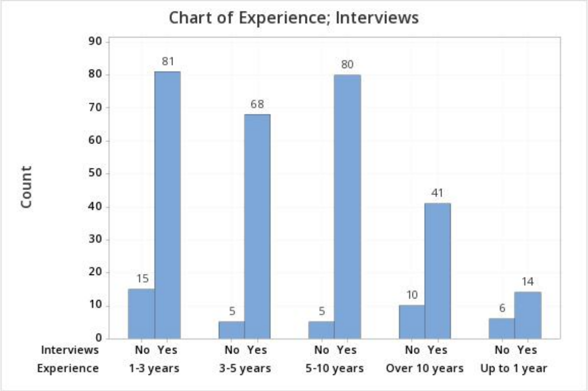


Fig. 12. Relationship between Experience and Interviews technique usage, **p = 0,008** (Chi-Square Test)

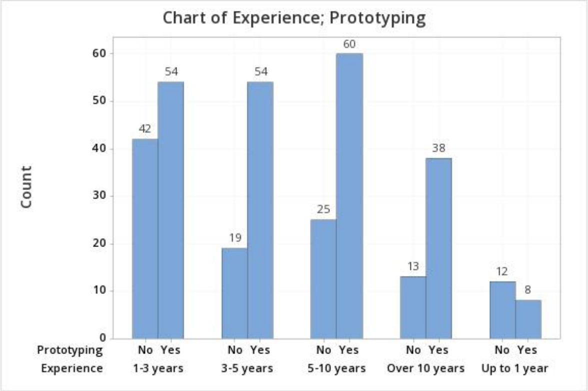


Fig. 13. Relationship between Experience and Prototyping technique usage, **p = 0,007** (Chi-Square Test)

IV. Conclusion

A survey study dedicated to the analysis of current state and selection of requirements elicitation techniques in different software project contexts has been conducted. The survey structure was built based on the worldwide known industrial standards. Attributes of project context were established to analyze influence the requirement elicitation techniques. The survey was conducted among practitioners from the Ukrainian IT and non-IT companies, 328 specialists (mainly business analysts and product owners) took part in the survey. The seven most used elicitation techniques were defined: Interview (used by 89% of respondents), Document analysis (87%), Interface analysis (72%), Brainstorming (70%), Prototyping(67%), Process analysis/Process modelling (67%). This result can be used as guidance or practical advice for selection a core set of elicitation techniques. The Chi-Square Test (Cross Tabulation) was applied to examine the relationships between project context and requirement elicitation technique usage. The statistically significant relationship were identified for the following project context attributes and elicitation techniques: Company type – Document analysis, Process Analysis/Process Modelling; Elicitation responsibility - Process Analysis/Process Modelling, Prototyping, Reuse database and guidelines, Design Thinking/Lean Startup; Methodology - Design Thinking/Lean Startup; Experience of business analyst - Workshops/Focus Groups, Interviews, Prototyping. The hypotheses about relation between certificates and experience, company size and responsible for the requirements elicitation, team distribution and particular technique usage were not confirmed by Chi-Square Test results. These dependencies can be used as guidance for selection supportive techniques or adjusting set of core elicitation techniques. Our study had several limitations. The list of techniques included in the survey is not exhaustive. Elicitation techniques may be applied alternatively or in conjunction with other techniques. Due to specific of project context business analysts are encouraged to modify techniques or engineer new ones. The survey result gathering via google survey engine and was intended to be anonymous (requiring personal data is problematic on legal and ethical grounds), therefore we cannot prove that respondents provided true information about project context and used elicitation techniques. Taking into account that the survey was limited to one country only, its results cannot extrapolated for worldwide software industry (even though IT industry in Ukraine is integrated in international environments, especially outsourcing and outstaffing companies, whose employees were the majority of respondents (65%). Several directions for future research can be considered. Other business analysis' tasks can be analyzed to define dependencies and recommendation regarding selection techniques for requirement specification and modelling, validation and verification. The factor analysis can be used to identify and assess variability among observed, correlated variables (project context attributes, and business analysis techniques).

References

[1] International Institute of Business Analysis, “A guide to the business analysis body of knowledge (BABOK Guide)” ver. 3, *IIBA,* 2015.

[2] J. Rubenss, “Business analysis and requirements engineering: the same, only different?” *Requirements Engineering* vol. 12(2), 2007, pp. 121–123, dx.doi.org/10.1007/s00766-007-0043-3

[3] M. Aoyama, “Bridging the requirements engineering and business analysis toward a unified knowledge framework” *in International Conference on Conceptual Modeling ,* Springer, Cham, 2016, pp. 149–160, dx.doi.org/10.1007/978-3-030-05719-0

[4] The Standish Group, “CHAOS Report,” The Standish Group, 2014.

[5] O. Sanchez, M. Terlizzi, “Cost and time project management success factors for information systems develop-ment projects, *International Journal of Project Management* vol. 35(8), 2017, pp. 1608-1626

[6] R. Nelson, “IT project management: Infamous failures, classic mistakes, and best practices, *MIS Quarterly executive* vol. 6(2), 2007

[7] International Institute of Business Analysis, “A Core Standard A Companion to A Guide to the Business Analysis Body of Knowledge (BABOK® Guide)" ver. 3, *IIBA,* 2017.

[8] ISO/IEC/IEEE, “Systems and software engineering - Life cycle processes - Requirements engineering", ISO/IEC/IEE, Standard 29148-2011, 2011, dx.doi.org/10.1109/ieeestd.2011.6146379

[9] Project Management Institute, “The PMI Guide to BUSINESS ANALYSIS", *PMI,*Newtown Square, Pennsylvania, 2017.

[10] K. Pohl, “Requirements engineering: fundamentals, principles, and techniques", Springer Publishing Company, 2010.

[11] S. Wagner, et al., “Status quo in requirements engineering: A theory and a global family of surveys", *ACM Transactions on Software Engineering and Methodology (TOSEM) ,* vol. 28(2), 2019, pp. 1–48.

[12] D. Fernandez, S. Wagner, “Naming the pain in requirements engineering: A design for a global family of surveys and first results from Germany", *Information and Software Technology,* vol. 57, 2015, pp. 616–643.

[13] O. Dieste , N. Juristo, “Systematic review and aggregation of empirical studies on elicitation techniques", *IEEE Transactions on Software Engineering ,* vol. 37(2), 2011, pp. 283–304.

[14] L. Wong, et al., “A systematic literature review about software requirements elicitation", *J Eng Sci Technol,* vol. 12(2), 2017, pp. 296–317.

[15] C. Pacheco, I. Garcia, “A systematic literature review of stakeholder identification methods in requirements elicitation", *J Syst Softw,* vol. 85(9), 2012, pp. 2171–2181.

[16] T. Ambreen, et al. , “Empirical research in requirements engineering: trends and opportunities", *Requirements Engineering,* vol. 23(1), 2018, pp. 63–95.

[17] H. Dar, et al. , “A systematic study on software requirements elicitation techniques and its challenges in mobile application development", *IEEE Access,* vol. 6, 2018, pp. 63859–63867.

[18] T. Lym, et al., “Elicitation Techniques for Internet of Things Applications Requirements: A Systematic Review", *Proceedings of the 2018 VII International Conference on Network, Communication and Computing,* 2018, pp. 182-188.

[19] F. Anwar, R. Razali, “A practical guide to requirements elicitation techniques selection-an empirical study", *Middle-East Journal of Scientific Research,* vol.11(8), 2011, pp. 1059-1067.

[20] D. Carrizo, et al., “Systematizing requirements elicitation technique selection", *Information and Software Technology,* vol.56(6), 2014, pp. 644-669.

[21] L. Wong, D. Mauricio, “New Factors That Affect the Activities of the Requirements Elicitation Process", *Journal of Engineering Science and Technology,* vol.13(7), 2018, pp. 1992-2015.

[22] D. Paul, et al., “Business analysis", *BCS, The Chartered Institute for IT,* 2014.

[23] F. Yates, et al., “The Practice of Statistics", *New York: W.H.Freeman,* 1st ed., 1999.

[24] F. Yates, “Contingency table involving small numbers and the Chi-squared test", *Suppl J R Stat Soc*, 1:217-35, 1934.

DOI: 10.15439/2020F174
ISSN 2300-5963 ACSIS, Vol. 21

Testbed for thermal and performance analysis in MPSoC systems

Michal Sojka*, Ondřej Benedikt*†, Zdeněk Hanzálek*
Czech Technical University in Prague, Czech Republic
*Czech Institute of Informatics, Robotics and Cybernetics
†Faculty of Electrical Engineering
Email: michal.sojka@cvut.cz

Pavel Zaykov
Honeywell International s.r.o.,
Advanced Technology Europe, Brno, Czech Republic
Email: pavel.zaykov@honeywell.com

***Abstract*—Many modern computing platforms in the safety-critical domains are based on heterogeneous Multiprocessor System-on-Chip (MPSoC). Such computing platforms are expected to guarantee high-performance within a strict thermal envelope. This paper introduces a testbed for thermal and performance analysis. The testbed allows the users to develop advanced scheduling and resource allocation techniques aiming at finding an optimal trade-off between the peak temperature and the achieved performance. This paper presents a new, open-source Thermobench tool for data collection and analysis of user-defined workloads. Furthermore, a methodology for shortening the time needed for the data collection is proposed. Experiments show that a significant amount of time can be saved. Specifically, time reduction from 60 minutes to 15 minutes is achieved with the i.MX8 MPSoC from NXP while running a set of user-defined benchmarks that stress CPU, GPU, and different levels of the memory hierarchy.**

I. Introduction

HIGH-PERFORMANCE computing platforms are composed of heterogeneous Multi-Processor System-on-Chip (MPSoC). The heterogeneity in the MPSoC is the key for delivering high-performance as each hardware (HW) component has its strengths for specific user workloads. Exemplary heterogeneous computing resources in an MPSoC are various types of Central Processing Units (CPUs), Graphical Processing Units (GPUs), and Field-Programmable Gate Arrays (FPGAs).

In recent years, safety-critical domains such as automotive and aerospace have experienced a significant increase in the Software (SW) complexity and functionality that led to the gradual adoption of the heterogeneous MPSoCs. Examples of successfully deployed heterogeneous MPSoCs are infotainment systems and autonomous driving computers in the recent car generations.

Apart from guaranteeing the high-performance, the safety-critical systems shall also operate under harsh environmental conditions such as dust, vibrations, and extended thermal ranges. In the context of safety and reliability, it is vital to preserve the MPSoC thermal envelope. Thus, it is necessary to keep the peak temperature under a predefined threshold. One of the most popular methods for thermal management is the active cooling that is commonly implemented by forcing airflow by CPU fans. The active cooling significantly complicates the mechanical design. In some cases, it might not be available as electronics are so closely placed that only a limited airflow is available. An alternative to the active cooling is the passive cooling, which is commonly implemented by heat-sinks. The passive cooling is less efficient than the active cooling and requires additional space and adds additional weight. Therefore, the safety-critical domains are interested in complementary techniques to reduce the peak temperatures in MPSoCs chips.

This paper paves the road towards the development of efficient peak-temperature reduction techniques based on scheduling and resource allocation. We introduce a testbed for thermal and performance analysis of various user-defined workloads executed on a selected MPSoC platform, NXP i.MX8QuadMax [1]. We focus our effort on building the testbed using a real hardware platform rather than on working with simulators. As the thermal behavior of the real platform is rich and influenced by a huge amount of factors like computer architecture, physical chip and board layout, and ambient environment, we propose tools and methods that shall be applicable to a wide-range of real-world conditions and hardware platforms.

More specifically, in this paper, we look for a reproducible way to measure platform temperatures while running various user-defined workloads. The goal is to eliminate various random temperature-influencing factors as much as possible without using expensive special-purpose equipment such as thermal chambers. A part of our study is an investigation of minimal experiment length that achieves both reproducible results and good precision. To that end, we derive a thermal model and try to use its knowledge to shorten the experiments.

The contributions of this paper are as follows:

1) We introduce Thermobench – a new open-source tool that helps with benchmarking, collection of statistical (performance and thermal) data, and their analysis. It also contains multiple ready-to-use user-defined workloads.
2) We propose a method to shorten the length of the measurements needed to predict the steady chip temperature (from 60 to 15 minutes), and we evaluate the precision of this method.
3) We present selected results measured on our testbed with the proposed tooling, showing the relation of thermal and performance properties.

IEEE Catalog Number: CFP2085N-ART ©2020, PTI

The remainder of the paper is organized as follows. In Section II, we analyze the works most related to ours. Section III introduces the components in our hardware setup and Section IV outlines the functionality of the Thermobench tool. In Section V, we provide details on the data analytics and outline the conducted experiments for the model fitting. Experimental results from the user-defined workloads are presented in Section VI. The paper concludes with Section VII.

II. Related work

Many researchers analyze the thermal properties of computer systems. A common approach is to create a model and examine the thermal behavior using system simulation [2]–[4]. The disadvantage of such an approach is that the simulation precision highly depends on the input parameters. Examples of input parameters are the floor plans of the chip and details about the computer architecture [2]. However, such input parameters are rarely available for the modern MPSoC chips.

An alternative approach, pursued in this paper, is an experiment-based analysis performed on real hardware. The experimental approaches can be divided into two groups based on the physical quantity being measured: i) electrical power/ energy and ii) temperature. These two quantities are related and measuring each one of them has its own advantages and disadvantages. For example, authors of [5]–[9] analyze the power consumption of various workloads aiming at decreasing it. The advantage of measuring the power consumption rather than the temperature is the instantaneous response. Nevertheless, there are also some disadvantages, especially when temperature is the primary quantity of interest:

- in case the power measurement circuitry is not integrated into the system, an invasive modification have to be made to the board [10], which is not always possible. An alternative is to measure the total input power, but this way, it is not possible to distinguish between the power consumers (e.g., MPSoC, displays and communication interfaces).
- power consumption has fluctuations. Thus, the peaks in the power consumption have to be captured with high sampling rate [6]. Execution of such high-frequency sampling on the target system increases the power consumption and execution on an external system makes it harder to correlate measurements with activities on the target system.

In contrast to the availability of power measurements, almost all modern MPSoCs have on-chip temperature sensors that are accessible without the need of any special hardware. Also, thermal measurements allow measuring interesting effects not visible when only the power is measured [11].

The thermal properties obtained as a result of this work can be used to optimize scheduling or resource allocation of the workload. This topic is addressed by other authors [12]–[15], but in most cases their work is not applicable to our platform and safety-critical requirements, either because they consider single-core platforms or because their scheduling model is not compatible with our target domain – avionics.

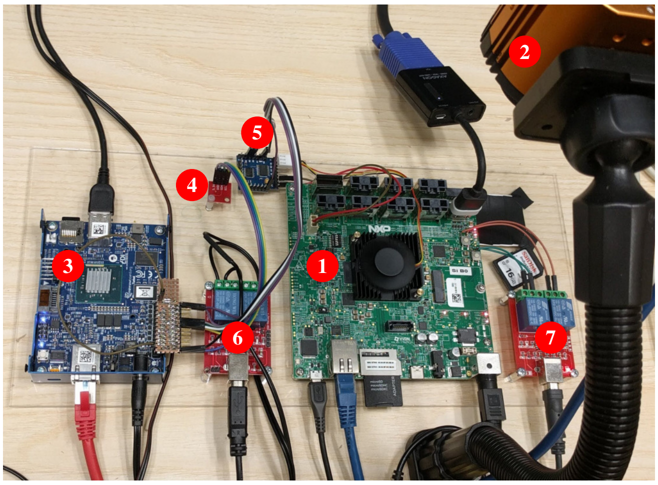


Fig. 1. Testbed for thermal measurements.

The Thermobench tool presented in this work already includes several ready-to-be-used benchmarks. Still, it is possible to use our tool with other benchmarks, such as Rodinia benchmark suite [16].

III. Testbed hardware setup

This section describes the hardware setup in the testbed. Figure 1 depicts the designed testbed where each label corresponds to one of the following hardware components:

1) i.MX8 Multi-sensory Enablement Kit (MEK) [1];
2) Workswell thermal camera WIC 336 [17];
3) MinnowBoard Turbot (x86 architecture) [18];
4) HTU21D ambient temperature sensor [19];
5) WeMos D1 mini TB6612FNG fan motor controller [20];
6) USB-controlled relay connected to the MinnowBoard;
7) USB-controlled relay connected to the i.MX8 board reset and power buttons.

The target device for our thermal and performance measurements is the i.MX8 MEK board by NXP [1] (hereafter referred to as i.MX8 board). We choose this particular MPSoC because it is the latest generation of the i.MX family, which has successfully demonstrated itself as a prominent computing platform for a wide range of applications, including the onboard infotainment systems in the automotive domain.

The MPSoC in the i.MX8 board is equipped with two CPU clusters. The first one has four ARM Cortex-A53 cores, while the second has two ARM Cortex-A72 cores. The MPSoC also contains two Vivante GC7000 GPUs.

For the convenience of the testbed users and to make the validation easier with various software stacks and OS kernels, we choose to boot up the i.MX8 board over the network rather than from the SD card. The network booting process relies on the features provided by the U-Boot bootloader. The Linux kernel is loaded via TFTP protocol, and the root file system

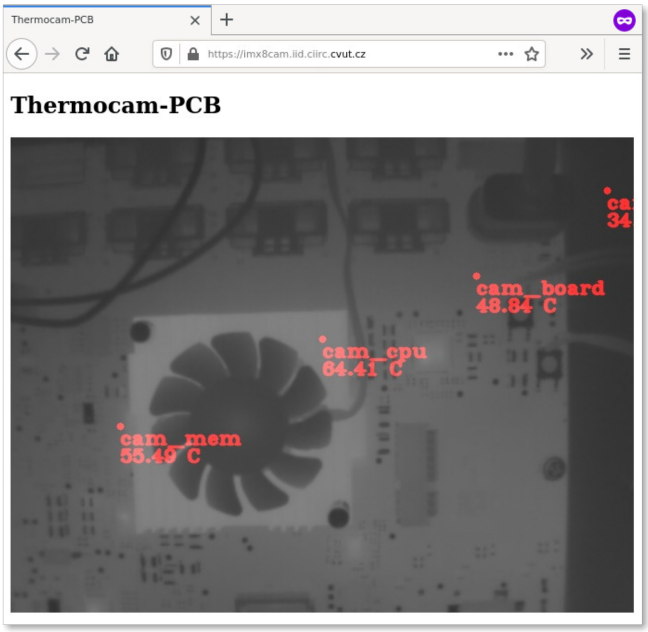


Fig. 2. *Thermocam* application – Web interface.

is mounted via the NFS protocol. The network boot process is automated with the help of the novaboot[1] tool.

We employ an external Pulse Width Modulation (PWM) motor controller to command the Revolutions Per Minute (RPM) for the on-board CPU fan as the i.MX8 board does not allow us to command the fan speed directly. The CPU fan speed is controlled from the software in the i.MX8 board by remotely running (via an SSH session) a command on the Turbot board. Then, the Turbot board controls the WeMos minifan controller to which the CPU fan is attached.

Many of the experiments are executed for prolonged periods of time. During this time, the ambient temperature changes, which may negatively impact the thermal benchmarks' precision. The temperature deviations caused by the fluctuations in the ambient temperature has to be compensated for. A way to achieve such compensation is to record the ambient temperature and consider its values in the latter analysis.

The ambient temperature sensor is attached to the Turbot development board. The software on the i.MX8 board may read the ambient temperature sensor by remotely running a command on the Turbot board. The command records the ambient temperature every 10 seconds.

The thermal behavior of the i.MX8 board is monitored by an external thermal camera. In our testbed, the thermal camera is attached to the Turbot board and controlled by a custom application referred to as *thermocam*. The *thermocam* application processes the images from the camera and makes them available over a web interface (see Figure 2).

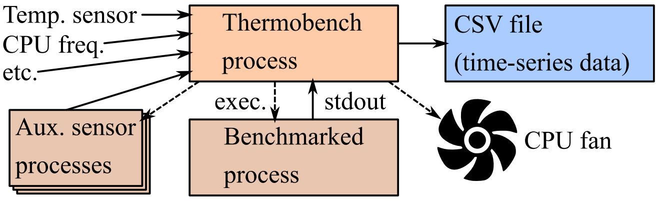


Fig. 3. Data flows in the Thermobench tool

Finally, the MinnowBoard Turbot board serves as a "gateway" for the i.MX8 board in the testbed. The MinnowBoard services provide access to the CPU fan, the ambient temperature sensor, the relays, and the thermal camera.

IV. Thermobench Tool

The Thermobench tool is an open-source software hosted on GitHub[2]. The Thermobench tool is developed to configure the testbed and capture the execution profiles of user-defined workloads as depicted in Fig. 3. Its three main components are designed to be portable and are described as follows:

- A C++ application that captures the execution profile of user-defined workloads and stores them in a file. Examples of execution profiles might be thermal and performance measurements.
- A Julia[3] package that has data analytics capabilities (refer to Section V for further details), and allows the user to generate various graphs.
- User-defined workloads that expose the thermal and performance profiles of the MPSoC under test. Currently, we provide the following examples of user-defined workloads: CPU micro-benchmarks, CPU memory subsystem benchmarks, and GPU benchmarks. Further details and results are provided in Section VI.

Other notable features of the Thermobench tool are:

- inserting a cool-down time between the execution of two consecutive user-defined workloads,
- controlling the fan speed, and
- the possibility to specify the collected statistics via an external *sensor file*. This feature makes it easier to collect data from all relevant sensors available on a given board.

In a nutshell, the Thermobench tool runs a user-defined workload and periodically collects its execution profile (most importantly measured temperatures), which are then stored to a file for later processing. Examples of recorded statistics are:

- Timestamps,
- Temperatures from Linux thermal-zone sensors,
- CPU clock frequencies,
- CPU load,
- Standard output of the benchmarked program,
- Output (or just selected values) from specific commands (e.g., reading temperatures from the ambient temperature sensor and from the thermal camera).

[1] https://github.com/wentasah/novaboot

[2] https://github.com/CTU-IIG/thermobench

[3] https://julialang.org/

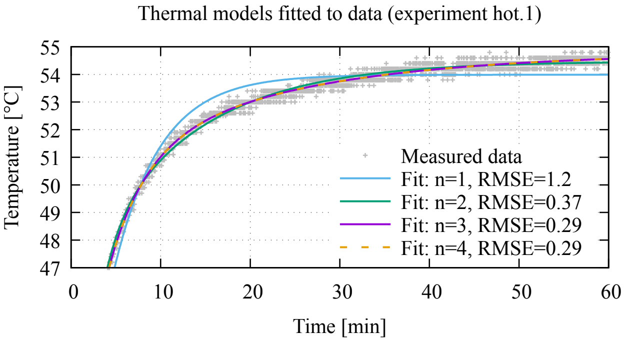


Fig. 4. Measured data and fitted models (data < 47 °C not visible). Exact equations of the fitted models are shown below as (3)–(6).

By default, the Thermobench tool records statistics once per second. With this setting, we have experimentally verified that the Thermobench tool's impact on the system's thermal and performance profile is negligible.

V. Data analytics

The data analytics in the Thermobench tool allows to compare the thermal and performance profiles of various user-defined workloads in the testbed. The thermal profile is based on the *heat flow* that is produced during the execution of user-defined workloads on the MPSoC. The heat flow is the amount of heat energy passed out of the MPSoC. The heat flow is denoted by $\dot{Q}$ and measured in Watts. Unfortunately, the heat flow cannot be directly measured. Therefore, we do estimate it by alternative means – namely using the temperature of the chip. In the rest of the Section, we describe how to estimate the heat flow and evaluate the estimation's precision.

It is well known [21] that the heat flow $\dot{Q}$ produced by a chip is proportional to the *relative temperature* ΔT_{ss}:

$$\dot{Q} \propto \Delta T_{\text{ss}} = T_{\text{ss}} - T_{\text{amb}}, \tag{1}$$

where T_{ss} is the steady-state chip temperature, and T_{amb} is the ambient temperature. Therefore, to compare the heat flows of various user-defined workloads in an MPSoC, it is sufficient to compare their ΔT_{ss}. However, waiting for the chip temperature to stabilize may require a prolonged period of time, while the user may expect quick and precise ΔT_{ss} estimates.

A naïve approach to estimate the ΔT_{ss} is to average multiple adjacent temperature samples. Unfortunately, the temperature readings are noisy and may significantly compromise the ΔT_{ss} precision. A more robust approach is proposed in the sections to follow, where we also discuss the influence of the ambient temperature on the ΔT_{ss} estimates. Finally, we conclude the section by proposing a way to shorten the time necessary for the data capturing of the temperature readings.

A. *Thermal model fitting*

Figure 4 visualizes the temperature measurements collected with the Thermobench tool with sampling period of 1 second for a 60-minutes experiment running arithmetic CPU computations. The temperature measurements are captured with a switched-off CPU fan as required by our target applications. To overcome the noise in the temperature measurements, the T_{ss} is estimated by fitting the thermal model to the measured data and by computing the T_{ss} as T_∞ from the thermal model described by (2). We use least-squares fitting implemented by Levenberg-Marquardt algorithm. In this paper, the applied *thermal model* [21] follows the evolution of the chip temperature as a function of time:

$$T_n(t) = T_\infty + \sum_{i=1}^{n} k_i e^{-\frac{t}{\tau_i}}, \tag{2}$$

where n is the order of the model and τ_i are the *time constants* of the model. The time constants specify "how fast" the temperature reacts to changes of the heat flow. We selected such a thermal model because it has the same result as a solution of a set of linear differential equations that is typically used in the modeling of thermal systems.

By fitting the thermal model (2) to the temperature data measured with the Thermobench tool, we manually select n and discover the constants T_∞, k_i, and τ_i.

In Figure 4, we demonstrate how the thermal models of different orders (n) may fit the data. The first and the second-order models do not fit well – see their root-mean-square errors (RMSE). The third and the fourth-order models fit better. The difference between them is negligible, thus we conclude that the 3rd order model is sufficient. Numerically, the models for different orders are as follows:

$$T_1(t) = 54.0 - 17.9e^{\frac{-t}{5.2}} \tag{3}$$

$$T_2(t) = 54.5 - 13.6e^{\frac{-t}{1.9}} - 8.4e^{\frac{-t}{11.6}} \tag{4}$$

$$T_3(t) = 54.8 - 7.3e^{\frac{-t}{0.9}} - 11.2e^{\frac{-t}{4.1}} - 4.6e^{\frac{-t}{20.1}} \tag{5}$$

$$T_4(t) = 54.8 - 0.3e^{\frac{-t}{0.02}} - 7.3e^{\frac{-t}{0.9}} - 11.2e^{\frac{-t}{4.1}} - 4.6e^{\frac{-t}{20.3}}. \tag{6}$$

We observe in (5) that for $n = 3$, the $T_\infty = 54.795 \pm 0.075$ °C (95% confidence intervals are the output of the fitting algorithm). Depending on the thermal model order, the estimated T_∞ differ by approximately one degree. For the third-order thermal model, the time constants are $\tau_1 = 0.91 \pm 0.08$, $\tau_2 = 4.1 \pm 0.3$ and $\tau_3 = 20.1 \pm 2$ minutes. The longest time constant τ_3 is particularly important, because it determines the experimental time for the temperature to reach a steady-state – the exponential term reaches 95% of its contribution k_i in $3 \cdot \tau_i$. In case of τ_3, the experimental time is ≈ 60 minutes. In Section V-C below, we examine how this time can be reduced.

B. *Suppression of ambient temperature changes*

The ambient temperature influences the on-chip temperature. The experiments may run for prolonged periods of time, and the ambient temperature may easily vary among the experiments or even during a single experiment. Such variations often result in non-reproducible temperature readings.

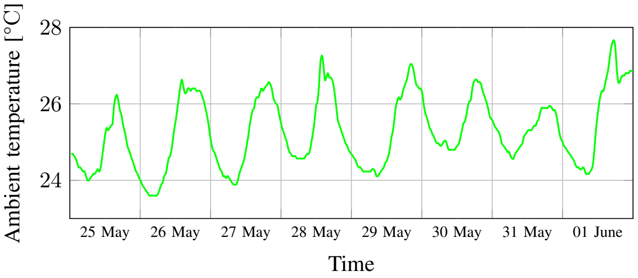


Fig. 5. Evolution of the ambient temperature over a period of one week.

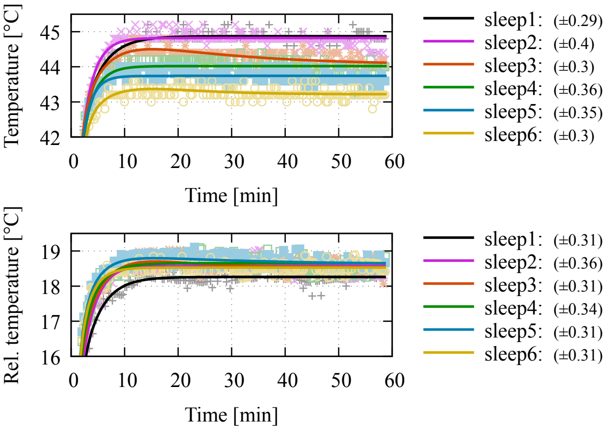


Fig. 6. Comparison of thermal model fitting with (bottom) and without (top) ambient temperature compensation on a series of identical experiments. Fit error (RMSE) is given after ± sign in parentheses.

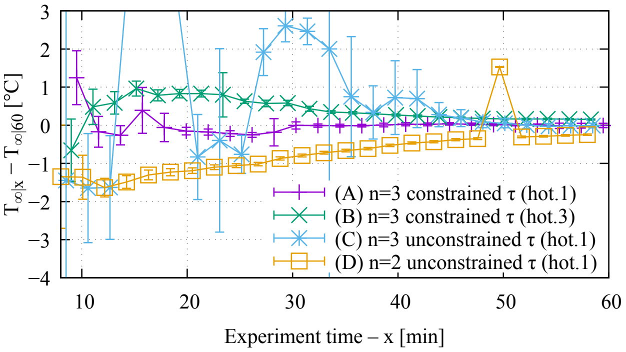


Fig. 7. Relation between the length of the experiment and estimation of T_∞. Vertical axis shows the difference between T_∞ estimated from data of length x ($T_{\infty|x}$) and from full 60 minutes of data ($T_{\infty|60}$). Error bars represent 95% confidence intervals.

TABLE I
PARAMETERS OF THE 3RD ORDER THERMAL MODEL WITH UNCONSTRAINED τ FOR EXPERIMENTS HOT.1 AND HOT.3.

	T_∞	k_1	τ_1	k_2	τ_2	k_3	τ_3
	[°C]	[°C]	[min]	[°C]	[min]	[°C]	[min]
hot.1	54.8	−7.3	0.9	−11.2	4.1	−4.6	20.1
hot.3	54.9	−3.4	0.6	−8.3	1.6	−7.8	7.1

Figure 5 visualizes the evolution of the ambient temperature as measured by our testbed over a period of one week.

To suppress the effect of ambient temperature changes, it might be best to have a model that counts the impact of the ambient temperature on the on-chip temperature readings and can produce delays of the heat propagation from the ambient environment to the chip. Such a model is referred to as a *transfer function* and can be estimated by system identification methods based on models such as OE, ARX, and ARMAX[4]. It is a well-known fact that for these methods to produce good results, it is necessary to have a high variation on the system inputs. Small changes in the ambient temperature over a long period of time (as in Figure 5), together with relatively high measurement noise, rendered these methods to be ineffective.

Due to the lack of a better model, we do compensate for the ambient temperature changes by simply subtracting the actual ambient temperature $T_{\text{amb}}(t)$ from the other measured temperatures. As a result, the T_∞ in model (2) represents the estimate of ΔT_{ss}. Figure 6 visualizes the results of the fitting thermal models with and without the ambient temperature compensation. The graphs show data from an experiment called "sleep", which was repeated six times. It can be seen that the standard deviation of T_∞ estimates is ≈ 0.6 °C without the compensation and ≈ 0.1 °C with it.

The proposed compensation for the ambient temperature changes also helps with the model fitting. The mean value of the fit errors (RMSE) from the examined experiments is 4% lower after the compensation, and the maximum fit error is even 12% lower.

C. Reduction of the experimental time

As we have demonstrated, the user-defined workloads have to run for at least 60 minutes, which turns to be impractical and time-consuming. In this section, we investigate the possibility of fitting the thermal model from a shorter experimental data set and estimating the T_∞ afterward. The difference between the estimates from short and full 60 minute experiments can be seen in Figure 7.

We compare three ways for the thermal model fitting:

- 3rd order model with known time constants, i.e., τ_i are constrained to $\pm 1\%$ of the values estimated from 60 minutes of data.
- 3rd order model with unconstrained time constants, i.e., time constants are fully estimated from the shorter data,
- 2nd order model with unconstrained time constants.

In Figure 7, the 3rd order model and the constrained τ is depicted by curve (A). Curve (A) suggests that the thermal model is able to predict the chip temperature with unsatisfactory precision ($> \pm 0.5$°C) for an experiment executed for less than 20 minutes. Curve (B) suggests that the same method applied to the same user-defined workload but executed 2

[4] https://www.mathworks.com/help/ident/ug/what-are-polynomial-models.html

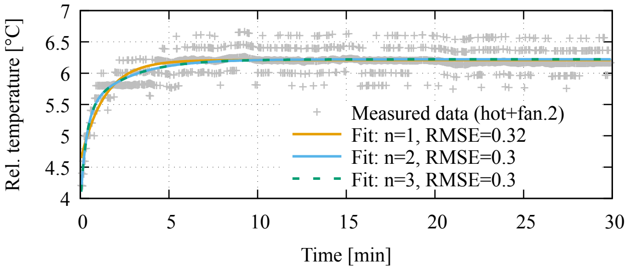


Fig. 8. Measured data and fits with the fan switched on. Note that the ripples in the measured data are caused by the compensation for the ambient temperature. The second order thermal model is $6.2 - 1.5e^{\frac{-t}{0.3}} - 0.9e^{\frac{-t}{2.4}}$.

hours later (hot.3) produces even worse results. A satisfactory estimate is achieved for an experiment longer than 30 minutes.

In Figure 7, curve (C) shows the results of fitting the "shortened" data without constraining the time constants close to the correct value. We observe a high number of outliers and convergence to the constrained case for experiments longer than 50 minutes. Also, the time constants have high variation when the fits of the different runs of the same experiments are compared – see Table I. We attribute this variation to the fact that our thermal model presents a lumped-parameter system, where the spatial distribution of heat production and transfer is ignored, whereas the thermal model in a real MPSoC is a distributed-parameter system where spatial dimension matters.

Finally, in Figure 7, curve (D) shows the results of fitting 2$^{\text{nd}}$ order model with a systematic estimation error.

To conclude, a satisfactory estimate T_∞ is achieved for experiments longer than $1.5 \max_i(\tau_i) \approx 30$ minutes. To have satisfactory temperature estimates for shorter experiments, it is necessary to decrease the time constants of the tested system. One of the possible ways is described in the following section.

D. Using the CPU fan to decrease time constants

In thermal models, the time constant τ can be computed as $\tau = RC$, where R is the *thermal resistance* between two objects with different temperatures, and C is the *thermal capacity* of the object whose temperature is being measured. In our case, the object is the MPSoC chip. The time constant (τ_i) can be reduced by:

- reducing the thermal resistance R. It can be achieved by regulating the CPU fan speed – higher RPM of the fan motor results in lower thermal resistance between the heat sink and the surrounding environment, or
- reducing the capacity C, e.g., by removing the heat sink from the MPSoC chip.

We decided to pursue the former alternative with the CPU fan as it is easy to implement. Figure 8 visualizes the temperature variations from the same workload as in Figure 4, but with the fan running at full speed. It can be seen that the steady-state temperature is only 6.2 °C above ambient temperature and that the time constants are much lower: 0.3 and 2.4 minutes, respectively. Also note that in this case, the 3$^{\text{rd}}$ order model is not necessary as it provides the same result as the 2$^{\text{nd}}$ order model.

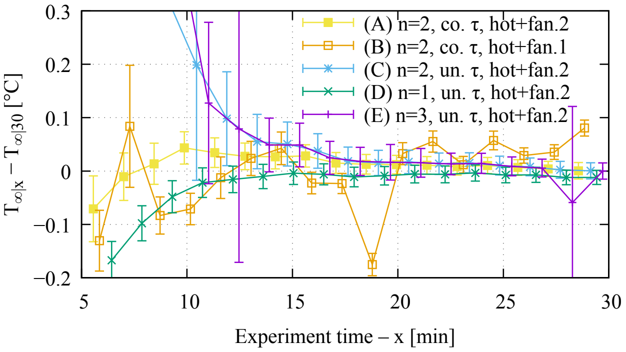


Fig. 9. Relation between the length of the experiment and estimation of T_∞. Vertical axis shows the difference between T_∞ estimated from data of length x ($T_{\infty|x}$) and from full 30 minutes of data ($T_{\infty|30}$). Error bars represent 95% confidence intervals.

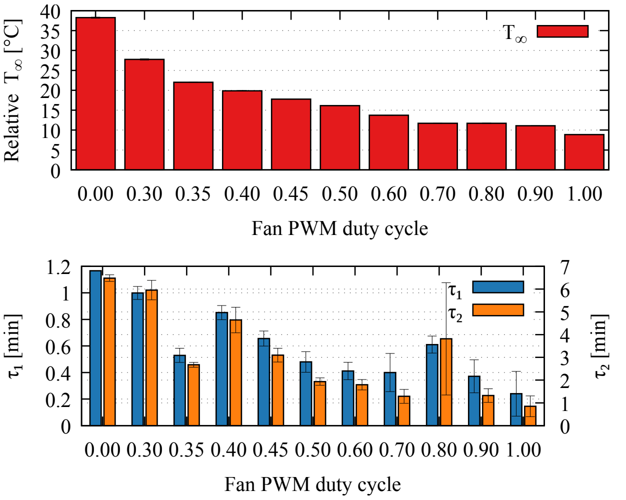


Fig. 10. Relation between different fan speeds and estimated thermal model parameters. Note that 0.3 is the lowest PWM duty cycle that makes the fan move. Error bars represent 95% confidence intervals.

When we try to estimate the T_∞ from a shorter period of time, we can see (Figure 9) that satisfactory results are obtained for experiments longer than 13 minutes with unconstrained (un.) τ estimations. Constraining (co.) τ constants leads to systematic errors for data from different experiments (curve B). The 1$^{\text{st}}$ order model error is slightly below zero even for $x \to 30$ (curve D). The 3$^{\text{rd}}$ order model (curve E) gives the same results as 2$^{\text{nd}}$ order model, but with few outliers.

It may happen that for less CPU/memory intensive user-defined workloads, the T_∞ is even lower than 6 °C. In that case, the resolution of the temperature sensors might not be sufficient to provide precise temperature estimates. A possible mitigation is to lower the CPU fan speed. The results of the 2$^{\text{nd}}$ order thermal model fitting with different CPU fan speeds are presented in Figure 10. The T_∞ temperature clearly decreases with increasing fan speed and the same trend can be observed for time constants τ_1 and τ_2 except for few outliers (0.35 and 0.8 PWM duty cycle). For some experiments, mostly with high CPU fan speeds (in Figure 10, it is visible from error bar size for 0.8 PWM duty cycle) the fitting algorithm is not able to precisely estimate τ_2, because the slow mode is almost not visible in the measured data, i.e., the 1st order model would be more appropriate there. From non-outlier experiments, we observe that τ_2 drops from 6 to about 0.8 minutes. The outliers in τ estimates are the reason why constraining the time constants to "correct" values during the model fitting, as described in Section V-C, does not always lead to satisfactory results.

The results presented in this section follow from laws of physics and as such, they should apply universally to different hardware platforms. We validated them on second platform – NVIDIA Jetson Xavier – and the measured trends were the same as those described above.

E. Summary

Analyzing the data from the Thermobench measurements is not fully automatic and may require a few manual steps. These steps are mainly related to the selection of the thermal model order, the experiment duration, and the appropriate CPU fan speed. Additionally, after fitting the thermal model, one has to check that the model fitting algorithm did not end up in a local minimum, which may result in imprecise T_∞ estimations. After these manual steps are complete, the methods described in this section give a reasonably precise estimate for the T_∞. The estimate T_∞ is proportional to the heat flow $\dot{Q}$ generated by the executed workload.

By fitting the thermal model to the measured data, we obtain more robust estimates for T_∞. Furthermore, we have demonstrated the role of the CPU fan to shorten the experimental time. With the CPU fan switched on, it is sufficient to run the experiments for as short as 15 minutes instead of the initial 60 minutes. Note that the results obtained from the fan-enabled testbed are still applicable to fan-less operation in target applications after scaling the temperatures and time constants up.

VI. Experimental results

The Thermobench repository contains multiple user-defined workloads that might be used to assess the thermal and performance characteristics of the selected MPSoC in the testbed. In what follows, we introduce three types of user-defined workloads. Each one of the workloads has been validated on the i.MX8 board.

A. CPU computation-intensive workloads

The `benchmarks/CPU/instr` folder contains various CPU micro-benchmarks that perform mostly arithmetic operations. With these benchmarks, we compare the thermal efficiency and the performance of the CPUs and the CPU clusters. Note that by thermal efficiency, we refer to heat flow, is proportional to steady-state temperature T_∞.

In Figure 11, we list the multiplication operations. The top row compares the single-core performance of A53 and A72 CPUs. The A72 core offers higher performance for non-SIMD and floating-point SIMD instructions. Surprisingly, the integer SIMD instructions are faster on A53. The CPU temperature does not depend significantly on a particular type of operation. On average, the A72 produces $51.1 \pm 5.7\%$ more heat than the A53, while delivering only $92.9 \pm 1.6\%$ of the A53 performance.

The bottom row compares the multi-core performance, where the same benchmark was running on all CPUs within a single cluster. For the A53 cores, the comparison between the single-core and the multi-core execution suggests that the performance increases four times while the temperature rises by only $31.9 \pm 8.0\%$. For the A72, the comparison between the single-core and the multi-core execution suggests that the performance is increased by $2\times$ while the temperature is raised by $27.8 \pm 4.4\%$. The experiments suggest that if all cores in a cluster are used, the A53 cluster is always faster than the A72 cluster, while the A72 cluster dissipates $46.4 \pm 8.6\%$ more heat.

B. CPU memory-intensive workloads

In the Thermobench repository, the CPU memory-intensive workloads are referred to as `membench` benchmark. The `membench` benchmark stresses each level of the memory hierarchy and measures the available memory bandwidth.

The memory bandwidth achieved by various numbers of CPUs can be seen in Figure 12. As expected, the highest bandwidth is for L1 cache memory, i.e., for working set size (WSS) $\leq$ 32 KiB, followed by the L2 cache bandwidth (32 KiB < WSS < 1 MiB) and the lowest bandwidth is, unsurprisingly, available for the DRAM accesses (WSS > 1 MiB). The drops in the performance are aligned with the size of the caches. The further the cache from the processor core is, the lower the performance is. One may also observe, that the DRAM bandwidth available to the 2x A72 cores is slightly lower than the DRAM bandwidth available to the 4x A53 cores.

Figure 13 visualizes the temperature effects of accessing different parts of the memory hierarchy. Clearly, the L1 cache accesses are the most thermal efficient, whereas the DRAM accesses are the least thermal efficient.

C. GPU-intensive workloads

User-defined workloads for the GPU are based on OpenCL-based benchmarks. In OpenCL, the compute work is divided into *work items*. The total number of work items is referred to as the *global size*. The work items are being worked on by *kernel* code running in so-called *work groups*. Each work group processes a certain number (called *local size*) of work items in parallel. The work groups are executed on the GPU either sequentially or in parallel, depending on its (local) size and the size of the GPU.

Figure 14 lists the results from the OpenCL `mandelbrot` (compute-bound) benchmark. The left graph shows the results

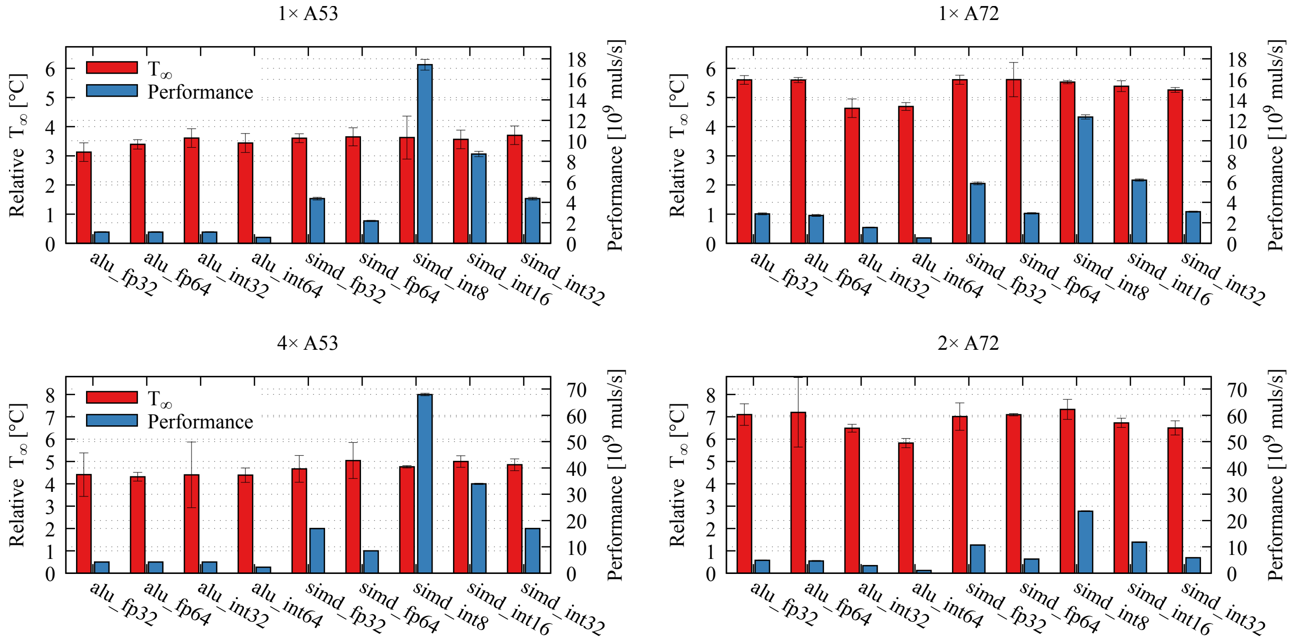


Fig. 11. Comparison of relative steady state temperature T_∞ and performance of multiplication instructions (number of performed multiplications per second) on different CPUs.

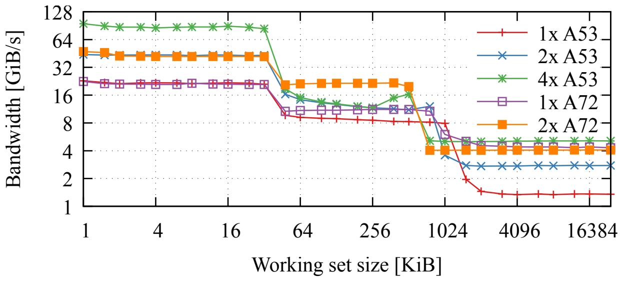


Fig. 12. Memory bandwidth with sequential access.

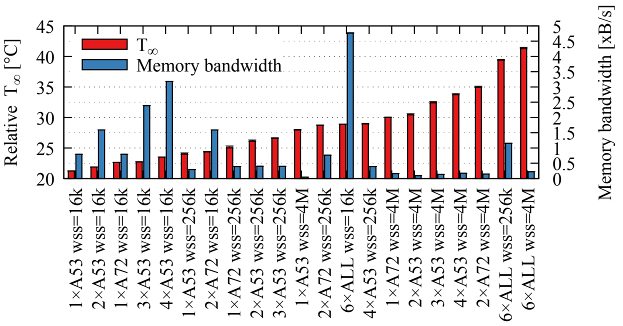


Fig. 13. Memory performance and temperature (measured with the fan switched off).

from the experiments with different global sizes and the same local size. The maximum performance is reached for a global size of 512 or higher (with negligible performance loss for a size of 1024). Smaller global sizes cannot reach the full GPU parallelism, and hence the computation takes a longer time. The steady-state temperature T_∞ decreases with decreasing performance.

In Figure 14, the graph on the right side shows that there is no significant difference in both temperature and performance when the same amount of work items is divided into differently sized work groups.

Figure 15 shows the results of the OpenCL memory-bound benchmark that reads dummy data from the DRAM memory. The left graph suggests that the increased parallelism (global size) leads to decreased performance because the memory bandwidth is the limiting factor. The temperature slightly decreases with performance reduction. The right graph shows that varying the local size makes no difference.

VII. Conclusions

We demonstrated the functionality of our testbed and of the Thermobench tool for processing the data gathered from the testbed. Presented experimental results with computation and memory intensive CPU and GPU benchmarks show which types of results can be obtained from the testbed.

In the future, we intend to leverage the findings of the current work and develop advanced scheduling and resource allocation techniques aiming at finding an optimal trade-offs between the dissipated heat and the achieved performance. The currently proposed testbed will be used for assessing the effectiveness of the proposed techniques.

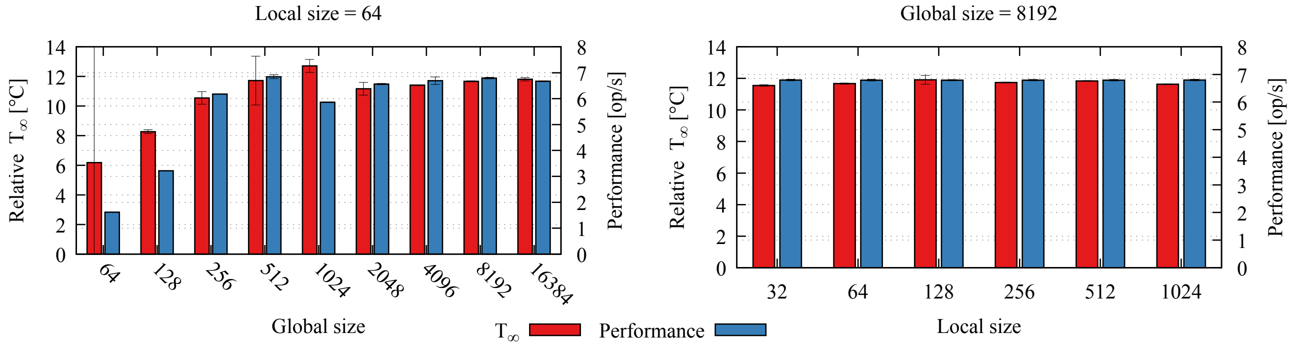


Fig. 14. Performance of a compute-bound GPU benchmark.

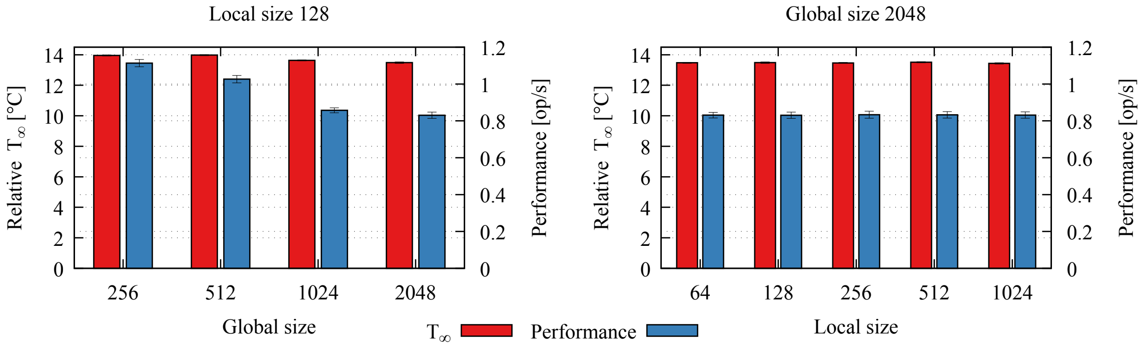


Fig. 15. Performance of a memory-bound GPU benchmark.

ACKNOWLEDGMENTS

This research has received funding from the Clean Sky 2 Joint Undertaking under the European Union's H2020 research and innovation programme under grant agreements No 807081 and No 832011.

REFERENCES

[1] NXP. (2020) i.MX 8QuadMax/QuadPlus Multisensory Enablement Kit. [Online]. Available: https://www.nxp.com/design/development-boards/i-mx-evaluation-and-development-boards/i-mx-8quadmax-multisensory-enablement-kit-mek:MCIMX8QM-CPU

[2] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan, "HotSpot: A compact thermal modeling methodology for early-stage VLSI design," vol. 14, no. 5, pp. 501–513.

[3] A. Kanduri, M. Haghbayan, A. M. Rahmani, M. Shafique, A. Jantsch, and P. Liljeberg, "adBoost: Thermal Aware Performance Boosting Through Dark Silicon Patterning," vol. 67, no. 8, pp. 1062–1077.

[4] Y. Chandarli, N. Fisher, and D. Masson, "Response Time Analysis for Thermal-Aware Real-Time Systems under Fixed-Priority Scheduling," in *2015 IEEE 18th International Symposium on Real-Time Distributed Computing*, pp. 84–93.

[5] N. Bombieri, F. Busato, and F. Fummi, "Power-aware Performance Tuning of GPU Applications Through Microbenchmarking," in *Proceedings of the 54th Annual Design Automation Conference 2017*, ser. DAC '17. ACM, pp. 66:1–66:6. [Online]. Available: http://doi.acm.org/10.1145/3061639.3062304

[6] E. Calore, A. Gabbana, S. F. Schifano, and R. Tripiccione, "Evaluation of DVFS techniques on modern HPC processors and accelerators for energy-aware applications," vol. 29, no. 12, p. e4143. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1002/cpe.4143

[7] J. Lucas and B. Juurlink, "MEMPower: Data-Aware GPU Memory Power Model," in *Architecture of Computing Systems – ARCS 2019*, ser. Lecture Notes in Computer Science, M. Schoeberl, C. Hochberger, S. Uhrig, J. Brehm, and T. Pionteck, Eds. Springer International Publishing, pp. 195–207.

[8] B. Johnston, B. Lee, L. Angove, and A. Rendell, "Embedded Accelerators for Scientific High-Performance Computing: An Energy Study of OpenCL Gaussian Elimination Workloads," in *2017 46th International Conference on Parallel Processing Workshops (ICPPW)*, pp. 59–68.

[9] F. Muslim, A. Demian, L. Ma, L. Lavagno, and A. Qamar, "Energy-efficient FPGA implementation of the k-nearest neighbors algorithm using OpenCL," in *Position Papers of the 2016 Federated Conference on Computer Science and Information Systems*, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 9, 2016, pp. 141–145.

[10] C. Schlaak, M. Fakih, and R. Stemmer, "Power and Execution Time Measurement Methodology for SDF Applications on FPGA-based MPSoCs." [Online]. Available: http://arxiv.org/abs/1701.03709

[11] K. Dev, I. Paul, W. Huang, Y. Eckert, W. Burleson, and S. Reda, "Implications of Integrated CPU-GPU Processors on Thermal and Power Management Techniques." [Online]. Available: http://arxiv.org/abs/1808.09651

[12] Y. Lee, K. G. Shin, and H. S. Chwa, "Thermal-Aware Scheduling for Integrated CPUs–GPU Platforms," in *EMSOFT'19*. [Online]. Available: https://rtcl.eecs.umich.edu/rtclweb/assets/publications/2019/yml-emsoft.pdf

[13] J. Perez Rodriguez and P. Meumeu Yomsi, "Thermal-aware schedulability analysis for fixed-priority non-preemptive real-time systems," in *2019 IEEE Real-Time Systems Symposium (RTSS)*, pp. 154–166, ISSN: 2576-3172.

[14] S. Hosseinimotlagh and H. Kim, "Thermal-Aware Servers for Real-Time Tasks on Multi-Core GPU-Integrated Embedded Systems," in *2019 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pp. 254–266.

[15] S. Pagani, H. Khdr, J. Chen, M. Shafique, M. Li, and J. Henkel, "Thermal Safe Power (TSP): Efficient Power Budgeting for Heterogeneous Manycore Systems in Dark Silicon," vol. 66, no. 1, pp. 147–162.

[16] S. Che, J. W. Sheaffer, M. Boyer, L. G. Szafaryn, L. Wang, and K. Skadron, "A characterization of the rodinia benchmark suite with comparison to contemporary CMP workloads," in *IEEE International Symposium on Workload Characterization (IISWC'10)*, pp. 1–11.

[17] Workswell. (2020) Workswell infrared camera. [Online]. Available: https://workswell-thermal-camera.com/workswell-infrared-camera-wic/specifications

[18] Intel. (2020) Minnowboard turbot. [Online]. Available: https://software.intel.com/content/www/us/en/develop/topics/iot/hardware/minnow-board-turbot.html

[19] TE Connectivity. (2020) Htu21d digital high accuracy rh/t sensor. Available: https://www.te.com/global-en/product-CAT-HSC0004.html

[20] Toshiba. (2020) TB6612FNG. [Online]. Available: https://toshiba.semicon-storage.com/ap-en/semiconductor/product/motor-driver-ics/brushed-dc-motor-driver-ics/detail.TB6612FNG.html

[21] F. T. Brown, *Engineering System Dynamics: A Unified Graph-Centered Approach, Second Edition*, 2nd ed. CRC Press.

4^{th} International Conference on Lean and Agile Software Development

THE evolution of software development life cycles is driven by the perennial quest on how to organize projects for better productivity and better quality. The traditional software development projects, which followed well-defined plans and detailed documentations, were unable to meet the dynamism, unpredictability and changing conditions that characterize rapidly changing business environment. Agile methods overcame these limits by considering that requirements are not static but dynamic, while customers are unable to definitively state their needs up front. However, the advent of agile methods divided the software engineering community into opposing camps of traditionalists and agilists. After more than a decade of debate and experimental studies a majority consensus has emerged that each method has its strengths as well as limitations, and is appropriate for specific types of projects, while numerous organizations have evolved toward the best balance of agile and plan-driven methods that fits their situation.

In more recent years, the software industry has started to look at lean software development as a new approach that could complement agile methods. Lean development further expands agile software development by adopting practices from lean manufacturing. Lean emphasizes waste elimination by removing all nonvalue-adding activities.

TOPICS

The objective of LASD is to extend the state-of-the-art in lean and agile software development by providing a platform at which industry practitioners and academic researchers can meet and learn from each other. We are interested in high quality submissions from both industry and academia on all topics related to lean and agile software development. These include, but are not limited to:

- Combining lean and agile methods for software development
- Lean and agile requirements engineering
- Scaling agile methods
- Distributed agile software development
- Challenges of migrating to lean and agile methods
- Balancing agility and discipline
- Agile development for safety systems
- Lean and agility at the enterprise level
- Conflicts in agile teams
- Lean and agile project production and management
- Collaborative games in software processes
- Lean and agile coaching
- Managing knowledge for agility and collaboration
- Tools and techniques for lean and agile development
- Measurement and metrics for agile projects, agile processes, and agile teams
- Innovation and creativity in software engineering
- Variability across the software life cycle
- Industrial experiments, case studies, and experience reports related to all of the above topics
- Gamification
- Affective Software Engineering

TECHNICAL SESSION CHAIRS

- **Przybyłek, Adam,** Gdansk University of Technology, Poland

PROGRAM COMMITTEE

- **Ahmad, Muhammad Ovais,** Karlstad University, Sweden
- **Akman, Ibrahim,** Atilim University, Turkey
- **Ali, Sikandar,** China University of Petroleum, China
- **Almeida, Fernando,** University of Porto & INESC TEC, Portugal
- **Alshayeb, Mohammad,** King Fahd University of Petroleum and Minerals, Saudi Arabia
- **Angelov, Samuil,** Fontys University of Applied Sciences, The Netherlands
- **Bagnato, Alessandra,** SOFTEAM R&D Department, France
- **Belle, Alvine Boaye,** École de Technologie Supérieure, Canada
- **Ben Ayed Elleuch, Nourchène,** Higher Colleges of Technology, ADW, United Arab Emirates
- **Bernhart, Mario,** Vienna University of Technology, Austria
- **Bhadauria, Vikram,** Texas A&M International University, United States
- **Binti Abdullah, Nik Nailah,** Monash University Malaysia, Malaysia
- **Biró, Miklós,** Software Competence Center Hagenberg and Johannes Kepler University Linz, Austria
- **Blech, Jan Olaf,** Aalto University, Finland
- **Borg, Markus,** SICS Swedish ICT AB, Sweden
- **Buchalcevova, Alena,** University of Economics, Prague, Czech Republic
- **Buchan, Jim,** Auckland University of Technology, New Zealand

- **Buglione, Luigi,** Engineering Ingegneria Informatica SpA, Italy
- **Chatzigeorgiou, Alexandros,** University of Macedonia, Greece
- **Cruzes, Daniela,** SINTEF ICT, Norway
- **Daszczuk, Wiktor Bohdan,** Warsaw University of Technology, Poland
- **Dejanović, Igor,** Faculty of Technical Sciences, Novi Sad, Serbia
- **Derezinska, Anna,** Warsaw University of Technology, Institute of Computer Science, Poland
- **Diebold, Philipp,** Fraunhofer IESE, Germany
- **DUTTA, ARPITA,** IIT Kharagpur, India
- **Escalona, Maria Jose,** Universidad de Sevilla, Spain
- **Essebaa, Imane,** Hassan II University of Casablanca, Morocco
- **Fagerholm, Fabian,** University of Helsinki, Finland and Blekinge Institute of Technology, Finland
- **García-Mireles, Gabriel Alberto,** Universidad de Sonora, Mexico
- **Ghofrani, Javad,** HTW Dresden University of Applied Sciences, Germany
- **Goczyła, Krzysztof,** Gdańsk University of Technology, Poland
- **GODBOLEY, SANGHARATNA,** NIT ROURKELA, India
- **Gonzalez Huerta, Javier,** Blekinge Institute of Technology, Sweden
- **Górski, Janusz,** Gdańsk University of Technology, Poland
- **Gregory, Peggy,** University of Central Lancashire, United Kingdom
- **Hanslo, Ridewaan,** Council for Scientific and Industrial Research, South Africa
- **Heil, Sebastian,** Chemnitz University of Technology, Germany
- **Hohenstein, Uwe,** Siemens AG, Germany
- **Hohl, Philipp,** ZF Friedrichshafen AG, Germany
- **Ikonen, Marko,** Projektivarikko Oy, Finland
- **Janes, Andrea,** Free University of Bolzano, Italy
- **Järvinen, Janne,** F-Secure Corporation, Finland
- **Jarzębowicz, Aleksander,** Gdansk University of Technology, Poland
- **Jovanović, Miloš,** University of Novi Sad, Serbia
- **Kakarontzas, George,** Aristotle University of Thessaloniki, Greece
- **Kaloyanova, Kalinka,** Sofia University, Bulgaria
- **Kanagwa, Benjamin,** Makerere University, Uganda
- **Karolyi, Matěj,** Institute of Biostatistics and Analyses, Faculty of Medicine, Masaryk University, Brno, Czech Republic
- **Karpus, Aleksandra,** Gdańsk University of Technology, Poland
- **Kassab, Mohamad,** Innopolis University, Russia
- **Katić, Marija,** Birkbeck, University of London, United Kingdom
- **Khlif, Wiem,** University of Sfax, Tunisia
- **Knodel, Jens,** Fraunhofer IESE, Germany
- **Kropp, Martin,** University of Applied Sciences and Arts Northwestern Switzerland, Switzerland
- **Kuvaja, Pasi,** University of Oulu, Finland
- **Laanti, Maarit,** Nitor, Finland
- **Lehtinen, Timo O. A.,** Aalto University, Finland
- **Lenarduzzi, Valentina,** Tampere University, Finland
- **Liebel, Grischa,** University of Gothenburg, Sweden
- **Luković, Ivan,** University of Novi Sad, Serbia
- **Lunesu, Ilaria,** Università degli Studi di Cagliari, Italy
- **Mahnič, Viljan,** University of Ljubljana, Slovenia
- **Mangalaraj, George,** Western Illinois University, United States
- **Marcinkowski, Bartosz,** Department of Business Informatics, University of Gdansk, Poland
- **Mazzara, Manuel,** Innopolis University, Russia
- **Mesquida Calafat, Antoni-Lluís,** University of the Balearic Islands, Spain
- **Miler, Jakub,** Faculty of Electronics, Telecommunications And Informatics, Gdansk University of Technology, Poland, Poland
- **miller, gloria,** Skema Business School, Germany
- **Misra, Sanjay,** Covenant University, Nigeria
- **Mohapatra, Durga Prasad,** NIT ROURKELA, India
- **Morales Trujillo, Miguel Ehecatl,** University of Canterbury, New Zealand
- **Mordinyi, Richard,** Vienna University of Technology, Austria
- **Muñoz, Mirna,** Centro de Investigación en Matemáticas, Mexico
- **Muszyńska, Karolina,** University of Szczecin, Poland
- **Ng, Yen Ying,** Nicolaus Copernicus University, Poland
- **Nguyen-Duc, Anh,** University College of Southeast Norway, Norway
- **Noyer, Arne,** University of Osnabrueck and Willert Software Tools GmbH, Germany
- **Oktaba, Hanna,** National Autonomous University of Mexico, Mexico
- **Ortu, Marco,** University of Cagliari, Italy
- **Oyetoyan, Tosin Daniel,** SINTEF Digital, Norway
- **Özkan, Necmettin,** Kuveyt Turk Participation Bank, Turkey
- **Panda, Subhrakanta,** Birla Institute of Technology and Science, Pilani, India
- **Pereira, Rui Humberto R.,** Instituto Politecnico do Porto - ISCAP, Portugal
- **Poth, Alexander,** Volkswagen AG, Germany
- **Przybyłek, Michał,** University of Warsaw, Poland
- **Ramsin, Raman,** Sharif University of Technology, Iran
- **Ristić, Sonja,** University of Novi Sad, Faculty of Technical Sciences, Serbia
- **Rossi, Bruno,** Masaryk University, Czech Republic
- **Rybola, Zdenek,** FIT CTU in Prague, Czech Republic
- **Salah, Dina,** Sadat Academy, Egypt
- **Salnitri, Mattia,** University of Trento, Italy

- **Schön, Eva-Maria,** HAW Hamburg, Germany
- **Sedeno, Jorge,** University of Seville, Spain
- **Senapathi, Mali,** Auckland University of Technology, New Zealand
- **Shkroba, Illia,** Polish-Japanese Academy of Information Technology, Poland
- **Sikorski, Marcin,** Polish-Japanese Academy of Information Technology, Poland
- **Śmiałek, Michał,** Politechnika Warszawska, Poland
- **Soares, Michel,** Federal University of Sergipe, Brazil
- **Soria, Álvaro,** ISISTAN Research Institute, Argentina
- **Spichkova, Maria,** RMIT University, Australia
- **Springer, Olga,** Gdańsk University of Technology, Poland
- **Stålhane, Tor,** Norwegian University of Science and Technology, Norway
- **Stettina, Christoph Johann,** Leiden University, The Netherlands
- **Szymański, Julian,** Gdańsk University of Technology, Poland
- **Taibi, Davide,** Free University of Bolzano, Italy
- **Tarhan, Ayca,** Hacettepe University Computer Engineering Department, Turkey
- **Theobald, Sven,** Fraunhofer IESE, Germany
- **Thomaschewski, Jörg,** University of Applied Sciences Emden/Leer, Germany
- **Torrecilla Salinas, Carlos,** University of Seville, Spain
- **Unterkalmsteiner, Michael,** Blekinge Institute of Technology, Sweden
- **Wardziński, Andrzej,** Gdańsk University of Technology, Poland
- **Weichbroth, Paweł,** Gdańsk University of Technology, Poland
- **Werewka, Jan,** AGH University of Sci. and Technology, Poland
- **Winter, Dominique,** University of Applied Sciences Emden/Leer, Germany
- **Wróbel, Michał,** Gdańsk University of Technology, Poland
- **Yilmaz, Murat,** Çankaya University, Turkey
- **Zarour, Nacer Eddine,** University Constantine2, Algeria
- **Łukasiewicz, Katarzyna,** Gdańsk University of Technology, Poland

DOI: 10.15439/2020F214
ISSN 2300-5963 ACSIS, Vol. 21

Machine Learning models to predict Agile Methodology adoption

Ridewaan Hanslo
Council for Scientific and
Industrial Research, South Africa.
Department of Information
Systems, University of Cape Town,
South Africa.
Email: rhanslo@csir.co.za

Maureen Tanner
Department of Information
Systems, University of Cape Town,
South Africa.
Email: mc.tanner@uct.ac.za

***Abstract*—Agile software development methodologies are used in many industries of the global economy. The Scrum framework is the predominant Agile methodology used to develop, deliver, and maintain complex software products. While the success of software projects has significantly improved while using Agile methodologies in comparison to the Waterfall methodology, a large proportion of projects continue to be challenged or fails. The primary objective of this paper is to use machine learning to develop predictive models for Scrum adoption, identifying a preliminary model with the highest prediction accuracy. The machine learning models were implemented using multiple linear regression statistical techniques. In particular, a full feature set adoption model, a transformed logarithmic adoption model, and a transformed logarithmic with omitted features adoption model were evaluated for prediction accuracy. Future research could improve upon these findings by incorporating additional model evaluation and validation techniques.**

Index Terms—Adoption, Agile methodologies, Machine learning, Scrum.

I Introduction

THE Scrum framework is one of many Agile software development methodologies [1]-[3]. The purpose of the Scrum framework is to develop, deliver, and maintain complex software products. The Scrum Guide [4] defines Scrum as *"a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value."*

Scrum remains the predominant Agile software development methodology used for project management according to the 13th annual State of Agile survey. According to the survey [5], Scrum and Scrum variants (such as Scrumban and the Scrum/XP hybrid) account for 72% of the Agile methodologies used.

The rise in popularity of Agile approaches has grown to other industries within the global economy. Some of these industries we are referring to are Transportation, Education, Energy, Healthcare and Pharmaceuticals, and Financial Services [5]. With this growth in Agile popularity, this study posits that incorporating predictive and prescriptive analytics with Agile methodologies' context of use is a start at unpacking the complex relationships between factors related to Agile project outcome.

Machine Learning (ML) can be defined as the study of a *"real world"* phenomenon implementing the scientific principle to iteratively validate and refine a model or hypothesis [6]. From literature as recent as 2015, there was a mention for the need to incorporate Agile and data science methodologies to see frequently realized gains to software development and applications [7].

The purpose of this paper was to use ML techniques to predict adoption of the Scrum Agile methodology. This research takes the reader through the data science lifecycle of the defined problem, data collection, data preparation and exploration, feature extraction, prediction model development, testing and evaluation, followed by the discussion of the findings.

The remainder of this paper comprises of the following sections: Sect. 2 discusses the research problem; Sect. 3 provides literature on incorporating machine learning techniques with Agile software development; Sect. 4 presents the research methodology including the statistical analysis techniques. The results of the machine learning predictive modelling are presented in Sect. 5 and a discussion of the research findings are provided in Sect. 6. Section 7 concludes the paper and provides recommendations for future research.

II Research Problem

There is plenty of literature on the benefits and success of Agile software development methodologies over traditional methodologies such as the Waterfall method [5], [8], [9]. However, literature also note that even when organizations use Agile methodologies and practices for software development projects, less than half of these projects were deemed successful.

The Standish Group's modern criteria for determining project outcomes are known as the triple constraint [9]. These constraints are:

Ridewaan Hanslo wishes to thank the Council for Scientific and Industrial Research (CSIR) for their support allowing the author to continue his scientific inquiry in the Agile software development and Artificial Intelligence research areas.

1. OnBudget – The project remained within the planned budget.
2. OnTime – The project was resolved within a reasonable time estimation.
3. A satisfactory result – The project delivered user and customer satisfaction even though changes were made to the initial scope.

The project outcome definitions taking the triple constraint into account can, therefore, be summarized as follows;

1. Successful – A project that has met all three constraints, OnBudget, OnTime, and with a satisfactory result.
2. Challenged – A project that has accomplished two of the three constraints upon project completion, for example, the project was delivered on budget with a satisfactory result but did not keep to the planned time-of-delivery.
3. Failed – A project that was cancelled before it could be completed, or completed but was not used.

The Standish Group's 2018 CHAOS report stated that 42% of the surveyed Agile projects succeeded, while 50% were challenged, and 8% were reported as failures [10]. While 42% success is not an ideal rate, it is nonetheless an improvement from previous Standish Group CHAOS reports. For example, the 2015 report for 2011 to 2015 had Agile project success as 39%, challenged projects at 52% and failed projects at 9%. When combining Agile and Waterfall projects, the successful project outcomes drops to a low 29% with projects that experienced challenges at 52% and failed projects at 19% [9]. The recent CollabNet VersionOne [5] annual global survey also stated that *"95% of respondents reported at least some of their agile projects have been successful with 48% reporting that most or all of their agile projects were successful"*.

The authors are, therefore, aware of the low success rate of software development project outcomes regardless of the industry, methodology, and project size. We are optimistic that the future project outcome success will have an upward trajectory, however, we are also aware that the acceleration of autonomous and converged technologies can deepen the problem.

We, therefore, posit that ML algorithms can be used to contribute towards improving the success of project outcomes. As a start to solving this complex problem, this research paper focused on developing ML models to predict Agile methodology adoption. Before the outcomes of the project are predicted, we think that predicting the adoption of an Agile methodology during the early stages of the methodologies inclusion in software development projects could contribute significantly to the future understanding and outcomes of Agile projects. In other words, we believe that by understanding the problem earlier at the adoption phase could allow the project team to implement strategies that could pivot the trajectory of future project outcomes.

III. Fusing machine learning with Agile methodologies

From an engineering perspective, ML involves developing software that implements scientific principles. This complex process can be simplified into three steps. The first step is to formulate a hypothesis about a phenomenon, which also includes the model selection. Secondly, collect data to test the hypothesis and validate the model. Lastly, iteratively refine the hypothesis for continuous model increments [6].

Both the Agile software development methodology and ML incorporates an iterative approach to providing solutions to complex challenges. Indeed, past studies have successfully utilized ML within the context of Agile software development. Kahles and others [11] applied ML to automate the root cause analysis in Agile software testing environments. The study was able to produce an ML model that could achieve a prediction accuracy of 88.9% by using artificial neural networks to either classify or pre-process the data for clustering, using manually labelled data.

Another research area within Agile software development where ML models are often used is in effort estimation. Software development effort estimation is the process of estimating the effort required by the software development team within the Agile environment to develop and maintain software [12]. The studies by [12], and [13] used ML algorithms for effort prediction. Satapathy and Rath [13] used ML algorithms such as Random Forest (RF), decision tree (DT), and stochastic gradient boosting (SGB) to improve upon the manual and tedious story pointing approach of effort estimation. The results indicate that RF, DT and SGB improved upon the story point approach, however, SGB outperforms the other two ML algorithms.

Moharreri and others [12] also developed an automated estimation methodology called *"Auto-Estimate"*. The study's ML model construction used supervised learning algorithms. The model was used to improve upon the manual Agile Planning Poker (PP) for effort estimation. PP involves all key stakeholders of the Agile planning team to estimate the effort required to complete a task, which usually makes use of playing cards with estimates using the Fibonacci series of numbers [12]. The results of the study indicate that the J48 Decision Tree (J48) and the Logistic Model Tree (LMT) ML algorithms outperformed PP. The results also suggest combining PP with J48 or LMT yields lower aggregate costs which could in future augment human effort estimation.

Other studies that combined Agile methodologies and ML include a study by [14], which successfully incorporated Agile practices in big data analytics, and the study by [15] which built a model using the J48 ML technique to predict software code defects during automated testing with 85% accuracy, drastically lowering the time needed to detect these potential problems. In addition, Schleier-Smith [7] incorporated Agile practices into data science real-time recommendation system development for benefits such as faster development cycles, quick feedback mechanisms and improved teamwork.

In summary, there is sufficient literature to be found on ML being used with Agile methodologies. The authors found more than 100 search results of ML being used with Agile methodologies on the Scopus and Web of Science citation databases alone.

IV. Methodology

Before the authors could develop and evaluate the ML predictive models a few preprocessing steps had to be undertaken. The first preprocessing step was to extract and synthesize Scrum and Agile adoption challenges within the literature. This was published in a paper entitled *"Scrum adoption challenges detection model: SACDM"* [16] in which a conceptual model was developed to test and evaluate challenges to Scrum adoption. A narrative review was conducted on the existing Agile and Scrum adoption challenges experienced globally and by practitioners in South Africa (SA). The synthesized challenges were used as the independent variables of the model. The first iteration of the Conceptual Framework (CF) known as SACDM generated 19 independent factors that are used to evaluate Scrum adoption as the dependent factor. Some of the independent factors included organizational structure, organizational culture, teamwork, experience, communication, collaboration, complexity, compatibility, and the relative advantage of the Scrum framework. This CF is a custom model adapted from the Diffusion of Innovation (DOI) theory and a study of the adoption of new technology by [17]. The descriptions of each of the independent and dependent variables can be obtained from the *"Scrum adoption challenges detection model: SACDM"* open-access paper [16].

To be able to identify the factors that contribute significantly to the adoption of the Scrum framework, there was a need for testing and evaluation of the CF. This was presented in another paper entitled *"Factors that contribute significantly to Scrum adoption"* [18] which described the process behind the three iterations of the CF that lead to the factors of significance. During the second iteration of the CF, SACDM was renamed as Scrum Adoption Challenges Conceptual Framework (SACCF). The online survey questionnaire serving as a Likert-type scale gathered response data from 78 questionnaire items. The Likert-type scale was used to record the perceived outcomes of Scrum adoption within the organisation, team and individually. The questionnaire design used in the previous paper is accessible online (https://bit.ly/scrumchallengessurvey). The sample consisted of Scrum practitioners working within South African organizations. The research design took the form of a narrative review and survey questionnaire. For the research analysis a set of 207 valid responses to this survey was used to perform Exploratory Factor Analysis (EFA) and Cronbach's alpha analysis, which confirmed the validity and reliability of the questionnaire as the measuring instrument. EFA further revealed that the factors can be reduced from 19 to 14 independent variables. Fig. 1 depicts the 14 factors with Scrum adoption as the dependent variable. The results from the correlational and multiple linear regression (MLR) statistics were used to identify factors that have a significant linear relationship with Scrum adoption. Factors revealed as significant were the management of the Scrum sprint, and the complexity and relative advantage of the Scrum framework. The details of the analysis results and findings can be obtained from the *"Factors that contribute significantly to Scrum adoption"* open-access paper [18].

Using quantitative analysis on Scrum adoption the authors were able to test nineteen research hypotheses in a chapter entitled *"Quantitative Analysis of the Scrum Framework"* [19]. Four hypotheses were shown to be statistically significant to Scrum. These hypotheses are the following;

1. Sprint Management: There is a significant linear (positive correlation) relationship between Sprint Management and Scrum adoption.
2. Change Resistance: There is a significant linear (negative correlation) relationship between Change Resistance and Scrum adoption.
3. Relative Advantage: There is a significant linear (positive correlation) relationship between Relative Advantage and Scrum adoption.
4. Complexity: There is a significant linear (negative correlation) relationship between Complexity and Scrum adoption.

The three papers just described allowing us to firstly, build a conceptual model and test the reliability and validity of the model as a CF. Secondly, the authors further found significant factors that contribute to Scrum adoption. These factors were quantitatively analyzed using correlation coefficients and MLR. Thirdly, thereafter, we could test the research hypotheses. To contribute further to the research field the authors want to incorporate predictive analytics on projects using Agile methodologies. This research paper, therefore, looks at developing the capability for teams and organizations to predict Scrum adoption using predictive analytics.

The factors discussed above form part of the feature engineering process, which is a pre-requisite to the ML model building. To build and test the ML models the sample data had to be split between the training set and test set. For both the training set and test set, Scrum adoption (dependent variable) and the features (independent variables) are added as arguments to the model. The following code sample adds the features and Scrum adoption to the *train_test_split* function of the scikit-learn machine learning library for Python (1).

```
X_train, X_test, y_train, y_test = train_test_split(     (1)
features, adoption, test_size=0.3, random_state=4)
```

In the code sample, the random state was set for testing and replicability (*random_state=4*). The dataset was split into a 69.57% training set and 30.43% test set. Before training the models it was important that the data was normalized and that all assumptions had been met. These assumptions are the assumption of normality of residuals, the assumption of no autocorrelation of residuals, the assumptions of linearity and homoscedasticity, and the assumption of no multicollinearity.

The Bayesian Information Criterion (BIC) is a model selection criterion for a finite list of models [20]. Weakliem [21] critiques BIC for excessively favouring simple models in practice, however, we used it because BIC is a widely used and popular criterion for model selection in linear regression. The lower the BIC value the better the model. The BIC equation can be defined as (2);

$$BIC = \ln(n)k - 2\ln(\hat{L}). \qquad (2)$$

1. k=the data points.
2. n=the number of parameters estimated by the model.
3. $\hat{L}$=the maximum value of the likelihood function of the model.

For this paper, the authors used three models to test the prediction accuracy; the transformed logarithmic (log) adoption model, full feature set adoption model and the transformed log with omitted features adoption model. Each of these three models are using the MLR ML statistical analysis technique using the 14 explanatory variables to predict Scrum adoption as mentioned earlier.

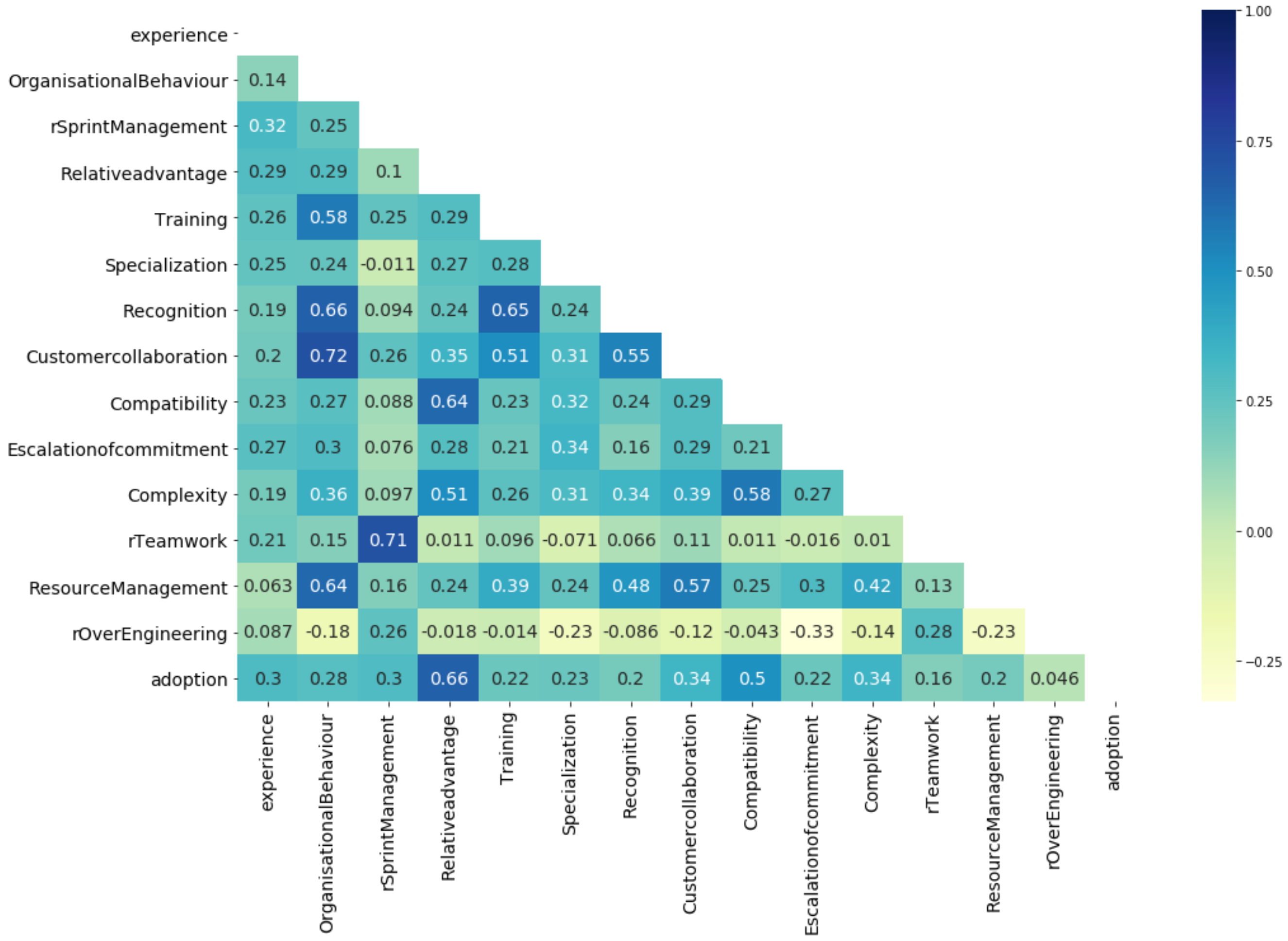
The full feature set model includes the fourteen features (over-engineering, relative advantage, recognition, experience, teamwork, specialization, escalation of commitment, compatibility, resource management, customer collaboration, complexity, training, sprint management, organizational behaviour) to predict Scrum adoption. The transformed log model normalized the skewed data and includes the full feature set to predict Scrum adoption. The transformed log with omitted features model also normalized the skewed data, however, three of the fourteen features (experience, recognition, and compatibility) have been excluded from the feature set to predict Scrum adoption. The BIC value is -0.88 for the log adoption model, and -15.72 for the log with omitted features adoption model.

V. Results

To remind the reader, the prediction of Scrum adoption referred to as adoption going forward, is the focus of this paper. Fig. 1 displays the correlations of the feature set. The stronger the correlation the darker the displayed colour. The negatively phrased questions of features sprint management, teamwork, and over-engineering were recoded (identified by the r prefix).

Some of the relationships between the features and their significance are discussed below.

1. A positive and significant relationship between Relative Advantage and Adoption (*r=0.66, p<0.001*). The correlation was moderate to strong in strength.
2. A positive and significant relationship between Recognition and Organizational Behaviour (*r=0.66, p<0.001*). The correlation was moderate to strong in strength.
3. A positive and significant relationship between Relative Advantage and Compatibility (*r=0.64, p<0.001*). The correlation was moderate in strength.
4. A positive and significant relationship between Customer Collaboration and Training (*r=0.51, p<0.001*). The correlation was moderate in strength.
5. A positive and significant relationship between Resource Management and Organizational Behaviour (*r=0.64, p<0.001*). The correlation was moderate in strength.
6. A positive and significant relationship between Teamwork and Sprint Management (*r=0.71, p<0.001*). The correlation was strong in strength.
7. A positive and significant relationship between Complexity and Relative Advantage (*r=0.51, p<0.001*). The correlation was moderate in strength.
8. A positive and significant relationship between Resource Management and Training (*r=0.39, p<0.001*). The correlation was weak to moderate in strength.
9. A positive and significant relationship between Resource Management and Recognition (*r=0.48, p<0.001*). The correlation was moderate in strength.
10. A positive and significant relationship between Compatibility and Adoption (*r=0.50, p<0.001*). The correlation was moderate in strength.



	experience	OrganisationalBehaviour	rSprintManagement	Relativeadvantage	Training	Specialization	Recognition	Customercollaboration	Compatibility	Escalationofcommitment	Complexity	rTeamwork	ResourceManagement	rOverEngineering	adoption
experience															
OrganisationalBehaviour	0.14														
rSprintManagement	0.32	0.25													
Relativeadvantage	0.29	0.29	0.1												
Training	0.26	0.58	0.25	0.29											
Specialization	0.25	0.24	-0.011	0.27	0.28										
Recognition	0.19	0.66	0.094	0.24	0.65	0.24									
Customercollaboration	0.2	0.72	0.26	0.35	0.51	0.31	0.55								
Compatibility	0.23	0.27	0.088	0.64	0.23	0.32	0.24	0.29							
Escalationofcommitment	0.27	0.3	0.076	0.28	0.21	0.34	0.16	0.29	0.21						
Complexity	0.19	0.36	0.097	0.51	0.26	0.31	0.34	0.39	0.58	0.27					
rTeamwork	0.21	0.15	0.71	0.011	0.096	-0.071	0.066	0.11	0.011	-0.016	0.01				
ResourceManagement	0.063	0.64	0.16	0.24	0.39	0.24	0.48	0.57	0.25	0.3	0.42	0.13			
rOverEngineering	0.087	-0.18	0.26	-0.018	-0.014	-0.23	-0.086	-0.12	-0.043	-0.33	-0.14	0.28	-0.23		
adoption	0.3	0.28	0.3	0.66	0.22	0.23	0.2	0.34	0.5	0.22	0.34	0.16	0.2	0.046	

Fig. 1 The Feature correlation heat map. Displays the relationship between the features and Scrum adoption.

The first model is the full feature set model which has an actual and predicted adoption correlation of 0.75. Fig. 2 displays the actual and predicted adoption and Fig. 3 depicts the residuals. The 95% prediction interval is 4.83 and 1.98 for the upper and lower bound in the full feature set adoption model, respectively.

For the second model, Fig. 4 displays the actual and predicted adoption correlation for the log transformation, while Fig. 5 displays the residual and predicted values. The actual and predicted log adoption correlation is 0.73. The 95% prediction interval is 3.80 and 3.01 for the upper and lower bound in the log adoption model, respectively.

The third model is transformed using log adoption and simplified by dropping three features, namely, experience, recognition, and compatibility, as mentioned earlier. Fig. 6 depicts the transformed and simplified model while the residuals of this model are displayed in Fig. 7. The actual and predicted log adoption with omitted features correlation is 0.73. The 95% prediction interval is 3.79 and 3.01 for the upper and lower bound, respectively, in the log adoption with omitted features model.

Table I displays the R-squared (R^2), and the Mean Squared Error (MSE) for each of the three ML predictive models. The R^2 is a statistical measure of the variance of the predicted values divided by the variance of the data [22]. A 0% R^2 value indicates that the models explain none of the variability of the data, in other words, it is worse than predicting the mean. A 100% R^2 value indicates that the model explains all the variability of the data. The R^2 equation divides the sum of the squares due to regression by the total sum of squares (3).

$$R^2 = 1 - \frac{SS_{regression}}{SS_{total}} \tag{3}$$

The MSE criterion calculates how close the regression line is to the data points by taking into account the predicted value of the observation and eliminates the arbitrariness associated with the residual sum of the squares [23]. Put in another way, the MSE equation measures the average squared error of our predictions where y_i is the actual output and $\hat{y}_i$ is the model's prediction (4). The lower the MSE value the lower the variance of error.

$$MSE = \frac{1}{n}\sum_{i-1}^{N}(y_i - \hat{y}_i)^2 \tag{4}$$

TABLE I.
ML PREDICTIVE MODELS

Adoption Model	R-squared	R-squared %	MSE
Transformed log model	0.527	52.7	0.039
Full feature set model	0.564	56.4	0.507
Transformed log with omitted features model	0.527	52.7	0.038

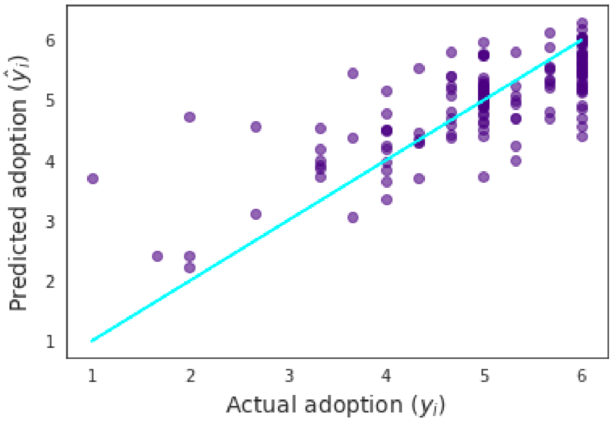


Fig. 2 The normal probability plot for the untransformed full feature set model. The assumption of normality of residuals was met because the actual and predicted adoption residuals were approximately linear.

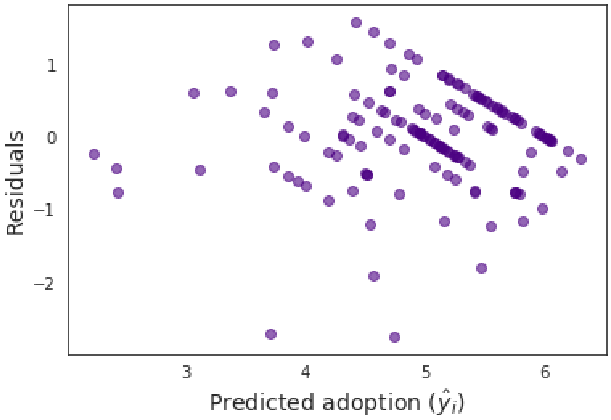


Fig. 3 The Scatterplot for the untransformed full feature set model. The assumptions of linearity and homoscedasticity were met because the residual and predicted values did not curve or funnel out.

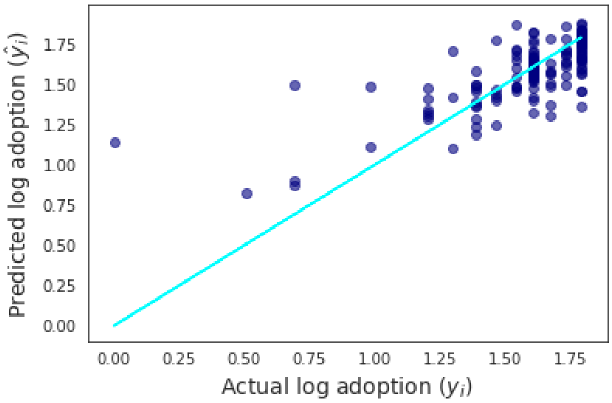


Fig. 4 The normal probability plot for the log adoption model. The assumption of normality of residuals was met because the actual and predicted adoption residuals were approximately linear.

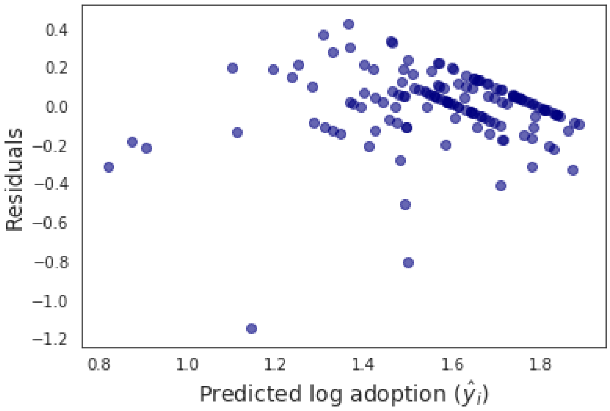


Fig. 5 The Scatterplot for the log adoption model. The assumptions of linearity and homoscedasticity were met because the residual and predicted values did not curve or funnel out.

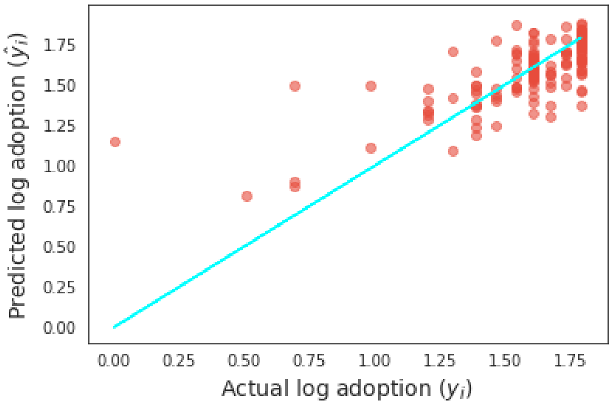


Fig. 6 The normal probability plot for the log adoption with omitted features model. The assumption of normality of residuals was met because the actual and predicted adoption residuals were approximately linear.

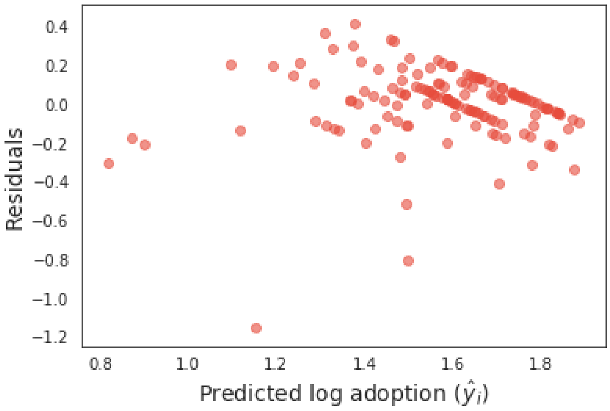


Fig. 7 The Scatterplot for the log adoption with omitted features model. The assumptions of linearity and homoscedasticity were met because the residual and predicted values did not curve or funnel out.

VI. Discussion of Findings

The preprocessing and feature engineering of the response data as described in the research methodology section allowed us to build and evaluate three machine learning (ML) models. The three models were not an exhaustive collection of predictive models as this approach was beyond the scope of this research paper. We wanted to investigate whether different models which include transformations and simplified feature sets can predict Scrum adoption with less variance and error, in other words, at what prediction accuracy.

The three models evaluated in this study was the log adoption model, full feature set adoption model, and the log with the omitted features adoption model. As mentioned in the results, the R-squared (R^2) value measures how close the data are to the regression line, and the Mean Square Error (MSE) measures the average of the square of the errors.

The full feature set adoption model has a moderate variance value of 0.564, explaining more than half of the model instances. The closer the R^2 is to 1 usually the greater the prediction accuracy. This model also has an MSE value of 0.507, indicating a high error rate as the error value is closer to one.

The log adoption model is a transformation of the full feature set model. This model has an R^2 value of 0.527 with a 0.039 MSE value. It is immediately evident that the log model is a better model for adoption prediction accuracy because of the MSE being closer to zero while the R^2 value is greater than 0.5 and less than 0.6, similar to the full feature set model.

The third model is the log with the omitted features adoption model. This model simplified the feature set by removing three of the fourteen features. The three features are experience (*p-value=0.929*), recognition (*p-value=0.969*), and compatibility (*p-value=0.820*) due to their high p-values. The higher the p-values, the less significant of a factor it is to adoption. With the three features removed the R^2 value is 0.527 and the MSE is 0.038.

The transformed log with omitted features model is, therefore, the best-fit prediction model even though it gives a marginally lower error level than the log model. We are fairly confident that we can improve upon the prediction accuracy with a greater randomized sample. Further, we can improve upon the best-fit model by developing a model with lower variance and MSE.

VII. Conclusion

This research paper reports on the development of ML models to predict the accuracy of Scrum adoption based on a feature set derived from a survey questionnaire's response data. The sample size of 207 response data was used to train and test the prediction models. Data cleaning and preprocessing was required before the models could be trained and tested. We trained each of the three models with approximately 70% of the dataset while 30% was used to test the models. The three prediction models was a full feature set adoption model, transformed logarithmic (log) adoption model, and a transformed log with omitted features adoption model. The adoption model with the highest prediction accuracy was the transformed log with omitted features model with an *R^2=0.527* and *MSE=0.038*. The full feature set model was the least accurate when looking at the combination of *R^2=0.564* and *MSE=0.507*. Implications of these findings, while still preliminary, allows researchers and practitioners to gain a better understanding into which features are potentially significant to predicting Scrum adoption. Researchers could compare our findings against their own and modify their modelling techniques.

Limitations of this research are threefold. Firstly, the training and test split used in this research paper for the dataset has been reported previously as displaying biases [24]. Secondly, the model evaluation measure of R^2 being used for goodness-of-fit of the models are one of many metrics used for prediction model evaluation. Thirdly, additional model validation techniques such as bootstrap sampling has not been used in this preliminary research.

Additional research, therefore, could implement the bootstrap aggregating technique to improve the stability and accuracy of the ML algorithms. Metrics such as max error, mean absolute error, the median squared error could be used to further evaluate prediction accuracy. Using a larger randomized sample would improve the predictive accuracy of the models used within this research paper. Further research could develop a logistic regression model with a larger dataset to predict Agile project outcomes, modifying the conceptual framework and methodology as required.

References

[1] A. Przybylek, D. Kotecka, “Making agile retrospectives more awesome,” *In Federated Conference on Computer Science and*

Information Systems (FedCSIS), Prague, Czech Republic, 2017. https://doi.org/10.15439/2017F423

[2] A. Przybylek, M. Zakrzewski, "Adopting Collaborative Games into Agile Requirements Engineering," *In 13th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE'18)*, Funchal, Madeira, Portugal, 2018. https://doi.org/10.5220/0006681900540064

[3] A. Przybyłek, M. Olszewski, "Adopting collaborative games into Open Kanban," *In Federated Conference on Computer Science and Information Systems (FedCSIS)*, Gdansk, Poland, 2016. https://doi.org/10.15439/2016F509

[4] K. Schwaber and J. Sutherland, "The Scrum Guide," *scrum.org*, https://www.scrum.org/index.php/resources/scrum-guide. 2020.

[5] CollabNet VersionOne, "13th Annual State of Agile Report," *collab.net*, https://www.stateofagile.com. 2020.

[6] A. Jung, "Machine Learning: Basic Principles," *arXiv: 1805.05052v11[cs.LG]*, https://arxiv.org/pdf/1805.05052.pdf. 2019.

[7] J. Schleier-Smith, "An architecture for agile machine learning in real-time applications," *In Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 2059-2068, ACM, 2015.

[8] R. Hoda and L.K. Murugesan, "Multi-level agile project management challenges: A self-organizing team perspective," *The Journal of Systems & Software*, 117, 245-257. 2016. https://doi.org/10.1016/j.jss.2016.02.049

[9] The Standish Group, "CHAOS Report 2015," *The Standish Group*, https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf. 2020.

[10] Vitality Chicago, "Agile Project Success Rates are 2X Higher than Traditional Projects," *VitalityChicago* https://vitalitychicago.com/blog/agile-projects-are-more-successful-traditional-projects/. 2020.

[11] J. Kahles, J. Torronen, T. Huuhtanen and A. Jung, "Automating Root Cause Analysis via Machine Learning in Agile Software Testing Environments," *In Proceedings - 2019 IEEE 12th International Conference on Software Testing, Verification and Validation*, ICST 2019, pp. 379-390. [8730163] Institute of Electrical and Electronics Engineers. https://doi.org/10.1109/ICST.2019.00047

[12] K. Moharreri, A.V. Sapre, J. Ramanathan and R. Ramnath, "Cost-effective supervised learning models for software effort estimation in agile environments," *In 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC),* vol. 2, pp. 135-140. IEEE. 2016.

[13] S.M. Satapathy and S.K. Rath, "Empirical assessment of machine learning models for agile software development effort estimation using story points," *Innovations in Systems and Software Engineering*, 13(2-3), pp.191-200. 2017.

[14] H.M. Chen, R. Kazman and S. Haziyev, "Agile big data analytics for web-based systems: An architecture-centric approach," *IEEE Transactions on Big Data*, 2(3), pp.234-248. 2016.

[15] L. Butgereit, 2019, "Using Machine Learning to Prioritize Automated Testing in an Agile Environment," *In 2019 Conference on Information Communications Technology and Society (ICTAS)*, pp. 1-6. IEEE. 2019.

[16] R. Hanslo and E. Mnkandla, "Scrum Adoption Challenges Detection Model: SACDM," *In Federated Conference on Computer Science and Information Systems (FedCSIS)*, Poznan, Poland: IEEE: 949–957. 2018.

[17] F. Sultan and L. Chan, "The adoption of new technology: the case of object-oriented computing in software companies," *IEEE transactions on Engineering Management*, 47(1): 106–126. 2000.

[18] R. Hanslo, E. Mnkandla and A. Vahed, "Factors that contribute significantly to Scrum adoption," *In Federated Conference on Computer Science and Information Systems (FedCSIS)*, Leipzig, Germany: IEEE: 821–829. 2019.

[19] R. Hanslo, A. Vahed and E. Mnkandla, "Quantitative Analysis of the Scrum Framework," *In: Przybyłek A., Morales-Trujillo M. (eds) Advances in Agile and User-Centred Software Engineering. LASD 2019, MIDI 2019*, Lecture Notes in Business Information Processing, vol 376. Springer, Cham. 2020. https://doi.org/10.1007/978-3-030-37534-8_5

[20] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," *In Proc. DARPA broadcast news transcription and understanding workshop*, vol. 8, pp. 127-132. Feb. 1998.

[21] D.L. Weakliem, "A critique of the Bayesian information criterion for model selection," *Sociological Methods & Research*, 27(3), pp.359-397. 1999.

[22] A. Gelman, B. Goodrich, J. Gabry and A. Vehtari, "R-squared for Bayesian regression models," *The American Statistician*, 73(3), pp.307-309. 2019.

[23] D.M. Allen, "Mean square error of prediction as a criterion for selecting variables," *Technometrics*, 13(3), pp.469-475. 1971.

[24] C. Tantithamthavorn, S. McIntosh, A. E. Hassan and K. Matsumoto, "An Empirical Comparison of Model Validation Techniques for Defect Prediction Models," *In IEEE Transactions on Software Engineering*, vol. 43, no. 1, pp. 1-18, Jan. 2017. https://doi.org/10.1109/TSE.2016.2584050

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 705–708 DOI: 10.15439/2020F62 ISSN 2300-5963 ACSIS, Vol. 21

Retrospective games in Intel Technology Poland

Dominik Mich
Gdansk University of Technology, Faculty of Electronics, Telecommunications and Informatics
Narutowicza 11/12, 80-233 Gdansk, Poland.
Email: dominik.mich@autonomik.pl

Yen Ying Ng
Nicolaus Copernicus University, Department of English Studies, Torun, Poland.
Email: nyysang@gmail.com

Abstract— **One of the Agile principles is that the team should regularly reflect on "how to become more effective, then tunes and adjusts its behavior accordingly". While the setup of a retrospective session is intuitive, in praxis, conducting successful retrospectives is challenging. This paper is a continuation of our previous work on the use collaborative games in addressing common retrospective problems. In addition to the replication of our previous action research in a new context, we aim to investigate whether preliminary anonymous idea generation mitigates negative social influences that have been identified as causes of poor performance of brainstorming. The obtained results confirms the previous findings that game-based retrospectives produces better results than the standard retrospective as well as improves participants' creativity, involvement, and communication. Our findings also suggest benefits to the preliminary anonymous idea generation.**

I. Introduction

ALTHOUGH agile software development has become mainstream in industry, changing to an agile mindset is still challenging for many companies [8, 13, 16]. However, in today's competitive business world, which creates demand for shorter cycle times and in which technology evolves rapidly [18], the need for agility has become even more important [6]. To adapt to environmental changes, mitigate the frequent problems with addressing customer's needs [10, 19] and adjust their processes accordingly, organizations implement process improvement initiatives [27]. Scrum provides organizations continuous process improvement by the Sprint Retrospective [7]. According to the Scrum Guide, retrospective is a time-boxed meeting held at the end of each sprint to reflect on the past iteration and creates plans for improvements to be enacted during the next iteration. Retrospectives are held as face-to-face meetings, which are the most common way of communication, both among the agile team and between the team and the stakeholders [11]. The aim of team reflexivity is to share experiences, learn from failures and successes, and adjust the way of working to become continuously better [9].

Although reflection is a fundamental aspect of agile software development, not all teams take it as seriously as they should. Babb et al. [2] found that in the hectic life of software development, where teams perform under sustained pressure to deliver the Increment, retrospectives are the meetings most likely to be skipped or compromised over time. Furthermore, several studies suggested that running an effective and enjoyable retrospective meeting is challenging [22]. This is because if the meeting is repeated according to the same pattern over and over again, it can cause a certain monotony and lack of motivation. In turn, when retrospectives become flat, they may be abandoned because they stop adding value. To address this challenge, Przybyłek & Kotecka [22] successfully refreshed retrospective meetings in three agile teams by adopting collaborative games. Collaborative games are designed to be engaging and support retrospectives by providing structure to the meeting, new exploration perspectives, encouraging equal participation and stimulating creativity [22].

Recently, Gaikwad et al. [5] pointed out further disadvantages of retrospective meetings: they are non-anonymous and time consuming. In fact, both issues have been long identified with face-to-face idea generation sessions [4]. It appears that participants may feel fear of negative evaluation from others, [17] and they also feel anxious that there may be negative social consequences of sharing ideas contrary to the ideas of higher-status others [3]. When not all participants feel free to contribute, potential good ideas are lost [3]. Besides, in a face-to-face group, participants are unable to express themselves simultaneously, but must take turns to express their ideas (production blocking) [4, 17]. Nunamaker et al. [14] found that these two inhibitory factors can be reduced by electronic idea generation sessions, in which the participants are anonymous (therefore mitigating evaluation apprehension) and in which participation is asynchronous (therefore mitigating production blocking). Similar findings were also obtained by Davis et al. [3].

Since anonymity has been demonstrated to mitigate negative group effects that are responsible for the productivity loss in face-to-face idea generation sessions, this paper is aimed to introduce anonymity in the idea-generation phase of the retrospective. We expect that anonymity will encourage participants to express their true feelings and critical thinking, which in turn will increase the quality and quantity of ideas generated [26]. Besides, we intend to replicate our previous studies [15, 25] in which we adopted the "game-based retrospectives" approach initially introduced by Przybyłek & Kotecka [22].

II. RELATED WORK

There has been lots of interest in adopting collaborative games to support agile teams. Przybyłek & Olszewski [21] defined an extension to Open Kanban, which consists of 12 collaborative games to help novice Kanban practitioners to understand the Kanban principles. Przybyłek and his team [24, 28] proposed a framework for extending Scrum with 9 collaborative games to enhance agile requirements engineering. Przybyłek & Kowalski [23] developed a web portal which provides 8 collaborative games to be used in agile software development. Przybyłek & Kotecka [22] adopted 5 retrospective games, which improved team members' creativity, involvement, and communication as well as produced better results than the standard retrospective. In our previous work [15, 25], we confirmed their findings, while this paper complements and extends this research area.

III. RESEARCH METHOD AND CONTEXT

Our study was carried out as Action Research. Action Research is aimed at solving an immediate business problem, while simultaneously expanding scientific knowledge [1]. The researcher is concerned to intervene in the studied situations for the explicit purpose of improving the situation. According to Avison et al. [1] terminology, our study followed research-driven initiation, i.e. our supervisor was in possession of a general theoretical approach to addressing a problem situation (which was specified as a proposal for a Master's thesis) and searching for settings that are characterized by such a problem. The first author of this paper had been a member of a Scrum team at Intel Technology Poland that was willing to participate in the research, so he undertake the Master's project. The team consisted of 11 developers, a team lead and a Scrum Master. Four senior developers, who had been in the team from the beginning, were imbued with higher status because of their knowledge and expertise. The team was responsible for validation of the Intel Ethernet Switch software. The software was very often updated and released to the external customer. Accordingly, it was very important to keep the required levels of quality. The requirements for the specific features were very void and changed easily over the time.

IV. ACTION RESEARCH IN INTEL TECHNOLOGY POLAND

A. Diagnosing

We started by conducting a focus group to inspect Scrum practices used by the participating team. Our aim was to investigate the practical implementation of Scrum and the ScrumButs. All team members attended the focus group. The discussion was structured around 5 questions, but in this paper we focus only on feedback pertaining to the Sprint Retrospective (all questions and the full feedback can be found in [12]). It turned out that all Scrum meetings, except Daily Scrum, were merged into one. Besides, retrospectives were often skipped as they "did not bring any useful information".

B. Action planning

Since we identified a lot of ScrumButs regarding the whole Scrum process, we decided to break the intervention into smaller steps, which would be implemented as separate Action Research cycles. We also concluded that the Sprint Retrospective should be fixed first. Accordingly, in the first cycle, we planed to implement all retrospective games that we used in our prior study [25] and in addition try a new game, namely Mountain Climbing. Besides, we decided to implement each game twice, first in non-anonymous and then in anonymous way.

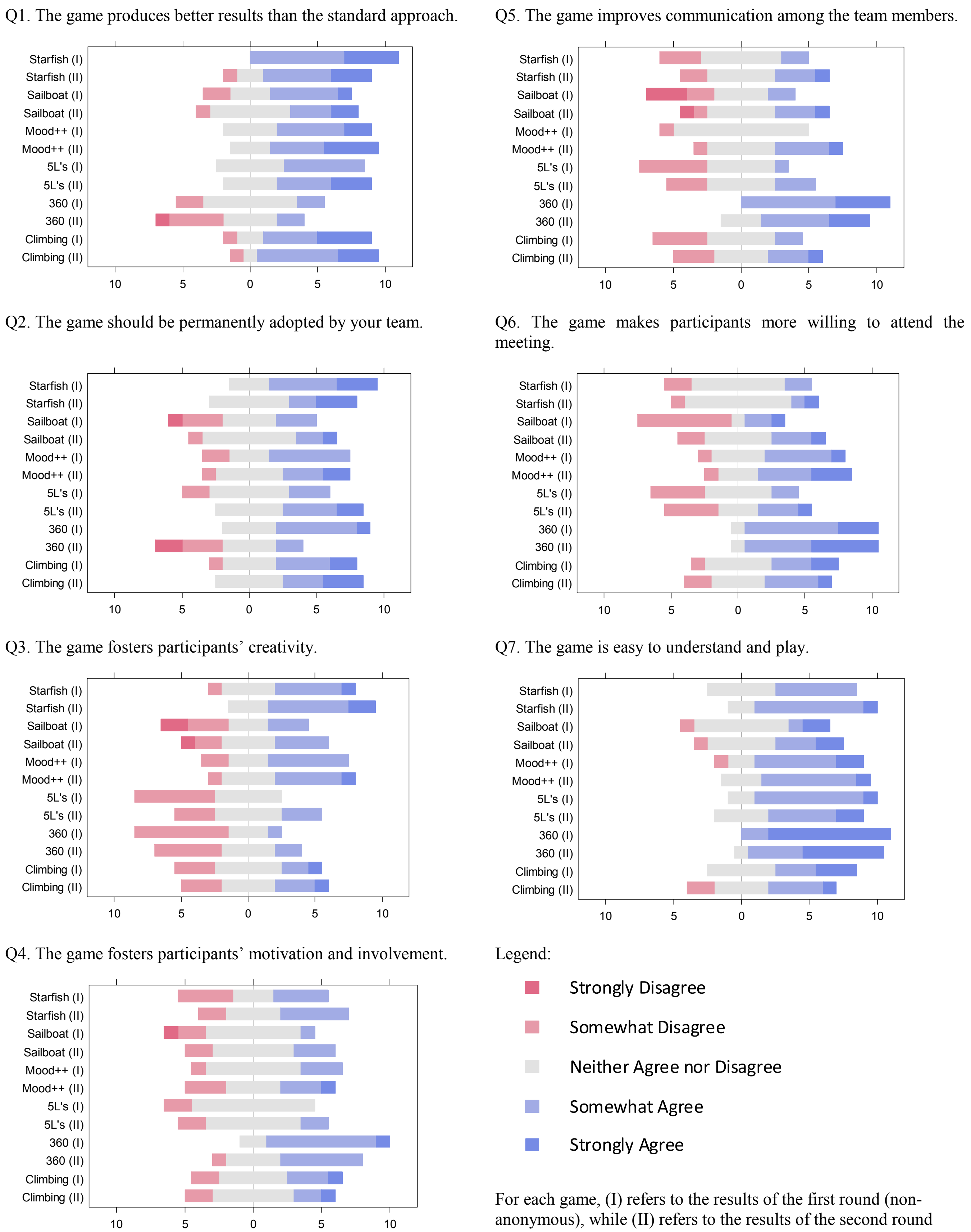
C. Action taking

Before a game was run for the first time, it was presented to the team. When introducing the first game, which was Sailboat, the participants felt that drawing on the board was like playing in kindergarten, so it was a waste of time. Nevertheless, during the meeting, they changed their minds and began to see the value in the game. After each game session, we issued a questionnaire to collect feedback from the participants. At the end of the day, the results were analyzed and discussed with the team.

D. Evaluating and specifying learning

Fig.1 summarizes the questionnaire results. The responses were made on a five-point Likert scale. Overall, **Starfish** and **Mood++** performed the best and were admired for covering all important aspects of the Sprint Retrospective. However, the majority of the participants agreed that all games except **360° Appreciation** produced better results than the standard approach and thus should be permanently adopted by the team. Nevertheless, as for the **Sailboat** adoption, the opinions were divided, because the game was very time-consuming. Unsurprisingly, **Mountain Climbing**, which is quite similar to **Sailboat**, was rated better in all aspects. In contrast to the remaining games, **360° Appreciation** cannot be considered as a standalone retrospective, so it received the lowest grades regarding the first question, but since it positively affects all other aspects except creativity, the team decided that the game should be permanently adopted.

When it comes to creativity, only Starfish and Mood++ performed well. In turn, *Motivation & Involvement* as well as *Communication* were boosted mainly by **360° Appreciation**. It is reasonable, as the game is to praise other teammates, which helps team members to be socially connected with the team and makes the collaboration easier. As for other games, the responses were divided between supporters, opponents and undecided. Furthermore, **Mood++**, **360° Appreciation** and **Mountain Climbing** made most of the participants more willing to attend the meeting. Finally, *Easiness of playing & Understandability* was the most positively rated aspect of all games.



For each game, (I) refers to the results of the first round (non-anonymous), while (II) refers to the results of the second round (with the preliminary anonymous idea generation).

Figure 1. Aggregated results

Although collaborative games are claimed to encourage equal participation, we observed that when the contributes were non-anonymous, senior developers dominated meetings by talking more and exerting control over the retrospective agenda. On the one hand, this kept the retrospectives on-task and focused; on the other hand, junior team members contributed less, because they refrained from disagreeing with higher-status others. Accordingly, in general, all games except **360° Appreciation** benefited from anonymity. Unsurprisingly, since **360° Appreciation** allows team members to express only positive feedback, the authors of this feedback preferred to be known. The aspects that gained the most were *Creativity* and *Communication*, while as for other aspects the improvements were rather slight.

V.Conclusions

This paper reports on an Action Research project conducted in Intel Technology Poland. The research objective was to replicate the previous studies on game-based retrospectives and to investigate whether the preliminary anonymous idea generation mitigates negative social influences. On the other hand, the practical objective was to audit and improve the working practices in the participating team. We confirmed that game-based retrospectives produce better results than standard retrospectives and lead to a variety of measurable societal outcomes. Accordingly, the team has continued to run them since the project finished. Taking into account the results from [15, 22] and this work, the most successful game is Starfish. Besides, we observed that higher status team members dominate meetings even though collaborative games are used. Our results also suggest that game-based retrospectives benefit from anonymity. The only exception is 360° Appreciation, which, in fact, cannot be considered as a standalone retrospective. Nevertheless, we intend to further investigate the effect of anonymity in a controlled experiment with settings similar to [3, 14, 20, 26].

References

[1] Avison, D., Baskerville, R., Myers, M.: Controlling Action Research Projects. In: Info. Technology and People, Vol. 14(1), pp. 28-45, 2001

[2] Babb, J., Hoda, R., Norbjerg, J.: Embedding Reflection and Learning into Agile Software Development. In: IEEE Soft., vol. 31(4), 2014

[3] Davis, J., Zaner, M., Farnham, S., Marcjan, C., McCarthy, B.P.: Wireless brainstorming: overcoming status effects in small group decisions. In: HICSS 2003. doi: 10.1109/HICSS.2003.1173812

[4] Diehl, M., Stroebe, W.: Productivity loss in brainstorming groups: Toward the solution of a riddle. In: Journal of Personality and Social Psychology, 53(1), 497–509, 1987

[5] Gaikwad, P.K., Jayakumar, C.T., Tilve, E., Bohra, N., Yu, W., Spichkova, M.: Voice-activated solutionsfor agile retrospective sessions. In: Procedia Computer Science, vol. 159, 2414–2423, 2019

[6] Hanslo, R., Tanner, M.: Machine Learning Models to Predict Agile Methodology Adoption. In: 2020 Federated Conference on Computer Science and Information Systems (FedCSIS'20), Sofia, Bulgaria, 2020

[7] Hanslo, R., Vahed, A., Mnkandla, E.: Quantitative Analysis of the Scrum Framework. In: Przybyłek A., Morales-Trujillo M. (eds) Advances in Agile and User-Centred Software Engineering. LASD 2019, MIDI 2019, Lecture Notes in Business Information Processing, vol 376. Springer, Cham. 2020. doi: 10.1007/978-3-030-37534-8_5

[8] Hanslo, R., Mnkandla, E., Vahed, A.: Factors that contribute significantly to Scrum adoption. In: Federated Conference on Computer Science and Information Systems (FedCSIS), Leipzig, Germany: IEEE: 821–829. 2019. doi: 10.15439/2019F220

[9] Ilyés, E.: Create your own agile methodology for your research and development team. In: 2019 Federated Conference on Computer Science and Information Systems, Leipzig, Germany, 2019

[10] Jarzębowicz, A., Ślesiński W.: What is Troubling IT Analysts? A Survey Report from Poland on Requirements-related Problems. In: KKIO 2018. Advances in Intelligent Systems and Computing vol. 830, pp. 3-19, Springer, 2018. doi: 10.1007/978-3-319-99617-2_1

[11] Jarzębowicz, A., Sitko, N., Communication and Documentation Practices in Agile Requirements Engineering: A Survey in Polish Software Industry. In: SIGSAND/PLAIS 2019, LNBIP vol. 359, pp. 147-158, Springer, 2019. doi: 10.1007/978-3-030-29608-7_12

[12] Mich, D.: Integrating innovative collaborative games into Scrum development. MSc thesis, Gdansk University of Technology, 2018

[13] Miler J., Gaida P.: Identification of the Agile Mindset and Its Comparison to the Competencies of Selected Agile Roles. In: Przybyłek A., Morales-Trujillo M. (eds) Advances in Agile and User-Centred Software Engineering. LASD 2019, MIDI 2019. LNBIP, vol 376, Springer, 2020. doi: 10.1007/978-3-030-37534-8_3

[14] Nunamaker, J.F., Applegate, L.M., Konsynski, B.R.: Facilitating Group Creativity: Experience with a Group Decision Support System. In: J. of Management Information Systems, vol. 3(4), pp. 5-19, 1987

[15] Ng, Y.Y., Skrodzki, J., Wawryk, M.: Playing the Sprint Retrospective: A Replication Study. In: Przybyłek A., Morales-Trujillo M. (eds) Advances in Agile and User-Centred Software Engineering. LASD 2019, MIDI 2019. LNBIP, vol 376, 2020

[16] Ozkan, N., Gök, M.Ş., Köse, B.Ö.: Towards a Better Understanding of Agile Mindset by Using Principles of Agile Methods. In: 2020 Federated Conference on Computer Science and Information Systems (FedCSIS'20), Sofia, Bulgaria, 2020

[17] Paulus, P.B, Dzindolet, M.: Social influence, creativity and innovation. In: Social Influence, 3:4, 228-247, 2008

[18] Przybyłek, A.: The Integration of Functional Decomposition with UML Notation in Business Process Modelling. In: Advances in Information Systems Development, Vol 1, pp. 85–99, 2007

[19] Przybyłek, A.: A Business-Oriented Approach to Requirements Elicitation. In: 9th International Conference on Evaluation of Novel Approaches to Software Engineering, Lisbon, Portugal, 2014

[20] Przybyłek, A.: An empirical study on the impact of AspectJ on software evolvability. In: Empirical Software Engineering, vol. 23(4), pp. 2018–2050, August 2018, doi: 10.1007/s10664-017-9580-7

[21] Przybyłek, A., Olszewski, M.: Adopting collaborative games into Open Kanban. In: 2016 Federated Conference on Computer Science and Information Systems (FedCSIS'16), Gdansk, Poland, 2016

[22] Przybyłek, A., Kotecka, D.: Making agile retrospectives more awesome. In: 2017 Federated Conference on Computer Science and Information Systems (FedCSIS'17), Prague, Czech Republic, 2017

[23] Przybyłek, A., Kowalski, W.: Utilizing online collaborative games to facilitate Agile Software Development. In: 2018 Federated Conference on Computer Science and Information Systems (FedCSIS'18), Poznan, Poland, 2018, doi: 10.15439/2018F347

[24] Przybyłek, A., Zakrzewski, M.: Adopting Collaborative Games into Agile Requirements Engineering. In: 13th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE'18), Funchal, Madeira, Portugal, 2018

[25] Wawryk, M., Ng, Y.Y.: Playing the Sprint Retrospective. In: 2019 Federated Conference on Computer Science and Information Systems (FedCSIS'19), Leipzig, Germany, 2019. doi: 10.15439/2019F284

[26] Weichbroth, P.: Facing the brainstorming theory. A case of requirements elicitation. Studia Ekonomiczne, 296, 151-162, 2016

[27] Weichbroth, P.: Delivering usability in IT products: empirical lessons from the field. In: International Journal of Software Engineering and Knowledge Engineering, 28(07), 1027-1045, 2018

[28] Zakrzewski, M., Kotecka, D., Ng, Y.Y., Przybylek, A.: Adopting Collaborative Games into Agile Software Development. In: Damiani et al. (eds) ENASE 2018. Comm. in Computer and Information Science, vol 1023. Springer, 2019. doi: 10.5220/0006681900540064

DOI: 10.15439/2020F47
ISSN 2300-5963 ACSIS, Vol. 21

Harmonizing IT Frameworks and Agile Methods: Challenges and Solutions for the case of COBIT and Scrum

Necmettin Ozkan
Information Technologies Research and Development Center Kuveyt Turk Participation Bank Kocaeli, Turkey
necmettin.ozkan@kuveytturk.com.tr

Ayca Kolukisa Tarhan
Computer Engineering Department Hacettepe University Ankara, Turkey
atarhan@cs.hacettepe.edu.tr

Burak Gören
Vice President of Agile Governance, DevOps and Lead Agile Coach
Akbank Kocaeli, Turkey
burak.goren@akbank.com

İsmail Filiz
Problem, Service Level and Quality Assurance Manager
Akbank Kocaeli, Turkey ismail.filiz@akbank.com

Enis Özer
Agile Coach Akbank Kocaeli, Turkey
enis.ozer@akbank.com

***Abstract*—Information Technology (IT) is a complex domain. In order to properly manage IT related processes, several frameworks including ITIL (Information Technologies Infrastructure Library), COBIT (Control OBjectives for Information and related Technologies), IT Service CMMI (IT Service Capability Maturity Model) and many others have emerged in recent decades. Meanwhile, the prevalence of Agile methods has increased, posing the coexistence of Agile approach with different IT frameworks already adopted in organizations. More specifically, the pursuit of being agile in the area of digitalization pushes organizations to go for agile transformation while preserving full compliance to IT frameworks for the sake of their survival. The necessity for this coexistence, however, brings its own challenges and solutions for harmonizing the requirements of both parties. In this paper, we focus on harmonizing the requirements of COBIT and Scrum in a same organization, which is especially challenging when a full compliance to COBIT is expected. Therefore, this study aims to identifying the challenges of and possible solutions for the coexistence of Scrum and COBIT (version 4.1 in this case) in an organization, by considering two case studies: one from the literature and the case of Akbank delivered in this study. Thus, it extends the corresponding previous case study from two points: adds one more case study to enrich the results from the previous case study and provides more opportunity to make generalization by considering two independent cases.**

***Index Terms*—agile, Scrum, information technology, COBIT, challenge, solution**

I. Introduction

THERE are many IT (Information Technology) process frameworks to guide organizations in their compelling environments. In order to properly manage IT related processes, several frameworks including ITIL (Information Technologies Infrastructure Library), COBIT (Control OBjectives for Information and related Technologies), IT Service CMMI (IT Service Capability Maturity Model) and many others have already emerged in recent decades. They commonly poses capabilities with a disciplined, sustainable, controlled, standard and consistent way of working for IT. Recently, there also exist Agile Software Development (ASD) methods and frameworks to add more agility to organizations in their complex software development processes. The use of such broad frameworks with their diverse and occasionally different characteristics in a same organization has a high possibility of emergence, posing some challenges on the side of practitioners.

As one of them, providing organizations mainly a level of control and assurance, COBIT has a domination in IT field for many years. With its more than 40 international integrated standards, it is a framework providing IT governance to help in delivering value from IT and managing risks associated with IT [4]. Many countries listed in [5] including USA, Canada, Australia, India, Japan, Brazil, Poland, Romania, South Africa, Turkey facilitate COBIT for their public sectors, governmental agencies and regulatory bodies. In particular, in Turkey, since May 2006, the banks have started to use COBIT, widely with control and audit ground. This control and audit based usage is the main driver to keep the COBIT v.4.1 as the valid version for the banks in Turkey and to cover in this particular study, rather than the further versions.

On the other side, adoption of the ASD is increasing, especially with its most widely used framework, Scrum [6]. Considered the coverage of Scrum and COBIT and Scrum's penetration especially in large organizations [2, 7], a coexistence of them in a same organization has a possibility of emergence [1]. However, the ASD approaches, defined with the ability to respond to change, have generally been regarded as contrary to the traditional (heavyweight, disciplined, predictive, plan-driven) approaches due to opposing viewpoints [8, 9]. COBIT, as a representative of heavyweight, disciplined, predictive, plan-driven approaches, has no exception in this regard. It has a co-occurrences and similarities with the rationalized, engineering-based approaches. Thus, melting COBIT and Scrum in the same organization can be intriguing yet challenging (especially when a full compliance with COBIT is required as in the case of the banks in Turkey) [1, 2], as shown with the results from this study.

Despite the challenges, the charm of being agile attracts organizations to go for the Agile transformations [2, 7] and manifests a need to study the possible challenges of and solutions for COBIT-Scrum coexistence to guide the organizations. COBIT defines itself as a framework and allows tailoring with specific needs of organizations. However, in some countries such as Turkey, COBIT is applied in a strict way such as a regulation rather than a guide, with less chance to tailor. One way or another, there is at least a need to address to which parts of COBIT are required to pay attention in a Scrum implementation.

In this study, we search for a proper coexistence of the ASD – as demonstrated by Scrum – with the realities of IT needs in an organization – as predefined by COBIT. Accordingly, this paper focuses on the identification of challenges and proposing solutions for the identified challenges for a possible Scrum implementation within a COBIT-driven environment, by considering two case studies: one from the literature and the case of Akbank delivered in this study. It extends the works of [1-3] from two points: adding one more case study, to enrich the results from previous case study and providing more opportunity to make a generalization by considering two independent cases. More specifically, this study has two research objectives, as the following:

- RO1: To elicit and identify challenges that arise in harmonizing the requirements of COBIT and Scrum,
- RO2: To propose and discuss solutions for the identified challenges.

The remaining of this work is organized as follows. We provide the related works in Section II. The research methodology is delivered in Section III. In Section IV, the challenges identified as relevant to RO1 are communicated. In Section V, the case study is delivered. In Section VI, solution suggestions relevant to RO2 are provided. In Section VII, the subject is evaluated with discussions. Finally, in Section VIII, conclusions and future work are delivered.

II. RELATED WORKS

In order to reach a set of related works, a search with the keyword of "Scrum COBIT" was conducted throughout of libraries of ACM Digital Library, IEEE Xplore, Web of Science, Science Direct and Scopus, with their default search settings and without any specific filter in the year range. Relatively a small number, 42 peer-reviewed works (from workshops, conferences, journals and book chapters) in total were returned in English and examined through their titles and, where necessary, abstracts and/or full texts. It is seen that some works covers COBIT and Scrum different from yet within related context to our study. Among them, Aguillar et. al [10] comes with a case study where the COBIT 5.0 Process Assessment Model was used to identify processes that need improvements within the studied company. It was then applied to Scrum to integrate internal activities and processes. Study [36] aims to eliminate some known challenges of COBIT 5 adoptions by providing a Scrum based methodology and demonstrates better results in terms of commitment from top management and alignment. Study [38] proposes a model type artifact for software development governance, mainly based on COBIT 5 and Scrum. Study [39] use COBIT through identification, description and evaluation of general roles and structures related to the notion of project, to unite the project related entities with Scrum that are not explicitly included in Scrum. In the study [40], the authors use the indicators of COBIT for measuring Scrum-based software development.

Directly related to our study, studies of [1, 37] aim to identify potential challenges in a possible Scrum and COBIT coexistence in an organization, without providing any solutions to the challenges. Studies of [2, 3] provide the experiences based on a single case of a bank in Turkey that operates with Scrum and COBIT co-occurrence, and therefore has some natural limitations and a specific window to the case. It should be noted that the challenges identified in [1, 2, 3, 37] are considered, refined, improved and justified in this study. The solutions proposed by [2] and [3] are considered, refined and improved, especially by another case study, in order to eliminate the context-based distortions of that particular case and aims to make the subject more generalizable.

III. RESEARCH METHODOLOGY

It is a fact that Scrum more or less touches all the COBIT processes that are 34 in number, with 210 control objectives and 990 control practices. As COBIT presents a huge area to work with, it is a must to focus on the processes that Scrum affects directly. In identifying the related COBIT processes with a direct and intense relevance with Scrum, the study of Ozkan [1] was taken into consideration as it provides the comprehensive COBIT process list in this regard and the reasons to select them.

Among the processes, regarding PO1 (Define a Strategic IT Plan), the study of Ozkan [1] covers two topics addressed in PO1.1 (IT Value Management) and PO1.2 (Business-IT Alignment) respectively. The mentioned issue within the scope of "IT Value Management" is the early warnings of any deviations from the plan, including cost, schedule or functionality, which can be possibly met by the frequent feedback and high transparency mechanisms in Scrum. As the second issue, establishing fair, transparent, repeatable and comparable evaluation of business cases and providing the business and IT alignment and integration as pointed out in COBIT are among the common issues regardless of the development method applied. It thus leads to exclude PO1 process from the list. The fact that both case studies covered in this study do not provide any clues about these matters reinforces this exclusion. The list of the remaining processes includes PO4 (Define the IT Processes, Organization and Relationships), PO7 (Manage IT Human Resources), PO10 (Manage Projects), AI1 (Identify Automated Solutions), AI2 (Acquire and Maintain Application Software), AI4 (Enable Operation and Use) AI7 (Install and Accredit Solutions and Changes).

After the identification of these seven COBIT processes, a comprehensive and thorough investigation of the challenges was done by the first author. In doing so, a profound reading of the Agile values and principles [20] and the Scrum Guide [11] on one side and COBIT 4.1 [4] on the other side was conducted. If further detail is needed for a particular COBIT process, COBIT Control Practices [19] sustaining each related control objectives with detailed control practices were used. For the notations used in this study, control objectives are remarked as in "AI2.1" and control practices as in "AI2.1.3" with parentheses in the appropriate places in the content. In identifying the challenges of the processes, additionally, the studies of [1], [2], [3] and [37] were considered. Besides, the case study of Akbank in Turkey was used to justify and, if needed, to update the list of challenges, by asking to the interviewees if the current identified challenges are valid and if there is any additional one. With this justification in the case study, one item - Alignment of the Audit Perspective with Agile Approaches - was added and the rest of the list was maintained. The final set of the challenges are delivered and explained in the next section.

After the identification of the challenges, the possible solutions were proposed based on, but not limited to, two case studies; one delivered in the studies of Ozkan, Tarhan and Kucuk [2] and Ozkan [3] and the second one from the case of Akbank delivered in our study. The first case was selected as it delivers the most comprehensive results, as far as we know, in the context of our study. Additionally, the first author's experiences, apart from these two case studies, were involved in proposing the final solutions. In the identification of challenges and corresponding solutions, however, this paper aims to eliminate the context-based distortions of these particular cases and to make the subject more generalizable.

For the case of Akbank delivered in our study, semi-structured interviews were conducted with three people from the bank by the first author of this paper. Being also the co-authors of this paper, one of the three involved people is the Vice President of Agile Governance, DevOps and Lead Agile Coach, the second one is an Agile Coach and the last one is the Quality Assurance Manager involving with COBIT intensively. Two meetings, which lasted three hours in total, were held, and during the meetings, the identified challenges were conveyed by the first author, and they were asked whether these challenges are valid for their case and whether there are any other challenges. The first author did not mention any solution suggestions in order to avoid bias, and they were asked to convey their own solutions to these challenges. The interview contents were noted down by the first author and then sent to the interviewees for the confirmation and then necessary updates were made.

IV. CHALLENGES

This part of the paper communicates the descriptions of challenges identified. The source of the challenges falls into two categories: 1) those coming from the alterations resulted from a Scrum adoption and COBIT has an emphasis on the same points that Scrum alters. This type does not necessarily create a conflict between the two sides yet organizations should pay extra attentions to meeting COBIT requirements especially during the Scrum transformation. This type of challenges is called as "concern" in this work. 2) The second type of challenges are those that COBIT and Scrum have different perspectives on. This type is a matter of a clear "conflict" that organizations should deal with. The category which is classified as "concern" is likely to be more in terms of number. Here, it was aimed to give a place to those that are fundamental and primary.

A. Steering Committee (Conflict)

PO4 (Define the IT Processes, Organization and Relationships) points out one or more steering committees to determine prioritization of IT-enabled investment programs. Schwaber and Sutherland [11] state for the same issue: "The Product Owner is one person, not a committee. The Product Owner may represent the desires of a committee in the Product Backlog, but those wanting to change a Product Backlog item's priority must address the Product Owner." According to the Scrum Guide, Product Owner is the ultimate decision point in prioritization of the projects, and it is the steering committee according to COBIT.

B. Segregation of Duties (Conflict)

Scrum recognizes no specific titles inside the development team other than developer, giving the accountability to the development team as a whole [11]. From the window of COBIT, this restricts the controls to preclude full segregation of duties as mentioned in PO4.11 such as in conducting functional tests [1]. Similarly, AI7.6.1 clearly states that "ensure that the testing is designed and conducted by a test group independent from the development team". Additionally, in (AI2.8) Software Quality Assurance, COBIT states that"...ensuring that reviewers are independent from the development team".

C. Human Resource Management (Concern)

Accountabilities and responsibilities of functions of teams related to personnel recruitment, retention (PO.7.1), termination (PO.7.8), competencies management (PO7.2), adhering to codes of ethics (PO7.3), dependence upon individuals (PO7.5), reliance on a single individual performing a critical job function (PO4.13) in the context of tacit knowledge, performance evaluation (PO7.7) and administrative operations largely directed and managed by line managers in the traditional methods should be addressed in Scrum [1]. However, Scrum does not specify techniques to address the human side of software development [41]. Regarding the performance evaluation, showing confidence in producing team's own data to reflect their own performance possibly in a way for the sake of personal favors may become a dilemma. Career path development (PO7.2.4) is a matter of Scrum that provides a flat structure of organization for teams, especially for those regarding people management experience valuable [1].

According to [1], regarding PO4.4.3, the workload and resource capacity management among and inside the Scrum teams has not been detailed out in terms of who is responsible for deciding on the staff capacity of the teams, in order to response adequately to business needs. PO.4.5.2 extends this issue by adding "the use of external contractors and flexible third-party" that may not have the Scrum capabilities. AI2.7.5 highlights the same issue by stating that "when third-party developers are involved with the applications development, establish that they adhere to contractual obligations and organizational development standards…"

D. Project Management (Conflict)

COBIT poses a traditional project management approach with requirements for assessing of schedule, budget and scope of projects (PO10.6.3), reviewing and approving cost, schedule, scope and quality changes in the project baseline (PO10.11) by key stakeholders and project sponsors (PO10.5.2), integrated project plan (PO10.7) with work breakdown structures and identification of critical paths, forming and acquiring a project team with its competent staff members (PO10.8) and a project governance structure including project office and project manager roles (PO10.3). However, in Scrum, the notion of project, project management and project manager role are deliberately left blank [12].

E. SDLC (Conflict, Concern)

As a "*Conflict*", in the SDLC processes in COBIT, it is expected to create certain document contents with a certain order (for instance, preparing detailed design before coding is initiated) and to approve them by relevant parties. Thus, COBIT has a potential to make Scrum's each sprint to resemble mini-waterfall in flow [2] by posing requirements of some check-points and a sequential flow in a usual iteration. Within the coexistence COBIT and Scrum, iterative and incremental development results in iterative and incremental development of the relevant documents/contents for multiple times including requirements and feasibility decision and approval (AI1.4), high level design (AI2.1), detailed design (AI2.2), design of application security and availability (AI2.4) and application control and auditability (AI2.3), test plans (AI7.2), (AI7.3) and implementation plan (AI7.3) along with their formal approvals by related business process owners and IT stakeholders (AI1.4.1), (AI2.1.5), (AI2.2.11), (AI2.9.3), (AI7.2.8), (AI7.3.2).

As a "*Concern*", when combining such COBIT requirements with the frequency of (relatively short) Scrum sprints, naturally, the continuous integration probably including performance, stress, usability, security, system, integration, user acceptance, operational readiness, backup and recovery tests (AI7.2.5) "within a secure test environment representative of the planned operations environment relative to security, internal controls, operational practices, data quality and privacy requirements, and workloads" (AI7.4) may be overloading and time consuming even with a right balance between automated scripted tests and interactive user testing (AI7.6.3).

Similarly, iterative and incremental growth of the system calls for the creation and integration of complete, accurate and usable supporting documents (AI4.2) with promptly updates to the existing environment in production for the use of end users (AI4.3), operations and support staff (AI4.4), and business management (AI4.2) along with the required trainings (AI7.1.2) [2].

F. Documentation (Concern)

Documentation takes a fundamental role for COBIT as a means of storing, sharing, conveying, replicating and backing-up knowledge, planning, codifying and standardizing of practice, and creating logs for further use [1]. On the other hand, although not stated theoretically, in practice, Agile approaches discourage documentation and they may consider documentation as a secondary activity [13].

G. Alignment of the Audit Perspective with Agile Approaches (Concern)

Depending on all these points above, the relevant methods of audit and control teams must change accordingly. This brings along the need for an alignment in the relevant audit and control perspectives and processes to the Agile approaches and vise verse.

V. CASE STUDY: AKBANK IT

Akbank is a leading bank in Turkey, which has long been applying Scrum in its IT. Akbank IT runs with 1400 people who are practicing Scrum with 161 Scrum Teams, 12 Product Group domains, 161 Scrum Masters, 109 Product Owners and four internal Agile Coaches. Agile Studio which is an in-house Agile coaching team and Agile Leaders team which is a transformation sponsor team at the highest-level lead for the Agile transformation and foster agility within the organization.

To justify the current challenges, identify additional ones and to propose solutions to the final set of the challenges, the first author of this paper conducted an interview with three experts from Akbank where all people participated. After reaching the final set of the challenges that were delivered in the previous section, the experts conveyed their own solutions to these challenges. The following sub-sections provide solutions applied by Akbank to the determined challenges.

A. Steering Committee

There is a two-stage flow to prioritize the projects. Running in the quarterly period, at the first stage, the senior executives of each business unit promote the projects they want to develop, to all other business unit participants. Product owners are not a member of this committee, since the product owners have a perspective on the product line and do not provide a comprehensive perspective that cuts multiple products and services horizontally. At this level, an enterprise level alignment and orientation towards the same goal are assured through some methods such as linking projects to the annually defined strategy areas and determining value and expected contributions of the project proposals in tangible terms. Towards the end of this stage, a prioritized project list is achieved, by means of the votes of all business unit stakeholders.

Following the ordering of the projects, at the second stage, the prioritized project list is conveyed to IT. With the participation of the product owners, portfolio managers, product group leaders, a meeting is conducted for the alignment and collaboration between the POs and discussing priorities, risks and inter-team dependencies of teams. In this meeting, the project list is evaluated in terms of capacity, high level planning and product management perspective. A master product owner is determined for each project and the dependencies between the relevant products are identified and evaluated. Finally, an agreement is reached between the relevant product owners, and if necessary, the orders of the projects is revised accordingly. At the end of the quarterly period, the ranking at the first stage is re-operated with a new list of that moment to response to the needs of the current time. The loop thus repeats, with the new list.

B. Segregation of Duties

The accountability with a customer-request-based testing is appointed to one specific person in the team to ensure that tests are carried out appropriately. This person can wear any role from the development team (developer, analyst, tester, etc.). Naturally, the person performing the functional tests should be different from the person who codes. For the software quality assurance, there is a separate team outside the development teams to control the flow and creation of documentations throughout the end-to-end process. Even if the quality assurance team is located online with the development teams, their approval is just before the transition to the production environment. The main reason for the positioning of this stage is that the content changes dynamically during the development, reaching its final state when getting closer to the transition.

C. Human Resource Management

There is no team manager in the new structure, yet there still exist department managers, newly positioned as product group leaders. Although the Human Resource Department exhibits no radical changes in its current practices after the transformation, the development teams have more voice in the human resource related processes. For instance, the teams can design their candidate profiles and run the recruitment process themselves along with the Human Resource Department.

The career path to the conventional line management have been replaced with the new path built on the expertise of the team members. In this way, the career path has been widened instead of narrowing down at a management level. There are two main legs feeding the performance management: team-based and individual-based evaluation. The majority of the score comes from the team-based score, and most of this team score is fed from metrics such as problem cases, interruption records. Additionally, the customers and associated teams evaluate the whole team as one unit. For the individual evaluation inside the teams, as the second leg, the department managers assess the individuals and additionally the members of the team assess the other members within his/her team. The team-based evaluation taking more weight on the final score is combined with individual-based evaluation to create the ultimate output. Throughout the all process, the targets are assigned team-based, customer-focused and on the final value.

In capacity management, considering teams' capacity as master, the teams are fixed around the products in terms of structure and do not organize dynamically around business projects. In case of a capacity shortage of the teams meaning that they will not satisfy business priorities for a particular period, some options emerge; if this shortage indicates a permanent situation, the static team structures are arranged accordingly. If the particular supply-demand imbalance is for a temporary case and an action is required for it, a separate temporary team is formed with the people gathered from the teams, and disbanded when the development of the project is finished. If the imbalance seems for a long-term situation, the product teams are re-arranged accordingly.

D. Project Management

The project notion is maintained, but there is no project manager role. If the project is within the scope of a single development team, the relevant product owner follows the project. In case of the distribution of the project to more than one team, the PO of the main team follows the project. In new projects, the first connection between the customer and the development team is established with the mediation of the portfolio managers. Afterwards, it is ensured that the product owner and the customer establish a one-to-one relationship between them. The portfolio managers then take a position that supports customers mostly about the progress of their projects in the portfolio.

E. SDLC

The SDLC processes flow over the product-based teams. For this reason, the Sprint 0 step, which is suitable in the project-based formations, is not located. The approvals of the design documents given by the business and IT have been taken to the end of the releases (instead of sprints). There is no sprint-based approvals of the development documents because during the sprints, the customers view the product itself, not via a proxy of it with some documents. Until the tool support receives, the business unit approvals are being given by the product owners. For the IT side, the approvals given by a member from the development team is sufficient. Additionally, the approvals given by the central bodies such as security, enterprise architect and infrastructure teams remain. A DevOps team is located to manage certain central operations such as code base management, promotion to production and as well as to penetrate the DevOps culture into the organization.

F. Documentation

There are mainly two tools in supporting documentation: Jira and Microsoft TFS. Jira manages the pipeline where the product is offered to the customer. TFS is like the kitchen of the developers. With the support of these tools, the document contents expected to be produced are followed up.

G. Alignment of the Audit Perspective with Agile Approaches

In finding the middle way between internal control systems and Agile approaches, it is helpful to expose that sometimes COBIT and Agile methods propose different ways to eliminate the same concern. In this regard, Agile methods in certain areas provide more advantages compared to the COBIT proposals. For example, for the software development processes, the associated COBIT controls with the purpose of not deviating from the customer requirements are met with the short iterations in Scrum. In such cases, the corresponding control objectives lose their meaning in the context of agile working. However, some controls (for example, obtaining approval from customers for the development documents) should stay valid and be maintained. For the alignment in this context, many trainings were provided to the internal control and audit teams to show and persuade for how the main concerns of the corresponding control or audit processes are met by the agile way of working.

VI. SOLUTIONS

This section suggests the solution proposals for the identified challenges in a unified view based the two case studies: the case of Akbank and the other one conveyed in the studies of [2, 3]. It is noted that while the solutions proposed are based on the case studies, they are not limited to them; the first author’s experiences gained during the case study delivered in [2, 3] were also integrated in building the solution proposals.

A. Steering Committee

COBIT mentions who constitutes the steering committee in general by stating the members from IT and business and then the product owners can be a member of the committee. Another point is that the fair position of POs and operations of processes under their reasonability play very critical role [1]. Similarly, PO4.4.2 (Organizational Placement of the IT Function) emphasizes to “define and fund the IT function in such a way that individual user group departments cannot exert undue influence over the IT function and undermine the priorities agreed upon by the IT steering committee”. With a structure of the balanced power of relevant parties reduces the risk POs bear who are solely one person according to the Scrum Guide [11]. Or, in a more-Scrum-way, “the product owner may represent the desires of a committee in the product backlog, but those wanting to change a product backlog item’s priority must address the product owner”.

B. Segregation of Duties

For the sake of the nature of the work, a developer should not be allowed to make functional tests for his/her code, as s/he can naturally be blind to what s/he codes. However, it does not mean to follow literally the COBIT’s suggestions in this regard. COBIT regards the test groups and development teams as separate bodies, which is not the case in Scrum teams. Similarly, the teams as a whole can be responsible for their software quality assurance activities as the process assurance covers the team wholly. Alternatively, the team may delegate this responsibility to another body outside of the team if a central body dedicated to the software quality assurance reviews is located, as in the case of Akbank.

C. Human Resource Management

Regarding the operations of human resource management including personnel recruitment, retention, termination, competencies management, adhering to codes of ethics, it is possible to delegate those activities to the certain parts of the organization, as seen in the both cases. The development teams will probably get more responsibilities than of those in the traditional approach. Such a distribution also helps to avoid possible narrow throats. Additionally, decision makers can be supported with inputs from evaluations of the teams inside and from parties around the teams. For the career path development, it can be possible to add positions not in hierarchy based on the new way of working which can be promoted by evaluating the experience, knowledge, skills and contributions, as proposed by [2].

Scrum is powerful in knowledge sharing and the rotation of team members enables the knowledge not monopolized by a few roles [14]. As mentioned in the “Documentation” section of this study, utilizing documentation, process and tool capabilities will also help in minimizing dependence upon key individuals such as PO who performs critical operations including maximizing the value (PO4.13 in relevant), managing relationships between IT and key stakeholders (PO4.15.2 in relevant). Considering PO is a sole person, as pointed out in (PO4.13.4), Scrum should additionally assure this critical person’s appropriate availability during time-off periods, vacations and leaves of absence (PO7.5) [1].

To facilitate IT functions to support the business with appropriate and flexible resource arrangements, even when involving external contractors and third parties, the concept of project and project management can be injected into Scrum in an agile way. The details of proposed solution are elaborated in “Project Management” section of this paper.

The personal-level performance measurement should not be preferred in Scrum because it inhibits the team spirit. In this manner, when the team-level performance measurement is preferred, the following points should be considered [2]:

- Teams are not fully isolated from their environments and there are inter-team boundaries at many cases. Thus, a special attention should be paid to identify and, if possible, to separate the borders between the teams in the performance measurement.
- Be aware of the dilemma of using the metrics both for performance measurement context and improving teams themselves meanwhile.
- It may not be always visible to identify low and high level performance of individuals in a team from a point outside of the team, then let the team identify them transparently.
- Intend to strike a balance between maintaining the team spirit and providing individual measurement visibility.
- When appropriate, use assessment of people instead of measurement of them.

D. Project Management

IT currently is dominated by a process-oriented approach proposed by COBIT, service-oriented approach proposed by ITIL (Information Technologies Infrastructure Library), and a project-oriented approach proposed by PMBOK [15]. Although the product-oriented development is de-facto in the industrial production, the case is different in the IT field. The thinking has shifted from the pure product focus to a combination of service and product [16], and the pure product concept is prone to disappear behind service, especially in the banking sector. Unlike the context of the industry, in the software field, the development process is complex and dynamic and it deserves an interest at least as much as the result itself. As a result, regardless of the frameworks, to manage this complex and dynamic development processes, the concept of project inevitably is a living phenomenon in IT [35]. The response of the Agile world to this fact reinforces it: tens of thousands of results including one by Schwaber [17], returning from Google Scholar search with "agile project management" keyword, even for the time being. In parallel, in the both case studies, the project phenomenon was seen to exist.

This study comes with the idea of having a proper project definition aligned with PMBOK [18] which states: “project is a temporary endeavor undertaken to create a unique product, service or result”. The term temporary means it has a defined beginning and end in time, and therefore defined scope and resources. This definition can fit with “Scrum-type-project-management”, in which the defined beginning and end in time, scope and resources can be dynamic with rules, policies etc., rather than with static numbers, at the run time based on (dynamic) value of the (other) projects.

Apart from the presence of the project phenomenon, organizations may prefer the project or product oriented team structures. While in the case study of [2, 3], both of these approaches are preferred, resulting in the product and project team existence together, while Akbank, although the phenomenon of project continues to exist, has settled only the product teams. Each of these options has its own advantages and disadvantages.

In fact, the Scrum approaches do not prefer a team structure shaped around the project concept, since project offers temporary teams that may damage the team spirit and thus not providing opportunity to establish a lasting basis for trust and performance of the teams. On the other hand, the management of the large-scale initiatives over the product teams may require distributing a whole (a customer epic) to the multiples development teams, if the specific initiative touches more than one product. This brings more dependency on the static entities (products) and weakens the flexibility that reinforces the agility. Such a disadvantage may imply the necessity for an abstract layer supported by the project notion with its encapsulation and unifying capabilities. Project enables gathering individuals around the project-specific teams rather than distributing a whole (a project) over the multiple teams, thus removing boundaries between the static product teams during the project. A project may also work for encapsulating the end-to-end solution developments, covering pre- and post-development stages including the project transition, trainings and creating user instructions and documentation materials [12]. Optionally, keeping the team members together during a project can be a way to unify people around a dedicated project team from probably different domains that are otherwise prone to become estranged to outer world and diverged from the central designs, structures and formations in time with their “self-sufficient” structures [15].

Project manager role can be in Scrum and it must be located when the circumstances call for it [12]. A separate project manager role might be beneficial even necessary in practice when it comes to larger projects [12]. Apart from the size, in hybrid environments, a project manager who works aligned with agile culture may [12]:

- Functions as a bridge for Scrum world to open it to places where Scrum does not exist, such as the rest of organization's classical structures.
- Plays a unifying role, free from methodology when Scrum is not preferably applied in all IT.
- When third party partners want to keep their own methodologies, plays as a unifying role for these two parts.

E. SDLC (Software Development Life Cycle)

The two frameworks mainly pose different approaches to the documentation requirements of SDLC, their approvals and level of discipline promoted by means of the processes. In order to reduce this tension between them, some points that can be useful may be as following [2]:

- It is possible to approach differently to the risk regarding the potential value at the future and the one posed to the production environments during the deployments. In this regard, depending on the risk appetite of the organization, the approvals of the design documents given by the business and IT can be taken to the end of the sprints or releases because the lost on the potential value can be at maximum as much as of a sprint/release length.
- It is helpful to organize the document requirements and their frequency based on projects, releases and sprints. However, the design documents should be iteratively and incrementally created and maintained per projects.
- If the project-based teams established, Sprint 0 as a step to identify and remove possible uncertainties around the project scope, cost, schedule and technical strategy can be beneficial for large and complex projects. There, adequate grooming can be made that is required for launching subsequent sprints. Apart from getting the big picture in design, this step can fortunately reduce the documentation overhead in the subsequent sprints.
- Organizations may locate additional roles for the approvals in the IT side including enterprise and domain architects to increase centralization.

Regarding the supporting documentations, adequate trainings and continuous integration/deployment of the system being promoted to production, it is clear that the COBIT requirements are independent from any kind of methodology, as the nature of the work calls for it. However, different from the design documents, those can be organized with a release frequency. For the continuous integration, test automation helps for the regression tests of corresponding increments. For frequent promotions to the production, providing secure and sanitized test environments as a representative of the future operating landscape that Scrum anticipates, is important [2]. For this reason, the frequency of updating the test environments with more frequently updated data is crucial. Moreover, DevOps initiatives can help to fasten and smooth the transition to the production.

F. Documentation

The documentation needs are valid for the developers [21] and the software to develop. If the software development requires documentation, this need of the development should be met. Thus, the Agile methods must decide where to place the balance in documentation [22]. In searching this place of balance, as members of the teams prefer simple and practical documentation techniques [23], lean (not necessarily agile) approaches aiming at avoiding unnecessary documentation should be preferred to reach 'just enough' documentation, that can be 'comprehensive', if required.

However, when considered the contrary natures of COBIT and Scrum in terms of the documentation approach, it requires a considerable effort to find a middle point between the documentary behaviors of Scrum's lightweight and dynamic characteristic and the COBIT's deterministic and massive documentation approach [2]. However, mitigating burdens on people coming with the COBIT documentation requirements arising with Scrum coexistence is necessary. It will result in more frequent documentation transactions to keep the documents updated. The use of the digital tools can help to manage such frequent document transactions and to lessen the burden on people to manifest their real productivity.

Similarly, the concept of tools and processes should be re-considered for a right balance of digitization to create the capabilities for all the variance of time (past, present and future time), size (small-to-large) and location (distributed, collocated etc.) axis. Scrum should keep pace with the requirements of the digital age and benefit from the advanced digitization capabilities of this era (such as e-collaboration, electronic boards [24] and online-meetings) for people, by utilizing the documentation, process and tool capabilities.

G. Alignment of the Audit Perspective with Agile Approaches

To align the audit perspective with Agile approaches, the agile mindset need to be understood and adopted by the internal control and audit teams. For this, such teams may need a mindset, process and technology transformation. Thus, it would be beneficial to include these stakeholders as part of the trainings, at least.

The control side of the organizations such as the compliance and audit teams should open a proper space, give a time for Scrum and be open for a negotiation during the transformation. Seeking solutions for the new case should also manifest a new interpretations of and changes to the COBIT regulations. It may even be a need or chance to reconsider performance-conformance equation of the organizations and to adjust it accordingly.

Table-I summaries the challenges, the category of the challenges, the solution proposals and the points against to COBIT and/or Scrum and additionally indicates the source of the solution proposals. "Case 1" refers to the case delivered in the studies of [2, 3], "Case 2" refers to the case of Akbank and "Author" refers to those proposed by the first author of this paper. If the proposal of the author is aligned with the application of the case(s), then "Author" statement was not added to the relevant item.

TABLE I. LIST OF CHALLENGES AND SOLUTIONS

Challenge Title	Challenge	Category of Challenge	Solution of this Study	Source of the Solution	Points Against to COBIT	Points Against to Scrum
Steering Committee	Conflict of interest between PO and steering committee	Conflict	Product owners can be a member of the committees	Author	-	"those wanting to change a product backlog item's priority must address the product owner" / -
		Conflict	Product owner may represent the desires of the committee	Partially Case 2	The ultimate decision point in prioritization of the projects is the steering committee according to COBIT	-
	PO, one person, to decide prioritization of IT-enabled investment programs with possible influences of others	Conflict	Product owners can be a member of the committees	Author	-	Scrum does not necessarily mandate product owners as a member of any committees
Segregation of Duties	Scrum's approach lets a person to make functional tests for the functions he/she codes	Conflict	A coder should not be allowed to make functional test for his/her code	Case 1, 2	-	-
	Testing and quality review conducted by an independent bodies	Conflict	The team as a whole can be responsible for software testing and quality assurance	(for testing) Case 1, 2, (for quality assurance) Author	Not an independent body	-
Human Resource Management	Delegation of people management operations across the team and organization	Concern	Distributing the activities to parts of the organization and development teams	Case 1, 2	-	-
	Career path development	Concern	Adding positions based on the new way of working	Case 1, 2	-	-
	Dependence upon individuals	Concern	Minimizing the dependence through knowledge capture, knowledge sharing and staff backup	Case 1, 2	-	Agile principles discourage documentation
		Concern	Utilizing documentation, process and tool capabilities	Case 1, 2	-	Agile principles discourage using documentation, process and tool
	Workload and resource capacity management	Concern	Project and project management injected into Scrum	Case 1, Partially Case 2	-	Project and project management are not being addressed in Scrum properly
	Performance systems in Scrum changes dramatically	Concern	Team level measurement and assessment should be preferred with high awareness	Case 1, 2	-	For self-organizing teams, creating their own data to reflect their own performance with the sake of personal favors may be possible
Project Management	Definition of project	Conflict	Appreciating the phenomenon of project and defining it properly for Scrum	Case 1, 2	-	Scrum does not define project properly
	A project management practices along with project office and project manager roles	Conflict	Unifying people around dedicated project teams from probably different domains and managing project dynamically in an agile way	Author, Partially Case 1, 2	Against to COBIT's heavy plan-driven, control based project management approach	"How" side of project management is blur in Scrum
		Conflict	Locating project manager role (not necessarily a project office) when circumstances call for it	Case 1	COBIT mandates project office	Project manager role is still controversial in Scrum
SDLC	Certain document contents must be produced in a certain order and must be	Conflict	Breaking the order of the creation of document contents by taking them to the end of	Case 1, 2	COBIT, in principle, opposes to break the order of the creation of	Scrum does not mandate such document contents,

	approved by related parties		the sprint/release with approvals by related parties		document contents by taking them to the end of the sprint/release along with their approvals.	their approvals and roles for approvals
		Conflict	Classifying document requirements and their frequency according to project, release and sprint base	Case 1, 2	-	Scrum does not provide relevant guideline for release, project or pre-project based cycles to organize such documents
		Conflict	Locating Sprint 0 phase	Case 1	-	The Scrum Guide does not include a Sprint 0 phase
		Conflict	Additional roles for common approvals in IT side	Case 1, 2	-	Scrum does not mandates such roles or their approvals. If this roles are located outside of the teams, it becomes against to self-organizing principle of the teams
	Supporting documents, trainings and continuous integration/deployment	Concern	Preparing relevant contents and activities based on sprint and release	Case 1, 2	-	Frequency of such activities that Scrum anticipate may be an overhead
		Concern	DevOps	Case 1, 2	Depending on the design details of the role assignments, DevOps may be against to segregation of duties	-
Documentation	Documentation takes a fundamental role for COBIT, on the other hand, although not stated theoretically, in practice, Agile approaches discourage documentation and they may consider documentation as a secondary activity	Conflict	Just enough documentation as much as needed, by the help of digitization	Case 1, 2	-	Scrum considers documentation, tools and processes as a secondary activity
Alignment of the Audit Perspective with Agile Approaches	The relevant methods of audit and control teams must change accordingly	Concern	The agile mindset need to be understood and adopted by internal control and audit teams	Case 2	-	-
		Concern	A new interpretation of and changes to COBIT regulations	Case 2	Could lead to calling for radical changes to COBIT regulations	-

VII. DISCUSSION

"Conflict" represents that there exists a different and somewhat opposite point of views from the two frameworks. It indicates organizations have to find proper solutions if they prefer to go with these two frameworks simultaneously. "Concern", on the other side, stands for there is need to pay attention to the COBIT requirements especially during a Scrum transformation. Considering there is a clear need to meet COBIT requirement uninterruptedly, such a transformation could be challenging especially with the gradual transitions from traditional models that are accustomed to COBIT. The transition, thus, needs to accomplish a successful transformation and satisfy the relevant COBIT requirements during and after it.

For the "Concern" type of challenges, COBIT is regarded as the dominant part and Scrum is taken a position to fulfill the relevant COBIT objectives, except for "a new interpretations of and changes to COBIT regulations". For this kind, it is realized that COBIT do not touch to "how" side to meet its objectives and proposes the natural way of working which is common for many organizations. For instance, COBIT expects the adequate operations for the competency management, adhering to codes of ethics, dependence upon individuals, high-level design for the solution developed and providing supporting documents and trainings of new systems for relevant parties. During the implementation of Scrum, organizations are to find their ways of satisfying such requirements when considered Scrum, with its current version, leaves such areas mostly blank and does not fulfill them intentionally or unintentionally.

For the “Conflict” type of challenges, there should be an endeavor in organizations to find solutions at the middle ground. The possible solutions may be slightly different from the out-of-box COBIT proposals that are mainly a representative of the traditional mentality and Scrum that is a representative of the modern way of development mentality.

As seen, the proposed solutions are not the pure Scrum or COBIT, rather located somewhere between them. Considering this picture, regarding Scrum and COBIT as two contrary sides will not end up with desired solutions. By preserving the two sides with their essence without any negotiations, the possible solutions would be difficult to reach, e.g., rigid regulations of COBIT would crush with the flexible structure of Scrum. Similarly, it would be pointless to try to relocate agility inside COBIT. Finding a middle ground would be possible with understanding the essence of COBIT and Scrum, identifying their contexts and expectations as aimed in this study, and harmonizing them with unique needs of the organizations.

It is worthy to note that each framework have contexts and assumptions. Organizations should be aware of such context and assumption constraints when applying them in their organizations’ unique contexts that can naturally be different from those of the frameworks. For instance, in COBIT, the development pipeline is designed for a long time development period, bearing the well-known disadvantages [25]. This is why COBIT proposes to get the approvals from the relevant parties to ensure that this long time development pipeline is on the way. However, such a long pipeline is not the case with the short iterations of Scrum. Additionally, COBIT comes with some sort of assumptions such as a linear progressing in the software development; however, software development does not mostly behave in a linear direction. Thus, organizations should be identifying and thus separating the issues coming from the COBIT’s own context that may become similar to the traditional development.

For the Agile way of development, there are context-based disadvantages to tackle as well. The ASD fits and is most likely to succeed within its own "home-ground" [26], called as the “comfort-zone” for Scrum in particular [27]. The illusion of staying at the comfort-zone may have led to thinking that the Agile methods have universal value, that they represent some ultimate recipe, the holy grail of software engineering [28]. However, there rationally may exist places where an absolute agility may be naturally needed, beyond such a “comfort zone” or “Agile sweet spot” [15]. This is why, although Agile methods enjoy their comfort-zone, some organizations have already started to push the agility in software development beyond this out-of-box comfort area [24, 29]. Injecting Scrum into an environment unfamiliar to it can be regarded such. Adoption and adaptation of Scrum within a COBIT environment may help in this manner, in a way to find a middle ground for organizations that need the different capabilities.

From a higher point of view, organizations should be able to take the advantages of varying adjectives (abilities) according to their needs [34]. While ‘agility’ in the current state of the software development enjoy its comfort-zone alone, organizations today still need to have some other abilities such as being disciplined and sustainable that the Agile world intentionally or unintentionally ignores [15]. It is suggested to search for the proper integration and harmony of agile mindset, theoretically and practically with other realities and needs of organizations, and the same goes in for the versa vise [34]. This approach is also parallel to the view of Conboy and Fitzgerald [30] whom study of experts’ opinion on Agile methods notes that “the very name agile suggests that the method should be easily adjusted to suit its environment”.

There has been already a clear polarization between the Agile and traditional management communities for many years. Apart from this, the sufficient integration of Agile and traditional approaches calls for the major and fundamental shifts in thinking. However, as an indication, the traditional IT frameworks and guidelines such as PMBOK and ITIL has started to incorporate more about Agile in their recent publications. For instance, the last version of PMBOK, the 6. version [31], and the Agile Practice Guide released by the Project Management Institute pose some minor yet considerable attempts to create a more integrated approach with Agile project management. ITIL 4 framework has also evolved to reflect some similarities with the Agile principles in its guiding principles [32]. For the latest version of COBIT, COBIT 2019, [33] states that it addresses new trends in technology such as DevOps and Agile development concepts.

These initiatives can be considered as a kind of response to the needs of organizations that want to have these two worlds at the same time. Therefore, the number of such initiatives is likely to increase. For today, while this combination remains at the level of inclusion of some concepts without the sufficient integration, this convergence in the near future will probably advance.

VIII. CONCLUSION AND FUTURE WORK

The study provides a picture via the list of challenges and solutions for Scrum and COBIT coexistence. The study reminds to consider the opportunities and benefits of Scrum along with the challenges in integrating it with the COBIT practices [1]. Though the agility is necessary for organizational adaptation, control is also necessary for high assurance of survival of organizations. After all, practitioners need to strike a balance between these two seemingly ”contrary” but also complementing interests in their work environments. Even though these two sides apparently seem “contrary”, the options are not a binary or mutually exclusive. Blending them with a right balance according to each organization’s unique context is a natural need of organizations. Since it is not possible for the frameworks to predict these unique needs of each organization up-front, it would be right for the organization to tailor them according to their needs.

As a solution to striking a balance, it is actually possible to tailor Scrum. Following the idea that the more a particular project's conditions differ from the home-ground conditions, the more risk in using one approach in its pure form [26], COBIT can accompany Scrum during its unknown and compelling journey. On the other side, with the question of how much COBIT (v.4.1) is suitable to support being agile in today’s environments, organizations should choose the option of tailoring COBIT in a more agile way, if possible.

We assume the contributions this work would be valuable especially for the practitioners before the implementation of Scrum in COBIT environments. COBIT are phone to apply in disciplined, large-scaled and traditional environments. The organizations bearing one of those characteristics yet not applying COBIT can also benefit from this study. As a future work, we plan to provide the similar work of this study with the newer versions of COBIT.

References

[1] N. Ozkan, "Risks Challenges and Issues in a Possible Scrum and COBIT Marriage", Software Engineering Conference (APSEC) 2015 Asia-Pacific, pp. 111-118, 2015.

[2] N. Ozkan, A. Tarhan and C. Kucuk, “Scrum at Scale in a COBIT Compliant Environment: The Case of Turkiye Finans IT”, XP2017, 2017.

[3] N. Ozkan, N., “Scrum Integrated SDLC Processes of Turkiye Finans IT in a COBIT Compliant Environment”, Turkish National Software Engineering Symposium (UYMS), pp. 126-134, 2017.

[4] "Cobit 4.1", ISACA, 2007.

[5] "COBIT Global Regulatory and Legislative Recognition", ISACA, 2014.

[6] "13th annual state of agile survey", 2019, [online] Available: https://stateofagile.com/#ufh-i-521251909-13th-annual-state-of-agile-report/473508

[7] T. Dingsøyr and N. B. Moe, "Towards principles of large-scale agile development", International Conference on Agile Software Development, 2014.

[8] B. Boehm and R. Turner, "Using risk to balance agile and plan-driven methods", Computer., vol. 36, no. 6, pp. 57-66, 2003.

[9] V. Kannan, S. Jhajharia, S. and S. Verma, “Agile vs waterfall: A Comparative Analysis”, International Journal of Science, Engineering and Technology Research (IJSETR), vol. 3, no. 10, pp. 2680-2686, 2014.

[10] D. A. Aguillar, I. Murakami, P. Manso, and P. T. Aquino, “Small Brazilian Business and IT Governance: Viability and Case Study”, Information Technology for Management. Ongoing Research and Development, pp. 173-193, 2017.

[11] J. Sutherland and K. Schwaber, "The Scrum guide. the definitive guide to Scrum: The rules of the game", 2017, [online] Available: https://www.scrum.org/resources/scrum-guide.

[12] N. Ozkan and C. Kucuk, "A Systematic Approach to Project Related Concepts of Scrum", Revista de Management Comparat International, vol. 17, no. 4, 2016.

[13] T. Clear, “Documentation and agile methods: striking a balance”, SIGCSE Bull, vol. 35, no. 2, pp. 12–13, 2003.

[14] S. Nerur, R. Mahapatra and G. Mangalaraj, "Challenges of migrating to agile methodologies", Commun. ACM, vol. 48, no. 5, pp. 72-78, 2005.

[15] N. Ozkan, and A. Tarhan, “An Investigation into Increased Agility by Balancing Agile and Traditional Process Approaches”, Turkish National Software Engineering Symposium (UYMS), 2018.

[16] G. Parry, L. Newnes, and X. Huang, “Goods, products and services”, Service design and delivery, pp. 19-29, Springer: Boston, MA, 2011.

[17] K. Schwaber, "Agile Project Management with Scrum", Redmond: Microsoft Press, 2004.

[18] PMBOK Guide, Project Management Institute, 2013.

[19] "COBIT Control Practices: Guidance to Achieve Control Objectives for Successful IT Governance 2nd edn", ISACA, 2007.

[20] K. Beck et al., "Agile manifesto", 2001, [online] Available: http://agilemanifesto.org.

[21] N. Sekitoleko, et. al., “Technical dependency challenges in large-scale agile software development”, International Conference on Agile Software Development, pp. 46-61, 2014.

[22] M. C Paulk, “Agile methodologies and process discipline”, Institute for Software Research, pp.15-18, 2002.

[23] C. D. O Melo, C. Santana, and F. Kon, “Developers motivation in agile teams”, 38th Euromicro Conference on Software Engineering and Advanced Applications, pp. 376-383, 2012.

[24] R. Hoda, P. Kruchten, J. Noble and S. Marshall, "Agility in context", ACM Sigplan Notices, vol. 45, no. 10, pp. 74-88, 2010.

[25] M. Stoica, M. Mircea and B. Ghilic-Micu, "Software development: Agile vs. traditional", Inform. Econ., vol. 17, pp. 64-76, 2013.

[26] B. Boehm, and R. Turner, Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-Driven Methods, Proc. the 26th International Conference on Software Engineering, pp. 718-719, 2004.

[27] R. Lyon and M. Evans, "Scaling up pushing scrum out of its comfort zone", Agile 2008 Conference, pp. 395-400, 2008.

[28] P. Kruchten, “Contextualizing agile software development”, Journal of Software: Evolution and Process, vol. 25, no:4, pp. 351-361, 2013.

[29] T. Dingsøyr and N. B. Moe, "Research challenges in large-scale agile software development", ACM SIGSOFT Software Engineering Notes, vol. 38, no. 5, pp. 38-39, 2013.

[30] K. Conboy and B. Fitzgerald, "The views of experts on the current state of agile method tailoring", IFIP International Working Conference on Organizational Dynamics of Technology-Based Innovation, pp. 217-234, 2007.

[31] Project Management Institute A Guide To The Project Management Body Of Knowledge (PMBOK-Guide) - Sixth version, Pennsylvania, USA:Project Management Institute, Inc, 2017.

[32] M: Corona, “ITIL 4, IT Service Management and Agile”, Axelos, 2019, [Online]. Available: https://www.axelos.com/case-studies-and-white-papers/itil-4-it-service-management-and-agile.

[33] J. Lainhart, “Introducing COBIT 2019: The Motivation for the Update”, 2018, [Online]. Available: https://www.isaca.org/resources/news-and-trends/newsletters/cobit-focus/2018/introducing-cobit-2019-the-motivation-for-the-update

[34] N. Ozkan, “Imperfections Underlying the Manifesto for Agile Software Development”, 1st International Informatics and Software Engineering Conference (UBMYK), 2019.

[35] N. Ozkan, A.K. Tarhan, “Investigating Causes of Scalability Challenges in Agile Software Development from a Design Perspective”, 1st International Informatics and Software Engineering Conference (UBMYK), 2019.

[36] A. C. Amorim, M. M. da Silva, R. Pereira and M. Gonçalves, “Using agile methodologies for adopting COBIT”, Information Systems, 101496, 2020.

[37] N. Ozkan, “People Management Issues in Scrum from COBIT Perspective”, the Workshop on Alternative Workforces for Software Engineering (WAWSE), pp. 54, 2015.

[38] C. Montenegro, and R. Arévalo, “Software development governance for VSE-SCRUM teams: Model and evaluation in a developing country”, International Conference on Software Engineering and Information Management, pp. 1-5, 2018.

[39] N. Ozkan and C. Kucuk, “Integrating Project Related Concepts with the Core of Scrum. International Management Conference, pp. 221-230, 2017.

[40] N. Zabkar and V. Mahnic, "Using COBIT indicators for measuring Scrum-based software development", WSEAS Transactions on Computers, vol. 7, no. 10, pp. 1605-1617, 2008.

[41] A. Przybyłek, W. Kowalski, Utilizing online collaborative games to facilitate Agile Software Development, Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, M. Ganzha, L. Maciaszek, M. Paprzycki (eds). ACSIS, Vol. 15, pp. 811–815, 2018, DOI: 10.15439/2018F347.

DOI: 10.15439/2020F46
ISSN 2300-5963 ACSIS, Vol. 21

Towards a Better Understanding of Agile Mindset by Using Principles of Agile Methods

Necmettin Ozkan
Information Technologies Research and Development Center
Kuveyt Turk Participation Bank
Kocaeli, Turkey
necmettin.ozkan@kuveytturk.com.tr

Mehmet Şahin Gök
Department of Business
Gebze Technical University
Kocaeli, Turkey
sahingok@gtu.edu.tr

Büşra Özdenizci Köse
Department of Business
Gebze Technical University
Kocaeli, Turkey
busraozdenizci@gtu.edu.tr

Abstract— **The right way to agility should start with a proper agile mindset instead of applying Agile methods directly. However, apart from the manifesto, it is unlikely to find a comprehensive set of Agile principles that can serve for an improved agile mindset. Our study intends to fulfill this gap in a systematic way: providing a list of the Agile methods along with their principles within a single source, in the way of providing a better understanding of the concept of agility from a wide and exhaustive perspective. To do so, the collected 105 principles were content-analyzed in order to group them into 32 categories for a higher-level abstraction. These categories then were subsumed into two main categories. The whole grouping process was reviewed by one expert and the list was adjusted accordingly. Then, based on the consolidated list of the categorized principles, analysis and evaluations were made by the authors. As a part of the evaluations, semi-structured interviews with two experts were conducted to evaluate the categorized principles in general, especially in terms of their contribution to agility.**

Keywords— agile mindset, agility, agile methods, agile frameworks, principles

I. Introduction

Effective agile individuals, teams and organizations require a particular attitude, way of thinking and behavior so called as agile mindset, beyond the given set of procedures, techniques and rituals [12]. By applying a set of Agile practices of (a) particular Agile method(s), there is no guarantee to utilize the agile mindset properly [1]. The practices offered by the Agile methods have some fundamental limitations in nature; they are not adequate to cover possible agility capabilities fully and also they are very static in providing the ability of adaptation to changing situations. Indeed, the right way to agility should start with a proper agile mindset instead of applying Agile methods directly. Principles, as “a basic belief, theory, or rule that has a major influence on the way in which something is done” (macmillandictionary.com), support any mindset more effectively than practices. This emphasizes a proper understanding and locating the principles first and foremost, before the practices.

In terms of providing Agile principles, the Agile Manifesto is the most well-known set in supporting the Agile mindset in a formal sense via its values and principles. Apart from the manifesto, it is possible to find different sets of Agile principles scattered in the literature, which makes it hard to reach a comprehensive list. The existing literature has been reviewed in our study to find out the most possible comprehensive list of the Agile methods. It shows that there has been no study aiming to disclose the principles or principle-like features (referred to as “principles” across the study, if not stated otherwise) of the methods. Our study in particular intends to fulfill this gap in a systematic way: providing the most possible complete list of the methods along with their principles within a single source from a wider and exhaustive perspective, in the way of providing a better understanding of the concept of agility.

In addition to exhibiting the known attributes of the methods, differently, our study aims to reveal some analysis through the consolidated list of the principles (Level 1/L1) such as grouping them into categories (Level 2/L2) along with the classification of these categories at a higher level (Level 3/L3), an analysis on the methods and their principles (L2), the contribution of principles (L2) to agility and more, with referring to expert opinions to get more sound basis. In particular, taking advantage of reaching out to such a scope, the principles provided by the methods (L2) and the principles of the Agile Manifesto are mapped, in terms of coverage. Consequently, the research questions are formed as follows:

RQ1: What are the Agile methods in the literature?

RQ2: What are the principles offered by these Agile methods?

RQ3: Is there a match between the principles of the methods with the manifesto principles or not?

RQ4: What do the principles (L2) represent in general, and also about their contribution to agility in particular?

The rest of this paper is organized as follows: Section 2 delivers the background for the methods and the definition of agility for software solution development. Section 3 elaborates related works and Section 4 depicts the research design. Section 5 delivers findings and analyses made for the set of the principles. Section 6 evaluates findings and analysis with the consideration of the feedback from the interviewees. Finally, Section 6 presents conclusions and future work.

II. Background

A. A Brief History of the Agile Methods

As a cornerstone of the Agile methods, iterative, evolutionary, and incremental development roots go back decades [2]. It grew from the 1930s’ work proposing a series of short “plan-do-study-act” cycles for quality improvement and was involved in software projects such as NASA’s Mercury in the early 1960s, with practices like time boxing, test-first development [2]. One of the early traces of the Agile principles were also witnessed in the work of the Tavistock Group, which conducted research on the self-organizing teams of British coal miners in the 1950s [3]. It is worth mentioning that, in 1976, Tom Gilb introduced evolutionary project management as the first clear flavor of the Agile methods [2].

One of the early known works defining the self-organizing teams is Takeuchi's study [4], inspired by the Toyota production system. In the study of Morgan [5], he argues that an organization can improve its ability to self-organize through the holographic brain metaphor. In 1988, Gilb published a new book, Principles of Software, which describes the Evo method (chronologically the first method in our study).

Apart from these prominent milestones, throughout the 1970s and 1980s, there are some other publications and specific projects partially integrating the Agile practices. While people from the previous decades incorporate a preliminary major specification stage with the teams utilizing iterations with minor feedback, differently in the 1990s, the mainstream of Agile initiatives became preferring less early specification work, rather a stronger evolutionary analysis approach [2]. In this decade, unlike the previous ones, the agile mindset and practices started to take a form within certain formal methods/frameworks (hereinafter and heretofore referred to as "method"), such as Scrum, Dynamic Systems Development Method (DSDM), Rational Unified Process (RUP), Extreme programming (XP), Feature-driven Development (FDD), which later referring collectively to Agile Software Development Methodologies.

The quest for a full-fledged agile mindset ended up in the meeting in a ski resort in Utah, in the year 2001, where the well-known techniques from some "Agile Software Development Methodologies" were combined within the manifesto for the Agile Software Development [6]. Those well-known methods that had an influence on the manifesto include DSDM, TDD (Test-Driven Development), ASD (Adaptive Software Development), D3 (Design Driven Development), Scrum, Crystal, XP, Pragmatic Programming, FDD [6]. From this period of time to today, the interest in Agile has continued increasingly and various other methods have been presented.

B. Back to the Basics: the Definition of Agility in Software Development Domain

The understanding of the word "Agile" varies [7], even among prominent Agile pioneers. For instance, Alistair Cockburn defines it as "being effective and maneuverable" [8]. Kruchten's [9] definition is "the ability of an organization to react to changes in its environment faster than the rate of these changes". Conboy and Fitzgerald [10] state agility as "the continual readiness of an entity to rapidly or inherently, pro-actively or reactively, embrace change, through high-quality, simplistic, economical components and relationships with its environment the continual readiness of an entity". Highsmith defines agility as "the ability to both create and respond to change in order to profit in a turbulent business environment; it is the ability to balance flexibility and stability [11]. Instead of using such existing definitions, we would rather like to present a revised definition of agility inspired from the definition in [42], to communicate a better understanding of the background of our mentality used throughout this study.

We see "responding to change" as the widely recommended feature of agility. At this point, questions arise: change of "what"; inconsistent customer requirements, analysis documentation or changes in the environment? Therefore, it is better to define agility based on the closest point to the source of the change, which is the reality itself, instead of from the view of customers, for instance. Users or customers are a kind of proxy of reality and the same as documentation as a proxy of the system being developed, not the reality itself. From another point of view, the definitions similar to of Kruchten ("the ability of an organization to react to changes in its environment faster than the rate of these changes") take us to a passive position of re-acting. Although information technology has traditionally taken a passive position throughout its history, as it has been seen as a business-driven body, it is not a common rule beyond ages. Thus, these two points bring us to a new definition of agile; "the ability to move quickly and easily" (where Cambridge, Oxford and Macmillan dictionaries achieve consensus for this part of the definition), to adapt to changes of the reality or to create changes becoming the reality, let us say in the domain of software solution development.

III. Related Work

There are plenty of studies reviewing the Agile methods, comparing them with their characteristics, strengths, weaknesses, similarities and differences, providing criteria to choose them according to the context of development, generally provided in an informative way. Our study rather intends to provide a complete list of the methods along with their principles. In addition to exhibiting the known attributes of the methods such as their principles, we aim to reveal some patterns through the consolidated list of the principles such as by grouping them into categories, the classification of categories, the frequency of principles, the contributions of principles to agility and more. Expert opinions are involved in interpretation-intense sections to get more sound determinations. Hence, this study provides a wider perspective to the concept of agility by revealing all possible Agile methods and their principles in a single picture along with analysis, resulting in inputs for a better understanding of the agile mindset.

The majority of the works on the agile mindset are satisfied by only mentioning the term as a "fixed concept" without actual descriptions, details, explanations or definitions [1]. As witnessed by the study of Mordi and Schoop [1] conducted in 2020, there are also relatively few studies on the agile mindset. Among these few papers, [1, 12, 43] aim to come with a list of the elements of agile mindset. Miler and Gaida [12] conducts a survey with 52 Agile practitioners who evaluate the importance of 26 selected elements of the agile mindset to the effectiveness of an Agile team. Miler uses the literature review to identify relevant elements, consisting mainly of books, web sites and hardly of peer-reviewed papers. By using a similar way to define the characteristics of the agile mindset, the study [1] conducts a review of the existing literature, including both scientific as well as practitioner publications, and interviews with practitioners. Study [43] identifies factors that affect the expansion of agile development in large organizations positively or negatively using interviews within multiple case studies then groups them in two categories: ”agile mindset” and ”contextual dependencies". When it comes to the difference between these types of studies and our study, our work focuses on the principles specifically that may contribute directly or indirectly to the understanding of the agile mindset, with their possible elements.

IV. Research Design

The well-known methods that had an influence on the manifesto include DSDM, TDD, ASD, D3, Scrum, Crystal, XP, Pragmatic Programming, FDD [6]. From this set of methods Scrum, XP, Crystal and FDD were used to form the search phrase, as these are better known than others do. Thus, the search was done with the keyword of "scrum 'Extreme Programming' crystal 'feature driven development'", in the dates between 28/01/2020 and 05/02/2020, without any specific filter in the year range, within the full text, in the libraries of IEEE Xplore, Web of Science, Science Direct and Springer, respectively. A total number of 368 works that are peer-reviewed and in English were returned from the search results. The researcher could not reach the full text of the 58 of them. The rest 310 works were examined through their full text to find and extract the methods mentioned. Considering that any new method name not encountered since after 83% of this search indicates that the search result set is sufficient in terms of the coverage.

After reaching the list of the methods, explicitly listed principles or principle-like attributes of each method along with their descriptions were extracted from the formal books (as indicated in Table 1) or from the formal web-site of the methods. Primarily, the principles were sought, if not found, the principle-like attributes were used. The principle-like attributes include philosophy, value, pillars, characteristics, and properties, respectively. To reach to this list of attributes (philosophy, value, pillars, characteristic, and properties), the concepts with a close relationship with dictionary meanings of "principle" were sought in multiple dictionaries. The relationships between these attributes are shown below. It demonstrates that the meaning of the principle, philosophy, value and pillars are related to each other by means of shared words in the descriptions of their meanings. By definition, properties and characteristics of a concept serve for effectively defining phenomenon under consideration [14], which make "characteristic" and "property" a proper candidate for inclusion.

- Principles: "a **basic** belief, theory, or rule that has a major influence on the way in which something is done" (macmillandictionary.com)
- Philosophies: "a system of **beliefs** that influences someone's decisions and behavior" (macmillandictionary.com)
- Values: "the **principles** and **beliefs** that influence the behavior and way of life of a particular group or community" (macmillandictionary.com)
- Pillars: an important idea, **principle**, or belief (macmillandictionary.com)
- Characteristics: "a typical or noticeable feature of someone or something" (dictionary.cambridge.org)
- Properties: "a quality or characteristic that something has" (oxfordlearnersdictionaries.com)

In the chain of the "reality-values-principles-practices", even though principles have a close relationship with practices in a way of influencing the pattern of practices done, the practices were not included in the set because of that they can/should be diverse and varying, with no limitation. Even though Agile methods share some common practices such as short time boxed iterations with adaptive and evolutionary refinements of plans, specific practices of the methods still vary [15], in this sense, it makes it difficult to collect all of them from all of the methods. Thus, this study internationally prefers to exclude practices from the list.

After reaching the list of principles of the methods, which is a straightforward process, the first author content-analyzed the principles' descriptions (L1) and grouped them into 32 categories (L2) based on his knowledge for a higher-level abstraction. These categories then were subsumed into two main categories (L3), by the same author. The whole grouping process was reviewed by one expert in Agile Software Development having both academic and sector background for 5 years in Agile Software Development particularly, and the list was adjusted accordingly (%18 of the items updated after two iterations). Then, over the consolidated list of the principles with their grouping (L2), analysis and evaluations were made by the authors. As a part of the evaluations, the first author conducted semi-structured interviews with two experts to evaluate the principle categories (L2), especially in terms of their contribution to agility. The notes taken were then reviewed by the interviewees and necessary corrections were made accordingly.

V. Findings and Analysis

A. Methods (RQ1)

The search mentioned in the Research Design Section to find out the Agile methods in the literature has ended with 28 methods listed in Table 1.

Regarding these methods, as one of them, **Evo**, the first Agile method in the list, provides a baseline for many Agile initiatives. As the most used method, **Scrum**, is designed for small self-organizing teams breaking their work into smaller parts that can be completed within time-boxed iterations that are no longer than one month. **DSDM** was initially proposed to build quality into RAD (Rapid Application Development). In the recent version, DSDM fixes time; functionality varies according to the need of stakeholders. It resembles Scrum in terms of practices such as time-boxing, iterative development, taking the customer in, staying mainly on the development layer, covering the world of a single team and increasing roles via the proxies. In a different way, DSDM adds a project layer on top with planning activities, encourages visualization through the concept of modeling and makes an emphasis on the quality aspects of the development.

DAD brings discipline in the implementation of Agile approaches and builds on the many practices from Scrum, AM, LSD, and others yet with the aim of moving beyond Scrum, which makes it a scaling framework as well. **DevOps** proposes a set of practices that combine software development (Dev) and operations (Ops) which aims to provide a continuous stream of integration and delivery. **FDD** focuses on the feature aspect of a project and development is organized based on the feature concept, posing a position located mainly on the first parts of the development pipeline. **XP** proposes software development engineering practices. **Crystal** family is a collection of the Agile methods proposing different sub-methods based on the individual project complexity and the team size that are measured mostly by the quantitative properties. Then, it recommends the implementation of certain roles and artifacts accordingly, representing a plan-driven approach to development to some extent.

An ancestor of considerable methods on the list, **RUP** draws attention with its object-oriented modeling, numerous roles and artifacts offering a descriptive and obsolete approach for today in terms of agility. **OpenUP** preserves the essential characteristics of RUP that include iterative development, use cases and scenarios driving development and architecture-centric approach yet adding some Agile aspects such as iterative development with feedback loops. Like RUP, **ICONIX** uses UML based diagrams turning to use case text into working code. **AM** was introduced to adapt modeling practices using an agile mindset and it covers only modeling. Sharing the same inventor, **ADM** focuses on the data aspects of development. Coined by the same inventor, **AUP** proposes a simplified version of RUP and, in 2012, was superseded by DAD.

ASD comes with some basic principles, lacking with implementation details, as a more iterative and shorter-interval version of the RAD. **ASP**, with an image of extinction with very few resources, describes concurrent development processes in the Japanese software industry, which already includes practices like dividing software into smaller parts, a time-fixed interval of delivery, close customer relations, and incremental construction of the system. Although **MSF** was not designed with a full Agile perspective at the first stage, it brought in an Agile template into the tool in 2005. **PSP&TSP** offers suggestions for individuals and teams to manage their own works and determines their competencies with a focus on measurement that they need to develop.

LSD focuses on optimizing the entire development process and reducing waste. **Kanban** focuses on continuous flow and continual delivery of work instead of iterating. **Scrumban** offers a structure that combines selected features of Kanban and Scrum.

OSSD is hardly to count as a pure Agile method, yet it can be considered similar to the Agile approach with sharing code freely, faster development cycles and such. **ISD** proposes development with small teams working in parallel and dependency management by using a combined spiral /waterfall model with daily builds aimed at developing a product with high speed.

TDD provides a set of practices for testing. **BDD** is an extension of test-driven development with a set of practices for testing. **PP** introduces a set of programming best practices in the form of the collection of short tips. These three (TDD, BDD and PP) are excluded from the list for further stages as they focus deeply on programming practices. **D3**, suffering from lack of sufficient resources, uses design as a part of processes to learn and better define requirements whereby design and user experience drive the development. For **APM**, there is similarly no sufficient resource for further investigation and thus, these two (D3 and APM) are excluded from the list for further stages.

While determining the “Obsolete” field in Table 1, three different parameters were looked at: 1- whether the main subject (such as UML modeling, object-oriented approach, spiral model) on which the method is based becomes obsolete in the Agile world for today, 2- whether superseded by another method, 3- no appearance in the VersionOne reports [13] from 2006 to 2019 (Obsolete) or disappearance towards the recent years (Nearly Obsolete), 4- the resources found during the authors' review on the methods belong to the far old years. These reasons for being obsolete as coded from 1 to 4 accordingly are also delivered in the list. For instance, AUP, ADM and AM are superseded by DAD and the ones using RUP as the foundation stone including OpenUp, ICONIX and AUP are out of date as RUP is so, at least for the Agile communities of today. For the rest of the methods that are referred to as “Alive”, it implies that their ideas are still valid and their names are included in the reports of Version One for at least the recent three years (2017, 2018, and 2019).

TABLE I. LIST OF AGILE METHODS

Method	Abb.	Release year	Vitality	Reasons of Being Obsolete	Principle Related Attribute	Main Reference
Evolutionary Project Management	Evo	1981	Obsolete	3,4	Principles	[17]
Dynamic Systems Development Method	DSDM	1995	Nearly Obsolete	3	Principles	[18]
Scrum	Scrum	1995	Alive	-	Pillars	[19]
Rational Unified Process	RUP	1996	Obsolete	1,3,4	-	[20]
Agile Software Process	ASP	1997	Obsolete	3,4	Characteristics	[21]
Open Source Software Development	OSSD	1997	Obsolete	3,4	-	[22]
Crystal	Crystal	1998	Obsolete	3,4	Properties	[23]
Adaptive Software Development	ASD	1999	Obsolete	3,4	Characteristics	[24]
Extreme Programming	XP	1999	Alive	-	Values	[25]
Feature-driven Development	FDD	1999	Nearly Obsolete	3	-	[26]
Internet Speed Development	ISD	1999	Obsolete	1,3,4	-	[27]
Pragmatic Programming	PP	1999	-	-	-	-
Agile Modeling	AM	2002	Nearly Obsolete	1,2,3	Values	[28]
Agile Data Method	ADM	2003	Obsolete	1,2,3	Philosophies	[29]
Lean Software Development	LSD	2003	Alive	-	Principles	[30]
Agile Unified Process	AUP	2005	Nearly Obsolete	1,2,3	Philosophies	[31]
Microsoft Solutions Framework	MSF	2005	Obsolete	3,4	Principles	[32]
Open Unified Process	OpenUP	2006	Obsolete	1,3,4	Principles	[33]
Behavior-Driven Development	BDD	2009	-	-	-	-
DevOps	DevOps	2009	Alive	-	Principles	[34]
Scrumban	Scrumban	2009	Alive	-	-	[35]
Kanban	Kanban	2010	Alive	-	Principles	[36]
Disciplined Agile Delivery	DAD	2012	Alive	-	Principles	[37]
Design Driven Development	D3	-	-	-	-	-
Personal Software Process & Team Software Process	PSP&TSP	1996	Obsolete	3,4	Principles	[38]

Agile Portfolio Management	APM	-	-	-	-	-
ICONIX	ICONIX		Obsolete	1,3,4	-	[39]
Test-driven development	TDD	-	-	-	-	-

B. *Agile Principles (RQ2)*

After reaching the list of the methods, the principles of each method (L1) were collected as described in the Research Design section and 105 (101 distinct in names) principles were achieved. These principles were then grouped into the high-level principles that are 33 in number (L2). During this stage, it is seen that some original principles (L1) can serve for multiple high-level principles (L2) then they are duplicated under different high-level L2 principles, yielding 114 L1 principles in total (duplicated ones are marked with a number inside the corresponding principle box). The L2 principles are also classified as People or Process-Relevant (L3), according to their descriptions. At this stage, if an L3 item includes both Process and People Relevant L2 item(s) then it was taken of those with a higher number (there was no equality encountered). As a note, this hybrid distribution was seen in the 5 of 33 L2 principles. All trees are depicted as below bearing principle name, relevant method(s), and the unique numbers if duplicated.

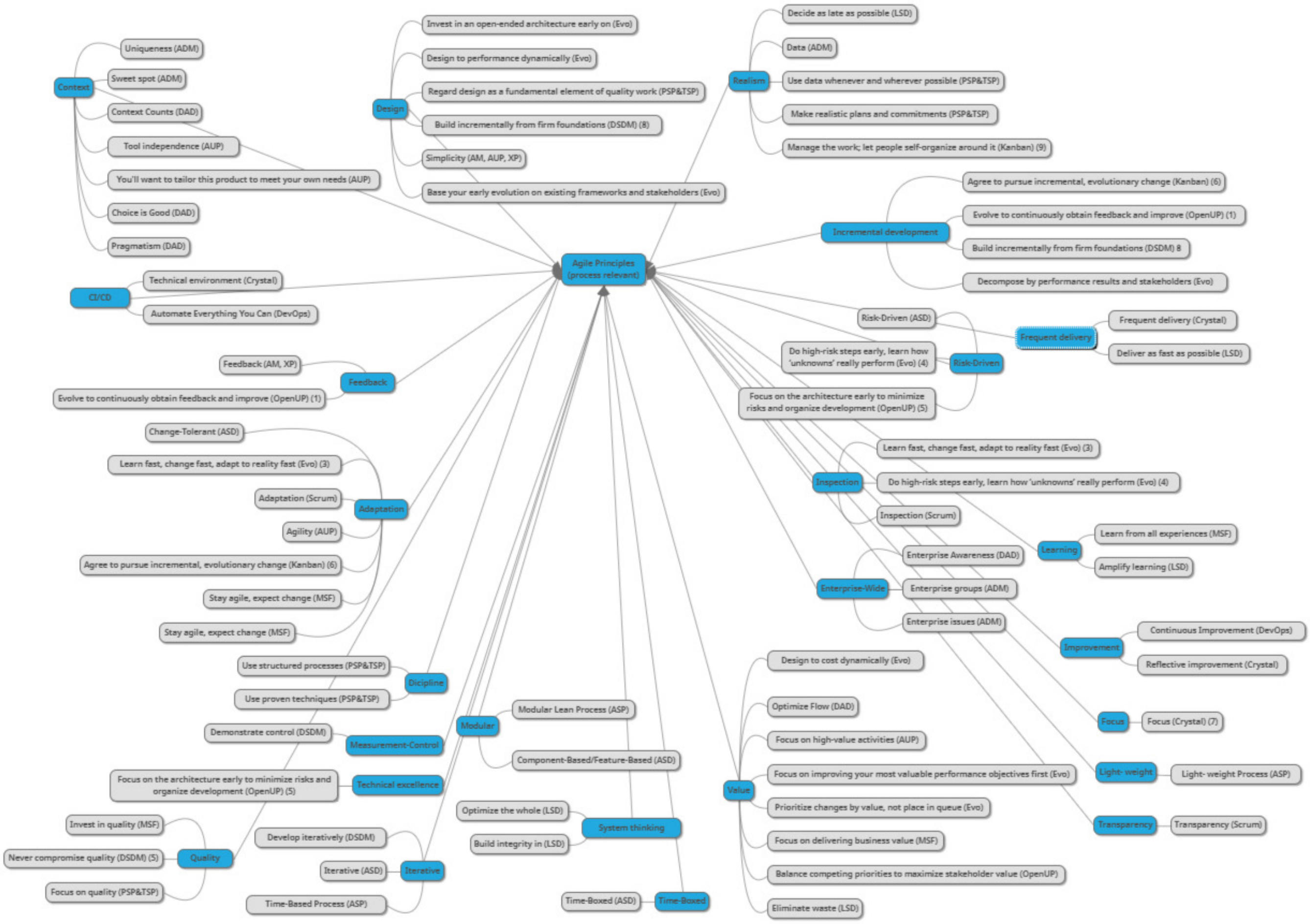

Agile Principles (process relevant)
Context
Uniqueness (ADM)
Sweet spot (ADM)
Context Counts (DAD)
Tool independence (AUP)
You'll want to tailor this product to meet your own needs (AUP)
Choice is Good (DAD)
Pragmatism (DAD)
Design
Invest in an open-ended architecture early on (Evo)
Design to performance dynamically (Evo)
Regard design as a fundamental element of quality work (PSP&TSP)
Build incrementally from firm foundations (DSDM) (8)
Simplicity (AM, AUP, XP)
Base your early evolution on existing frameworks and stakeholders (Evo)
Realism
Decide as late as possible (LSD)
Data (ADM)
Use data whenever and wherever possible (PSP&TSP)
Make realistic plans and commitments (PSP&TSP)
Manage the work; let people self-organize around it (Kanban) (9)
Incremental development
Agree to pursue incremental, evolutionary change (Kanban) (6)
Evolve to continuously obtain feedback and improve (OpenUP) (1)
Build incrementally from firm foundations (DSDM) 8
Decompose by performance results and stakeholders (Evo)
CI/CD
Technical environment (Crystal)
Automate Everything You Can (DevOps)
Feedback
Feedback (AM, XP)
Evolve to continuously obtain feedback and improve (OpenUP) (1)
Risk-Driven
Risk-Driven (ASD)
Do high-risk steps early, learn how 'unknowns' really perform (Evo) (4)
Focus on the architecture early to minimize risks and organize development (OpenUP) (5)
Frequent delivery
Frequent delivery (Crystal)
Deliver as fast as possible (LSD)
Adaptation
Change-Tolerant (ASD)
Learn fast, change fast, adapt to reality fast (Evo) (3)
Adaptation (Scrum)
Agility (AUP)
Agree to pursue incremental, evolutionary change (Kanban) (6)
Stay agile, expect change (MSF)
Stay agile, expect change (MSF)
Inspection
Learn fast, change fast, adapt to reality fast (Evo) (3)
Do high-risk steps early, learn how 'unknowns' really perform (Evo) (4)
Inspection (Scrum)
Learning
Learn from all experiences (MSF)
Amplify learning (LSD)
Enterprise-Wide
Enterprise Awareness (DAD)
Enterprise groups (ADM)
Enterprise issues (ADM)
Improvement
Continuous Improvement (DevOps)
Reflective improvement (Crystal)
Dicipline
Use structured processes (PSP&TSP)
Use proven techniques (PSP&TSP)
Measurement-Control
Demonstrate control (DSDM)
Technical excellence
Focus on the architecture early to minimize risks and organize development (OpenUP) (5)
Modular
Modular Lean Process (ASP)
Component-Based/Feature-Based (ASD)
Quality
Invest in quality (MSF)
Never compromise quality (DSDM) (5)
Focus on quality (PSP&TSP)
Iterative
Develop iteratively (DSDM)
Iterative (ASD)
Time-Based Process (ASP)
System thinking
Optimize the whole (LSD)
Build integrity in (LSD)
Time-Boxed
Time-Boxed (ASD)
Value
Design to cost dynamically (Evo)
Optimize Flow (DAD)
Focus on high-value activities (AUP)
Focus on improving your most valuable performance objectives first (Evo)
Prioritize changes by value, not place in queue (Evo)
Focus on delivering business value (MSF)
Balance competing priorities to maximize stakeholder value (OpenUP)
Eliminate waste (LSD)
Focus
Focus (Crystal) (7)
Light- weight
Light- weight Process (ASP)
Transparency
Transparency (Scrum)

Fig. 1. Process-relevant Principles

With **iterative** development along with **frequent delivery**, a big bunch of development is divided into smaller functional increments to understand functionality better, to manage **risk** effectively and to get **feedback** from customers and end users early. Iterative development encourages experimentation and **learning**. Through feedback and learning cycles, teams can identify areas for **improvements**. To the short cycles of iterations, a **fixed schedule** accompanies [in some methods] to reach a high level of predictability. With iterative development, the accumulation is not by default in additive kind. The system developed can yield **incremental** progress thus an organic growth of the system is achieved as required to **adapt** to changes.

In order to manage the complex world of **reality** along with its **context** variations, the human-beings who have equally complex abilities is brought up against it. Human-made proxy products, such as processes, documents, fixed plans, are neither capable of representing the actual ability of the human nor the reality itself. Instead of these intermediate solutions, by reducing their significance, people are in the foreground to counterbalance the reality. And, therefore it is aimed to be close to the customer who is relatively close to reality. As being close to the front sides, customer and end user are the real owner and user of requests, which reminds being **focusing on the customer, value, quality and goal**.

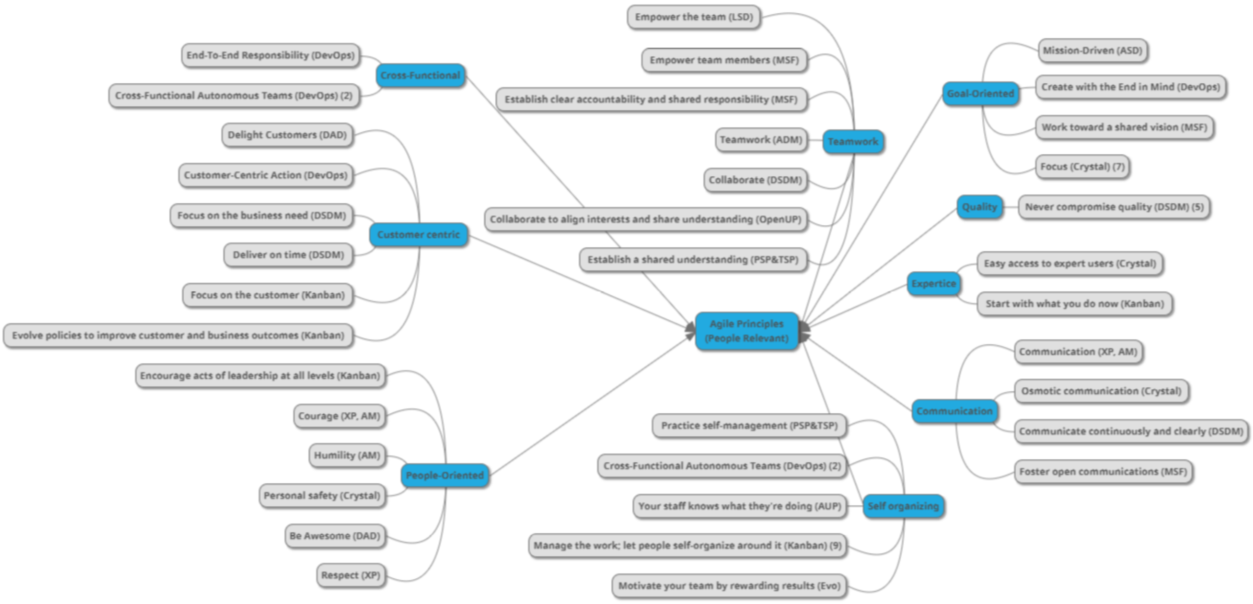

Agile Principles (People Relevant)
Cross-Functional
End-To-End Responsibility (DevOps)
Cross-Functional Autonomous Teams (DevOps) (2)
Customer centric
Delight Customers (DAD)
Customer-Centric Action (DevOps)
Focus on the business need (DSDM)
Deliver on time (DSDM)
Focus on the customer (Kanban)
Evolve policies to improve customer and business outcomes (Kanban)
People-Oriented
Encourage acts of leadership at all levels (Kanban)
Courage (XP, AM)
Humility (AM)
Personal safety (Crystal)
Be Awesome (DAD)
Respect (XP)
Teamwork
Empower the team (LSD)
Empower team members (MSF)
Establish clear accountability and shared responsibility (MSF)
Teamwork (ADM)
Collaborate (DSDM)
Collaborate to align interests and share understanding (OpenUP)
Establish a shared understanding (PSP&TSP)
Goal-Oriented
Mission-Driven (ASD)
Create with the End in Mind (DevOps)
Work toward a shared vision (MSF)
Focus (Crystal) (7)
Quality
Never compromise quality (DSDM) (5)
Expertice
Easy access to expert users (Crystal)
Start with what you do now (Kanban)
Communication
Communication (XP, AM)
Osmotic communication (Crystal)
Communicate continuously and clearly (DSDM)
Foster open communications (MSF)
Self organizing
Practice self-management (PSP&TSP)
Cross-Functional Autonomous Teams (DevOps) (2)
Your staff knows what they're doing (AUP)
Manage the work; let people self-organize around it (Kanban) (9)
Motivate your team by rewarding results (Evo)

Fig. 2. People-relevant Principles

In the association with the reality, people often use investigation, **inspection, learning and feedback loops** to get to know more about the reality. People abandon the passive position of classical methods and take on a more active role. Effective learning includes learning from mistakes. At this process, one of the things people need is **courage** needed for change including changing one's own self, with a feeling of being safe and having relatively high tolerance against mistakes that require **personal safety**. This calls for that both the team and the members of the team **respect** each other. Respect strengthens communication channels, supports coloration and accepting feedback. Respect assures for individuals a suitable safe place for trial and learning. Courage is also important to hearten people to make critical decisions to be able to change direction for adaptation.

"To move quickly", the information should flow quickly inside and between the teams. This is mainly why Agile teams are preferably co-located and cross-functional. Thus, with close and intense **communication**, the interaction of information increases and the information itself becomes agile: it is updated, corrected, accelerated, shared to gain experience and to develop new ideas throughout and beyond **enterprises**. **Transparency** plays as a facilitator for communication. Communication enables learning, including from the developed solution itself. It is necessary to communicate with the developed solution itself to see its behavior, listen to what it says (the process is successful, throws an error, etc.). Moreover, along with shared goals, communication also supports collaboration and **teamwork**.

Cross-functionality reduces the cost of communication by gathering the necessary competencies into the team and enables rapid action. With the contribution of cross-functionality, **self-organization** enables teams to operate around varying cases of the complex world of reality.

Agile processes are additionally equipped with **technical excellence, continuous integration and deployment, system thinking, design** capabilities, **disciplined approaches and measurement-control mechanisms**, by some of the methods.

Numerically speaking, process- relevant items (of L1) cover 64% (73/114) of the whole. Among the people- relevant items (of L2), depending on the definition of agility, adaptation to realism to create value comes into prominence. However, the enterprise-wide perspective is relatively underestimated to create this value (of the organization). This may be because of the people who created these methods having more developer backgrounds. Design may come to the fore with an effect of a similar situation. Quality emphasis has a moderate place unlike in the manifesto that gives no place for it. Although discipline in Agile approaches is hardly addressed, we see that some methods include this dimension. Time-box, frequent delivery, and iterative development practices applied by many methods are rather less apparent at the principles level. However, when considered incremental and iterative development, frequent delivery and continuous integration and deployment (CI/CD) together, they take considerable place. It is observed that the Lean approach, of which the main focus is not agility, but the literature counts it as an Agile method, creates a unique field and does not receive much support from other methods for System Thinking.

In the people-relevant dimension, we see that human and team relevant principles come to the fore. It is natural in this sense that the channels of communication equipping human abilities are seen at a high level of principles. The context dimension, which needs human abilities to manage rather than the ability of the process dimensions, takes an important place. Parallel to reality-driven, customer orientation is also located at higher levels at the people-relevant side. However, the expertise of individuals who are expected to pose a parallel level with the context dimension and being crucial for self-organizing teams takes a lower place. Similarly, cross functionality, which is proposed by many methods, is relatively at a low level. We see that this principle is supported by DevOps, which is bounded by this principle very profoundly.

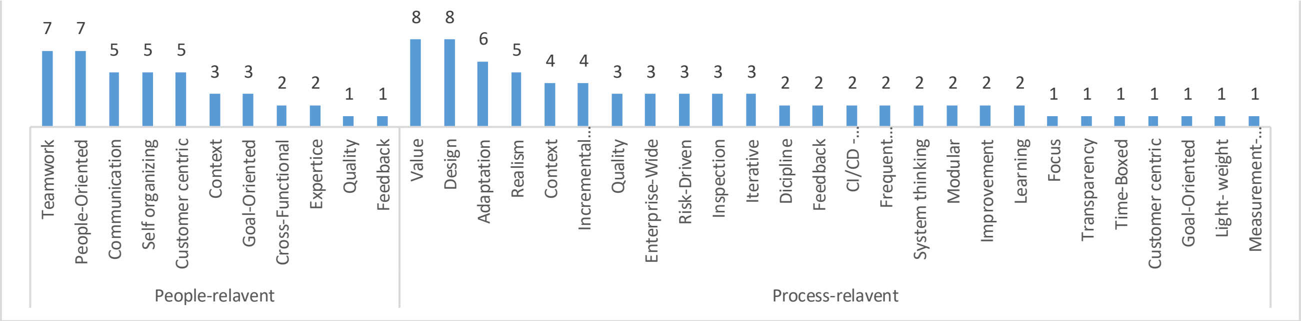


Fig. 3. Count of Principles

Unsurprisingly, we can say that those methods with a process expression in their names such as Agile Software Process (ASP) or in their definitions such as of Scrum outweigh the process side. As the first instance of Agile methods, Evo approaches agility mainly from the process side. Lean Software Development poses a very process-oriented image with its focus on the waste in the processes.

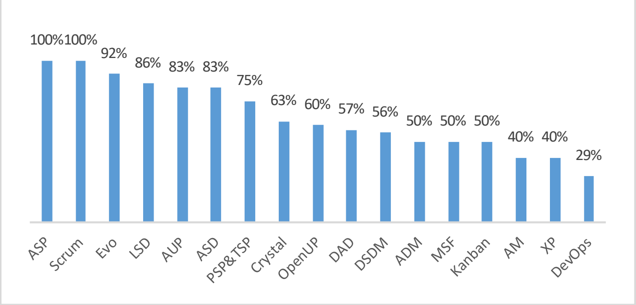


Fig. 4. % of Process-Relevance of Principles

Although DevOps has many aspects that touch processes, it is remarkable that DevOps is at the forefront of people's dimension. Agile Modeling bases on XP in defining its values. In the context of this study, since XP and AM are included with values instead of principles and as values are more people-oriented, it can be considered normal that these two are seen at the forefront of people-relevant dimension.

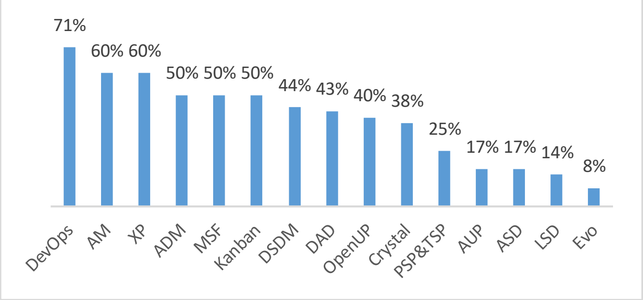
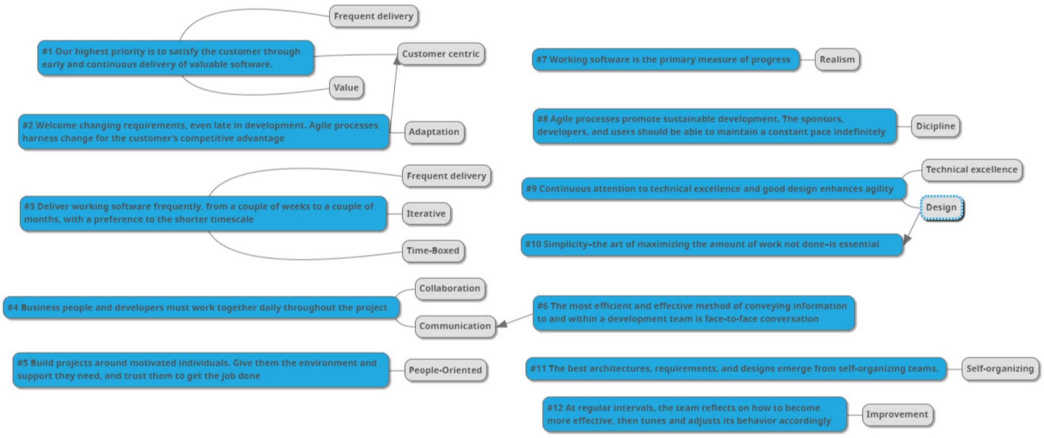


Fig. 5. % of People-Relevance of Principles

C. *Comparison of Principles with the Manifesto (RQ3)*

When looking at the degree of overlap of the principles (L2) with those of the manifesto, it is seen that more than half of the determined principle categories are touched by the manifesto. Cohen argues that all Agile methods follow the four values and twelve principles of the Agile Manifesto [16], yet they provide more principles than the manifesto in terms of the coverage.


#1 Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

#2 Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage

#3 Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale

#4 Business people and developers must work together daily throughout the project

#5 Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done

#7 Working software is the primary measure of progress

#8 Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely

#9 Continuous attention to technical excellence and good design enhances agility

#10 Simplicity–the art of maximizing the amount of work not done–is essential

#6 The most efficient and effective method of conveying information to and within a development team is face-to-face conversation

#11 The best architectures, requirements, and designs emerge from self-organizing teams.

#12 At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly

Fig. 6. Map for the Manifesto Prıncıples

However, even if the feedback is not explicitly stated, it is assumed to receive feedback on the delivery of the product with the early delivery, providing inspection accordingly. Similar logic can be put forward for incremental development in relation to iterated progress. Cross-functionality can be seen as a prerequisite for self-organizing teams. Similarly, although transparency is not explicitly stated, it can be considered as a capability gained automatically by establishing intensive (especially on a daily basis) communication. Principles relating to goal-oriented, focus and modular may not be seen as primary to be included in the manifesto, within a dedicated mentioning.

D. *Evaluation of the Principle by Experts (RQ4)*

The first author conducted semi-structured interviews with two experts to evaluate the principle categories (L2), especially in terms of their contribution to agility. Expert A has 15 years of experience in total, of which 5.5 years as a product owner in a bank in Turkey applying Scrum. Expert B has 13 years of experience in total mainly from two different banks in Turkey, of which 4 years in a Scrum development team. The following statements directly convey the views of experts on the principles.

Expert A states that each of these principles determined supports the agility. Using these principles together in the whole picture will be beneficial for maintaining balanced, healthy and sustainable agility. According to her, although adaptation is important, the market has a lot of emphasis on it, which can lead to an unbalance in some other points. For example, in some cases of adaptation without a balance, quality, enterprise-wide, risk-driven, systematic, realistic (adaptation to realistic changes) approaches and sufficient inspection phases may be damaged. This approach may lead to the emergence of unsustainable structures that will not benefit the customers in the long run. Teams that move away from the holistic picture with the effect of adaptation pressure can result in isolations across the teams themselves, such as happening in impact analysis mostly conducted in non-sufficient and isolated ways. In addition to agility, the necessity of elements such as expertise and discipline to support it manifests itself. Expertise for instance is important enough, as becoming a prerequisite for self-organizing teams to be able to self-organize. Unstable teams and teams with a low level of expertise unlikely to become self-organized. In addition to adaptation pressure, time-boxing may lead to compromise on quality and value with a similar effect.

She states although value and customer orientation are important, a blindfolded dedication to the customers may cause human values of teams to be ignored and remained in the background. With a customer-driven approach, development teams come to a more passive position, and customer demands that do not go through enough filters of the customers put more pressure on the teams. Considering these situations, principles such as system thinking, organization-wide, quality and realism stand out for sustainable agility. In addition, incremental and iterative development, teamwork, cross-functionality come into prominence in a way supporting agility fundamentally.

Expert B asserts that transparency contributes to reality by supporting open and clear environments, in a way of reducing reworks. She adds that frequent delivery increases quality. Frequent delivery, on the hand, cannot be possible in some cases depending on the nature of the project. Progressing iteratively reduces the risk for the users and developers as the users see the increment at the early stages and give feedbacks. For developers working the design up-front as much as possible reduces the risk as well.

According to her, teams with a deadline coming with the iteration time-box can have positive and negative effects depending on the situation. In both cases, determining the end of the iteration by the teams supports self-organization. It supports meetings to be more productive. However, for self-organized individuals, time-boxing will be meaningless. Daily meetings and time- boxing will be effective in a positive way with pressure for non-self-motivated individuals. However, this pressure can also have a negative effect on some people.

She says it is usually expressed that organizations trust the Scrum teams, yet it is a utopia to trust the team in an absolute manner. Factors and rules outside the teams do not allow the teams to be truly self-organizing. Self-organization can also be a problem, especially in the setup stages of Scrum. Scrum does not respect the context dependencies much. Depending on the context, it may be difficult to set up Scrum with its factory settings, especially during the transformation stages or in disciplined environments like in a bank.

She adds that Agile [Scrum] comes with a customer-oriented process setup. Customer feedback directs the development. What the customers want is accepted as master and generally does not go through a filter. Project-based team structures eliminate the need to work on a modular basis. Cross-functionality is thus provided for the project via such temporary teams. It is actually a structure that supports context diversity and process flexibility.

VI. Discussion

Among the **methods**, some of them focus on project management (like Scrum and DSDM), while some others focus mainly on software development activities (like XP, Crystal), mostly on the team level, ignoring organization-wide perspective. The main reason for being mostly on the team level may be that the creators of the methods mostly come from the software development background. While those such as DevOps and Kanban provide a continuous stream for delivery (continuous planning, integration, delivery, feedback etc.), some others like DSDM, Scrum uses segmented units of the timeline to manage the pipeline. Thinking the time within the segmented iterations like a sprint in Scrum can be an advancement for a big bunch of development lines of plan-driven approaches of yesterday, yet it cannot be regarded as a contemporary method of today. Contrary to the agile logic, handling these static time frames of iterations with strict planning and expecting a concrete product at the end is very instance of a plan-driven approach. Instead, to keep with fluctuations of the complexity of the reality that is at very atomic level of granularity, a continuous approach to development providing a very mutual and natural atomic level of reflections may be needed. This is probably why DevOps, Scrumban and Kanban stay alive among those a few, by providing a continuous stream for the pipeline.

Staying alive among those a few, XP and DevOps take place as focusing on people-related issues in terms of principles. However, being human oriented and being-system oriented seem to be a binary choice within the methods. While many models establish their main structure on the roles of people, there are some methods such as Kanban, LSD that focus the system rather than people.

As the most used method, an interesting issue with Scrum that takes a process-oriented approach to development, assertively delegates this duty of process-orientation to its a few basic roles of people. And as it is expected, this intense process orientation is prone to be derailed by people who are naturally far from providing a standard approach to what these intense processes require.

The aim of LSD is to approach the zero (waste) point. Agility leans more on the expansion of perspectives; learning (fail fast), reworks (creating features only to understand customers better at the earliest) and so on. This "haste” to respond quickly in Agile may “make waste", implying that Lean and Agile approaches can be contrast serving in two different directions. However, there is a Lean perspective in the manifesto by advocating just enough documentation, reducing “ineffective communication” occurring in the hierarchy, tools and processes. This shows us that the Lean and Agile approaches are used together in the manifesto, maybe with confusion, even if they contain some contradictions.

When it comes to the **manifesto**, interestingly, we see no quality-related emphasis on it. Another interesting point in the manifesto is that the agility of Agile Software Development is considered a separate and isolated body, not directly connected with the organization wide perspectives. The main reason for this may be that the manifesto writers come from the software development background, too. In the context of software development, it will not be enough to include the customer in the processes. Therefore, considering the Agile Software Development separate from the whole organization come with some issues. Another important issue in the manifesto appears in contextualization. It is usual for the reality to vary depending on the context, which calls for each unique practitioner to define a space for their context and to shape their own agility within this space. However, the manifesto does not explicitly refer to the context dependencies, which is an important dimension of realism, nor is there any concern about the expertise of people, which is a crucial factor to deal with the context, in the vertical dimension. Although in Agile approaches, T-shaped specialization is recommended instead of general specialization with assuming that it contributes to collaboration within cross-functional teams, yet it only provides a horizontal dimension to the context-related issues, which should remind us not to ignore the issues related to the depth of the context, especially of the complex world.

Some of the determined **principles** (L2) are close to each other (such as teamwork and self-organizing or incremental and iterative development, frequent delivery and CI/CD), some are closely supporting each other (such as feedback, communication and transparency). Others are not open to debate, as they make an absolute positive contribution (such as improvement and learning). We will discuss here debatable ones, in general, without mentioning the differences between those close to each other.

As one of the principles, moving within iteration is an old school tradition, seen mostly in the first generation of the methods. Maintaining this tradition with building walls (with fixed times) and trying to live agility within limitations of these walls of iterations are a kind of reduction to and conflict for people who have more atomic level, more sensitive, stronger agility capabilities in themselves. As an excuse, fixing iterations with a “deadline” to speed up the development to assure the fulfillment of the customer's top present needs, or using fixed iterations for motivating development teams (as stated by Expert B) are just expected benefits. Using a combination of adaptation and iteration with time-boxes may create artificial pressures on teams causing compromise on some other values (Expert B). It implies that this artificial ‘solution’ produced for indirect problems (not being motivated, not being value oriented) creates a cause for another problem; trying to imprison the reality of the future by artificial parameters of time. However, the reality of future is so dominant and free that it does not fit in an artificial frame of time (like sprints with a fixed end), then it gets out of iteration limits, enforces obedience of all other parameters. For example, towards the end of the fixed iteration which does not progress according to the plan enforces a situation where the scope or quality will be compromised. It is reminded that time is one of the strongest among parameters, then people should learn to get along with it instead of imprisoning it.

The term artificial means iteration is not in a pure form of time itself rather a kind of proxy of it, a sort of representative of the time at a different platform. In this sense, it takes the process away from reality. Moreover, iteration-based planning means adding determinism into the complexity of the future, especially if it comes with a fixed end time. This approach indicates an attempt to manage non-deterministic software development with deterministic methods. Using iterations as a batch feedback method with some static rituals is to communicate with an artificial cycles as well. For instance, an issue at the beginning of the sprint may, not necessarily but most probably, delay to the review or retrospective meetings that are located at the end of the sprints. Fixed rituals break the natural flow of the reality (such as in getting feedback when it is ready). So, it is recommended to synchronize the loop of feedbacks with its own cycle of the realism instead of an artificial one. Thus, with iterations saved from fixed events, the sooner the solution is delivered, the sooner feedback can be received.

Realism is to be driven by the reality itself instead of the proxy of it. For example, processes to organize real operations aiming to be a projection of the reality, with trying to represent it or even direct it by going ahead of it are also a sort of artificial proxies. However, a process is not the reality itself. It is a kind of artificial entities produced by humans. After all, models are human-made products, and every human-made product (software, hardware, ideas etc.) is defective. Like in the time parameter, the reality as the master dominates the static [process] frameworks, models and methodologies that try to be real.

Self- organization increases the ability to respond to change while decreasing the speed of response for decision-making in quickly, easily and adequately manner (as stated by Expert A). Advantageously, it strengthens the concept of “move” in the definition of agility by means of delegating the work to those who know it closely and expanding decision capabilities, yet it should not be regarded as a way that contributes to agility in absolute terms. Cross-functionality reduces the cost of communication by gathering the necessary competencies into the team and enables rapid actions. However, self-sufficient (!) teams weaken their abilities in the holistic picture with their possible estrangement. Even though the ability, speed and convenience of moving increase inside the teams, these capabilities may be in danger in the context of multiples teams (as stated by Expert A).

Agility is easier when managed in the abstract dimension, which calls for more design up-front. Managing the solution with a “concrete running software" may be costly, hand-binding, and waste. If the customer's need is "discoverable" a bit from the front, the up-front investigation should be located. Agility is also more sustainable when combined with system thinking, quality and organization-wide perspectives (Expert A) and discipline (Expert A, B).

Software developers develop software mostly for people, with people. However, the human is not a pure representative of the deepest level of the reality that is in a perpetual state of change. As a proxy, they cannot perceive and convey the reality as it is, sometimes deliberately and they add their natural interpretations, perspectives, and limitations of their context to the reality, making them a very strong decrement point in transmitting it. Hence, driving the change solely by people may be misleading (as partially stated by Expert A).

In parallel, Parnas and Clements [40] states (as paraphrased by [2]) that a system's users seldom know exactly what they want and cannot articulate all they know. Even if they could state all requirements, there are many details that we can only discover once we are well into implementation. Brown's study [41] reports three different perspectives about the same project varying dramatically with the role of people. The customer will of course be a mediator of the change. The important thing here is to be the seeker of the reality together with the customer and not regarding customers sacrosanct and accepting them as the absolute point of the reality. There is less visible yet another crucial layer between the customer and the reality to discover with them together.

VII. Conclusion and Future Work

The study does not attempt to redefine agility in the software solution development in a full-fledged way. It rather makes an evaluation based on the principles, considering a particular approach to the definition of the agility, with some threats to validity when considered low level validation by experts. Even so, the study may provide specific contributions, especially with its progressive position that has two faces: locating the principles on the center, looking at the relationship between the principles and the methods and examining how these principles support agility. In this sense, as future work, it can be investigated to what extent a specific method supports agility through these principles. However, as the next study, we prefer to improve these principles by combining results from other related studies and examine how and to what extent each element in the final set supports agility.

REFERENCES

[1] A. Mordi, and M. Schoop, “Making It Tangible–Creating A Definition Of Agile Mindset”, ECIS, 2020.

[2] C. Larman and V. R. Basili, “Iterative and incremental developments: a brief history”, Computer, vol. 36, pp.47–56, 2003.

[3] E. Trist, “The evolution of socio-technical systems”, Occasional paper, vol. 2, 1981.

[4] H. Takeuchi, and I. Nonaka, “The new new product development game”, Hardvard Business Review, vol. 64, no.1 1986.

[5] G. Morgan, Images of organization, Sage Publications: Beverly Hills, 1986.

[6] P. Hohl, J. Klünder, A. van Bennekum, R. Lockard, J. Gifford, J. Münch, and K. Schneider, “Back to the future: origins and directions of the “Agile Manifesto”–views of the originators,” Journal of Software Engineering Research and Development, vol. 6, no.1, 2018.

[7] N. G. Abbas, A. M. Gravell and G. B. Wills, “Historical roots of agile methods: Where did “Agile thinking” come from?, International conference on agile processes and extreme programming in software engineering, pp.94-103, 2008.

[8] A. Cockburn and J. Highsmith, “Agile Software Development: The Business of Innovation”, Computer vol. 34, no.9, pp.120–127, 2001.

[9] P. Kruchten, “Contextualizing agile software development”, Journal of Software: Evolution and Process, vol. 25, no. 4, pp. 351-36, 2013.

[10] K. Conboy, and B. Fitzgerald, “Toward a conceptual framework of agile methods: a study of agility in different disciplines” ACM workshop on Interdisciplinary software engineering research, pp.37-44, 2004.

[11] J. Highsmith, Agile Project Management, Boston: Addison-Wesley. . 2004.

[12] J. Miler and P. Gaida, “On the agile mindset of an effective team–an industrial opinion survey”, Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 841-849, 2019.

[13] https://stateofagile.com/

[14] R. Suddaby, “Editor's comments: Construct clarity in theories of management and organization”, 2010.

[15] C. Larman, “Agile and Iterative Development: A Manager's Guide”, C. Alistair, H. Jim, (eds.), Pearson Education: London, 2004.

[16] D. Cohen, M. Lindvall and P. Costa, “An Introduction to Agile Methods”, Advances in Computers, pp.1–66, 2004.

[17] T. Gilb, “Evolutionary Development", SIGSOFT Softw. Eng. Notes, vol. 6. No.2, 1981.

[18] J. Stapleton, DSDM, dynamic systems development method: the method in practice, Harlow: England, 1997.

[19] K: Schwaber and J. Sutherland, “The scrum guide”, Scrum Alliance, 2011.

[20] https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf

[21] M. Aoyama, "Agile Software Process model," 21st International Computer Software and Applications Conference, 1997.

[22] E.S. Raymond, The Cathedral and the Bazaar, O'Reilly: Cambridge, 1999.

[23] A. Cockburn, Surviving object-oriented projects: a manager's guide, Addison-Wesley: Longman Publishing, 1998.

[24] J. A. Highsmith, Adaptive Software Development: A Collaborative Approach to Managing Complex Systems, New York: Dorset House, 2000.

[25] K. Beck, Extreme programming explained: embrace change, Addison-Wesley Professional, 2000.

[26] P. Coad, J. D. Luca and E. Lefebvre, Java modeling color with UML: Enterprise components and process with Cdrom, Prentice Hall PTR, 1999.

[27] M.A. Cusumano and D. B. Yoffie, “Software development on Internet time.” IEEE Computer, vol. 32, no.10, pp.60-69, 1999.

[28] S. Ambler, Agile modeling: effective practices for extreme programming and the unified process, John Wiley & Sons, 2002.

[29] S. Ambler, Agile database techniques: Effective strategies for the agile software developer, John Wiley & Sons, 2003.

[30] M. Poppendieck, T. Poppendieck, Lean Software Development: An Agile Toolkit, Addison-Wesley Professional, 2003.

[31] http://www.ambysoft.com/unifiedprocess/agileUP.html

[32] M. Turner, Microsoft solutions framework essentials: building successful technology solutions, Microsoft Press, 2006.

[33] P. Kroll, B. MacIsaac, Agility and Discipline Made Easy: Practices from OpenUP and RUP, Pearson Education, 2006.

[34] G. Kim, J. Humble, P. Debois, and J. Willis, “The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations”, IT Revolution, 2016.

[35] C. Ladas, “Scrumban-essays on kanban systems for lean software development”, Lulu.Com, 2009.

[36] D. J. Anderson, Kanban: successful evolutionary change for your technology business, Blue Hole Press, 2010.

[37] S. W. Ambler, and M Lines, Disciplined agile delivery: A practitioner's guide to agile software delivery in the enterprise, IBM press, 2012.

[38] Watts, Using a defined and measured Personal Software Process, https://www.amazon.com/Introduction-Software-Process-Watts-Humphrey/dp/020147719X.

[39] D. Rosenberg, M. Stephens and M. Collins-Cope, Agile development with ICONIX process, New York: Editorial Apress, 2005.

[40] D. L. Parnas and P.C. Clements, “A rational design process: How and why to fake it”, IEEE transactions on software engineering, vol.2, pp. 251-257, 1986.

[41] A. D. Brown, “Narrative, politics and legitimacy in an IT implementation, Journal of Management Studies, vol. 35, pp.35-58, 1998.

[42] N. Ozkan, “Imperfections Underlying the Manifesto for Agile Software Development”, 1st International Informatics and Software Engineering Conference (UBMYK), 2019.

[43] H. van Manen, H. van Vliet, "Organization-Wide Agile Expansion Requires an Organization-Wide Agile Mindset", Product-Focused Software Process Improvement. Ed. by A. Jedlitschka, P. Kuvaja, M. Kuhrmann, T. Männistö, J. Münch, M. Raatikainen, pp. 48–62, 2014.

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 731–737 DOI: 10.15439/2020F186 ISSN 2300-5963 ACSIS, Vol. 21

Scaling agile on large enterprise level with self-service kits to support autonomous teams

Alexander Poth
Volkswagen AG
D-38440 Wolfsburg,
Germany
alexander.poth@volkswagen.de

Mario Kottke
Volkswagen AG
D-38440 Wolfsburg,
Germany
mario.kottke@volkswagen.de

Andreas Riel
Grenoble Alps University
G-SCOP Laboratory
F-38031 Grenoble,
France
andreas.riel@grenoble-inp.fr

Abstract — **Organizations are looking for ways of establishing agile and lean delivery processes. In this paper, we propose a particular way which based on self-service kits (SSK's). The SSK approach can be used to share expert knowledge in an agile and scalable way to the teams by offering them approaches, methods and tools with background information about the addressed topic. An SSK is provided as a digital bundle of artifacts that help solving an issue related to agile teams. Built upon the pull-principle, it supports team autonomy during teams' delivery procedures. An SSK addresses generic as well as domain specific topics. As all SSK's share a common structured approach to supporting an agile organization, they help systematically scaling expert knowledge. This leverages establishing best practices elaborated by experts in a large scale organization in a native agile manner. As an SSK is structured as a "how-to" guide including templates for learning by doing, it helps emphasizing quality aspects too. We demonstrate an example of the systematic application of the SSK approach as well as its scaling in the Volkswagen Group IT.**

I. Introduction

To achieve the agile transition of large enterprises, approaches beyond coaching are needed for non-linear scaling. As coaches are limited resources that cannot be easily increased on demand, new ways for scaling agile know-how, methods and tools have to be identified and implemented. The challenge is that people involved in the transition have to learn and understand the new agile mindset with their specific -values and principles [1, 2] and its characteristic approaches. An inherent job of coaches is to facilitate these learning activities and the agile mindset adoption. In this context, the term self-service kit (SSK) shall denote an approach to enabling teams to handle specific topics of their product and service related work. The way of facilitation isagile without team external persons (coaches etc.) by providing relevant knowledge and artifacts in digital form and in a pull-based manner.

The objective of this work is to propose and evaluate such an approach within a large corporate environment. Based on observations of daily business during the facilitation of transitions, we derived the following requirements for approaches which support scaling without direct team-external human integration and interaction:

R1) The scope of the scaled facilitation, deliveries have to be designed as to offer a valuable outcome to the teams.
R2) To ensure scaling, the deliveries have to be completely digitalized and offered anytime (24*7).
R3) Guidance is needed for the teams during application and learning.
R4) Teams need background knowledge about the facilitation delivery to be able to make adaptations to their specific context.
R5) A feedback loop is needed to request an expert like coaches for additional support.
R6) Quality has to be built in the delivery procedure to avoid scaling of errors.

These requirements lead to a combination of different learning and facilitation approaches having to be considered during the development of a solution. In order to do so, we use the design science approach [3], taking into account the R1 to R6 systematically.

Section II introduces related work, section III provides an overview of the SSK approach and section IV characterizes examples of selected SSK's. Section V elaborates an experience report about the SSK application, while section VI concludes and section VII shows next steps and future work.

II. Related work

This section identifies related work based on key topics. The literature research has been conducted in alignment with Webster & Watson [4]. As the term SSK has not been used in literature so far, the search structure has been aligned with related concepts .

A. Blended Learning

Blended learning combines different web-based technologies with various pedagogical approaches. It integrates different instruction approaches and brings together working and training [5]. One of the web-based technologies of e-learning are labs [6]. Labs are used for practical training guided by instructions. However, labs are experimentation environments that normally represent only a limited set of

This work was not supported by any public organization or funding.

real-world scenarios and their contexts. In the case of SSK's, the lab is replaced by the real-life context. Therefore, it is important that both the problem identification and the solution guidance is appropriate in order to avoid significant failures [7] leading to harm [8] either by misguidance, misuse or even by accident. Consequently and according to Bloom's model of learning [9], the minimum SSK objective has to be "applying" rather than "understanding" or even lower, which is typically the minimum learning target for Web-based trainings (WBT). WBT are established approaches to train people online. While WBT's transfer knowledge [10], they do not have the objective of guiding the transfer and re-contextualization of the transferred knowledge to a specific task or entire project. From that perspective, SSK's have learning objectives and maturity expectations that are significantly superior to common WBT. This further augments the need of setting SSK's into an adequate design [11] context, which depends on a lot of influencing factors.

B. Problem Based Learning

Problem-Based Learning (PBL) [12] is a topic related to SSK's because the latter address particular problems while supporting the SSK applicants in solving them. Approaches to providing guidance are analyzed in [13]. The SSK, however, does not pose a particular problem but rather provides the appropriate set of questions to ask to identify a problem in practice, and leverages on this problem identification process to propose methods that help in the problem resolution process motivated by [14]. As design patterns are widely used in industry [15] there is a difference in the application of a pattern to build a product, service or process flow by standardized patterns. The understanding and learning to be able to adopt the methods and tools is offered in SSKs is the additional objective.

C. Learning by Doing

Learning-by-doing is a useful approach in practice and industry [16]. SSK's foster the learning-by-doing method based on goal-based scenarios [17] by adopting a guided approach through the combination with other learning concepts, in particular blended learning. There exist many different blended learning approaches [11]. In this context, the focus is mostly on self-paced and asynchronous formats [18] extended by synchronous online formats for the online meetings of groups to work together on a topic.

D. Scaling Agile

Scaling Agile focuses on establishing a set of agile methods for a building complex systems within an organization [19]. Existing many different approaches for scaling agile with their specific benefits and issues [20]. However most of the established scaling methods and framework do no scope the how to establish the knowledge about the agile mindset and methods in the teams they focus on the demands for methods like [21] and their implementation order like in ASM [22]. Knowledge sharing and improvement is still a topic in scaling agile [23]. Coaching is the preferred knowledge transfer approach like in SAFe [24] with the certificated trainers and role specific trainings [25].

E. Agile Teams Demands

For example, the SAFe Lean-Agile Principle #8 recommends autonomy for employee engagement [26]. Other agile approaches emphasize T-shape [27] skills to form interdisciplinary, independent and autonomous teams. Team autonomy in large-scale corporate organizations is efficient if goals are well defined and transparent on a team-level [28]. For SSK's to be most effective, this implies that they have to support setting and achievement of goals in a effective way [29]. Furthermore, autonomy and self-organizing teams come together and need cross-functionality, which is based on sharing of knowledge [30] that is available both with and outside the teams.

F. Quality and Life-Cycle Management

To assure the quality of learning materials, embedding the latter in a life-cycle is useful [31]. Quality assurance is an established habit for learning materials for distance learning artifacts [32] like for the curriculum and instructions. To achieve organization-wide standardization, a systematic governance has to be established [33]. International standards have been elaborated [34] like for open and distance universities with UNIQUe.

III. Self-Service Kit Approach

To scale agile in an organization without explicit time intensive coaching of all teams SSKs are an alternative know-how transfer approach to the teams. In our context, an SSK is a combination of a web-based training (WBT) [35] and a digital tutorial [36] provided by domain-experts to a large number of – in general – geographically distributed users [37]. A WBT facilitates the delivery of specific knowledge to people needing it or asking for it. This pull of SSKs know-how by the teams supports autonomy. Furthermore the setting supports the agile mindset with the support of develop adoption know-how to enable the teams to enhance SSKs for their specific demands.

An SSK is designed to support teams to do their work with a high quality. To realize this, each SSK has to ensure that the relevant knowledge needed to perform the work is delivered to the team. The SSK approach supports autonomous teams in applying SSK's by its design. This lead to the point that SSK's can be used for autonomous knowledge scaling and as a key element of a flywheel approach for agile transitions. Depending on the individual scope of a specific SSK, the knowledge has to be identified, documented and integrated into supporting artifacts like checklists and other tools. As SSK's shall be used many times and in several different teams and places, assuring a high quality level of SSK's is important to avoid mistakes on a large scale. To this aim, SSK's need a rigorous design, production and delivery procedure, which experts of the specific SSK topic perform. As experts are not always good trainers and educators, they can themselves get support from SSK's for their SSK development. Figure 1

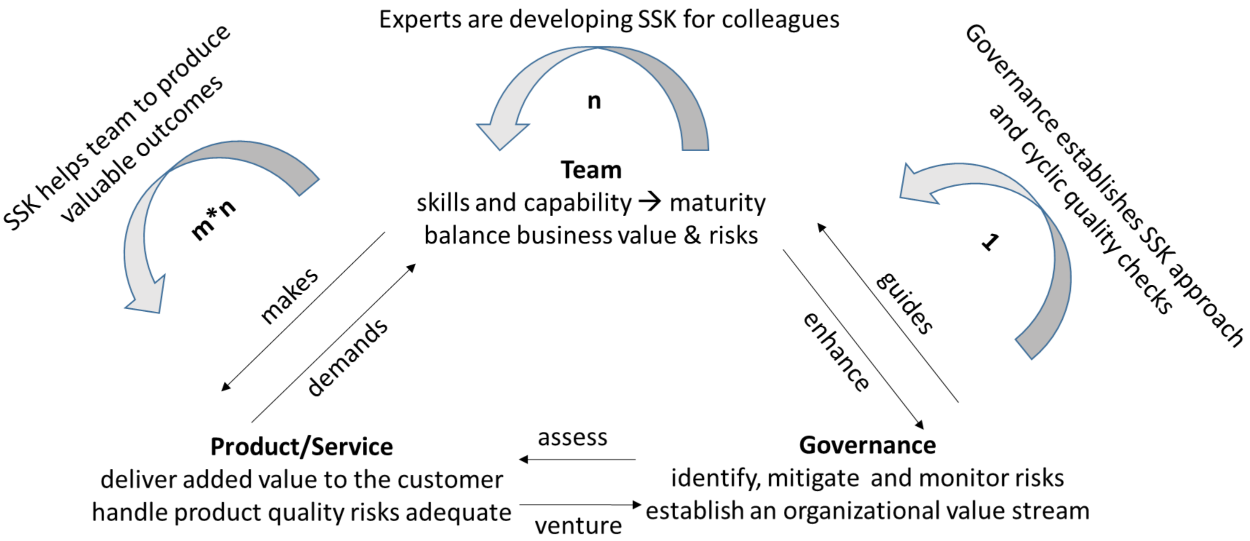

Experts are developing SSK for colleagues
n
Team
skills and capability → maturity
balance business value & risks
SSK helps team to produce valuable outcomes
m*n
Governance establishes SSK approach and cyclic quality checks
1
makes
demands
guides
enhance
Product/Service
deliver added value to the customer
handle product quality risks adequate
assess
venture
Governance
identify, mitigate and monitor risks
establish an organizational value stream

Fig. 1: Value chain of SSK delivery approach by one governance, n SSK's and n*m outcomes.

shows the relation between one (1) governance to a few (n) experts that develop a particular SSK, as well as many (m) applications of that very SSK. The basic structure with governance, team and product/service has been introduced in the context of the enterprise transition approach [38] and is enhanced to the SSK approach for autonomous scaling in this context.

The governance establishes the SSK approach with its development and delivery procedures. This includes templates and platforms for digital delivery of SSK's. Experts of different teams form a *development* team to develop an SSK for a specific topic. As the experts are "grounded" in normal teams of the organization they know about the latter's demands and issues and therefore can address them by design during the SSK development. For different SSK's, different experts work together in expert teams. They also have to ensure the cycle *updates* of the SSK (R5). These updates address feedbacks (R5) for improvement and the alignment with the development of the state-of-the-art (R6). The governance regularly checks that these updates are actually made for all SSK's which are in *delivery*. In case that an SSK has no experts for adequate maintenance, the SSK is marked as "*retired*" by the governance to show all users that they should not use this SSK anymore. Based on this generic approach with the life-cycle states for SSK *development, deliver, update* and *retired,* a framework is established to provide SSK's to the organization (R2).

This setting makes it easy for an organization to start with one lean governance for the SSK framework and scale to as many SSK offers as there are experts who produce and maintain SSK's. The instantiation of each SSK is independent of these in general highly limited human resources as long the SSK is delivered in a digital way to its consumers (users).

From a quality perspective and with respect to the objectives they want to help achieve, three types of SSK's shall be distinguished (R1):

- Product quality: the SSK's objective is to improve the product or service with its outcomes.
- Process quality: the SSK's objective is to improve the process of a service or product delivery.
- Team quality: the SSK's objective is to improve the team who produces and delivers a product or service.

All types of SSK's have as common objective to facilitate scaling knowledge within the organization in an agile manner. However, each type has some specific aspects to focus on. The following section presents examples for each type.

IV. Self-Service Kit

All SSK's shall include the following artifacts (R3):

- Introduction: a template for all SSK's to ensure their common structure including: scope, context, outcomes, application and references to further artifacts of the SSK.
- Working artifacts: one or more working artifacts are in an SSK. They are highly specific to the scope of that SSK. They are designed with the purpose to guide the teams during the outcome production.
- Background information: provides to the users information about the design requirements and constraints of the SSK and the development approach and evaluation context of the SSK. Furthermore it offers detailed descriptions of the working artifacts design.

All artifacts have information about the producer (author) and a version. Based on these three artifact types, all SSKs are build. However depending on their scopes, the specific instantiation is different (Table I). All SSK's have to be designed to offer the teams the opportunity to adopt the SSK to their specific demand by addressing Bloom's taxonomy domains with high learning objectives (R4). This is also important because the teams are working and learning by doing in a real life lab and should be able to see risks by mis-

TABLE I.
DIFFERENTIATION OF SSK TYPES ABOUT PRODUCT/PROCESS/TEAM-QUALITY

Aspect	Product	Process	Team
Scope of the SSK	Technology	Workflows and activities	Behavior
Outcomes of the SSK	Questions and checklists	Questions and methods	Questions and indicators
Evidences of (correct) usage of the SSK in the final instantiation	Objective evidences often persistent	Evidences depending on implementation and often temporary	Impressions often subjective (no/weak evidences) in a specific setting – non deterministic behavior
Bloom's taxonomy cognitive domain	*Evaluation* of product characteristics	*Evaluation* of adequate sequences of activities	*Evaluation* of adequate improvement action for the team
Bloom's taxonomy affective domain	*Organizing* of usage, features, capabilities of products	*Organizing* of workflows and activities for a specific purpose	*Organizing* the behavior and knowledge of the team to identify improvements
Bloom's taxonomy psychomotor domain	*Origination* of usage, features, capabilities of products	*Origination* of workflows and activities for a specific purpose	*Adaptation* of interacting/working methods to fit team potentials
Problem-based learning	Problems on tangible objects are good to measure and improve	Problems are mostly visible on their interface of activity outcomes and interactions	Problems are often related to behavior and their actions - outcomes can be used as indicators

using SSK's (R6). The following sections are showing examples for the three different quality types. Table I shows the product, process and team quality with the learning aspects within a SSK.

A. Product Quality

The development of product quality related SSK's is driven by outcomes for a specific product or service. These products are driven by technology that has to be handled adequately by the teams. To support the usage and adoption on a large scale of specific technologies that are new to the organization, such as machine learning [39] or serverless [40], SSK's can be useful. As presented in Table I, the SSK guides with questions about the technology adoption and offers checklists about the technology usage. As a product is a "real outcome", the valuable product related outcomes of the SSK are mostly persistent and measurable evidences. Mapped to Bloom's taxonomy, a product quality related SSK has to enable users in the cognitive domain for *evaluation* of product characteristics. This high learning level is not needed in every usage, however it is the objective of the SSK to support up to this level. In the affective domain, the high level of *organizing* of the product usage and its features or capabilities is a supporting objective. Furthermore, the psychomotor domain with *origination* is a valid objective to enable the agile teams to develop new ways of usability and interactions with the software. Not all product related SSK's need these high learning curve in all domains, but every SSK design has to check how much learning is needed (R4) to reach the expected outcomes (R3). With a problem-based learning view, a product related SSK makes it easy to learn as they related to tangible objects which typically can be measured and improved by observation of change impacts.

B. Process Quality

The development of process quality related SSK's is driven by outcomes that build workflows or activities in procedures. For example, our Level of Done approach derives organization specific procedures to be aligned with regulation [41]. In the context of our hybrid SSK for the systematic elicitation of product quality risks [42], a design thinking process is used to ideate specific product characteristics while being part of our Level of Done approach. As presented in Table I, the SSK guides with questions about workflows and activity adaption and offers methods to development and adoption. As a process is a "logical outcome", the valuable outcomes are descriptions and interfaces of workflows and activities. Depending on the implementation, the evidences are temporary (i.e., an interaction between individuals) or persistent (e.g. workflow logging). Mapped to Bloom's taxonomy, a process quality related SSK has to enable users up to the cognitive domain for *evaluation* of workflow sequences or activities. In the affective domain, the high level of *organizing* of the process workflow usage and its activities is a supporting objective. Furthermore, the psychomotor domain with *origination* is a valid objective to enable agile teams to develop new ways of usability and interactions with their workflows and procedures. Not all product related SSKs need such a high learning curve in all domains, however every SSK design has to check how much learning is needed (R4) to achieve the expected outcomes (R3). With a problem-based learning view, achievements of a process related SSK are mostly observable and measurable thanks to their interfaces and activity outcomes.

C. Team Quality

We address team quality aspects with agile Team Work Quality (aTWQ) [43]. As presented in Table I, the SSK guides with questions about the indicators of behavior and interactions between individuals. Both behavior and interactions underlying subjective observations and impressions, the evidences are rather indicators. Furthermore, behavior is often specific for a situation or setting which makes it non-deterministic. Mapped to Bloom's taxonomy, a team quality related SSK has to enable users up to the cognitive domain for *evaluation* of adequate improvement action for the team. In the affective domain, the high level of *organizing* of the team's behavior and knowledge to identify improvements is a supporting objective. Furthermore, the psychomotor domain with *adaptation* is a valid objective to enable the agile teams to leverage the potential for better fitting interactions and working methods to the specific team. Not all product related SSK's need such a high learning curve in all domains, however every SSK design has to check how much learning is needed (R4) to achieve the expected outcomes (R3). With a problem-based learning view, a team related SSK does not make this easy because only the outcomes of behavior or interactions can be observed. This is an indirection rather than a direct measure. However, the outcomes are what is used in the real life too. In this case, the intention of the behavior or interaction is not the fact that matters; only the outcome is the valuable factum. For learning, this indirection can be difficult in case of missing openness between the interacting people (in case of lack of trust etc.).

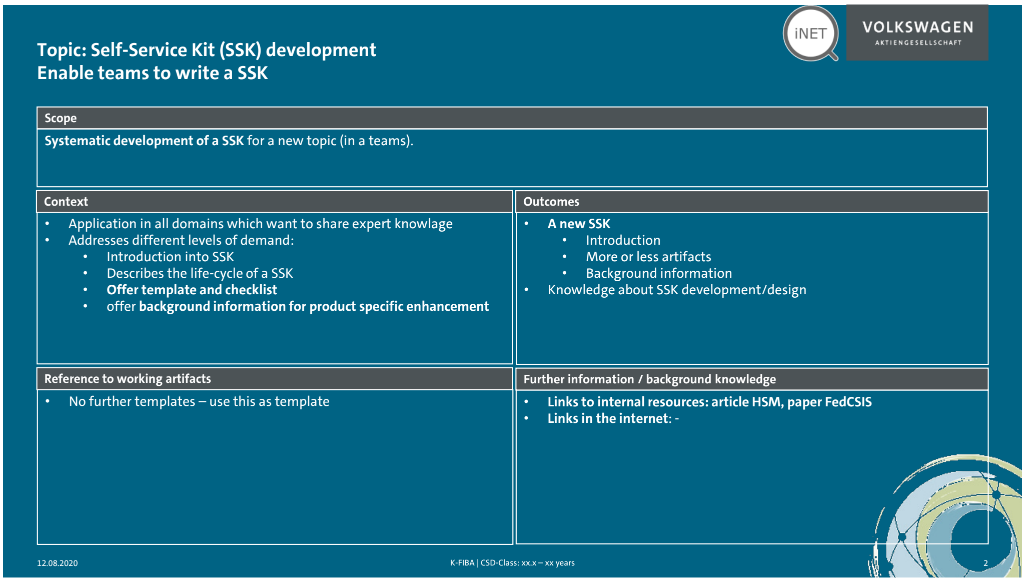
These three SSK types have proven useful to support the entire agile transition approach of Figure 1. The product quality SSK's support the product/service development. The team quality SSK's facilitate the teams by their maturity. The process quality SSK's are useful to establish processes and integrate those in the organizational governance. This leads to opportunities for the entire organization to scale all relevant parts at the same time thanks to the holistic SSK approach. SSK deployment in different organizations implies the challenge of identifying all relevant topics at the right time to have the SSK's developed just in time as they are demanded and needed by the organization and their teams. This has to be realized by the experts and innovators which are both producers and consumers ("prosumers") in cooperation with the governance as enabler and supporter of the SSK approach.

V. Experience Report

A. Evaluation

The Volkswagen Group IT has instantiated the SSK approach and has been actively using it for more than three years. The governance is established within the ACE [44] and supported by the Quality innovation NETwork (QiNET) [45]. An established internal wiki-like tooling is used as delivery platform for the digital SSK's. To ensure maintenance, SSK teams perform regular updates, a process that is verified by the governance through quality checks. The governance also checks for blind spots in the SSK portfolio and initiate the setup of SSK teams via Community of Practices (CoP) to close the blind spots. An additional point of the governance is to facilitate the integration of the SSKs into established procedures like the integration into trainings of the Group Academy.

The SSK teams are founded in a prosumer fashion. Each team member wanting to share some know-how in the organization can be part of an SSK team which produces the SSK content. Experts for a particular topic typically volunteer to create SSK initiatives and teams. Experts are organized in hierarchy lines like competence centers (example ACE), communities or networks (example QiNET). Both are sources for experts who are willing and able to develop an SSK. The SSK team typically is also the team that handles the updates over the life-cycle of the SSK. The SSK team is supported by the SSK for SSK development. This ensures that SSKs looking "similar" and reduces the work of the SSK team by using the templates and how-to's which are included in the SSK for SSK development. In the case that all relevant information and content for the SSK under development exists (typically a SSK is based on artifacts, which are used by teams for their work and now are "packaged" by the SSK for multiplication into the organization) an new SSK can be built by the SSK team in a few hours. Than the initial



Topic: Self-Service Kit (SSK) development
Enable teams to write a SSK

Scope

Systematic development of a SSK for a new topic (in a teams).

Context

- Application in all domains which want to share expert knowlage
- Addresses different levels of demand:
  - Introduction into SSK
  - Describes the life-cycle of a SSK
  - **Offer template and checklist**
  - offer **background information for product specific enhancement**

Outcomes

- **A new SSK**
  - Introduction
  - More or less artifacts
  - Background information
- Knowledge about SSK development/design

Reference to working artifacts

- No further templates – use this as template

Further information / background knowledge

- **Links to internal resources: article HSM, paper FedCSIS**
- **Links in the internet**: -

12.08.2020 K-FIBA | CSD-Class: xx.x – xx years 2

Fig. 2: Overview page of the SSK for SSK development.

application of the new SSK should be done under observation of an SSK team member to see that everything works as intended. Focus of the observation is that the usage is as intended and the time to understand and learn about the application is short. For a fast learning the SSK how-to template is the key to focus on the application and is supported by the offered templates. Most SSKs are ready for a first application by a "new product team" in less than one hour. If everything look good the SSK is ready for publishing. More details about the content of a SSK is shown in [42] and the associated conference presentation which is based on the SSK artifacts an impression gives Figure 2. More about the detailed structure of SSKs is described in [46] which leads to the SSK for SSK development.

All employees of the Volkswagen AG can consume any time any SSK offered by the platform by simple download and use, or by adaptation to the specific context of the product or service offered by the team. Moreover, each consumer can improve any SSK with feedbacks anytime.

Three years ago, the Group IT started with the development of the first SSK. Over time, the iterations of improvement and enhancement of established SSK's – SSK versions up 6 are released - accompanied by the development of new SSK's has led to a holistic SSK approach implementation – the SSK for SSK development. This "meta" SSK is offered to scale the SSK approach itself by its own approach (recursively). This shows that the SSK approach is continuously improved and enhanced. Currently, there is a two-digit amount of SSKs in the portfolio. The trend to more digitalization and blended learning will further propel the SSK approach and produce a bigger portfolio. An important point at the beginning was that the SSK development could be initiated bottom up without big resource allocation and funding. The SSK approach is an agile approach by design: an autonomous team of experts can be the initial spark to enflame an organization by its first SSK.

B. Limitations

The application was conducted in an enterprise with mostly European culture. Other cultures may behave differently. The feedback mechanism for improvement is weakly implemented through voluntary feedbacks. However, the "sound of silence" [47] in this case indicates that there are no significant issues with the implemented approach. Furthermore, the views/downloads figures are weak metrics for the learning impact and application intensity, since not every download leads to a valuable outcome. Moreover, the approach has been developed continuously and improved with the design science approach. However it is difficult to demonstrate explicit effectiveness of SSKs in the agile scaling of the organization because there are many other parameters impacting the scaling. This highly applied and productive context provided a constrained space to change design parameters and observe their impacts. On the other hand, this setting has been facilitating the SSK approach's development and adoption synchronized with the organization's digitalization and agile transition.

VI. Conclusion

The presented SSK approach combines different learning and training approaches to a specialized learning approach for agile organizations by focusing on agile values and mindset by design. The SSK approach offers an agile way to scale agile transitions in an organization. It offers a systematic learning by doing and gives the background information for adoption to specific demands of the application domain of its users. This leads to knowledge and experience creation in the teams. Furthermore, the approach values mature agile teams as prosumers who are able to improve not only their teams with established methods like the retrospective. In addition, they can improve the organization with their experience, knowledge sharing and elaboration artifacts for SSK's. This is an essential element for an agile organization that needs to step from self-organization of teams to self-organization of organizations in the long-term. The SSK approach which supports all the three quality dimensions from product, process to the team provides a key lever to achieving this goal.

VII. Next steps and future work

Future work will address current blind spots and limitations of the current SSK approach to evolve them further. In a next step, the limitation of the voluntary feedbacks for improvement will be investigated [48]. Also, we want to determine how useful metrics like downloads or views of SSK are to derive the impact of a specific SSK in the organization. Furthermore, metrics for the establishment of the self-organizing organization has to be developed to make the current state of the agile transition transparent and to measure the impact of specific contributions to the transition goal.

References

[1] K. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, and J. Kern. "Manifesto for Agile Software Development": https://agilemanifesto.org/; 2001.

[2] J. Miler, and P. Gaida. "On the agile mindset of an effective team–an industrial opinion survey". In 2019 Federated Conference on Computer Science and Information Systems (FedCSIS) (pp. 841-849). IEEE.

[3] A. Hevner, S. March, J. Park, and S. Ram. "Design science in information systems research", MIS Quarterly, Vol. 28, no. 1, pp. 75–105, 2004..

[4] J. Webster, and R. T. Watson. "Analyzing the past to prepare for the future: Writing a literature review," MIS Quarterly, 2002, 26(2):13-23

[5] M. Driscoll, "Blended learning: Let's get beyond the hype." E-learning 1.4 (2002): 1-4.

[6] A. Dukhanov, M. Karpova, and K. Bochenina. "Design virtual learning labs for courses in computational science with use of cloud computing technologies." Procedia Computer Science 29 (2014): 2472-2482.

[7] C. Raspotnig, and A. Opdahl. "Comparing risk identification techniques for safety and security requirements." Journal of Systems and Software, 86(4), 1124-1151. 2013.

[8] IEC 61508, 2008. Functional safety of electrical/electronic/programmable electronic safety-related systems. International Electrotechnical Commission, 2nd ed.

[9] B. S. Bloom, D. R. Krathwohl, and B. B. Masia. "Bloom taxonomy of educational objectives." In Allyn and Bacon. Pearson Education. 1984.

[10] S. W. Williams, "Instructional Design Factors and the Effectiveness of Web-Based Training/Instruction." 2002.

[11] N. Hoic-Bozic, V. Mornar, and I. Boticki, “Blended Learning Approach to Course Design and Implementation” IEEE Transactions on Education, vol. 52, No. 1, February

[12] W. Hung, D. H. Jonassen, and R. Liu. “Problem-based learning.” Handbook of research on educational communications and technology, 3(1), 485-506. 2008.

[13] C. E. Hmelo-Silver, and H. S. Barrows, “Goals and strategies of a problem-based learning facilitator”. Interdisciplinary journal of problem-based learning, 1(1), 4. 2006

[14] L. Brodie, “Problem based learning in the online environment-successfully using student diversity and e-education.” In Proceedings of the 2006 Annual Conference on Internet Research 7.0:(IR 7.0): Internet Convergences. Association of Internet Researchers.

[15] Beck, K., Crocker, R., Meszaros, G., Coplien, J. O., Dominick, L., Paulisch, F., & Vlissides, J. (1996, March). Industrial experience with design patterns. In Proceedings of IEEE 18th International Conference on Software Engineering (pp. 103-114). IEEE.

[16] K. J. Arrow, “The economic implications of learning by doing.” In Readings in the Theory of Growth (pp. 131-149). Palgrave Macmillan, London. 1971.

[17] R. C. Schank, T. R. Berman, and K. A. Macpherson. “Learning by doing. Instructional-design theories and models: A new paradigm of instructional theory”, 2(2), 161-181. 1999.

[18] H. Latchman, C. Salzmann, D. Gillet and H. Bouzekri, "Information technology enhanced learning in distance and conventional education", IEEE Trans. Educ., vol. 42, no. 4, pp. 247-254, Nov. 1999.

[19] D.J. Reifer, F. Maurer, and H. Erdogmus“Scaling agile methods.” IEEE software, 20(4), pp.12-14. 2003

[20] M. Alqudah, and R. Razali. "A review of scaling agile methods in large software development." International Journal on Advanced Science, Engineering and Information Technology 6, no. 6 (2016): 828-837.

[21] M. Kalenda, P. Hyna, and B. Rossi, “Scaling agile in large organizations: Practices, challenges, and success factors.” Journal of Software: Evolution and Process, 30(10), p.e1954. 2018.

[22] S.w. Ambler, “The agile scaling model (ASM): adapting agile methods for complex environments. Environments,” pp.1-35. 2009.

[23] T. Dingsøyr, and N.B. Moe, “Research challenges in large-scale agile software development.” ACM SIGSOFT Software Engineering Notes, 38(5), pp.38-39. 2013.

[24] https://www.scaledagileframework.com/safe-program-consultant/ (last checked on 14. August 2020)

[25] https://www.scaledagile.com/certifications/which-course-is-right-for-me/ (last checked on 14. August 2020)

[26] SAFe – principals: https://www.scaledagileframework.com/safe-lean-agile-principles/ (last checked on 3. July 2020)

[27] D. L. Johnston. “Scientists Become Managers-The "T"-Shaped Man.” IEEE Engineering Management Review, 6(3), 67–68. 1978. doi:10.1109/emr.1978.4306682

[28] N. B. Moe, B. Dahl, V. Stray, L. S. Karlsen, and S. Schjødt-Osmo. “Team autonomy in large-scale agile.” In Proceedings of the 52nd Hawaii International Conference on System Sciences. 2019

[29] I. F. Oskam. “T-shaped engineers for interdisciplinary innovation: an attractive perspective for young people as well as a must for innovative organisations.” In 37th Annual Conference–Attracting students in Engineering, Rotterdam, The Netherlands (Vol. 14). July 2009.

[30] R. Hoda, and L. K, Murugesan. “Multi-level agile project management challenges: A self-organizing team perspective.” Journal of Systems and Software, 117, 245-257. 2016

[31] I. Grützner, S. Weibelzahl, and P. Waterson. "Improving courseware quality through life-cycle encompassing quality assurance." Proceedings of the 2004 ACM symposium on Applied computing. 2004.

[32] D. Kirkpatrick. “Quality assurance in open and distance learning.” 2005.

[33] R. Oliver. “Quality assurance and e-learning: blue skies and pragmatism.” ALT-Journal, 13(3), 173-187. 2005.

[34] U. D. Ehlers. “Quality assurance policies and guidelines in European distance and e learning.” Quality assurance and accreditation in distance and e-learning, 79-90. 2012.

[35] T. Olson, and R. A. Wisher. "The effectiveness of web-based instruction: An initial inquiry." The International Review of Research in Open and Distributed Learning 3.2. 2002.

[36] C. Kelleher, and R. Pausch. "Stencils-based tutorials: design and evaluation." Proceedings of the SIGCHI conference on Human factors in computing systems. 2005.

[37] L. Rajabion, N. Nazari, M. Bandarchi, A. Farashiani, and S. Haddad. “Knowledge Sharing Mechanisms in Virtual Communities: A Review of the Current Literature and Recommendations for Future Research”. Journal Human Systems Management, pp. 365 – 384. January 2019.

[38] A. Poth, M. Kottke, and A. Riel. "Scaling Agile–A Large Enterprise View on Delivering and Ensuring Sustainable Transitions." Advances in Agile and User-Centred Software Engineering. Springer, Cham, pp. 1-18. 2019

[39] A. Poth, B. Mayer, P. Schlicht, and A. Riel. “Quality Assurance for Machine Learning – an approach to function and system safeguarding“, Int. Conference on IEEE Software Quality, Reliability and Security, in print, 2020.

[40] A. Poth, N. Schubert, and A. Riel. “Sustainability Efficiency Challenges of Modern IT Architectures – A Quality Model for Serverless Energy Footprint”. In: Yilmaz M., Niemann J., Clarke P., Messnarz R. (eds) Systems, Software and Services Process Improvement. EuroSPI 2020. Communications in Computer and Information Science, vol 1251. Springer, Cham.; 2020. https://doi.org/10.1007/978-3-030-56441-4_21

[41] A. Poth, J. Jacobsen, and A. Riel. “A systematic approach to agile development in highly regulated environments”, In: Proceedings of the 21st International Conference on Agile Software Development, Copenhagen, Denmark. Lecture Notes in Business Information Processing; M. Paasivaara and P. Kruchten (Eds.): XP 2020, LNBIP 396. https://doi.org/10.1007/978-3-030-58858-8_12

[42] A. Poth, and A. Riel. “Quality requirements elicitation by ideation of product quality risks with design thinking.” In: Proceedings of the 28th IEEE International Requirements Engineering Conference (RE’20), Zürich, Switzerland, pp. 238- 249, 2020. IEEE. DOI 10.1109/RE48521.2020.0003

[43] A. Poth, M. Kottke and A. Riel. " Evaluation of Agile Team Work Quality." In: Proceedings of the 21st International Conference on Agile Software Development (XP 2020), Copenhagen, Denmark. Lecture Notes in Business Information Processing; Lecture Notes in Business Information Processing; M. Paasivaara and P. Kruchten (Eds.): XP 2020, LNBIP 396. https://doi.org/10.1007/978-3-030-58858-8_11

[44] A. Poth. "Effectivity and economical aspects for agile quality assurance in large enterprises." Journal of Software: Evolution and Process, 28.11 pp. 1000-1004. 2016.

[45] A. Poth, and C. Heimann. "How to Innovate Software Quality Assurance and Testing in Large Enterprises?." European Conference on Software Process Improvement. Springer, Cham, 2018.

[46] A. Poth, M. Kottke, and A. Riel, “Digital Self-Service Kits for Scaling Knowledge, and Fostering Team Autonomy and Distant Collaboration in a Large-Scale Corporate Context” in Human System Management (HSM) Journal, 2020, in print.

[47] C. Dellarocas, and C. A. Wood. "The sound of silence in online feedback: Estimating trading risks in the presence of reporting bias." Management science 54.3. pp. 460-476. 2008.

[48] E. W. Morrison. “Employee voice and silence.” Annu. Rev. Organ. Psychol. Organ. Behav., 1(1), 173-197. 2014.

6th Workshop on Model Driven Approaches in System Development

FOR many years, various approaches in system design and implementation differentiate between the specification of the system and its implementation on a particular platform. People in software industry have been using models for a precise description of systems at the appropriate abstraction level without unnecessary details. Model-Driven (MD) approaches to the system development increase the importance and power of models by shifting the focus from programming to modeling activities. Models may be used as primary artifacts in constructing software, which means that software components are generated from models. Software development tools need to automate as many as possible tasks of model construction and transformation requiring the smallest amount of human interaction.

A goal of the proposed workshop is to bring together people working on MD languages, techniques and tools, as well as Domain Specific Languages (DSL) and applying them in the requirements engineering, information system and application development, databases, and related areas, so that they can exchange their experience, create new ideas, evaluate and improve MD approaches and spread its use. The intention is to target an interdisciplinary nature of MD approaches in software engineering, as well as research topics expressed by but not limited to acronyms such as Model Driven Software Engineering (MDSE), Model Driven Software Development (MDSD), Domain Specific Modeling (DSM), and OMG's Model Driven Architecture (MDA).

1st Workshop on MDASD was organized in the scope of ADBIS 2010 Conference, held in Novi Sad, Serbia. From 2012, MDASD becomes a regular bi-annual FedCSIS event.

TOPICS

- MD Approaches in System Design and Implementation – Problems and Issues
- MD Approaches in Software Process Models
- MD Approaches in Databases and Information Systems
- MD Approaches in Software Quality and Standards
- Metamodeling, Modeling and Specification Languages
- Model Transformation Languages
- Model-to-Model, Model-to-Text, and Model-to-Code Transformations in Software Process
- Transformation Techniques and Tools
- Domain Specific Languages (DSL) and Domain Specific Modeling (DSM) in System Specification and Development
- Design of Metamodeling and Modeling Languages and Tools
- MD Approaches in Requirements Engineering and Business Process Modeling
- MD Approaches in System Reengineering and Reverse Engineering
- MD Approaches in HCI development
- MD Approaches in GIS development
- MD Approaches in Document Engineering
- Model Based Software Verification
- Theoretical and Mathematical Foundations of MD Approaches
- Organizational and Human Factors, Skills, and Qualifications for MD Approaches
- Teaching MD Approaches in Academic and Industrial Environments
- MD Applications and Industry Experience

TECHNICAL SESSION CHAIRS

- **Luković, Ivan,** University of Novi Sad, Serbia

STEERING COMMITTEE

- **Gray, Jeff,** University of Alabama, United States
- **Mernik, Marjan,** University of Maribor, Slovenia
- **Ristić, Sonja,** University of Novi Sad, Faculty of Technical Sciences, Serbia
- **Tolvanen, Juha-Pekka,** MetaCase, Finland

PROGRAM COMMITTEE

- **Amaral, Vasco,** The New University of Lisbon, Portugal
- **Bryant, Barrett,** University of North Texas, United States
- **Budimac, Zoran,** Faculty of Sciences, Univ. of Novi Sad, Serbia
- **Chen, Haiming,** Chinese Academy of Sciences, China
- **Erradi, Mohammed,** ENSIAS, Mohammed-V University, Morocco
- **Fertalj, Krešimir,** University of Zagreb, Croatia
- **Härting, Ralf-Christian,** Hochschule Aalen, Germany
- **Ivanović, Mirjana,** University of Novi Sad, Serbia
- **Janousek, Jan,** Czech Technical University, Czech Republic
- **Karagiannis, Dimitris,** University of Vienna, Austria
- **Kardaş, Geylani,** Ege University International Computer Institute, Turkey
- **Kern, Heiko,** University of Leipzig, Germany
- **Kollár, Ján,** Technical University of Kosice, Slovakia
- **Kordić, Slavica,** University of Novi Sad, Faculty of Technical Sciences, Serbia

- **Kosar, Tomaž,** University of Maribor, Slovenia
- **Krdzavac, Nenad,** University of Cambridge, Cambridge Centre for Advanced Research and Education, Singapore
- **Liu, Shih-Hsi Alex,** California State University, United States
- **Maćoš, Dragan,** Beuth University of Applied Sciences, Germany
- **Mazzara, Manuel,** Innopolis University, Russia
- **Melo de Sousa, Simão,** University of Beira Interior, Portugal
- **Milosavljević, Gordana,** University of Novi Sad, Faculty of Tecnical Sciences, Serbia
- **Porubän, Jaroslav,** Technical University of Kosice, Slovakia
- **Rangel Henriques, Pedro,** Universidade do Minho, Portugal
- **Selic, Bran,** Malina Software Co., Canada
- **Sierra Rodríguez, José Luis,** Universidad Complutense de Madrid, Spain
- **Slivnik, Boštjan,** University of Ljubljana, Slovenia
- **Vangheluwe, Hans,** University of Antwerp, Belgium
- **Varanda Pereira, Maria João,** Instituto Politecnico de Braganca, Portugal
- **Wimmer, Manuel,** Johannes Kepler University Linz, Austria

Proceedings of the Federated Conference on Computer Science and Information Systems pp. 741–750 DOI: 10.15439/2020F133 ISSN 2300-5963 ACSIS, Vol. 21

RE4TinyOS: A Reverse Engineering Methodology for the MDE of TinyOS Applications

Hussein M. Marah
International Computer Institute
Ege University, Izmir, Turkey
hussein.marah@gmail.com

Moharram Challenger
Department of Computer Science
Univeristy of Antwerp and Flanders Make, Belgium
moharram.challenger@uantwerpen.be

Geylani Kardas
International Computer Institute
Ege University, Izmir, Turkey
geylani.kardas@ege.edu.tr

***Abstract*—In this paper, we introduce a tool-supported reverse engineering methodology, called RE4TinyOS to create or update application models from TinyOS programs for the construction of Wireless Sensor Networks. Integrating with an existing model-driven engineering (MDE) environment, use of RE4TinyOS enables the model-code synchronization where any modification made in the TinyOS application code can be reflected into the application model and vice versa. Conducted case studies exemplified this model-code synchronization as well as the capability of creating application models completely from already existing TinyOS applications without models, which is crucial to integrate the implementations of the third party TinyOS applications into the MDE processes. Evaluation results showed that RE4TinyOS succeeded in the reverse engineering of all main parts of two well-known TinyOS applications taken from the official TinyOS Github repository and generated models were able to be visually processed in the MDE environment for further modifications.**

***Keywords*—Model-Driven Engineering, Reverse Engineering, Wireless Sensor Network, TinyOS, RE4TinyOS.**

I. Introduction

WIRELESS Sensor Networks (WSN) have gained significant popularity and implemented in different areas (e.g. health systems, field monitoring, transportation, military applications and environmental sensing) to control both the status of physical objects and the surrounding circumstances like sound, pressure, vibration, light, temperature, and motion according to the type of the sensors used in the network [1]. WSNs use low-power micro-controllers and devices due to the power consumption constraints that must be adhered to.

One of the widely used operating systems for WSNs is TinyOS [2]. TinyOS is an open-source operating system for WSNs, developed in the University of California, Berkeley. It is a lightweight and flexible operating system that offers a set of services such as communication, timers, sensing, storage and these services can be reusable to compose larger applications. These features make TinyOS a reliable and efficient system for programming, configuring and running lower-power wireless devices [2][3]. However, especially the requirement of managing the power constraints makes TinyOS different from ordinary systems and hence building WSNs with TinyOS can be a challenging and time-consuming task. Moreover, the developers need to have deep knowledge and skills in the special programming language of TinyOS, called nesC to implement such systems [3]. Adoption to this language may be difficult and again time-consuming for the programmers.

As successfully applied in many other domains, model-driven engineering (MDE) can provide a convenient way of developing TinyOS applications for WSNs by leveraging the abstraction level before delving into programming with nesC. Within this context, in our previous work [4], we introduced the use of a domain-specific modeling language (DSML), called DSML4TinyOS, for the MDE of TinyOS applications. A metamodel for TinyOS was derived and a graphical modeling syntax was formalized from this metamodel to lead modeling TinyOS applications. nesC code of the modeled applications can be automatically generated with the model-to-code transformations again defined in DSML4TinyOS. However, this mechanism lacks the synchronization between a TinyOS application model and the generated code when any change is made in this code. Mostly, the auto-generated code is modified to completely meet with the requirements of the TinyOS application. Furthermore, the application may evolve according to changing requirements in the future. After the code modifications are performed, related changes will make models at different levels asynchronous and inconsistent [5]. Thus we need to propagate these changes to the other models and ensure a proper model synchronization [6]. In order to provide this synchronization which is missing in the MDE of TinyOS applications, in this paper, we introduce a tool-supported reverse engineering methodology, called RE4TinyOS. RE4TinyOS enables retrieving TinyOS application models from any existing nesC code. In addition to support the reverse engineering of such applications, use of RE4TinyOS also integrates with the current MDE process brought by DSML4TinyOS language to construct a complete model-driven roundtrip engineering [7] process for TinyOS applications. As depicted in Figure 1, evolution of the TinyOS models can be managed within this roundtrip MDE process which is a combination of the forward and reverse engineering of TinyOS models. TinyOS models can be created with using DSML4TinyOS language and the corresponding TinyOS code can be automatically generated. When this code is modified and becomes TinyOS code', RE4TinyOS reverse engineering methodology can be applied on this modified code to retrieve the corresponding modified model (still an instance of TinyOS metamodel) which properly reflects the changes in the application code.

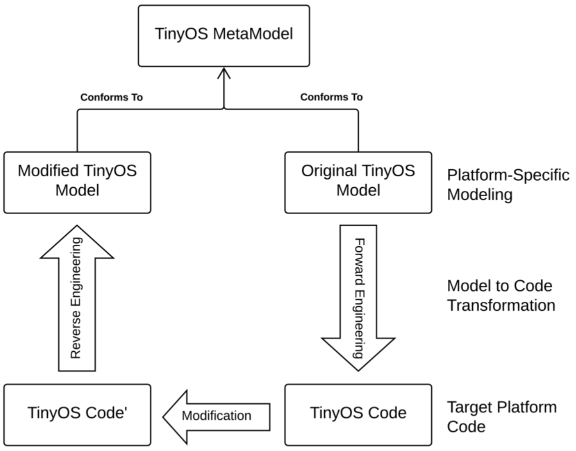


Fig. 1: Forward and reverse engineering for TinyOS applications

The remainder of the paper is organized as follows: Section 2 discusses the related work in this area. RE4TinyOS methodology and supporting parser and interpreter tools are introduced in Section 3. The usability of the methodology is demonstrated and evaluated in Section 4. Section 5 concludes the paper.

II. Related Work

In recent years, there is a significant interest of the researchers to apply MDE and its techniques for WSN and IoT development. The main goal of applying MDE approach is to facilitate the task of developing, building and deploying different WSN and IoT applications. Malavolta and Muccini [8] and Essaadi et al. [9] present good overviews of applied MDE approaches for this domain.

For example, ScatterClipse, a generative plugin-oriented tool-chain, is proposed in [10] to develop WSN applications running on the ScatterWeb sensor boards by using MDE. The tool aims to automate and standardize the generation of application system families for these sensor boards. Thang and Geihs [11] address the problem of optimizing power consumption and memory usage in the application design process and introduces an approach that integrates Evolutionary Algorithms with MDE where the system metamodels are generated to select the optimal model according to some performance criteria. Another modeling framework [12] allows developers to model separately the WSN software architecture and the features of the low-level hardware as well as the physical environment of the nodes of a WSN. The framework is capable of generating code from the created models which can be used for specific purposes such as analysis.

The study in [13] brings an MDE approach for prototyping and optimization of WSN applications while Veiset and Kristensen [14] introduce the use of Coloured Petri Net models for generating TinyOS protocol software. Likewise, the use of a domain-specific language (DSL), called SenNet, for WSN application development is proposed in [15] to prepare WSN applications using multi-abstraction levels. Finally, Rodrigues et al. [16] aim at facilitating the development tasks required for Wireless Sensor and Actuator Network (WSAN) applications via an MDA-based process. The proposed infrastructure is composed of a platform-independent model (PIM), a platform-specific model (PSM), and a transformation process which allows modeling and generation of these applications.

The above mentioned studies provide various noteworthy approaches both for modeling WSN applications in different abstraction levels and code generation for WSN development, mostly assisted with tools. Moreover, some of them specifically support the development of TinyOS applications within the MDE perspective. However, none of them considers the reflection of changes made after in the generated code to the corresponding application models, i.e. an approach for constructing the synchronization between WSN model and code does not exist. We believe that RE4TinyOS reverse engineering methodology, introduced in this paper, may contribute to these efforts by filling this gap as well as supporting the roundtrip engineering of TinyOS WSN applications within a toolchain consists of both generating code from TinyOS application models and retrieving models from the existing codes automatically.

Taking into consideration of applying reverse engineering in the context of MDE, various adoptions exist for different domains as surveyed in [17]. Perhaps one of the most popular approaches is MoDisco [18], which follows the MDE concepts and techniques to represent the legacy software systems in a different formalism by using reverse engineering. The infrastructure of MoDisco introduces generic components that can be used in the model-driven reverse engineering process (e.g., generic metamodels, model navigation, model transformation and model customization). Favre et al. [19] describe an operation for generating MDA models that combines the process of static and dynamic analysis. Model recovery is illustrated with the reverse engineering of Java code to get class and state diagrams. Fruitful applications of model-driven reverse engineering can also be seen in e.g. transforming legacy COBOL code into models [20], model discovery from Java source code to extract the business rules [21], generating GUI models of the explicit layouts especially for Java Swing user interfaces [22], restoring extended entity-relationship schema from NoSQL property graph databases [23] and even achieving reusable and evolvable model transformations [24]. However, reverse engineering of WSN applications is not addressed again in all these studies.

III. RE4TinyOS Methodology

Figure 2 represents the use of RE4TinyOS methodology for the MDE-based reverse engineering of WSN applications running on TinyOS. The figure gives a straightforward depiction of how reverse engineering works according to MDE concepts to convert the TinyOS code to a TinyOS model for any application.

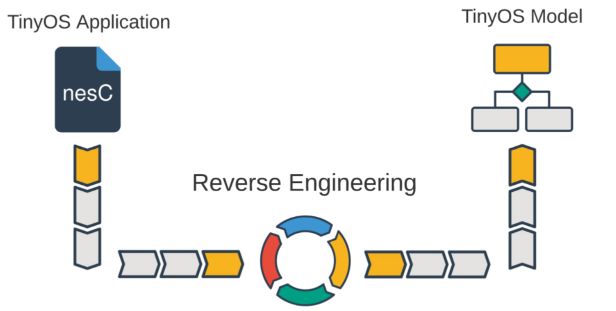


Fig. 2: Overview of the proposed reverse engineering approach

TinyOS applications are written in a special programming language, called nesC [25] for networked embedded systems. The nesC programming model combines the features of C programming language with the special needs in the WSN domain such as event-driven execution and component-oriented design [25]. In this study, we introduce the RE4TinyOS tool, which is designed to read any TinyOS application code written in nesC as the input and automatically generate the counterpart domain model representing this TinyOS application.

To recognize the syntax and all the valid components (symbols, characters and expressions) of a particular programming language, a language recognizer or language interpreter is needed to read the elements and differentiate them from other normal statements of this language. The language recognizer is used for different purposes like building a compiler or maybe analyze parts of code to perform some operations [26] [27]. Parsing is the process of syntax analysis and breaks down the syntax of the language into smaller structures of symbol strings conforming to the formal rules and the grammar that govern the language. Also, parsers or syntax analyzers provide the identification of the languages. Since our aim is to retrieve the model of the WSN application from its program code, parsing is an essential process to identify and analyze the input TinyOS code.

We followed a two-step method to create the environment required to the reverse engineering of TinyOS applications. The first step is to design the parser, called TinyOS parser, that can read any TinyOS code, and by parsing the input, we can obtain the useful or desired parts of the TinyOS code in order to use them to build the model. The second step is implementing this parser design as a Java application that can read any TinyOS application code and extract the main elements and components from the code and hence build the TinyOS model.

In this study, ANTLR was chosen to build the TinyOS parser. ANTLR (ANother Tool for Language Recognition) is a well-known computer-based language recognition tool, or more specifically a parser generator [28] [26] [27].

During a parser design, writing the grammar is a very crucial phase. It is the phase where the parser designers write the rules (Lexer and Parser rules) depending on analyzing the target system for their domains which in our case is the TinyOS system (i.e., the rules are written according to what type of input that will be parsed and what are the important information and parts are needed to be extracted). The next listing (Coding 1) includes a small fragment from the parser rules we created by using ANTLR. In this parser implementation, more than 300 lines of grammar were prepared besides the lexer rules.

Coding 1: Excerpts from TinyOS parser rules

```
compilationUnit
: (includeDeclarationModule* componentDeclaration)?
↪  (includeDeclarationConfiguration*
↪  componentDeclaration EOF) ;
includeDeclarationModule
: '#' INCLUDE qualifiedName ;
includeDeclarationConfiguration
: '#' INCLUDE qualifiedName ;
qualifiedName
: singleLine ;
componentDeclaration
: moduleDeclaration
| configurationDeclaration ;
//This part is for the module file
moduleDeclaration
: moduleSignature  moduleImplementation ;
moduleSignature
: MODULE moduleName '('? ')'?  moduleSignatureBody
↪  ;
moduleName
: singleLine ;
moduleSignatureBody
: '{' usesOrProvides* '}' ;
usesOrProvides
: usesState
| providesState ;
usesState
: USES INTERFACE usesInterfaceDescription* ';'
| USES '{' (INTERFACE usesInterfaceDescription
↪  ';')* '}' ;
providesState
: PROVIDES INTERFACE providesInterfaceDescription*
↪  ';'
| PROVIDES '{' (INTERFACE
↪  providesInterfaceDescription ';')* '}' ;
```

The above excerpts show the general structure of the written parser rules. For instance, the line that starts with “compilationUnit”, is considered as the start point of the whole parsing process. It states that two options exists; the first for the model and the second for the configuration that ends with “EOF” condition. The “componentDeclaration” line includes two main parts which are “moduleDeclaration” and “configurationDeclaration” respectively. The separator character ‘|’ declares that when the parsing process starts it has two options, module or configuration as they are the two main files of any TinyOS application. “moduleDeclaration” contains the details of the declaration. It has two parts which are “moduleSignature” and “moduleImplementation” respectively. It is worth indicating that these two parts are not separated by the ‘|’ character, which means that any module should have both signature and implementation.

Since our aim is to build models by parsing TinyOS programs, the metamodel for TinyOS, which we previously introduced in [4], was considered as the main reference model and the TinyOS Parser was written and designed with consistency to the TinyOS metamodel.

The next step after creating the TinyOS Parser is using this parser and benefiting from its features. ANTLR has the property to transform or, in more specific words, generate codes from ANTLR-based parsers to several commonly-used programming languages like Java, Python, JavaScript, Go, C++ and Swift [27]. In our case, the target language is Java. An overview of the constructed TinyOS parser is shown in Figure 3.

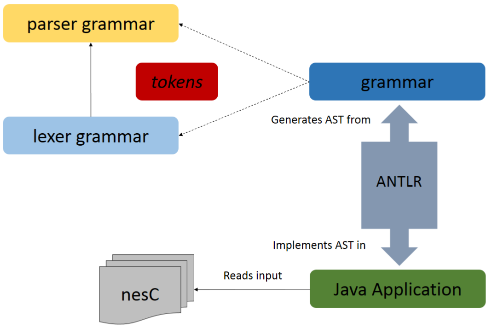


Fig. 3: Parsing process for TinyOS applications

As depicted in the previous figure, our TinyOS Parser is taking the produced tokens from the Lexer and constructs a data structure known as Abstract Syntax Tree (AST) for the parsed TinyOS code. The created AST here records how the input structure and the components have been recognized by the TinyOS Parser. By default, the runtime library in ANTLR provides a mechanism for walking through the constructed AST and this operation is called a tree-walking. In our approach, the primary provided parse-tree-walker mechanism called “Parse-Tree Listener” [27] was used to walk the built tree of the TinyOS applications. Finally, the “Parse-Tree Listener” is integrated and implemented in a Java application-specific code which reads TinyOS programs (nesC codes) as input and calls every node in the constructed tree of the parsed TinyOS code by providing a subclass for every TinyOS Parser grammar that enables the application to enter and exit from every triggered node in order to obtain and extract the required information to build theTinyOS model from the code.

Since the Eclipse Modeling Framework (EMF) uses the XML Metadata Interchange (XMI) standard to express models by mapping their corresponding information and write all this information into the XMI file extension, this standard was utilized to build the TinyOS models inside the developed Java application. The Java application could extract all the required and important information from the input files (nesC code) and convert this information into a TinyOS model, i.e. XMI file containing a representation of the TinyOS application according to the TinyOS metamodel.

Above described processes of using TinyOS parser and the Java application are combined together to create the TinyOS Interpreter executed by the RE4TinyOS tool (Figure 4).

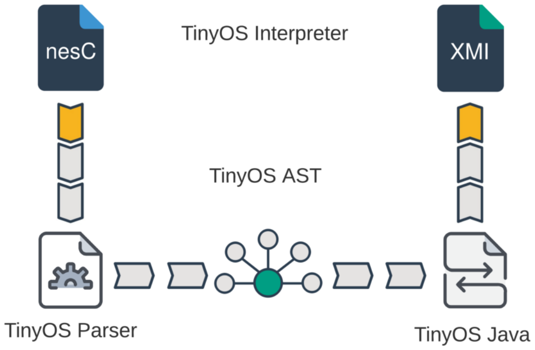


Fig. 4: TinyOS Interpreter structure

The generated XMI files containing the model representations of the input TinyOS applications can be opened inside the DSML4TinyOS modeling tool without any human intervention. Hence, these model instances conforming to the TinyOS metamodel, can be visually seen and ready for modifications if needed.

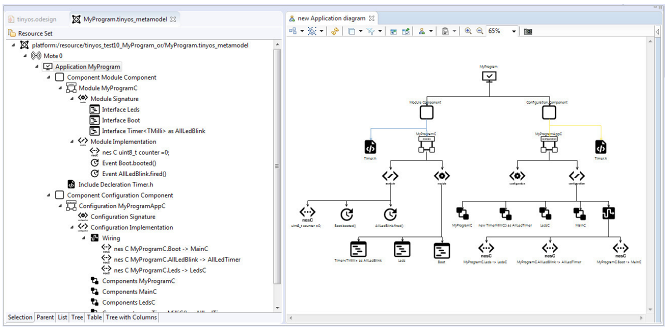
DSML4TinyOS is a tool-supported DSML which facilitates the development of TinyOS applications according to MDE principles and techniques. The tool enables TinyOS developers to develop applications from scratch by visually modelling these applications and generate code as the final artefact. DSML4TinyOS uses the TinyOS metamodel introduced in [4] as the abstract syntax. It has an EMF-based graphical syntax and the graphical modeling environment required for creating DSML4TinyOS models according to DSML4TinyOS syntax and semantics definitions. DSML4TinyOS modeling environment (see Figure 5) was built on the widely used Sirius platform. Table 1 lists the graphical notations used for the concrete syntax of the DSML4TinyOS language. TinyOS application models can be created by simply adding the language elements from the menu of the DSML4TinyOS tool. Implementation of the modeled applications can be automatically achieved via the code generation. DSML4TinyOS benefits from the features of Acceleo code generator to parse instance TinyOS models and create the templates of the implementation files.

As mentioned above, TinyOS application models, conforming to the TinyOS metamodel, are stored as XMI files and they can be modified inside the DSML4TinyOS tool by adding or removing components. These changes are automatically reflected into the corresponding application code again by the tool. Similarly, the TinyOS application models retrieved by the RE4TinyOS interpreter from the existing implementations can also be shown and processed again inside DSML4TinyOS tool. Hence, the synchronization of the system model and the

Concept	Notation	Concept	Notation
Mote	((•))	Application	🖥
Module	module	Configuration	configuration
Components	⧉	Interface	☰
Module_Signature	<⚙> module	Configuration_Signature	<⚙> configuration
Component	□	nesC	<···> nesC
Function	⚙	Wiring	⎍
Event	↻	Command	>_
Module_Implementation	</> module	Configuration_Implementation	</> configuration
Task	▰	IncludeDeclaration	</>
		Helper_Function	⚙

Table. 1: DSML4TinyOS concrete syntax notations

existing implementation is realized in case of any modification made on the model or the code.



```
platform:/resource/tinyos_test10_MyProgram_or/MyProgram.tinyos_metamodel
  Mote 0
    Application MyProgram
      Component Module Component
        Module MyProgramC
          Module Signature
            Interface Leds
            Interface Boot
            Interface Timer<TMilli> as AllLedBlink
          Module Implementation
            nes C uint8_t counter =0;
            Event Boot.booted()
            Event AllLedBlink.fired()
        Include Decleration Timer.h
      Component Configuration Component
        Configuration MyProgramAppC
          Configuration Signature
          Configuration Implementation
            Wiring
              nes C MyProgramC.Boot -> MainC
              nes C MyProgramC.AllLedBlink -> AllLedTimer
              nes C MyProgramC.Leds -> LedsC
            Components MyProgramC
            Components MainC
            Components LedsC
```

Fig. 5: DSML4TinyOS graphical modeling environment

To summarize, by applying the RE4TinyOS methodology, the software model of an existing TinyOS application can be achieved automatically. For this purpose, a developer only needs to give the code file of the related TinyOS application as the input for our RE4TinyOS tool. The built-in interpreter generates the corresponding model. This model is XMI serialized and can be opened and visually edited inside the DSML4TinyOS tool. If needed, any change made in the model is reflected into the code without any developer intervention.

IV. Case Studies

In order to demonstrate and evaluate the usability of RE4TinyOS methodology and its tool, a multi-case evaluation study has been performed. The first case study exemplifies how the synchronization between TinyOS models and the corresponding code can be provided with the use of both DSML4TinyOS and RE4TinyOS tools together within a model-driven roundtrip engineering process. The remaining two case studies consider the usability of RE4TinyOS methodology within the scope of the reverse engineering of already existing TinyOS applications publicly available from the official TinyOS repository in Github.

A. Supporting model - code synchronization

This section discusses the MDE of an application for a TinyOS mote, which displays the light emitting diodes (LEDs) on this mote when needed. The application, simply called MyProgram for the demonstration purposes, uses the "Boot" interface, executes the event "Boot.booted()" and calls the three LEDs via commands. In the "Boot.booted()" event, the command "AllLedBlink.startPeriodic(1000)" will be called. This command initializes a timer that gives interrupts for every 1000 milliseconds. Also, the application displays a counter on the three LEDs of the mote. It uses the timer interface "Timer<TMilli>as AllLedBlink" and executes the second event by firing the timer in the event "AllLedBlink.fired()". Inside this event, the three commands are called. The event will call the command "Leds.led0On()", "Leds.led1On()", and "Leds.led0On()" one by one corresponding to each "Counter" value.

The Above described TinyOS application was modeled graphically with using DSML4TinyOS and nesC code of this application was automatically generated.

Coding 2: nesC Module code auto-generated from the original application model

```
#include "Timer.h"
module MyProgramC @safe(){
    uses interface Leds;
    uses interface Boot;
    uses interface Timer<TMilli> as AllLedBlink;
}
implementation {
    uint8_t counter =0;
    event void Boot.booted() {
    /* Turn the three leds on */
    call Leds.led0On();
    call Leds.led1On();
    call Leds.led2On();
    /* call the timer every 1000 milliseconds */
    call AllLedBlink.startPeriodic( 1000 );
        }
    event void AllLedBlink.fired() {
    counter++;
    if (counter & 0x1) {
        call Leds.led0On(); }
    else { call Leds.led0Off();}
    if (counter & 0x2) {
        call Leds.led1On();}
    else { call Leds.led1Off();}
    if (counter & 0x4) {
        call Leds.led2On(); }
    else { call Leds.led2Off();}
   }
}
```

Coding 3: nesC Configuration code auto-generated from the original application model

```
#include "Timer.h"
configuration MyProgramAppC {
}
implementation {
    components MyProgramC;
    components MainC;
    components LedsC;
    components new TimerMilliC() as AllLedTimer;
    MyProgramC.Boot -> MainC;
    MyProgramC.AllLedBlink -> AllLedTimer;
    MyProgramC.Leds -> LedsC;
}
```

The previous two listings include the code fragment generated from this model for the module part (Coding 2) and the configuration part (Coding 3) of the TinyOS application. Also, the Figure 6 shows the model of the MyProgram application (as a DSML4TinyOS instance), the instance model represents the two parts of code ’Module’ and ’Configuration’ for the application in a single model.

When any change made in the application code, these can be reflected to the corresponding model with using the RE4TinyOS tool. Now, let us suppose that a developer wants to modify the above program with adding three new timers and a task. In the modified application, every interface will blink just one specific led: “Timer<TMilli>as RedLedBlink” will blink the red led, “Timer<TMilli>as GreenLedBlink” will blink the green led and “Timer<TMilli>as YellowLedBlink” will blink the yellow led respectively. Hence, every event will be triggered independently: “RedLedBlink.fired()” will trigger the red led timer, “GreenLedBlink.fired()” will trigger the green led timer and “YellowLedBlink.fired()” will trigger the yellow led timer. Inside “Boot.booted()” event, a ”for loop” with including an ”if statement” is added to the code to test the counter, call one of the timers that will be fired and call the command to turn on the LED. Also, a new task is added and it will be called in “Boot.booted()” event. Following code listings (Coding 4 and Coding 5) include the modified versions of the module and configuration components of our TinyOS program in which the added / changed parts are highlighted in cyan color.

Coding 4: Modified nesC Module code of the application

```
#include "Timer.h"
#include "printf.h"
module MyProgramC @safe() {
  uses interface Leds;
  uses interface Boot;
  uses interface Timer <TMilli> as AllLedBlink;
  uses interface Timer <TMilli> as RedLedBlink;
  uses interface Timer <TMilli> as GreenLedBlink;
  uses interface Timer <TMilli> as YellowLedBlink;
}
implementation {
  uint8_t counter;
 task void printTask() {
  printf("Print task\n");}
  event void Boot.booted() {
   for (counter = 0; counter <= 31; counter++) {
   if (counter == 10) {
    call RedLedBlink.startOneShot(counter);}
   else if (counter == 20) {
    call GreenLedBlink.startOneShot(counter);}
   else if (counter == 30) {
    call YellowLedBlink.startOneShot(counter);}
   else { printf("It will not blink any led\n");}
   }
    call AllLedBlink.startPeriodic(50);
    dbg("MyProgramC", "Application booted.\n");
    post printTask();
  }
  event void AllLedBlink.fired() {
    call Leds.led0On();
    call Leds.led1On();
    call Leds.led2On(); }
  event void RedLedBlink.fired() {
    printf("Blink the red led\n");
    call Leds.led0Toggle();}
  event void GreenLedBlink.fired() {
    printf("Blink the green led\n");
    call Leds.led1Toggle();}
  event void YellowLedBlink.fired() {
    printf("Blink the yellow led\n");
    call Leds.led2Toggle();}
}
```

Coding 5: Modified nesC Configuration code of the application

```
#include "Timer.h"
#include "printf.h"
configuration MyProgramAppC {}
implementation {
  components MyProgramC, MainC, LedsC;
  components new TimerMilliC() as AllLedTimer;
  components new TimerMilliC() as RedLedTimer;
  components new TimerMilliC() as GreenLedTimer;
  components new TimerMilliC() as YellowLedTimer;
  MyProgramC.Boot - > MainC;
  MyProgramC.AllLedBlink - > AllLedTimer;
  MyProgramC.RedLedBlink - > RedLedTimer;
  MyProgramC.GreenLedBlink - > GreenLedTimer;
  MyProgramC.YellowLedBlink - > YellowLedTimer;
  MyProgramC.Leds - > LedsC;
}
```

To propagate above code modifications to the model of the application, RE4TinyOS tool was used. New version of the program was given as input to the RE4TinyOS and the tool successfully produced the serialized file for the model. This model was opened in the DSML4TinyOS modeling environment (see Figure 7) and it was examined that RE4TinyOS

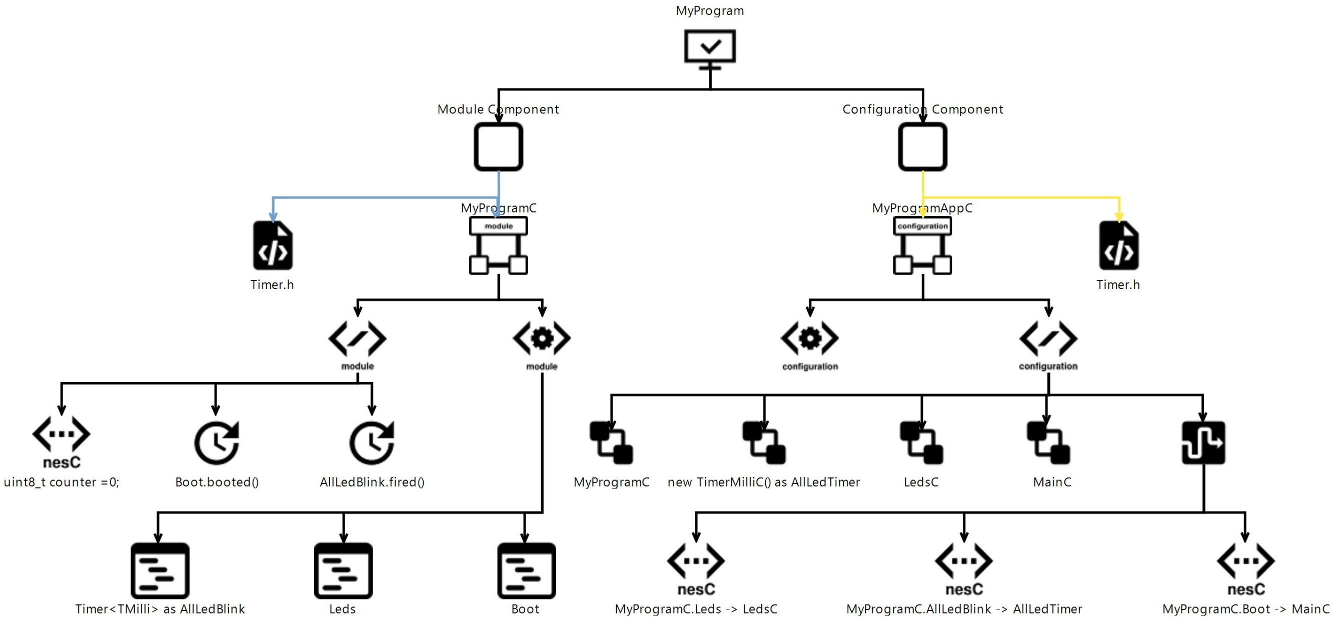


Fig. 6: Graphical model of the original TinyOS application

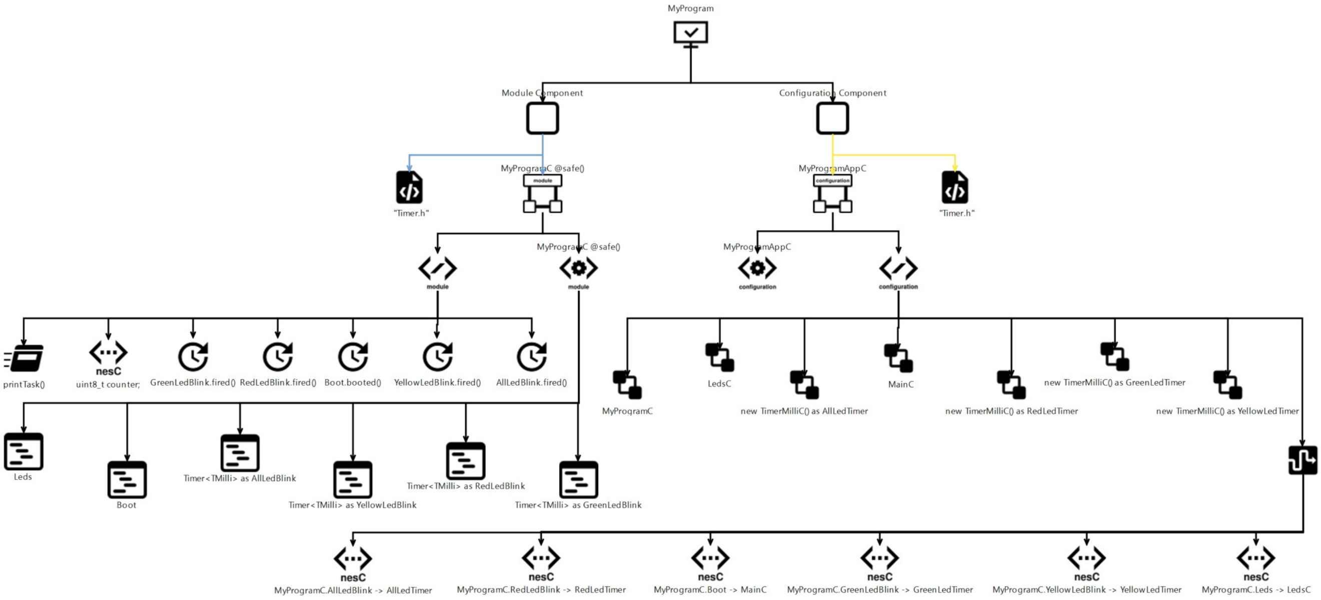


Fig. 7: Graphical model of the modified TinyOS application

maintained the synchronization between the model and the code by automatically inserting new model elements and changing existing elements (e.g. “Boot.booted()” event was changed due to its new function implementation). As can also be seen from figure 7, the modifications were seamlessly integrated into the modified and new model with preserving the unchanged model components.

B. *Integrating already existing implementations into modeling*

Although the previous case study shows how RE4TinyOs tool enables retrieving TinyOS application models from the code and updating the model when the code is modified, we also need to evaluate the capability of creating application models completely from already existing code which is crucial to integrate the implementations of the third party applications into the MDE processes. In here, already existing code means the application was not previously designed and implemented with using DSML4TinyOS and RE4TinyOS tool chain. Hence, it does not own an application model to be used as an input for further system developments. For the purpose of evaluating this capability of RE4TinyOS, we considered the reverse engineering of two existing TinyOS applications which are well-known and publicly available from the official TinyOS repository in Github. In the following, first these two applications and the generated models are introduced briefly, then the qualitative assessment results are discussed.

1) AntiTheft WSN: AntiTheft is an application for detecting thefts, that uses various aspects of TinyOS and its services. AntiTheft application can detect a theft by monitoring two events:

1) The change in the light level: It assumes that a stolen mote will be situated in a dark place.
2) The change in the acceleration rate: When thieves steal anything, they usually move too fast and run.

So, the application will report the theft by:

- Alerting via turning on the light (e.g. a red LED)
- Also making a beep sound
- Reporting to the other nodes within the range by broadcasting messages, and nodes will also turn on their red LEDs.
- Reporting to a central node using a multi-hop routing algorithm.

The complete nesC code of the AntiTheft application, accessed from TinyOS Github repository [29], was given as input to RE4TinyOS tool and the serialized model file was generated. When this file was opened in DSML4TinyOS modeling environment, the graphical model of the application was shown successfully (see figure 8). Parts of the TinyOS application including components, interfaces, commands, and events are now represented in DSML4TinyOS notation as the result of the applied reverse engineering methodology.

2) Sense WSN: The Sense is another application also available in the main TinyOS Github repository. As its name denotes, it is a simple sensing application that periodically samples data from the sensors by initializing a timer which will signal a "read event" and displays the bits of the sampled readings on the LEDs of the nodes. Similar to AntiTheft application, the complete code of the Sense application achieved from the Github repository [30] was processed by RE4TinyOS tool and the model of the application was generated without any error. Figure 9 shows this model opened in the DSML4TinyOS modeling environment.

C. Discussion

First case study, conducted for the MDE of a TinyOS LED display application, demonstrated the use of RE4TinyOS methodology and its tool to support the model-code synchronization where the application model is kept up-to-date in each modification made in the application code. The case study also exemplified the use of DSML4TinyOS and RE4TinyOS tool chain leading the roundtrip engineering of the TinyOs applications.

The remaining case studies enabled the assessment of the proposed reverse engineering methodology brought by RE4TinyOS especially for the already existing TinyOS applications which were not previously designed and implemented with using DSML4TinyOS and/or RE4TinyOS tools. Moreover, the fact that the code of these applications are publicly available in TinyOS Github and written by other developers, contributed to the objectiveness of the performed evaluation.

When the complete code of both Anti-Theft and Sense applications, which are ready to be executed, was given as input to RE4TinyOS, the embedded parser of the RE4TinyOS was able to automatically generate serialized versions of the TinyOS software models of these applications, and the produced models were processed and successfully opened in the DSML4TinyOS IDE. This also confirms that, if needed, RE4TinyOS tool can also be used independently from the MDE tool chain, i.e. the TinyOS application that will be processed by the RE4TinyOS tool could be previously implemented via using any other method and environment. The developers can achieve software models of these existing applications. Furthermore, it is straightforward to visually work on these recovered models at a higher level of abstraction, make modifications on them and then reflect these changes to the exact implementations.

Finally, it is worth indicating that RE4TinyOS succeeded in retrieving the models for all main parts of AntiTheft and Sense applications, including "event", "task", "component", "interface", "Command", "Helper-function", and "Wiring" (see figures 8 and 9). Although, block structures of the application events were also retrieved, internal specifications of some of these events could not be fully represented in the output model since corresponding meta-entities and relations are missing in the TinyOS metamodel currently used by the RE4TinyOS parser. However, these unconverted specifications were still kept as annotations inside the serialized model and when any changes made to the model in the visual editor, these specifications were automatically integrated with the new code generated from the modified model.

V. Conclusion

A reverse engineering methodology and its tool, both called RE4TinyOS, have been introduced in this paper. RE4TinyOS enables retrieving the application models from TinyOS programs written in nesC, which paves the way for using these models inside an MDE toolchain. Hence, any modification made in the application code can be reflected into the application model and vice versa. Conducted case studies showed that both model-code synchronization and the integration of existing TinyOS applications which do not have system models previously, into the proposed MDE are possible with using RE4TinyOS. However, the achieved results also showed that some of the internal TinyOS event specifications of these existing applications can not be represented in the newly generated models since corresponding meta-entities are missing in the

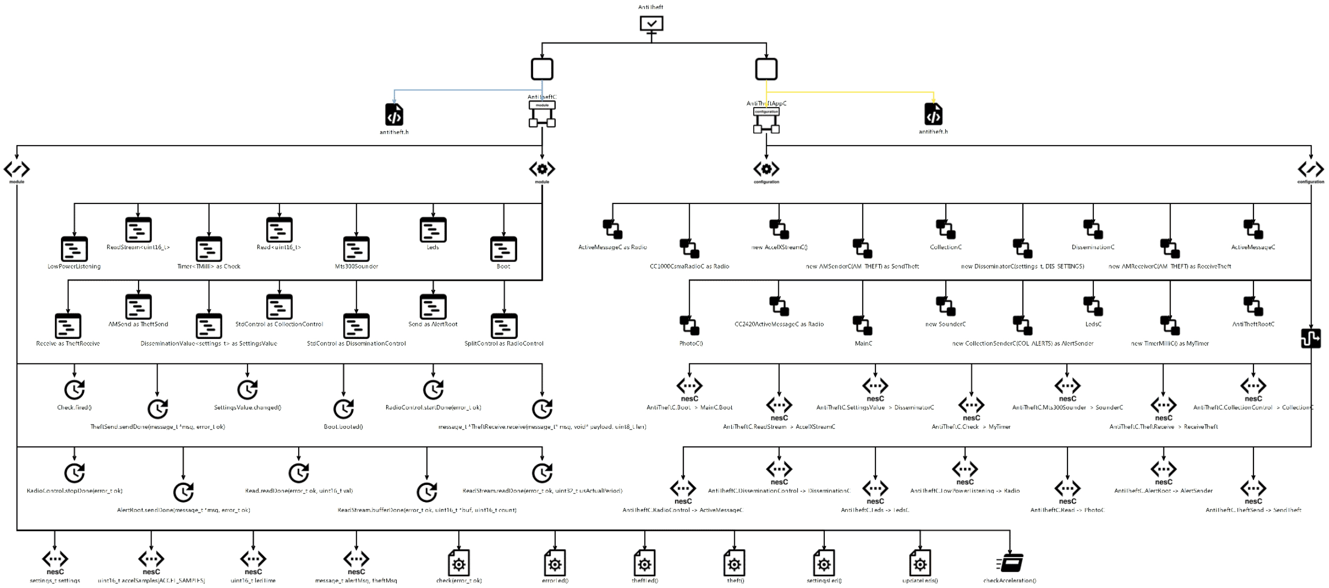

AntiTheft
AntiTheftC
module
antitheft.h
AntiTheftAppC
configuration
antitheft.h
module
module
configuration
configuration
LowPowerListening
ReadStream<uint16_t>
Timer<TMilli> as Check
Read<uint16_t>
Mts300Sounder
Leds
Boot
Receive as TheftReceive
AMSend as TheftSend
DisseminationValue<settings_t> as SettingsValue
StdControl as CollectionControl
StdControl as DisseminationControl
Send as AlertRoot
SplitControl as RadioControl
Check.fired()
TheftSend.sendDone(message_t *msg, error_t ok)
SettingsValue.changed()
Boot.booted()
RadioControl.startDone(error_t ok)
message_t *TheftReceive.receive(message_t* msg, void* payload, uint8_t len)
RadioControl.stopDone(error_t ok)
AlertRoot.sendDone(message_t *msg, error_t ok)
Read.readDone(error_t ok, uint16_t val)
ReadStream.bufferDone(error_t ok, uint16_t *buf, uint16_t count)
ReadStream.readDone(error_t ok, uint32_t usActualPeriod)
nesC
settings_t settings
uint16_t accelSamples[ACCEL_SAMPLES]
uint16_t ledTime
message_t alertMsg, theftMsg
check(error_t ok)
errorLed()
theftLed()
theft()
settingsLed()
updateLeds()
checkAcceleration()
ActiveMessageC as Radio
CC1000CsmaRadioC as Radio
new AccelXStreamC()
new AMSenderC(AM_THEFT) as SendTheft
CollectionC
new DisseminatorC(settings_t, DIS_SETTINGS)
DisseminationC
new AMReceiverC(AM_THEFT) as ReceiveTheft
ActiveMessageC
PhotoC()
CC2420ActiveMessageC as Radio
MainC
new SounderC
new CollectionSenderC(COL_ALERTS) as AlertSender
LedsC
new TimerMilliC() as MyTimer
AntiTheftRootC
AntiTheftC.Boot -> MainC.Boot
AntiTheftC.ReadStream -> AccelXStreamC
AntiTheftC.SettingsValue -> DisseminatorC
AntiTheftC.Check -> MyTimer
AntiTheftC.Mts300Sounder -> SounderC
AntiTheftC.TheftReceive -> ReceiveTheft
AntiTheftC.CollectionControl -> CollectionC
AntiTheftC.RadioControl -> ActiveMessageC
AntiTheftC.DisseminationControl -> DisseminationC
AntiTheftC.Leds -> LedsC
AntiTheftC.LowPowerListening -> Radio
AntiTheftC.Read -> PhotoC
AntiTheftC.AlertRoot -> AlertSender
AntiTheftC.TheftSend -> SendTheft

Fig. 8: Graphical model of the AntiTheft application

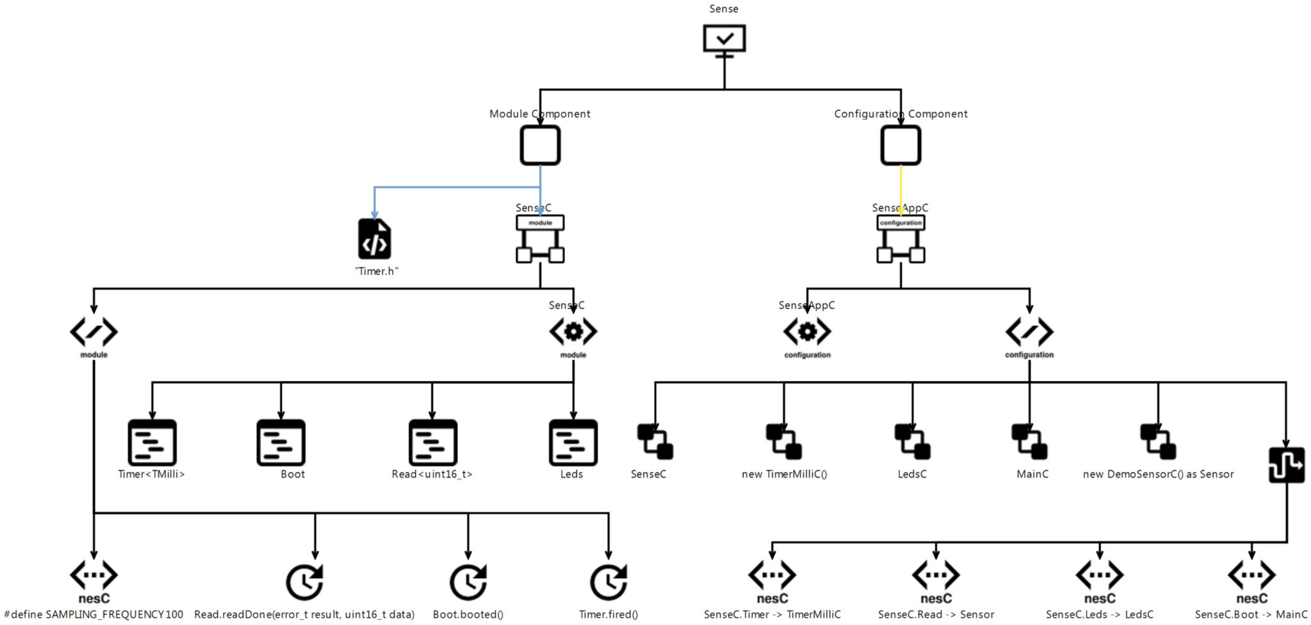


Fig. 9: Graphical model of the Sense application

current TinyOS metamodel used by the RE4TinyOS parser. In our future work, we aim at first extending this metamodel to cover all event internals while keeping the abstraction level and then improving the parser features with the utilization of this new metamodel.

Acknowledgment

Hussein Marah would like to thank Turkish government for Turkiye Scholarships (YTB) program. This research was partially supported by Flanders Make, a Flemish strategic research center for the manufacturing industry.

REFERENCES

[1] M. A. Matin and M. Islam, “Overview of wireless sensor network,” *Wireless Sensor Networks-Technology and Protocols*, pp. 1–3, 2012.

[2] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler, “TinyOS: An operating system for sensor networks,” in *Ambient Intelligence*, W. Weber, J. M. Rabaey, and E. Aarts, Eds. Springer Berlin Heidelberg, 2005, pp. 115–148. doi: https://doi.org/10.1007/3-540-27139-2_7

[3] P. Levis and D. Gay, *TinyOS Programming*. Cambridge University Press, 2009.

[4] H. M. Marah, R. Eslampanah, and M. Challenger, “DSML4TinyOS: Code Generation for Wireless Devices,” in *ACM/IEEE 21st International Conference on Model Driven Engineering Languages and Systems (MODELS), Model-Driven Engineering for the Internet-of-Things (MDE4IoT)*, 2018, pp. 509–514.

[5] T. Hettel, M. Lawley, and K. Raymond, “Model synchronisation: Definitions for round-trip engineering,” in *Theory and Practice of Model Transformations*, ser. Lecture Notes in Computer Science, A. Vallecillo, J. Gray, and A. Pierantonio, Eds. Springer Berlin Heidelberg, 2008, pp. 31–45.

[6] H. Giese and R. Wagner, “From model transformation to incremental bidirectional model synchronization,” *Software & Systems Modeling*, vol. 8, no. 1, pp. 21–43, 2009. doi: 10.1007/s10270-008-0089-9

[7] L. Favre, *Model Driven Architecture for Reverse Engineering Technologies: Strategic Directions and System Evolution*. Engineering Science Reference, 2010, google-Books-ID: e4RLuAAACAAJ.

[8] I. Malavolta and H. Muccini, “A study on MDE approaches for engineering wireless sensor networks,” in *2014 40th EUROMICRO Conference on Software Engineering and Advanced Applications*, 2014, pp. 149–157, ISSN: 2376-9505. doi: https://doi.org/10.1109/SEAA.2014.61

[9] F. Essaadi, Y. Ben Maissa, and M. Dahchour, “MDE-based languages for wireless sensor networks modeling: A systematic mapping study,” in *Advances in Ubiquitous Networking 2*, ser. Lecture Notes in Electrical Engineering, R. El-Azouzi, D. S. Menasche, E. Sabir, F. De Pellegrini, and M. Benjillali, Eds. Springer, 2017, pp. 331–346. doi: https://doi.org/10.1007/978-981-10-1627-1_26

[10] M. A. Saad, E. Fehr, N. Kamenzky, and J. Schiller, “ScatterClipse: A model-driven tool-chain for developing, testing, and prototyping wireless sensor networks,” in *2008 IEEE International Symposium on Parallel and Distributed Processing with Applications*, 2008, pp. 871–885, ISSN: 2158-9208. doi: https://doi.org/10.1109/ISPA.2008.22

[11] N. X. Thang and K. Geihs, “Model-driven development with optimization of non-functional constraints in sensor network,” in *Proceedings of the 2010 ICSE Workshop on Software Engineering for Sensor Network Applications*, ser. SESENA ’10. ACM, 2010, pp. 61–65. doi: https://doi.org/10.1145/1809111.1809128

[12] K. Doddapaneni, E. Ever, O. Gemikonakli, I. Malavolta, L. Mostarda, and H. Muccini, “A model-driven engineering framework for architecting and analysing wireless sensor networks,” in *Proceedings of the Third International Workshop on Software Engineering for Sensor Network Applications*, ser. SESENA ’12. IEEE Press, 2012, pp. 1–7. doi: https://doi.org/10.1109/SESENA.2012.6225729

[13] R. Shimizu, K. Tei, Y. Fukazawa, and S. Honiden, “Model driven development for rapid prototyping and optimization of wireless sensor network applications,” in *Proceedings of the 2Nd Workshop on Software Engineering for Sensor Network Applications*, ser. SESENA ’11. ACM, 2011, pp. 31–36. doi: https://doi.org/10.1145/1988051.1988058

[14] V. Veiset and L. M. Kristensen, “Transforming platform independent CPN models into code for the TinyOS platform: A case study of the RPL protocol,” in *PNSE+ModPE*, 2013.

[15] A. Salman, “Reducing complexity in developing wireless sensor network systems using model-driven development,” phdthesis, University of Salford, 2017. doi: http://usir.salford.ac.uk/44127/

[16] T. Rodrigues, F. C. Delicato, T. Batista, P. F. Pires, and L. Pirmez, “An approach based on the domain perspective to develop WSAN applications,” *Software & Systems Modeling*, vol. 16, no. 4, pp. 949–977, 2017. doi: 10.1007/s10270-015-0498-5

[17] C. Raibulet, F. A. Fontana, and M. Zanoni, “Model-driven reverse engineering approaches: A systematic literature review,” *IEEE Access*, vol. 5, pp. 14 516–14 542, 2017. doi: 10.1109/ACCESS.2017.2733518

[18] H. Brunelire, J. Cabot, G. Dup, and F. Madiot, “MoDisco: A model driven reverse engineering framework,” *Information and Software Technology*, vol. 56, no. 8, pp. 1012–1032, 2014. doi: 10.1016/j.infsof.2014.04.007

[19] L. Favre, L. Martinez, and C. Pereira, “MDA-based reverse engineering of object oriented code,” in *Enterprise, Business-Process and Information Systems Modeling*, ser. Lecture Notes in Business Information Processing, T. Halpin, J. Krogstie, S. Nurcan, E. Proper, R. Schmidt, P. Soffer, and R. Ukor, Eds. Springer, 2009, pp. 251–263. doi: https://doi.org/10.1007/978-3-642-01862-6_21

[20] F. Barbier, S. Eveillard, K. Youbi, O. Guitton, A. Perrier, and E. Cariou, “Model-driven reverse engineering of cobol-based applications,” in *Information Systems Transformation*. Elsevier, 2010, pp. 283–299.

[21] V. Cosentino, J. Cabot, P. Albert, P. Bauquel, and J. Perronnet, “A model driven reverse engineering framework for extracting business rules out of a java application,” in *Rules on the Web: Research and Applications*, ser. Lecture Notes in Computer Science, A. Bikakis and A. Giurca, Eds. Springer, 2012, pp. 17–31. doi: https://doi.org/10.1007/978-3-642-32689-9_3

[22] . Sanchez Ramon, J. Sanchez Cuadrado, and J. Garcia Molina, “Model-driven reverse engineering of legacy graphical user interfaces,” *Automated Software Engineering*, vol. 21, no. 2, pp. 147–186, 2014. doi: 10.1007/s10515-013-0130-2

[23] I. Comyn-Wattiau and J. Akoka, “Model driven reverse engineering of NoSQL property graph databases: The case of neo4j,” in *2017 IEEE International Conference on Big Data (Big Data)*, 2017, pp. 453–458. doi: https://doi.org/10.1109/BigData.2017.8257957

[24] J. Snchez Cuadrado, E. Guerra, and J. de Lara, “Reverse engineering of model transformations for reusability,” in *Theory and Practice of Model Transformations*, ser. Lecture Notes in Computer Science, D. Di Ruscio and D. Varr, Eds. Springer International Publishing, 2014, pp. 186–201. doi: https://doi.org/10.1007/978-3-319-08789-4_14

[25] D. Gay, P. Levis, R. Von Behren, M. Welsh, E. Brewer, and D. Culler, “The nesC language: A holistic approach to networked embedded systems,” *Acm Sigplan Notices*, vol. 38, no. 5, pp. 1–11, 2003.

[26] T. Parr and K. Fisher, “LL(*): The foundation of the ANTLR parser generator,” in *Proceedings of the 32Nd ACM SIGPLAN Conference on Programming Language Design and Implementation*, ser. PLDI ’11. ACM, 2011, pp. 425–436, event-place: San Jose, California, USA. doi: https://doi.org/10.1145/1993498.1993548

[27] T. Parr, *The Definitive ANTLR 4 Reference*, 2nd ed. Pragmatic Bookshelf, 2013.

[28] T. J. Parr and R. W. Quong, “Antlr: A predicated-ll (k) parser generator,” *Software: Practice and Experience*, vol. 25, no. 7, pp. 789–810, 1995.

[29] TinyOS_Github_Repository, “Tinyos antitheft application,” 2013. doi: https://github.com/tinyos/tinyos-main/tree/master/apps/AntiTheft

[30] TinyoS_Github_Repository, “Tinyos sense application,” 2013. doi: https://github.com/tinyos/tinyos-main/tree/master/apps/Sense

DOI: 10.15439/2020F176 ISSN 2300-5963 ACSIS, Vol. 21

The Syntax of a Multi-Level Production Process Modeling Language

Marko Vještica, Vladimir Dimitrieski, Slavica Kordić, Sonja Ristić, Ivan Luković
University of Novi Sad, Faculty of Technical Sciences,
Trg Dositeja Obradovića 6, 21102 Novi Sad, Serbia
Email: {marko.vjestica, dimitrieski, slavica, sdristic, ivan}@uns.ac.rs

Milan Pisarić
KEBA AG Linz, Gewerbepark Urfahr Reindlstraße 51,
4041 Linz, Austria
Email: pisa@keba.com

Abstract—**The fourth industrial revolution introduces changes in traditional manufacturing systems and creates basis for a lot-size-one production. The complexity of production processes is significantly increased, alongside the need to enable efficient process simulation, execution, monitoring, real-time decision making and control. The main goal of our research is to define a methodological approach and a software solution in which the Model-Driven Software Development (MDSD) principles and Domain-Specific Modeling Languages (DSMLs) are used to create a framework for the formal description and automatic execution of production processes. In that way production process models are used as central artefacts to manage the production. In this paper, we propose a DSML which can be used to create production process models that are suitable for automatic generation of executable code. The generated code is used for automatic execution of production processes within a simulation or a shop floor.**

I. Introduction

ADVANCED technologies in the form of smart resources and smart products are the basis for the fourth industrial revolution as they enable changes in factories and production. Industry 4.0 introduces primarily IT-driven changes in existing production systems in order to enable production of individualized products while preserving all beneficial economic characteristics of mass production [1].

Producing highly individualized products in traditional production facilities requires multiple production lines or, in case of a single production line, stopping the production to allow reconfiguration of machines which causes additional costs. To enable a flexible, individualized, lot-size-one production that is economically viable, the production needs to be carried out without stopping a production line for machine reconfiguration [2]. Therefore, it is necessary to solve the problem of tedious machine adaptation to frequent production changes that are common in the context of Industry 4.0. Additionally, there is a problem of frequent location changes of human workers in a factory [3]. Due to decreasing number of workers and increasing level of automation in factories, the workers are performing different tasks within a factory. Frequently changing worker's tasks increases production dynamics and requires fine coordination of workers in a factory so their work can be optimized, and production downtime avoided. As worker's tasks are often changed, a fast knowledge transfer is required so they do not lose time when changing workplaces.

To enable production of individualized products at the lower cost, a solution for production orchestration at a higher abstraction level can be utilized [4]. This solution would require a formal method to specify production processes and create process models that are suitable for automatic generation of instructions that are executed on smart resources. A smart resource represents a machine or a human worker that receives generated instructions and execute them on materials and products.

In this context, it is possible to apply a Model-Driven Software Development (MDSD) approach in which a centralized representation of knowledge would exist in a form of production process models. Therefore, in our previous work [5], we proposed a novel MDSD approach for production process modeling and automatic production process execution. The MDSD approach aims to reduce the gap between individual customer needs and the ability to produce required products. The main goals of the proposed MDSD approach are to: (i) enable easier adaptation of machines to dynamic changes of production processes, (ii) improve coordination of human workers and machines in factories and (iii) enable automatic execution of production processes. A formal specification of a production process is the crucial part of the proposed approach. Existing process modeling languages are not tailored to model production processes [6]. Currently, production processes are specified using different models like Bill of Materials (BOM), Flow Process Chart (FPC) and Failure Mode and Effects Analysis (FMEA) sheets. These models have different syntaxes and semantics. Therefore, it is hard to combine and reason production details from them in order to enable automatic execution of production processes.

To the best of our knowledge, there is no unified formal language aimed at modeling all production process aspects required for an automatic execution. Therefore, we decided to create a new Domain-Specific Modeling Language (DSML) aimed at production process modeling. Our MDSD

This paper is supported by KEBA AG Linz and by the Ministry of Education, Science and Technological Development through the project no. 451-03-68/2020-14/200156: "Innovative scientific and artistic research from the FTS domain".

approach, overviewed in Section 2, would enable flexible manufacturing with a help of Orchestrator software that manages production processes using a knowledge base and models created with the DSML. Orchestrator is a software running on a cluster of industrial computers that enables orchestration, detection and configuration of new and existing smart resources [7].

In this paper, we present abstract and concrete syntaxes of the DSML based on our previous research [5]. The Multi-Level Abstraction Approach (MLAA) is employed to develop the DSML. MLAA refers to representing objects at multiple levels of abstraction hierarchies. Due to the application of MLAA, we denote the language as Multi-level Production process modeling Language (MultiProLan). The higher level of abstraction enables easier production process modeling by specifying only production process steps, and the lower level of abstraction enables modeling of all the execution details dependent on a production system. MultiProLan allows process and quality engineers to collaborate on the specification of a production process by using a common language. In this paper, we denote process and quality engineers together as process designers. A process designer is a person in charge of transforming a valuable idea or experiment into an industrial process in a way to fulfil not only originality, efficiency, quality and sustainability criteria, but to consider a large number of often contradictory constraints.

MultiProLan enables modeling of production processes suitable for automatic execution. It can be used in a flexible and orchestrated production to facilitate the lot-size-one production. Supported with MultiProLan, our MDSD approach should increase the degree of factory automation by enabling easier adaptation of machines to dynamic production changes and by increasing coordination of resources in factories. Models expressed by the concepts of MultiProLan are simple enough for a human comprehension and can be also used as means of knowledge transfer to new workers or to workers that change their workplace frequently. Modeling production processes is important so human workers and supervisors could understand the processes better, eliminate potential modeling errors and optimize the processes.

Besides Introduction, this paper is structured as follows. An overview of the MDSD approach for modeling and automatic execution of production processes and the MultiProLan basic concepts are presented in Section 2. The related work that includes different modeling languages and approaches is summarized in Section 3. Abstract and concrete syntaxes of MultiProLan are described in Section 4. Conclusions and the future work are presented in Section 5.

II. An Overview of the MDSD Approach for Modeling and Automatic Execution of Production Processes

In the Model-Driven (MD) paradigm, models represent a central artefact at all stages of system development. A system developed by following the MD paradigm includes models that are connected and organized at different abstraction levels. An MDSD approach is a part of the MD paradigm and some of its goals are to: (i) increase software system developing speed through automatization and centralized representation of knowledge, (ii) increase software quality through formalization, (iii) increase reusability of models and (iv) lower system complexity through abstraction levels [8]. In MDSD approaches, DSMLs can be used and their purpose is to bring modeling concepts closer to users familiar with an application domain, so they can specify their solution with less time in comparison to General-Purpose Modeling Languages (GPMLs) [9]. Therefore, our opinion is that an MDSD approach and DSMLs will have significant role in enabling flexible, orchestrated and highly automated production. This is why we proposed a novel MDSD approach for modeling and automatic execution of production processes [5].

In Fig. 1, a system for automatic production orchestration and process execution is presented. The components that support the main steps of the MDSD approach are numerated and grouped within dashed rectangles. The proposed MDSD approach comprises the following steps: (i) specification of technological process models performed by process designers, (ii) automatic enrichment of technological process models with details needed for the execution, performed by Orchestrator on the basis of semantics gathered from Knowledge Base, (iii) generating the executable code performed by Code Generator and (iv) execution of generated instructions performed by Executor that forwards instruction to Digital Twin, and to the smart factory shop floor indirectly. A digital twin represents virtual model of a physical object. It can simulate the object behavior and the object can respond to changes made in the simulation [10]. The main part of the proposed system is MultiProLan created for the domain of hardware production. By using MultiProLan it is possible to create models that are suitable for automatic code generation and execution. The generated code represents human-readable or machine-

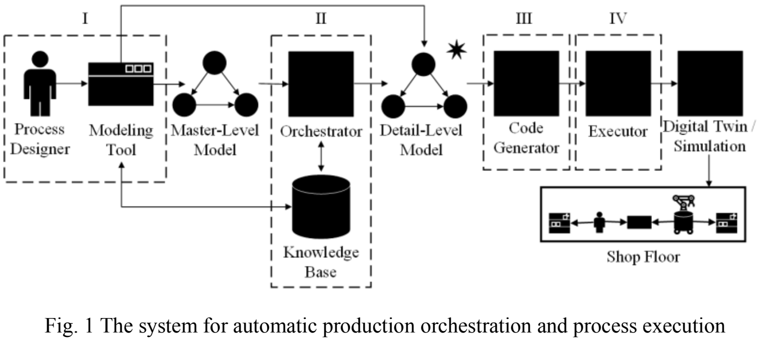


Fig. 1 The system for automatic production orchestration and process execution

readable instructions that are to be executed by smart resources. More detailed description of this approach is given in the rest of the section.

Specification of technological processes. The first step of the MDSD approach represents specification of production process models by using MultiProLan. These models include process steps without details required for automatic production, such as: smart resources required to execute process steps; production logistic activities; specific storages in which products and parts are stored; and machine configuration activities. A graphical modeling tool is implemented to allow the modeling of production processes using MultiProLan. Modeling Tool is used by process designers to model production processes at the higher level of abstraction. Such models are called Master-Level (ML) models. These models represent technological description of production processes and they include: (i) process steps, (ii) required capabilities, i.e. skills required to execute a process step, with their parameters and constraints (iii) input and output products, i.e. transformed resources like raw materials, components or finished goods, with constraints, (iv) workflows, i.e. sequence, parallelism, selection and iteration patterns, and (v) collaboration between process steps. A collaboration between smart resources, both humans and machines, is crucial in the context of Industry 4.0 [11] and it needs to be modeled. ML models do not depend on a specific technological platform, i.e. on a factory in which modeled production processes will be executed. Therefore, ML models can be considered as Platform-Independent Models (PIMs).

Enrichment of ML production process models with details needed for the execution. A production process will be executed within a given production system, e.g. some factory. To use an ML model for automatic code generation and execution, it is necessary to place additional information in it. This information refers to elements of a given production system. The information include: (i) specific resources like robots, machines and humans, that are to perform process steps, (ii) production logistic activities, which represent transportation of products and resources, and (iii) configuration of machines and robots like software setup, changing grippers, and plugging into a charger or a workstation. ML models enriched with aforementioned information are called Detail-Level (DL) models. DL models can be considered as Platform-Specific Models (PSMs) as they are enriched with details that are specific to a production system in which the models will be executed. The notions of ML and DL are introduced in this paper to better facilitate description of different modeling levels, and we did not come across them in surveyed literature.

DL models can be created manually or automatically. Manual DL creation is conducted by a process designer. A process designer can make additional changes to the existing ML/DL model or create a DL model from scratch using Modeling Tool. However, in our vision of the Industry 4.0 production process modeling, a production system and the production process models should be separated to enable a high level of a product customization. Thus, automatic creation of DL models is supported in our MDSD approach. The automatic DL creation from the existing ML model is conducted by the means of Orchestrator software. In the following text, automatic DL creation process is explained.

Knowledge Base needs to provide all the necessary information about a given production system for Orchestrator to be able to automatically generate DL models from ML models. Every process step specified in an ML model contains a capability, i.e. a skill that is required so that a process step can be executed. It is necessary to add the information about a resource that is to execute the process step within the given production system. This cannot be just any resource, but the resource that has the required capability in its set of offered capabilities. By using Knowledge Base, Orchestrator can match a capability that is required in a process step with a capability that a specific resource offers and, in that way, matches the process step with the resource. A capability of one process step could be matched with a capability of multiple resources. Orchestrator needs to use optimization techniques and scheduling mechanisms to choose one resource for every process step and to optimize work of resources in a factory. A process step that is ready to be executed is composed of: (i) input products, (ii) a capability needed to execute the process step, (iii) a smart resource that is to perform the capability on input products, and (iv) output products.

Orchestrator also needs to take care of production logistics. Orchestrator needs to add storages in which required products are stored and to add process steps that facilitate transportation of products and movement of resources between storages and workstations. Production logistic activities have a big impact on production processes as they require a lot of time [12], so it is very important to organize these activities well. Orchestrator also takes care of machine configurations. Based on knowledge gathered from Knowledge Base, Orchestrator can infer whether the machine configuration step needs to be added to the process to enable further activities.

For Orchestrator to be able to reach the aforementioned conclusions, Knowledge Base needs to contain knowledge of production system elements, such as: (i) smart resources with their set of capabilities, (ii) smart products with their attributes like dimensions and weight (iii) process steps with required products and capabilities, (iv) production logistics and (v) configuration process steps that are required by some resources prior or after execution of another process step. In this paper, we look at Orchestrator as a black box. It is presented just to provide context in which MultiProLan is used. An internal structure of Orchestrator that is used in the MDSD approach can be found in our previous work [7].

An ML model exists independently from a production system that will be the execution platform for the modeled production process. At this high abstraction level, a process designer does not need to take care for the specific details of a given production system. These details must be specified within Knowledge Base before the specification of a DL model begins. DL models can contain only those capability, product, resource, storage, constraint and parameter details that are already specified in Knowledge Base. In that context, a DL model is specified whenever an execution-ready production process model is needed, and it is dependent on a production system.

Generating the executable code. The third step of the MDSD approach represents code generation from DL models. It is possible to send DL models into Code Generator so it could automatically generate instructions that can be executed by human workers or machines. More details on Code Generator can be found in [7].

Execution of generated code instructions. Executor forwards generated instructions to Digital Twin, which represents both simulation and command proxies to the shop floor. In our case, the Digital Twin component could be used for the simulation only or it could also forward instruction to shop floor smart resources through embedded proxies and mobile devices [7]. Using a digital twin in the simulation-only mode could decrease production failures, provide insight into badly modeled process steps and enable optimization of resources and processes [13]. By running simulations it is possible to predict an influence of process steps to a final product [14].

III. Related Work

Production processes should be digitally supported in Industry 4.0 [15] so they can be integrated within a smart factory. Modeling production processes is very important in industrial informatics [16], but it is not enough to document processes and store them in a factory database. Production processes should be modeled to lead the production. Process models should be ready for automatic production, but also not too complex for a human comprehension. In this section, different production process modeling approaches and languages are presented, as well as their capabilities to fulfill the aforementioned needs.

Companies mostly use manufacturing process charts and BOMs to specify production processes, but none of these specifications provide enough data to facilitate automatic execution. BOM specifications are not enough to understand a production flow [17]. On the other hand, Bill of Materials and Operations (BOMO) [18] specifications cover the production flow, but are insufficient to specify selection and iteration patterns or smart resources. There is also Korean manufacturing process chart standard KS A 3002 [19], but a tooling support and a possibility to automatically execute models are missing [17]. Unified Modeling Language (UML) activity diagrams are used to describe production processes, but models are not suitable for the automatic execution, they are not intuitive for process designers and they could be complex [20].

By using conceptual process modeling languages like UML activity diagram, Business Process Modeling and Notation (BPMN) and Petri nets, it is difficult to model production processes primarily as they are not created for that purpose. These difficulties are even more noticeable whenever the languages need to cover all production process concepts required for the automatic execution [6]. To solve this problem researches usually extend existing languages to add missing semantics. However, these extensions are not enough to solve the problem due to the wide application domain of a language. Therefore, researches often try to create new domain-specific languages instead of extending existing general-purpose languages [21].

Zor et al. proposed BPMN extensions to model production processes [22], however it is difficult to model a material flow [23] and the whole context of production domain is not covered due to the absence of uniformity [17]. BPMN extensions are also proposed by Ahn and Chang for production process similarity measurements [17], however it is not possible to model selection and iteration patterns or to specify smart resources. According to Lütjen and Rippel [23], some languages like DELMIA Process Engineer, Systems Modeling Language (SysML) and Petri nets lack in possibility to specify the material flow. To overcome the usual lack of the material flow modeling concept, the same authors proposed a novel material flow-oriented process modeling language – GRAMOSA, but the material flow-oriented approach was complex [23].

Meyer et al. [24] proposed BPMN extensions to model Internet of Things (IoT) devices and create IoT-aware process models. Besides humans that participate in business process executions, IoT-aware processes also include IoT devices that can do some of tasks in a smart factory. Likewise, Petrasch and Hentschke [25] proposed IoT-Aware Process Modeling Method (IAPMM) using UML use cases and BPMN extensions in order to model IoT-aware processes. The goal of this method is to enable modeling of software systems and software applications like sensing and actuation. The same authors extended IAPMM and created Industry 4.0 Process Modeling Language (I4PML) [26] by adding extensions like Cloud Computing applications. Using this language, it is not possible to model all the technological details, as its purpose is to model production processes in a requirements specification and analysis phase. According to Schönig et al. [27], none of the aforementioned languages and approaches provides details on how to execute models. This is the reason why they proposed an approach for integration of IoT objects with business process models ready for an execution. They extended BPMN to enable integration of IoT objects with process models, but also to preserve a possibility to execute the models in existing Business Process Management (BPM) execution systems. However, it would be difficult to specify the material flow, smart resources, products, capabilities and constraints, and thus the full automatization, in which both humans and machines participate, would be hard to achieve. Because of these insufficiencies, Orchestrator would not be able to manage the production based on the models.

Witsch and Vogel-Heuser [20] presented Manufacturing Execution System Modeling Language (MES-ML) whose purpose is to specify MES through different views so that model complexity could be reduced. MES-ML is based on BPMN and covers the modeling of a technical system, production processes and MES/IT functions. By using links, it is possible to connect process steps with production system elements, i.e. smart resources, that will execute the steps. This way a dependency between production process models and a production system is created. Due to this dependency, a process designer needs to take care how to connect process steps with production system elements during the production process modeling. This makes the production process modeling significantly more difficult and could lead to higher number of created errors during the modeling and higher model complexity.

According to Weissenberger et al. [28], MES-ML does not support creation of generic production processes as the semantics of process tasks are insufficiently specified and process models are not suitable for code generation. To enable the modeling of machine-usable MES specifications suitable for code generation, the same authors implemented a DSML by extending MES-ML. The goal of this language is to enable higher independency of production process models from a production system during process modeling. Instead of the link that is used to connect a process step with a resource of a production system, the authors proposed a list of links to be used. At the runtime, resources that execute process steps will be determined. However, the dependency between process steps and production system resources still exists and it is ambiguous which resources will execute process steps until the runtime.

Similar to the previous work, Fallah et al. [4] presented a framework to model a modular MES using SysML. However, the framework is not implemented. Neither a code generator for model transformation into executable code nor an interpreter for direct model execution are implemented.

Because of the dependency between production process models and a production system, we decided to create the language with two levels of abstraction. In this way, process designers do not need to take care of production system elements during the production process modeling and they can be entirely focused on modeling process steps. Production process models become more generic by separating a production system from them. It is possible to automatically connect process steps with smart resources in the runtime without additional load to process designer by using Orchestrator. As we could not find any formal language that allows creation of generic production process models suitable for automatic execution, we decided to create a novel DSML. This DSML unifies all production process aspects, as mentioned in Section 1, and thus enables the specification of DL models that are used for automatic code generation and production process execution. ML models are separated from a production system so that process designers could model them in more generic way.

IV. Abstract and Concrete Syntaxes of MultiProLan

In this section we present abstract and concrete syntaxes of MultiProLan for modeling production processes suitable for automatic code generation and execution. We use an Ecore meta-meta-model, which is a part of Eclipse Modeling Framework (EMF) [29], to create the abstract syntax of MultiProLan. Also, we use the Eclipse Sirius framework [30] to create the graphical concrete syntax and to enable simple implementation of a prototype tool.

A. The Abstract Syntax of MultiProLan

Two levels of abstraction are needed to ease the modeling performed by process designers, but also to fully prepare models for an execution phase. A higher abstraction level – ML separates production process models from a production system, while a lower abstraction level – DL enables creation of production process models that are executable within a given production system. Based on these levels of abstraction, we divided the meta-model into two parts. This was also done because the meta-model is more concise and easier to understand.

The ML part of the MultiProLan meta-model is depicted in Fig. 2 and it represents production process modeling concepts needed at the higher level of abstraction. These concepts are used by process designers to create ML models. A production process is modeled by the *Process* class which represents the root model element. A process version must be specified as models are stored in a knowledge base and can be changed or reused at any time. A process is composed of process elements (*ProcessElement*), which can

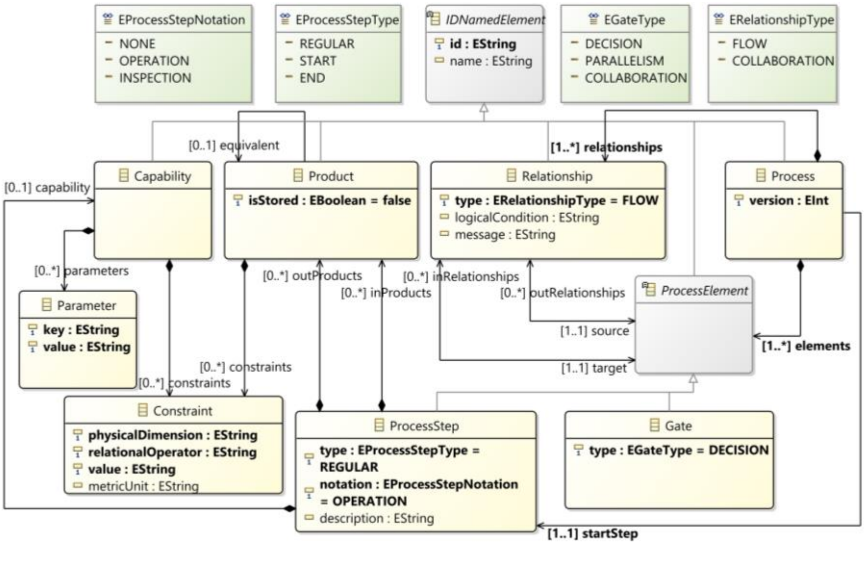


Fig. 2 The first part of the meta-model used for ML model creation

be process steps (*ProcessStep*) or gates (*Gate*), and relationships (*Relationship)* between them. The start process step must be referenced from a process (*startStep*) as knowledge of the execution starting point is needed. There are two types of relationships (*ERelationshipType)*: (i) flow – representing a workflow between process elements, and (ii) collaboration – representing a message flow between process steps. Relationships have the message attribute specified whenever a message needs to be sent between collaboration process steps. Also, relationships have the logical condition specified whenever they are used in selection or iteration patterns.

A process step is composed of a capability (*Capability*) and products (*Product*) on which the capability is to be performed. Input products (*inProducts*) represent products on which a capability is performed, i.e. raw materials, and output products (*outProducts*) represent products that are the result of the capability usage, i.e. finished goods. Process steps can be of different types (*EProcessStepType*): (i) start – the first process step, (ii) end – the last process step or (iii) regular – other process steps that contain capabilities that must be performed on products. Start and end process steps do not have any capability or product, and only one start process step and only one end process step have to exist per each production process model. A process step has a notation (*EProcessStepNotation*) which has one of the following values: (i) none – for start and end process steps, (ii) operation – an activity that changes input products and creates output products and (iii) inspection – an activity to check quality of products.

A material flow should be specified for every product. An input product can be equivalent (*equivalent*) to an output product of the previous process step, or it can be brought from a storage. An output product can be used in following process steps or it can be stored in a storage. Every product and capability have constraints (*Constraint*) such are dimensions, color and weight that will be considered by Orchestrator when it decides which smart resource is able to perform a process step. Some capabilities require parameters (*Parameter*) to be specified, e.g. to drill a hole, the drilling position must be specified.

Besides process steps, there are also gates that are used as process elements. Gates are elements that are needed in order to create: (i) selection and iteration patterns – flow control in processes, (ii) parallelism – two or more process steps need to be executed in parallel and (iii) collaboration – two or more process steps need to be executed in parallel, but one process step must not start or finish its activity before gets a message that another process step finished its activity. Finally, most of the presented classes inherit the *IDNamedElement* class comprising *id* and *name* attributes.

The DL part of the MultiProLan meta-model is depicted in Fig. 3 and it represents production process modeling concepts needed at the lower level of abstraction. This part of the meta-model is an extension of the ML part and together they are used to create DL models. Process step notations are extended by (i) transportation – production logistic activities, (ii) configuration – activities to configure resources and (iii) delay – necessary waiting activities. A process step is extended with a resource that will execute it by using a required capability. A resource (*Resource*) can be an actuator – an active resource, i.e. one that performs different activities during the production, or a storage – a passive resource, i.e. one that stores products. A resource can be both an actuator and a storage, e.g. there are robots that can execute different tasks, but also have a place to temporarily store products. A resource can be a human worker or a machine (*EResourceType*) and it can also represent an actuator or a storage. Depending on the resource type, human-readable or machine-readable instructions will be generated for every process step. Also, a resource could be of type *NONE* which means that it is neither a human nor a machine, e.g. a regular storage shelf, with no smart devices or sensors attached. Products are extended with a specific storage that must be defined for every input product brought from the storage and for every output product placed in the storage. When extended with active and passive resources, production logistics and configuration activities, process steps are ready for the automatic code generation and execution.

B. *The Concrete Syntax of MultiProLan*

There are two types of concrete syntaxes – textual and graphical, but there is no general answer which one is more suitable [31]. We decided to create the graphical syntax for MultiProLan to make the modeling easier for production

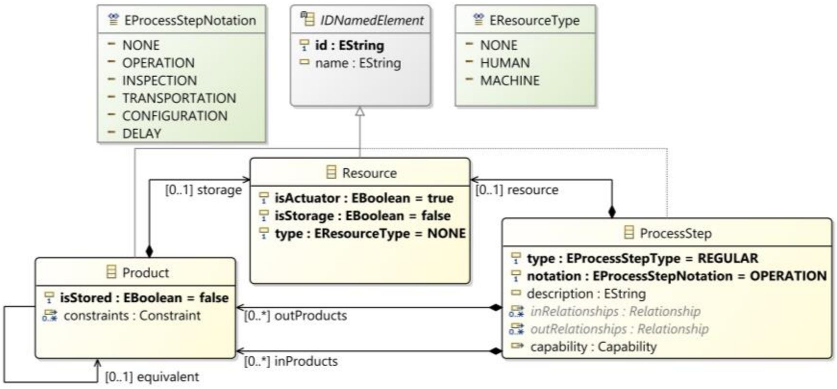


Fig. 3 The second part of the meta-model used for DL model creation

process designers as they are already familiar with other graphical languages, such as FPC. The decision was also made to enable visualized process monitoring, as well as to enable visualization of detected errors during the production. As BPMN [32] is commonly used to model different kind of processes and as it is easy to interpret its models [33], some BPMN concepts, such as activities and gates, are used in the graphical syntax of MultiProLan. The graphical syntax is also inspired by American Society of Mechanical Engineers (ASME) FPCs [34] as process designers are used to these charts. Some of FPC elements are used in process step notations, such as: operation, transportation, inspection, and delay. Also, the storage element is used within a product, indicating that a product should be gathered from a storage or placed in a storage. The symbols used for the MultiProLan concrete syntax are presented in Fig. 4.

The concrete syntax is described within production process model examples presented in Fig. 5 and Fig. 6. These two examples represent a process of a wooden box production at ML and DL of abstraction, respectively. The box is composed of four wooden planks that represent different sides of the box, and of a thin wooden back side. The four wooden planks can be assembled into a frame using wooden pins, and the wooden back side needs to be hammered into the frame, creating the box. The production of the wooden box is installed in a smart factory composed of: (i) the smart shelf – storage in which wooden planks are stored, (ii) the first assembly table – storage that is used to assemble four wooden sides, (iii) the second assembly table – storage that is used to hammer the back side into the frame, (iv) the recycle bin – storage for impaired boxes, (v) the finishing area – storage for finished boxes and (vi) human workers and industrial mobile robots – smart resources that are able to perform required activities.

The ML model of the wooden box production is presented in Fig. 5. The presented ML model is composed of six parts: (i) the *start* process step, (ii) parallel process steps of assembling left-bottom and right-upper sides, after which these two assembled sides should also be assembled into the frame, (iii) collaboration process steps of holding the frame and hammering the back side into the frame, (iv) inspection of the box, (v) decision whether the box needs to be stored or discarded, depending on results of the *inspection* process step and (vi) the *end* process step. The process step of assembling the left-bottom side represents an operation as it is depicted with a circle icon at the left side of the process step name. It has two input products, left and bottom sides of the frame, both gathered from a storage. The inverted triangle icon at the left side of a product name represents that an input product should be gathered from a storage, or that an output product should be placed in a storage. Two input products have two constraints, width and height, that will be considered by Orchestrator when it assigns a smart resource that is able to pick the plank of these dimensions. The same process step has the *assemble* capability with parameters that represent two wooden pins with the space between them of 0.07m. The output product of this process step is the assembled left-bottom side, which will not be stored, but will be used by the next process step. Assembling the right-upper side is an equivalent process step to assembling the left-bottom side process step. Both process steps need to be executed in parallel, as they are modeled between two parallelism gates (*PAR*). The next process step requires to assemble the frame and it has two input products, which are left-bottom and right-upper sides from the previous two process steps. These input products are not gathered from a storage but are equivalent to the previous process steps output products, as it is depicted by directed dashed lines in the process diagram. This process step has the *assemble* capability and the frame as the output product. The same frame is held in the next process step. This process step is a part of the collaboration activities, represented between two

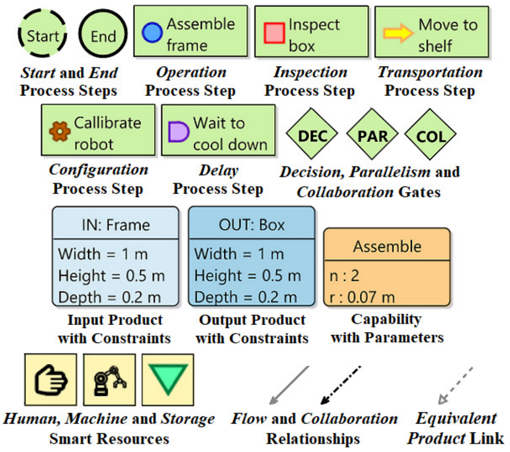

Start
End
Assemble frame
Inspect box
Move to shelf
Start and End Process Steps
Operation Process Step
Inspection Process Step
Transportation Process Step
Callibrate robot
Wait to cool down
DEC
PAR
COL
Configuration Process Step
Delay Process Step
Decision, Parallelism and Collaboration Gates
IN: Frame
Width = 1 m
Height = 0.5 m
Depth = 0.2 m
OUT: Box
Width = 1 m
Height = 0.5 m
Depth = 0.2 m
Assemble
n : 2
r : 0.07 m
Input Product with Constraints
Output Product with Constraints
Capability with Parameters
Human, Machine and Storage Smart Resources
Flow and Collaboration Relationships
Equivalent Product Link

Fig. 4 Symbols of the MultiProLan concrete syntax

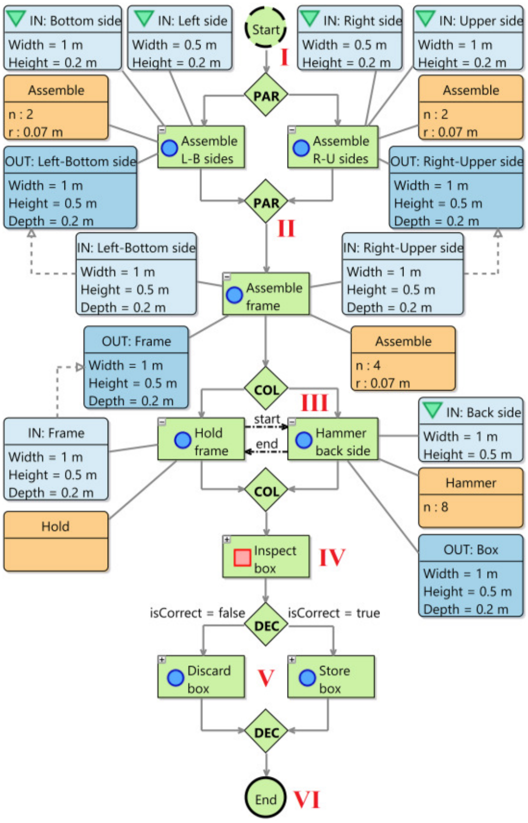


Fig. 5 The ML model of the wooden box production example

collaboration gates (*COL*). It does not have an output product as it is the same as the input product. Another process step of the collaboration activities is to hammer the back side into the frame that is held. Hammering the back side should not start before the message arrives that the frame is being held. The frame should be held until the message arrives that the hammering is finished. This is presented in the process diagram with dotted-line relationships between those two process steps. The input product of the hammering process step is the back side that should be gathered from a storage and the output product is the box. The *hammer* capability has predefined number of nails that should be hammered, e.g. eight, and after the hammering is finished, the message is sent to the *hold* process step. After the collaboration process steps are finished, the box is inspected for any deformation. The *inspection* process step and process steps between decision gates also have input and output products and a capability, but they are hidden from the diagram using the +/- button at the top left corner of process steps. The decision of storing or discarding the box should be made depending on whether the box passes all checks. These process steps are modeled between two decision gates (*DEC*). The process is finished after it reaches the *end* process step.

Based on the presented ML model and knowledge from Knowledge Base, Orchestrator generates the DL model of the wooden box production, which is presented in Fig. 6. Due to the paper length limitations, products and capabilities are depicted just for process steps in the left parallelism branch, while for other process steps they are modeled, but not presented on the diagram. Like the presented ML model, the generated DL model is composed of the same six parts, but the model is extended with additional details and new process steps, like production logistic activities and mobile robot configurations. These new process steps are needed to automatically produce the box. In the rest of this subsection, we describe some of the process steps, while others are extended in the similar way. The *assemble left-bottom side* and the *assemble right-upper side* process steps are assigned in parallel to a human worker and an industrial mobile robot, respectively. In both parallel branches transportation process steps have been added, which are depicted with the arrow icon at the left side of the process step name. To assemble the left-bottom side, the human worker needs to move to the smart shelf, pick left and bottom sides, move to the first assembly table and assemble these two sides. Transportation process steps only have the *move* capability with the *location* parameter, as products for these steps do not exist. The *pick* process steps have a capability and an input product, but an output product does not exist. Unlike the ML model in which input products have general storages as an indicator that they need to be gathered, the DL model input products have the specific storages, e.g. smart shelf, from which the products need to be gathered. These specific storages are depicted by inverse triangle objects set on input products. By selecting a storage, it is possible to specify values of the storage attributes, but this is not presented in the diagram due to the paper length limitations. Similar could be done with resources set on process steps. As for the *assemble* process step input products, they are equivalent to previously picked products, which is denoted with the directed dashed lines between equivalent products. The capability and the output product of this process step are the same as in the ML model.

Another parallel branch represents assembling of the right-upper side by the industrial mobile robot. Process steps in this branch are similar to process steps of the previously described branch, except of the configuration process steps. As the industrial mobile robot assigned to these process steps is not equipped with the machine vision modules, therefore it must be calibrated after each movement to

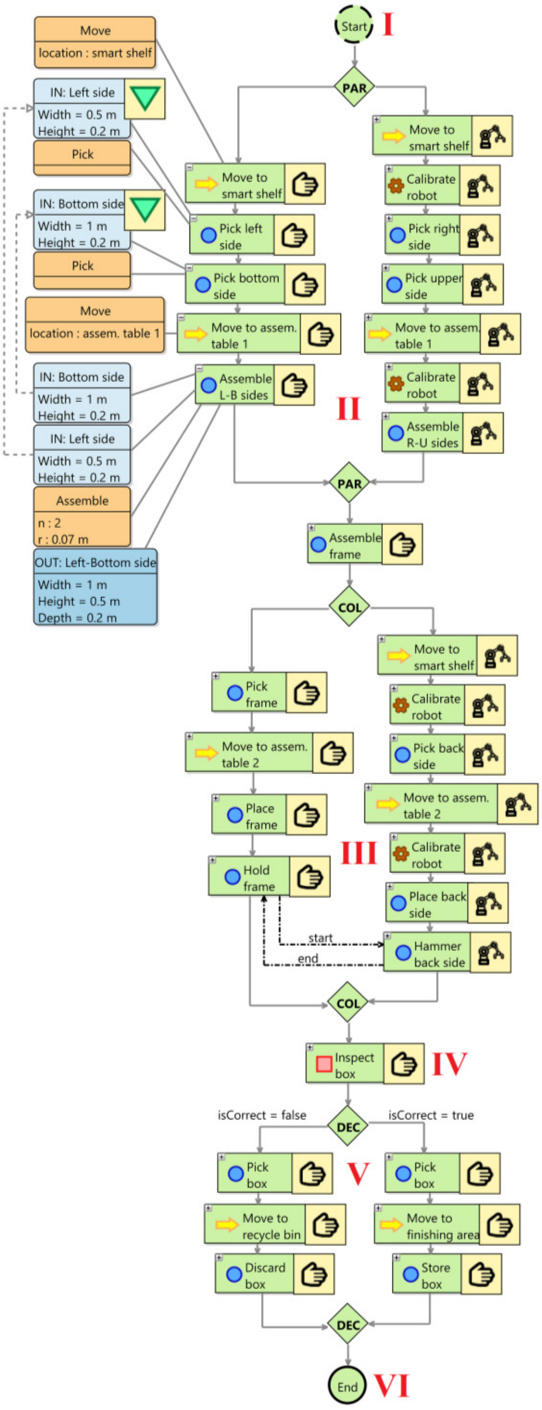


Fig. 6 The DL model of the wooden box production example

determine its position. Configuration process steps can be differentiated from other process steps by the gear icon at the left side of the process step name. After the left-bottom and right-upper sides are assembled, the same human worker needs to assemble the frame. This activity does not require any transportation process steps as the human worker and the required input products are already at the first assembly table. The assembled frame is used in the collaboration process steps that are extended with transportation and configuration process steps in the similar way. The frame should be transported to the second assembly table and the back side should be gathered from the smart shelf and transported to the same table. Hammering the back side into the frame should not start before the frame is transported and placed on the second assembly table and is being held. Also, holding the frame should not end until the hammering is finished, and the box is produced. The human worker then visually inspects the box for any deformations. Via a mobile device the human worker gets detailed instructions generated from the *description* attribute and checks whether the box passes the inspection. The decision must be made whether the box should be transported and discarded into the recycle bin or should be transported and stored into the finishing area. Any of these two cases will be done by the human worker.

The presented DL model is suitable for automatic execution. Code Generator will generate instructions from the DL model and Executor will send the instructions to smart resources and wait for their response. After the response arrives, Executor will send subsequent instructions until the production is finished. Code Generator generates generic instructions that are passed to Digital Twin in an appropriate protocol. Digital Twin receives and transforms messages into human-readable or machine-specific commands and passes them for execution. Digital Twin also updates the digital footprint of all resources it contains.

V. Conclusion and Future Work

In this paper we presented the DSML for modeling hardware production processes suitable for automatic execution. The goal of the language is to support the modeling of all production details required for automatic execution, but not to be too complex for a human to comprehend. To achieve this goal, two levels of abstraction are implemented so that production processes could be modeled in a generic way. By creating two levels of abstraction, production process models become independent from the production system details and thus efforts needed during the production process modeling are reduced. According to our experience from the industry, a process designer still needs to have the knowledge about the production system. Consequently, it is hard to make strict separation between production process models at PIM and PSM levels. However, we aim to achieve this separation by creating ML models and automatically generating DL models from them by using Orchestrator and the domain knowledge represented in a machine-readable way. Thus, the presented research leads one step closer to this goal. The language also allows process and quality engineers to collaborate on the creation of production process models. Created models could be used as a central artefact in a smart factory and thus lead the production automatically. Such language is implemented in a formal way and thus should increase consistency during modeling and decrease the amount of time needed for modeling. Integrating the language within the proposed MDSD approach should increase the production flexibility and contribute to the faster lot-size-one production.

One of the key future steps of our research will be to conduct the evaluation of the presented language. Using Modeling Tool, the language is tested by industrial process designers within an industrial use case [35], but we plan to systematically conduct the language evaluation that will include researchers and students from the academic community and process designers from the industry. During the initial MultiProLan validation, process designers were able to easily model the entire production process they needed and send the models to Orchestrator for execution. The evaluation should verify whether the language with multiple abstraction levels could make the modeling of production processes suitable for automatic execution easier comparing with other languages and approaches. Also, the evaluation should verify whether the language contributes to increasing the factory automation degree.

We will expand the language with concepts of quality assurance and error handling, as an occurrence of any failure requires error handling that needs to be carefully carried out and modeled [2]. Modeling production errors will cover all the basic attributes of FMEA documentation as the FMEA sheets will be automatically generated from process models. Also, an automatic generation of user manuals is needed. These documents contain a textual description of every process step and images on how to execute these steps. Currently, our Code Generator only generates human-readable or machine-readable instructions for the automatic process execution and should be extended with a feature to generate FMEA sheets, user manuals, BOMs and FPCs.

In addition to the error modeling, we plan to extend the language with: (i) subprocesses – to lower complexity of graphical process models, (ii) unordered process steps – as some activities could be executed in any order, e.g. in Fig. 6, the *pick left side* and the *pick bottom side* process steps should be unordered process steps and (iii) process variations – when the same result could be done by executing different process steps.

As the language is currently designed to model a hardware production, it could be extended to support the modeling of: (i) process production, e.g. breweries, sugar factories, pharma factories, (ii) software production and (iii) provision of service processes, e.g. banks, health care, education. Also, currently there is only the graphical syntax of the language. A textual syntax should also be implemented as some process designers could find it easier to use than the graphical syntax, or they could use the combination of these two syntaxes.

As a part of the future work, we also plan to further investigate the usability of MultiProLan. The emphasis will be on the collaboration between various participants and artefacts during the specification of a production process model.

REFERENCES

[1] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Bus. Inf. Syst. Eng.*, vol. 6, no. 4, pp. 239–242, Aug. 2014, doi: https://doi.org/10.1007/s12599-014-0334-4.

[2] K. Dorofeev, S. Profanter, J. Cabral, P. Ferreira, and A. Zoitl, "Agile Operational Behavior for the Control-Level Devices in Plug&Produce Production Environments," in *Proceedings of 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Zaragoza, Spain, 2019, pp. 49–56, doi: https://doi.org/10.1109/ETFA.2019.8869208.

[3] D. Gorecky, M. Schmitt, M. Loskyll, and D. Zuhlke, "Human-machine-interaction in the industry 4.0 era," in *Proceedings of 2014 12th IEEE International Conference on Industrial Informatics (INDIN)*, Porto Alegre RS, Brazil, Jul. 2014, pp. 289–294, doi: https://doi.org/10.1109/INDIN.2014.6945523.

[4] S. M. Fallah, S. Wolny, and M. Wimmer, "Towards model-integrated service-oriented manufacturing execution system," in *2016 1st International Workshop on Cyber-Physical Production Systems (CPPS)*, Vienna, Austria, Apr. 2016, pp. 1–5, doi: https://doi.org/10.1109/CPPS.2016.7483917.

[5] M. Vještica, V. Dimitrieski, M. Pisarić, S. Kordić, S. Ristić, and I. Luković, "Towards a formal description and automatic execution of production processes," in *Proceedings of 2019 IEEE 15th International Scientific Conference on Informatics*, Poprad, Slovakia, Nov. 2019, pp. 463–468, doi: https://doi.org/10.1109/Informatics47936.2019.9119314.

[6] M. Vještica, V. Dimitrieski, M. Pisarić, S. Kordić, S. Ristić, and I. Luković, "Towards a Formal Specification of Production Processes Suitable for Automatic Execution," *Open Comput. Sci.*, p. 20, May 2020, to be published.

[7] M. Pisarić, V. Dimitrieski, M. Vještica, and G. Krajoski, "Towards a Non-Disruptive System for Dynamic Orchestration of the Shop Floor," in *IFIP Advances in Information and Communication Technology (AICT)*, Novi Sad, Serbia, 2020, vol. 592, pp. 1–8, doi: https://doi.org/10.1007/978-3-030-57997-5_54.

[8] V. Dimitrieski, "Model-Driven Technical Space Integration Based on a Mapping Approach," Ph.D. Thesis, University of Novi Sad, Faculty of Technical Sciences, Serbia, 2017.

[9] M. Mernik, J. Heering, and A. M. Sloane, "When and how to develop domain-specific languages," *ACM Comput. Surv.*, vol. 37, no. 4, pp. 316–344, Dec. 2005, doi: https://doi.org/10.1145/1118890.1118892.

[10] Q. Qi and F. Tao, "Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: 360 Degree Comparison," *IEEE Access*, vol. 6, pp. 3585–3593, 2018, doi: https://doi.org/10.1109/ACCESS.2018.2793265.

[11] C. Leyh, S. Martin, and T. Schäffer, "Industry 4.0 and Lean Production – A Matching Relationship? An analysis of selected Industry 4.0 models," in *Proceedings of 2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, Sep. 2017, vol. 11, pp. 989–993, doi: https://doi.org/10.15439/2017F365.

[12] T. Qu, S. P. Lei, Z. Z. Wang, D. X. Nie, X. Chen, and G. Q. Huang, "IoT-based real-time production logistics synchronization system under smart cloud manufacturing," *Int. J. Adv. Manuf. Technol.*, vol. 84, no. 1–4, pp. 147–164, Apr. 2016, doi: https://doi.org/10.1007/s00170-015-7220-1.

[13] S. Vaidya, P. Ambad, and S. Bhosle, "Industry 4.0 – A Glimpse," in *Procedia Manufacturing*, Maharashtra, India, 2018, vol. 20, pp. 233–238, doi: https://doi.org/10.1016/j.promfg.2018.02.034.

[14] J. Wan, H. Cai, and K. Zhou, "Industrie 4.0: Enabling Technologies," in *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things*, Harbin, 2015, pp. 135–140, doi: https://doi.org/10.1109/ICAIOT.2015.7111555.

[15] L. D. Xu, E. L. Xu, and L. Li, "Industry 4.0: state of the art and future trends," *Int. J. Prod. Res.*, vol. 56, no. 8, pp. 2941–2962, Apr. 2018, doi: https://doi.org/10.1080/00207543.2018.1444806.

[16] L. D. Xu, "Enterprise Systems: State-of-the-Art and Future Trends," *IEEE Trans. Ind. Inform.*, vol. 7, no. 4, pp. 630–640, Nov. 2011, doi: https://doi.org/10.1109/TII.2011.2167156.

[17] H. Ahn and T.-W. Chang, "Measuring Similarity for Manufacturing Process Models," in *IFIP Advances in Information and Communication Technology (AICT)*, Cham, Aug. 2018, vol. 536, pp. 223–231, doi: https://doi.org/10.1007/978-3-319-99707-0_28.

[18] J. Jiao, M. M. Tseng, Q. Ma, and Y. Zou, "Generic Bill-of-Materials-and-Operations for High-Variety Production Management," *Concurr. Eng.*, vol. 8, no. 4, pp. 297–321, Dec. 2000, doi: https://doi.org/10.1177/1063293X0000800404.

[19] Korean Standards Service Network (KSSN), "KS A 3002 Standard." https://www.kssn.net/en/ (accessed Apr. 05, 2020).

[20] M. Witsch and B. Vogel-Heuser, "Towards a Formal Specification Framework for Manufacturing Execution Systems," *IEEE Trans. Ind. Inform.*, vol. 8, no. 2, pp. 311–320, May 2012, doi: https://doi.org/10.1109/TII.2012.2186585.

[21] A. Wortmann, O. Barais, B. Combemale, and M. Wimmer, "Modeling Languages in Industry 4.0: An Extended Systematic Mapping Study," *Softw. Syst. Model.*, vol. 19, pp. 67–94, Jan. 2020, doi: https://doi.org/10.1007/s10270-019-00757-6.

[22] S. Zor, D. Schumm, and F. Leymann, "A Proposal of BPMN Extensions for the Manufacturing Domain," in *Proceedings of the 44th CIRP International Conference on Manufacturing Systems*, Madison, Wisconsin, USA, 2011, pp. 1–7.

[23] M. Lütjen and D. Rippel, "GRAMOSA framework for graphical modelling and simulation-based analysis of complex production processes," *Int. J. Adv. Manuf. Technol.*, vol. 81, no. 1–4, pp. 171–181, May 2015, doi: https://doi.org/10.1007/s00170-015-7037-y.

[24] S. Meyer, A. Ruppen, and L. Hilty, "The Things of the Internet of Things in BPMN," in *Advanced Information Systems Engineering Workshops. CAiSE 2015. Lecture Notes in Business Information Processing*, Stockholm, Sweden, 2015, vol. 215, pp. 285–297, doi: https://doi.org/10.1007/978-3-319-19243-7_27.

[25] R. Petrasch and R. Hentschke, "Towards an Internet-of-Things-aware Process Modeling Method - An Example for a House Suveillance System Process Model," in *Proceedings of 2nd Management and Innovation Technology International Conference (MITiCON2015)*, Bangkok, Thailand, 2015, pp. 168–172.

[26] R. Petrasch and R. Hentschke, "Process modeling for industry 4.0 applications: Towards an industry 4.0 process modeling language and method," in *Proceedings of 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, Khon Kaen, Thailand, Jul. 2016, pp. 1–5, doi: https://doi.org/10.1109/JCSSE.2016.7748885.

[27] S. Schönig, L. Ackermann, S. Jablonski, and A. Ermer, "IoT meets BPM: a bidirectional communication architecture for IoT-aware process execution," *Softw. Syst. Model.*, Mar. 2020, doi: https://doi.org/10.1007/s10270-020-00785-7.

[28] B. Weissenberger, S. Flad, X. Chen, S. Rosch, T. Voigt, and B. Vogel-Heuser, "Model driven engineering of manufacturing execution systems using a formal specification," in *Proceedings of 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA)*, Luxembourg, Sep. 2015, pp. 1–8, doi: https://doi.org/10.1109/ETFA.2015.7301430.

[29] D. Steinberg, F. Budinsky, M. Paternostro, and E. Merks, *EMF: Eclipse Modeling Framework*, 2nd ed. Upper Saddle River, NJ, USA: Addison-Wesley Professional, 2008.

[30] "Eclipse Sirius Documentation." https://www.eclipse.org/sirius/doc/ (accessed Mar. 19, 2020).

[31] I. Dejanovic, M. Tumbas, G. Milosavljevic, and B. Perisic, "Comparison of Textual and Visual Notations of DOMMLite Domain-Specific Language," in *Local Proceedings of the Fourteenth East-European Conference on Advances in Databases and Information Systems*, Novi Sad, Serbia, Sep. 2010, pp. 131–136.

[32] Object Management Group, "Business Process Model and Notation, Version 2.0.2," Technical Report, 2014.

[33] M. Kocbek, G. Jost, M. Hericko, and G. Polancic, "Business process model and notation: The current state of affairs," *Comput. Sci. Inf. Syst.*, vol. 12, no. 2, pp. 509–539, 2015, doi: https://doi.org/10.2298/CSIS140610006K.

[34] American Society of Mechanical Engineers. Special committee on standardization of therbligs, process charts, and their symbols, *A.S.M.E. standard operation and flow process charts*. New York, N.Y., The American society of mechanical engineers, 1947.

[35] M. Vještica, V. Dimitrieski, M. Pisarić, S. Kordić, S. Ristić, and I. Luković, "An Application of a DSML in Industry 4.0 Production Processes," in *IFIP Advances in Information and Communication Technology (AICT)*, Novi Sad, Serbia, 2020, vol. 591, pp. 1–8, doi: https://doi.org/10.1007/978-3-030-57993-7_50.

Author Index