On the Performance Prediction Capabilities of the eBPMN-based Model-driven Method for Business Process Simulation

Paolo Bocciarelli
Andrea D'Ambrogio
Emiliano Paglia

Department of Enterprise Engineering
University of Rome Tor Vergata
Via del Politecnico 1
00133, Rome, Italy
{paolo.bocciarelli, dambro, emiliano.paglia}@uniroma2.it

Andrea Giglio

Guglielmo Marconi University
Dept. of Innovation and Information Engineering
Via Plinio 44
00193, Rome, Italy
a.giglio@unimarconi.it

***Abstract*—The analysis of business processes may take considerable advantage by the adoption of simulation-based techniques from the early phases of process lifecycle. Despite the fact that Modeling & Simulation (M&S) approaches have been widely recognized as a valuable solution, the remarkable effort required for their implementation and their essential complexity may limit their use in practice. In previous contributions we have proposed a model-driven method and a domain specific language, named eBPMN, for enabling the automated development and execution of business process simulations. In this paper, we analyze the performance prediction capabilities of the current implementation of the eBPMN-based model-driven method, by means of a comparison with the same capabilities provided by similar existing tools, such as BIMP and Bizagi, for a reference business process.**

***Index Terms*—Business Process Model and Notation (BPMN), Business Process Management, Modeling and Simulation, Business Process Analysis**

I. INTRODUCTION

Modern organizations have to face a constantly evolving market, characterized by rapidly changing demands and increasing levels of competition. In such a context, it becomes crucial for organizations to get a deep knowledge of their operational business model and thus have the ability to dynamically improve it, in order to gain a competitive advantage and better exploit market opportunities. Specifically, a deep understanding of the operational business processes is essential to enhance organization efficiency and improve the quality of delivered products and/or provided services.

As a consequence, the vast majority of organizations have embraced Business Process Management (BPM) approaches, which include methods, techniques, and tools to support the design, analysis, enactment and improvement of operational business processes [42]. A *business process (BP)* can be defined as a set of interrelated activities that are executed by one or more organizations working together to achieve a common business purpose [31], [25].

From an operational perspective, several modeling languages have been introduced in the last years to specify business processes. Among them, the *Business Process Model and Notation (BPMN)* [34], provided by OMG, is playing a primary role as the *de facto* standard in the BPM domain. BPMN is widely adopted by users with different backgrounds and roles, ranging from business analysts to IT experts [32].

Since the complexity of modern BPs is constantly increasing, it becomes essential to use innovative methodologies and techniques to effectively support BP analysis activities right from the first phases of the BP lifecycle.

In this respect, the use of *Modeling and Simulation* (*M&S*) techniques has been identified as an effective BP analysis approach [43], providing a valid support at both design time and operations time.

Although M&S approaches are widely recognized as extremely valuable [20], their practical use is not yet widespread as one would expect, due to two main factors. On the one hand, the cost to set up and to maintain a M&S environment is often not-negligible. On the other hand, the use of M&S-based techniques requires skills and know-how which most business analysts are not familiar with.

In previous contributions, we have introduced a framework that largely reduces the effort and the cost for carrying out simulation-based BP analysis activity. The proposed framework exploits standards and technologies introduced in the model-driven development field to ease and automate the simulation-based BP analysis [6]. The framework introduces a domain-specific language, named eBPMN, for the specification and execution of BP simulation models.

eBPMN allows one to carry out BP analysis activities by addressing both performance (i.e., efficiency) and reliability (i.e., failure-free behavior) properties. This paper specifically focuses on the performance prediction capabilities of the current implementation of eBPMN. The paper discusses the effectiveness and validity of such predictions by means of a comparison between the key performance indicators and the results provided by eBPMN and those provided by other well-know and widely used tools, such as BIMP and Bizagi, for a reference business process.

The comparison allows us to conclude that eBPMN enables business analysts to get the same performance predictions of commonly available BP simulation tools, with the additional advantages of:

- not requiring the manual specification and/or implementation of the simulation model, which can be automatically generated from BP models specified by use of BPMN or other BP modeling languages;
- providing both sequential and distributed simulation execution engines, which can be easily integrated into MSaaS (M&S as a Service) platforms.

The rest of this paper is organized as follows: Section II provides the necessary background and discusses related works. Section III outiles the aforementioned features of eBPMN and illustrates the comparison approach, while Section IV introduces the case study that has been used as a reference scenario for BP performance analysis. Section V discusses the related results and, finally, Section VI gives final remarks.

II. BACKGROUND

The reasons for adopting simulation techniques in the BPM domain have been investigated in [38], [37]. Typically, simulation techniques can answer questions such as [23], [26]: what are the total process time and the maximum throughput of the process, what are the equipment and technology requirements to meet the service demand, what are the waiting times, what is the best assignment of resources to task and how to deal with unexpected situations or emergencies.

However, the potential of M&S approaches is not fully exploited yet, mainly due to the fact that the several existing simulation tools require conceptual and technical skills beyond those generally available in business organizations [2]. In this regard, a survey conducted among potential BP simulation users reveals that almost 80% of respondents do not use simulation techniques, while actual users report as main motivations the ability of simulation-based analysis to support extensive experimentation and get a deeper understanding of complex process interactions [33].

A systematic review of business process simulation tools can be found in [29], in which the authors evaluate their applicability, underline some limitations and derive recommendations for further research. The review analyzes business process modeling tools that may be applicable for simulation (e.g., Protos, ARIS), tools that expose simulation capabilities (e.g., FLOWer, FileNet) and general purpose simulation tools (e.g., Arena, Colored Petri-Net tools). The evaluation is based on modeling capabilities (ease of model building, formal semantics and verification of correctness, workflow patterns, resource and data perspective, level of detail, transparency and suitability for communication), simulation capabilities (performance dimensions, distributions, animation, scenarios) and output analysis capabilities (statistics, format, what-if analysis, decision-making support).

An updated review on the role of simulation techniques can be found in [28], in which authors report that discrete-event simulation is by far the most investigated technique (about 40% of the literature contributions), followed by system dynamics (about 15%), hybrid techniques and Monte-Carlo simulation, among the others.

An additional comparison of BP simulation tools is described in [27], in which authors classify the reviewed tools (e.g., JBoss jBPM simulation tool, Oracle Business Process Analysis Suite, SigmaFlow modeler, Arena simulation modeling tool) in terms of licensing options and addressed application domains.

Beyond the various features and capabilities of the several BP simulation tools reviewed in the aforementioned contributions, what is considered as an actual impediment to the effective use of M&S-based approaches in the BPM domain is the semantic gap that often exists between the BP-based conceptual model and the BP simulation model.

From a practical perspective, in [16] the authors observe that a main focus of research is to provide simulation building blocks that are as close as possible to the elements of the conceptual model. The authors point out that a slight difference between the conceptual model and the simulation language can lead to difficulties in translating the model and, in some cases, to the impossibility to carry out such translation. In this respect, in [14] the author argues that most of the currently available simulation tools provide hard-coded simulation capabilities, failing to explicitly define the underlying simulation modeling formalism by use of, e.g., a well defined metamodel.

Attempts to bridge the gap between conceptual and simulation models are documented in [40], [30], [14], [39]. Such attempts are based either on the provision of specific simulation languages, such as the event-driven Simulation Activity Diagram (SAD) language introduced in [40], or on more flexible and effective approaches based on model-driven development and automated model transformations, such as in [30], which proposes a model transformation approach that can be used to translate conceptual process models, expressed in the event-driven process chain (EPC) notation, to different simulation software systems, and in [14], which presents a framework that exploits metamodeling and model transformations in order to translate conceptual elements into executable simulation components. Similarly, in [39] a model-driven approach is used to transform BPMN models into DEVS models and then to Java classes used for the simulation with the DSOL (Distributed Simulation Object Library) library.

To summarize, the analysis of the literature reveals a considerably wide offer of M&S tools for BP simulation, but still a limited use of the potential offered by M&S approaches for BP analysis. Further, several existing simulation tools require specific know-how in M&S, as well as software engineering skills, far beyond those generally available in business organizations. As such, model-driven development can be considered one of the most effective approaches to provide the required degree of automation, which helps to reduce the effort and cost of M&S-based BP analysis.

In this respect, we have proposed a full featured model-driven framework that introduces a set of automated model transformations to translate BP models into corresponding BP simulation models and simulation model implementations ready to be executed on top of sequential or distributed simulation engines [10], [9]. The framework includes a domain-specific language, named eBPMN, to specify and execute BP simulation models automatically generated from BP models specified in a given BP modeling language, such as BPMN or UML.

III. THE eBPMN DOMAIN-SPECIFIC LANGUAGE FOR BP SIMULATION

A. eBPMN Overview

eBPMN is a domain-specific simulation language based on the execution semantics defined in the BPMN 2.0 specification [34]. It has been originally introduced in [12] and further extended in [9], [10], [15], [1].

The eBPMN language has been built on top of *SimArch* [21], a layered software architecture which gives users the ability to specify event-driven simulation models that can be transparently executed either in local or distributed simulation environments. A detailed description of *SimArch* is given in [22].

eBPMN allows users to simulate BPs consisting of a single participant as well as complex process collaborations. A BP element (task or activity) makes use of a single resource or more resources to perform its job. In order to specify the resources behavior and the non-functional properties of BPMN elements, eBPMN exploits a lightweight BPMN extension named PyBPMN (Performability-enabled BPMN), briefly summarized in next sub-section and detailed in [7].

The eBPMN language implements a token abstraction to simulate the execution of a BPMN process. A token is generated by a *Start* node and can be considered as a reference to the execution of a process instance. The time interval between two subsequent token generation events follows a given probability distribution. After creation, the token goes throughout the process nodes, guided by the sequence flows, and each eBPMN element handles the token according to its execution semantics. At the *End* node the token is destroyed and the simulation environment gathers information about the process traversal.

The eBPMN simulation language provides the following performance metrics for process elements and collaborations:

- *service time* (mean and variance) and *waiting time* (mean and variance) for resources and tasks;
- *resource utilization*, i.e., the time spent by a resource in executing service requests, considering the amount of parallel working units and thus evaluating bottleneck issues;
- number of tokens processed by each element and by each resource as a measure of the *throughput*;
- tokens managed on each branch of gateways as a measure of the *usage* of each possible business process flow;
- *cycle time* (mean and variance) as the time spent by a token to complete the collaboration from token generation at start node to token termination at end node.

Although eBPMN allows to simulate both the performance and the reliability behavior of the business process [6], in this paper we specifically focus on performance properties. This is due to the fact that the proposed approach for performance prediction analysis is based on comparing the eBPMN results with those obtained with other existing BPMN simulators which do not provide process reliability simulation.

B. eBPMN-based Model-Driven Method

One of the main advantages of eBPMN is that the executable code can be directly obtained in an automated fashion from standard BPMN models by use of a model-driven method based on automated model transformations [10], [9].

As aforementioned, such transformations rely on a lightweight extension of the BPMN metamodel, named PyBPMN, that has been defined in previous works [7] as a semantics preserving extension that does not alter the original BPMN metamodel. PyBPMN gives the ability to annotate standard BPMN models with performance-oriented BP properties. The annotated BPMN model is then directly mapped to eBPMN executable code.

The PyBPMN metamodel introduces specific metaclasses for the following components:

- *workload definition*: responsible for modeling the workload related to the whole business process or to the tasks associated to the process (i.e., the execution of single activities);
- *performance properties definition*: responsible for specifying the performance properties associated to both the process and the single task. The most common performance properties are the service demand (service time), the time spent to accomplish the demand (response time), and the throughput;
- *reliability properties definition*: responsible for modeling the reliability related properties of the resources involved in a process or associated with a task. The most common reliability properties are the occurrence rate of the failure, the occurrence distribution of the failure, the mean time to failure (MTTF) and the mean time to repair (MTTR);
- *resource management*: responsible for specifying the actual resource which is used to execute an activity. PyBPMN allows to define non functional properties for atomic resources, as well as groups of resources consisting of concurrent or alternative resources.

The eBPMN-based model-driven method has been specified using languages and tools introduced by the OMG's MDA (Model Driven Architecture) incarnation of model-driven development principles, and implemented within the

Eclipse Modeling Framework (EMF) platform [18], [41] and the Eclipse Modeling Project [19], [24]. The following Eclipse plugins have been implemented in order to provide a tool-chain that eases the production of the eBPMN-based simulation code from a standard BPMN model:

- a plugin implementing the *PyBPMN metamodel*;
- a plugin implementing the *BPMN to PyBPMN* model-to-model transformation;
- a plugin implementing the *PyBPMN to eBPMN* model-to-text transformation.

The tool-chain is being ported and deployed into cloud-based environments, according to the MSaaS paradigm. The proposed MSaaS platforms includes the aforementioned model transformation services, a *SimArch*-based simulation execution engine, which allows users to execute eBPMN simulations on top of either sequential or distributed simulation environments, and modeling services to directly specify eBPMN simulation models, as described in [11], [8], [5].

C. eBPMN Performance Prediction

In order to analyze the eBPMN performance prediction capabilities, an indirect approach has been adopted. The analysis is carried out by comparing the results provided by eBPMN with the outputs provided, for the same process and input parameters, by two well-known tools that offer similar features, i.e., *Bizagi Process Modeler* [4] and *BIMP* [3].

The Bizagi Process Modeler (Bizagi in short) is a business process modeling and documentation tool, compliant with the BPMN 2.0 specification, that allows to visualize, model and document business processes using BPMN. The Bizagi Process Modeler also provides simulation parameterization capabilities that conform to the BPSim specification [44].

BIMP is a simulator of business process models, free for academic use, available either through a web-based interface or according to a simulation as a service paradigm.

The comparison is carried out with respect to some specific performance measures, or Key Performance Indicators (KPIs), which are able to capture the essential aspects of the business process performance behavior.

In this respect, a significant effort has been spent in order to deal with the definition of appropriate KPIs for BPM [13], [35], [36]. A typical performance metric of interest for business processes is the cycle time, i.e., the time taken to handle one token (one process instance), addressed in terms of maximum value, average value or its variation over instances. Further, cycle time can be analyzed through its constituent measures such as the service time (the time spent to actually handling the token) or the waiting time (the time spent in idle mode, either in queue or waiting for synchronization) [17].

For the comparison of results provided by eBPMN, BIMP and Bizagi, the following KPIs have been taken into consideration:

- process cycle time;
- resource utilization;
- activity waiting time.

IV. CASE STUDY FOR PERFORMANCE VALIDATION

The process considered for eBPMN validation is a health care process, specifically a process dealing with diabetes care. Diabetes of type 1 must be treated with medicines throughout life. Affected patients should take insulin to avoid excessive glycemic peaks. Pharmacological treatment aims at controlling the symptoms of diabetes and preventing serious complications.

The case study refers to the request of essential products for diabetes care. The start event consists of the expiration of the periodic treatment of a patient, which results in the need to contact the salesman to acquire new medicines. The salesman, following the patient necessary quantities, prepares a new medicines plan, which has then to be approved by the reference doctor (i.e., the diabetologist). After the plan approval, the salesman forwards the order to the pharmaceutical industry that packages the medicines. The pharmaceutical industry then sends the medicines to the main hospital, which locates the authorized medical center closest to the patient facility and forwards the medicines to it. Finally, the medical center notifies the patient of the availability of the requested medicines and the process ends. Both the medicines plan and the order can be subjected to refusal (with a 15% probability), which leads to rework of the request.

The BPMN model of the reference process is depicted in Figure 1.

Each pool in the BPMN model defines a resource role involved in the process. Each role is associated with one or more actual resources (performers) which are able to execute the activity. Table I summarizes the resources roles and the corresponding amount of performers considered for the reference process.

TABLE I
RESOURCES (WITH QUANTITIES) AVAILABLE FOR THE REFERENCE PROCESS.

Resource	Quantity
Salesman	1
Doctor	1
Pharmaceutical Industry	3
Hospital	2
Medical Center	1

For each activity in the process, Table II reports both the average service time and the resource performing the activity.

V. RESULTS

The performance validation has been carried out simulating the reference process with respect to two different scenarios A and B, which differ for the expected workload in terms of patients requests average inter-arrival time: 4 hours for scenario A and 3 hours for scenario B. Thus, scenario B deals with a more intensive workload.

The simulation has been set-up considering exponential probability distributions for inter-arrival and activity times.

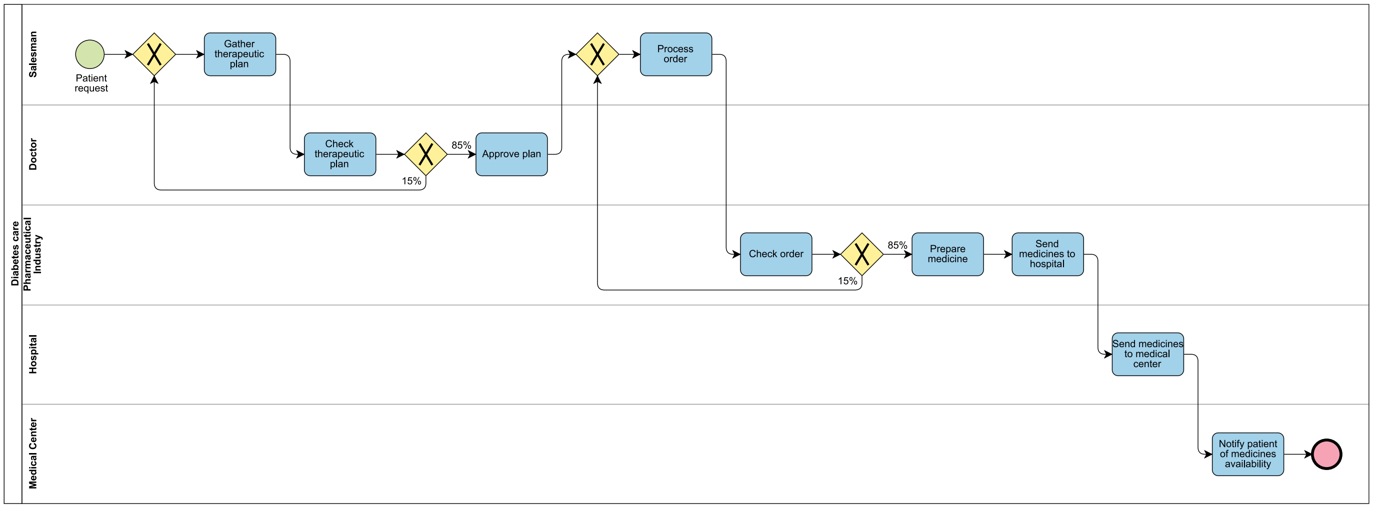

Fig. 1. Reference BP model for performance prediction capabilities analysis.

TABLE II
AVERAGE ACTIVITY EXECUTION TIMES AND RESOURCES IN CHARGE.

Activity	Avg. time [min]	Resource
Gather therapeutic plan	15	Salesman
Check therapeutic plan	45	Doctor
Approve plan	20	Doctor
Process order	60	Salesman
Check order	5	Pharmaceutical Industry
Prepare medicine	360	Pharmaceutical Industry
Send medicines to hospital	60	Pharmaceutical Industry
Send medicines to medical center	180	Hospital
Notify patient of medicines availability	60	Medical Center

As for the simulation duration, both eBPMN and Bizagi allow to stop the simulation after a given simulated time, while BIMP simulations are stopped when all the tokens generated by start nodes are terminated at the end nodes. The use of probability distributions for inter-arrival time, activities duration and gateways routing policies introduces some uncertainty on precision of the provided results. Indeed, if the termination condition is the maximum simulated time, the exact number of processed tokens is not exactly predictable; on the other end, if the termination condition is the number of tokens, the actual simulated time is not exactly predictable.

However, if the simulated time is long enough, all the results tend towards the steady-state conditions of the process. This allows to compare the results of different tools even if they have simulated a different amount of process instances.

For the purpose of this paper, the simulation stops after 365 simulated days for eBPMN and Bizagi (about 2000 tokens processed per run) and after 3000 tokens for BIMP (to meet tool constraints on simulation duration). Further, eBPMN and Bizagi automatically execute 50 runs of the process to reduce statistical error. BIMP does not allow to specify the number of runs so the simulation has been manually re-run to evaluate average values.

The aforementioned quantities have been chosen as a reasonable trade-off between minimizing the statistical error and limiting the simulation complexity (eBPMN and Bizagi provide results in about 2 minutes on a typical desktop PC with 4th-generation Intel i5 CPU and 8 GByte of RAM).

Moreover, it should be noted that eBPMN and Bizagi provide repeatable deterministic simulations while BIMP does not, so repeating the BIMP simulation could lead to different values.

The KPIs provided by eBPMN, BIMP and Bizagi for scenario A are depicted in Figure 2, while Figure 3 presents the same KPIs for scenario B. Corresponding figures use the same scale to make easier the comparison of the KPI on workload variation.

The simulation of the two scenarios shows comparable performance predictions of process times, resources utilization and waiting times for eBPMN, BIMP and Bizagi. Differences in some values are likely due to pseudo-random number generation for probability distribution approximation.

As expected, increasing the workload (i.e., switching from scenario A to scenario B) leads to higher resource utilization. The pharmaceutical industry has the highest utilization (from 60% for scenario A to 80% for scenario B). Under these conditions, the resource is working near its maximum capacity and the waiting times for activities performed by such

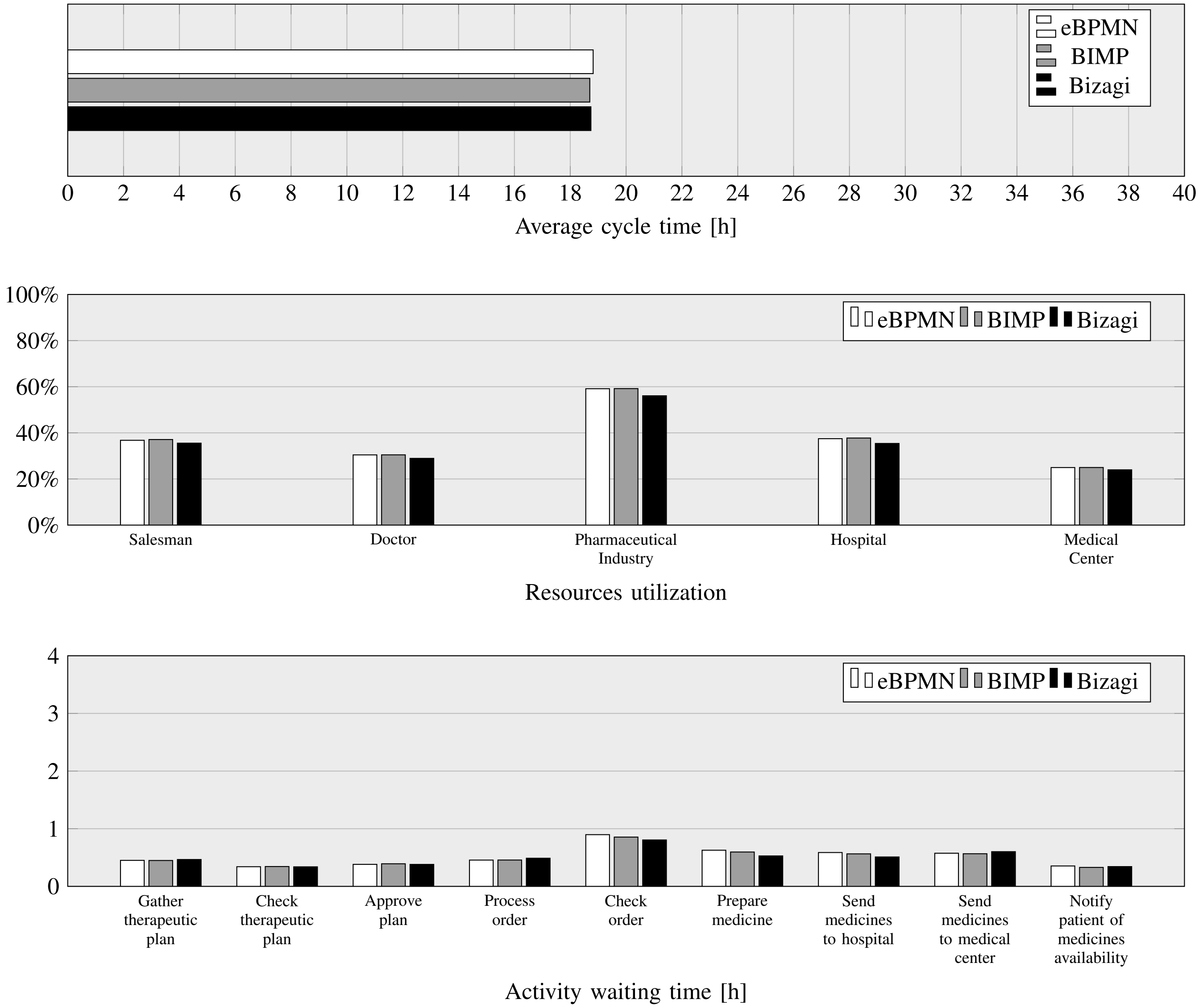

Fig. 2. Comparison of KPIs for reference process considering patient requests inter-arrival time of 4 hours.

resource quickly go up. This behavior is captured, with minor differences, by all the considered tools (for the *Check order activity* eBPMN shows higher waiting times than the other two tools, while for *Prepare medicines* and *Send medicines to hospital* Bizagi predicts lower waiting times than eBPMN and BIMP).

VI. CONCLUSIONS

The paper presents an analysis of the performance prediction capabilities for the current implementation of eBPMN, a domain-specific simulation language for BP simulation. The main advantages of eBPMN is its formal specification, compliant with the execution semantic of BPMN, and the ability to automatically generate the eBPMN executable code applying a model-driven tool-chain to the BPMN model.

The analysis of the BP performance prediction has been carried out by comparing the results of eBPMN with those provided by similar tools, i.e., BIMP and Bizagi Process Modeler, for a reference BP. The validation procedure has been applied to two scenarios for the reference process with different workload characterization. The analysis of the results reveals comparable performance predictions for eBPMN, BIMP and Bizagi for cycle time, resource utilization and waiting times.

The comparison has showed that eBPMN enables business analysts to get the same performance predictions of similar BP simulation tools, with the additional advantages of not requiring the manual specification and/or implementation of the simulation model, which can be automatically generated from BP models specified by use of BPMN or other BP modeling languages, and using a sequential/distributed simulation execution engine, which can be easily integrated into MSaaS (M&S as a Service) platforms.

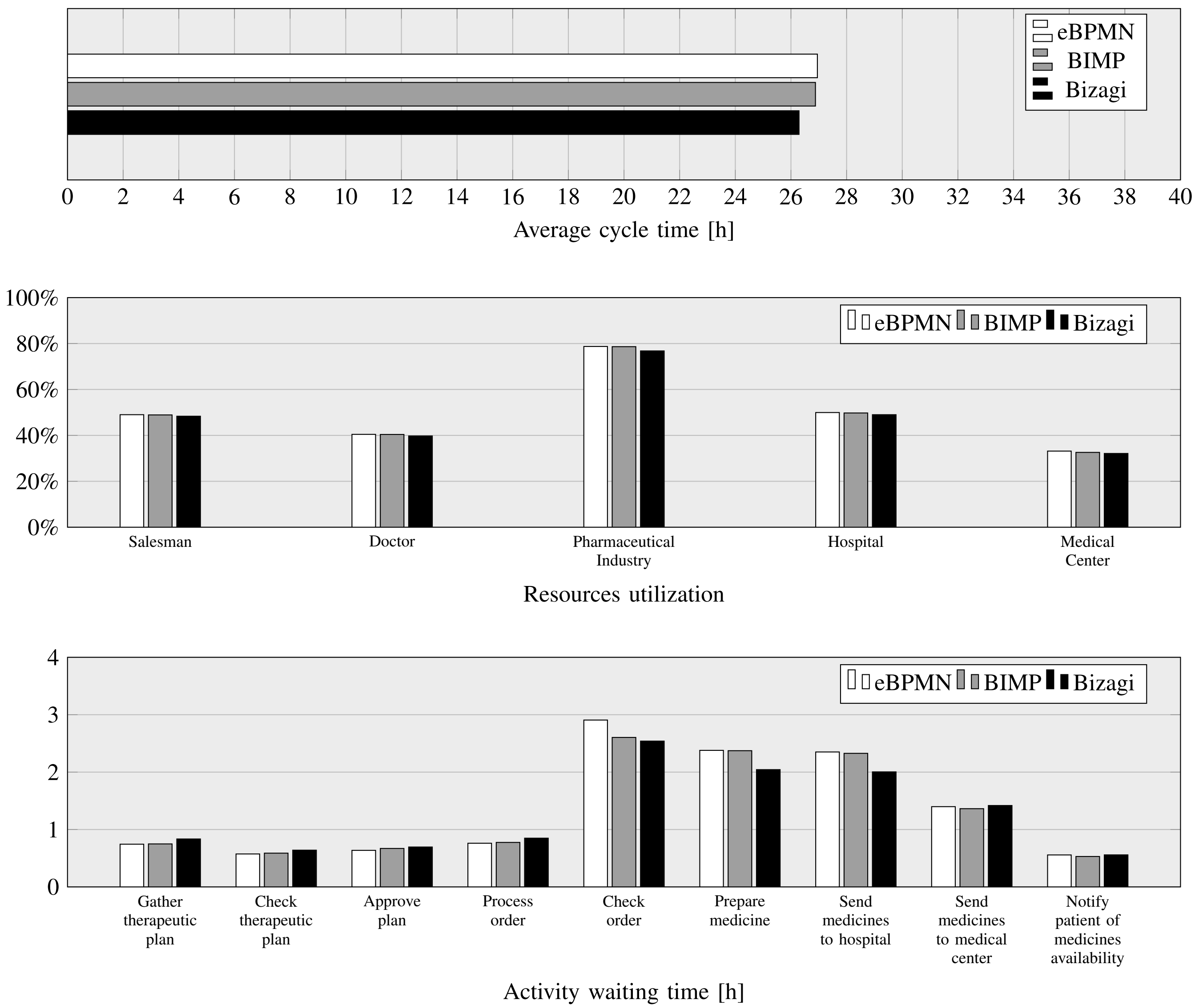

Fig. 3. Comparison of KPIs for reference process considering patient requests inter-arrival time of 3 hours.

REFERENCES

[1] G. Antonacci, A. Calabrese, Andrea D'Ambrogio, A. Giglio, B. Intrigila, and N.L. Ghiron. A bpmn-based automated approach for the analysis of healthcare processes. In Gaaloul W. Reddy S.M., editor, *Proceedings - 25th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE 2016*, pages 124–129. Institute of Electrical and Electronics Engineers Inc., 2016.

[2] K. D. Barber, F. W. Dewhurst, R. L. D. H. Burns, and J. B. B. Rogers. Business-process modelling and simulation for manufacturing management: A practical way forward. *Business Process Management Journal*, 9(4):527–542, 2003.

[3] BIMP. The Business Process Simulator, oct 2016.

[4] Bizagi. Bizagi Process Modeler, oct 2016.

[5] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, and Antonio Mastromattei. Automated Development of Web-based Modeling Services for MSaaS Platforms. In *International Workshop on Model-Driven Approaches for Simulation Engineering (Mod4Sim 2017), part of the 2017 Spring Simulation Multiconference (SpringSim 2017)*, Virginia Beach, VA, USA, 2017.

[6] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, and Emiliano Paglia. Simulation-Based Performance and Reliability Analysis of Business Processes. In *Proceedings of the 2014 Winter Simulation Conference*, pages 3012–3023, Savannah, Georgia, 2014. IEEE Press.

[7] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, and Emiliano Paglia. A BPMN Extension to Enable the Explicit Modeling of Task Resources. In *Proceedings of the INCOSE Italy Conference on Systems Engineering*, Turin, Italy, 2016.

[8] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, and Emiliano Paglia. Model Transformation Services for MSAAS Platforms. In *Proceedings of the 2018 Spring Simulation Conference (SpringSim'18)*, Baltimore, MD, USA, 2018.

[9] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, Emiliano Paglia, and Daniele Gianni. A Transformation Approach to Enact the Design-Time Simulation of BPMN Models. In *2014 IEEE 23rd International WETICE Conference*, pages 199–204. IEEE, 2014.

[10] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, Emiliano Paglia, and Daniele Gianni. Empowering Business Process Simulation Through Automated Model Transformations. In *Proceedings of the Symposium on Theory of Modeling & Simulation - DEVS Integrative*, DEVS '14, pages 39:1–39:9, Tampa, Florida, 2014.

[11] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, Emiliano Paglia, and Tommaso Panetti. A Cloud-based Service-oriented Architecture for Business Process Modeling and Simulation. In *Proceedings of the 3rd*

INCOSE Italy Conference on Systems Engineering, pages 87–93, Naples, Italy, 2017.

[12] Paolo Bocciarelli, Andrea D'Ambrogio, and Emiliano Paglia. A Language for Enabling Model-Driven Analysis of Business Processes. In *Proceedings of the 2nd International Conference on Model-Driven Engineering and Software Development*, MODELSWARD'14, Lisbon, Portugal, 2014. SciTePress.

[13] Evellin Cristine Souza Cardoso. Challenges in Performance Analysis in Enterprise Architectures. *17th IEEE International Enterprise Distributed Object Computing Conference Workshops*, pages 327–336, 2013.

[14] Deniz Cetinkaya, Alexander Verbraeck, and Mamadou Seck. Applying a Model Driven Approach To Component Based Modeling and Simulation. In *Proceedings of the 2010 Winter Simulation Conference*, pages 546–553, 2010.

[15] Andrea D'Ambrogio, Emiliano Paglia, Paolo Bocciarelli, and Andrea Giglio. Towards performance-oriented perfective evolution of BPMN models. In *6th International Workshop on Model-Driven Approaches for Simulation Engineering*, Pasadena, CA, USA., 2016.

[16] Gert Jan De Vreede, Alexander Verbraeck, and Daniel T. T. Van Eijck. Integrating the conceptualization and simulation of business processes: A modeling method and arena template. *Simulation*, 79(1):43–55, 2003.

[17] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Fundamentals of Business Process Management*. Springer-Verlag, 2013.

[18] Eclipse Foundation. Eclipse Modeling Framework (EMF), sep 2016.

[19] Eclipse Foundation. Eclipse Modeling Project, sep 2016.

[20] A Falcone, A Garro, and A Tundis. Modeling and simulation for the performance evaluation of the on-board communication system of a metro train. In *13th International Conference on Modeling and Applied Simulation, MAS 2014*, pages 20–29, 2014.

[21] Daniele Gianni, Andrea D'Ambrogio, and Giuseppe Iazeolla. A layered architecture for the model-driven development of distributed simulators. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, page 61. ICST, 2008.

[22] Daniele Gianni, Andrea D'Ambrogio, and Giuseppe Iazeolla. A software architecture to ease the development of distributed simulation systems. *Simulation*, 87(9):819–836, 2011.

[23] Bruce Gladwin and Kerim Tumay. Modeling Business Processes with Simulation Tools. In *Proceedings of the 26th Conference on Winter Simulation*, pages 114–121, 1994.

[24] Richard Gronback. *Eclipse Modeling Project: a Domain-Specific Language Toolkit*. 2009.

[25] Michael Hammer and James Champy. Reengineering the corporation: A manifesto for business revolution. *Business Horizons*, 36(5):90–91, 1993.

[26] Geoffrey Hook. Business Process Modeling and Simulation. In *Proceedings of the 2011 Winter Simulation Conference*, pages 773–778, 2011.

[27] Ibrahim Imam, Nihal Nounou, Alaa Hamouda, and Hebat Allah Abdul Khalek. Survey of business process simulation tools: A comparative approach. *Proceedings of SPIE - The International Society for Optical Engineering*, 8285(July):0–7, 2011.

[28] Mohsen Jahangirian, Tillal Eldabi, Aisha Naseer, Lampros K. Stergioulas, and Terry Young. Simulation in manufacturing and business: A review. *European Journal of Operational Research*, 203(1):1–13, 2010.

[29] Monique H. Jansen-Vullers and Mariska Netjes. Business Process Simulation - A Tool Survey. *Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN*, 834:77–96, 2006.

[30] Oliver Kloos, Volker Nissen, and Mathias Petsch. From Process to Simulation - A Transformation Model Approach. In *Proceedings of the 3rd International Workshop on Enterprise Modelling and Information Systems Architectures*, pages 83–96, 2009.

[31] Ryan K L Ko. A computer scientist's introductory guide to business process management (BPM). *Crossroads*, 15(4):4:11—-4:18, 2009.

[32] Matteo Magnani and Danilo Montesi. BPDMN: A conservative extension of BPMN with enhanced data representation capabilities. *CoRR*, abs/0907.1978, 2009.

[33] Nuno Melão and Michael Pidd. Use of business process simulation: A survey of practitioners. *Journal of the Operational Research Society*, 54(1):2–10, 2003.

[34] OMG. *Business Process Modeling Notation (BPMN), version 2.0*. 2011.

[35] Wang Pan and He Wei. Research on Key Performance Indicator (KPI) of Business Process. In IEEE, editor, *2012 Second International Conference on Business Computing and Global Informatization*, pages 151–154, 2012.

[36] David Parmenter. *Key performance indicators: developing, implementing, and using winning kpis*. John Wiley & Sons, 2007.

[37] Ray J. Paul, George M. Giaglis, and Vlatka Hlupic. Simulation of Business Processes. *American Behavioral Scientist*, 42(10):1551–1576, 1999.

[38] Ray J. Paul, Vlatka Hlupic, and George M. Giaglis. Simulation modeling of business processes. In *Proceedings of the 3rd UK Academy of Information Systems Conference*, pages 311–320, 1998.

[39] Igor Rust, Deniz Cetinkaya, Mamadou Seck, and Ivo Wenzler. Business Process Simulation for Management Consultants: a Devs-Based Simplified Business Process Modelling Library. In *Proceedings of the 23rd European Modelling and Simulation Symposium (EMSS 2011)*, 2011.

[40] John Ryan and Cathal Heavey. Process modeling for simulation. In *Computers in Industry*, volume 57, pages 437–450. Elsevier Science Publishers, 2006.

[41] Dave Steinberg, Frank Budinsky, Marcelo Paternostro, and Ed Merks. *EMF Eclipse Modeling Framework*. 2008.

[42] Arthur ter Hofstede, Wil van der Aalst, and Mathias Weske. Business Process Management: A Survey. In Mathias Weske, editor, *Business Process Management*, volume 2678 of *Lecture Notes in Computer Science*, page 1019. Springer Berlin / Heidelberg, 2003.

[43] Wil M. P. van der Aalst. Business Process Management: A Comprehensive Survey. *ISRN Software Engineering*, 2013:1–37, 2013.

[44] WfMC. *Business Process Simulation Specification*. 2016.