alsoMATH - A Database for Mathematical Algorithms and Software

Wolfgang Dalitz
Zuse Institute Berlin
Takustr. 7, D-14195 Berlin
dalitz@zib.de

Wolfram Sperber
FIZ Karlsruhe
Franklinstr. 11, D-10587 Berlin
wolfram@zbmath.de

Moritz Schubotz
FIZ Karlsruhe
Franklinstr. 11, D-10587 Berlin
moritz.schubotz@zbmath.de

Hagen Chrapary
FIZ Karlsruhe
Franklinstr. 11, D-10587 Berlin
hagen@zbmath.de

Abstract

Mathematical publications are an important resource for the development of machine-based methods for mathematical knowledge management. This article describes the publication-based approach to improve the information and the access to two important classes of mathematical research, mathematical software and mathematical algorithms. The publication-based approach is based on analyzing links and the structure of mathematical publications. It has been used to build the swMATH service which provides comprehensive information about mathematical software and algorithms.

1 Introduction

Linking of data creates additional value as Amazon, Google, and Facebook have shown. But this also applies to scientific information. For mathematics the swMATH database [swMATH] provides a link between mathematical publications and mathematical software [Greul]. The citation and description of the software used is an essential information for the user. But of course all context information is also of interest, the algorithm behind the software, real-world problems and models which can be solved by software. In the following, a publication-based approach to a repository of mathematical algorithms and its relation to software is implemented.

In: C. Kaliszyk, E. Brady, J. Davenport, W.M. Farmer, A. Kohlhase, M. Kohlhase, D. Müller, K. Pąk, and C. Sacerdoti Coen (eds.): Joint Proceedings of the FMM and LML Workshops, Doctoral Program and Work in Progress at the Conference on Intelligent Computer Mathematics 2019 co-located with the 12th Conference on Intelligent Computer Mathematics (CICM 2019), Prague, Czech Republic, July 8–12, 2019, published at http://ceur-ws.org

2 Analyzing zbMATH

Algorithms and software are closely connected. Algorithms describe the theoretical concept for solving a problem, software is an implementation of an algorithm in a programming language. But identification of algorithms and software is often difficult:

- The terms "algorithm" and "software" in publications operate on the one hand interchangeably and on the other hand distinctively, e.g., the articles in "Transactions on Mathematical Software" [TOMS] have often the titles "Algorithm..." and describe the method and an implementation.
- Publications cover different aspects of algorithms and software, ranging from theoretical considerations to practical issues of the implementation.
- Not all publications contain precise information about software. Some publications contain formulations as "numerical experiments", "simulations", etc. So, it is not clear whether and which software has been used.

In recent years, a publication-based approach has been successfully used to setup a database of mathematical software. The database swMATH analyzes the bibliographic database zbMATH [zbMATH] for software and contains currently (2019-04-08) 25,690 software-related objects with 339,683 references in 186,917 zbMATH articles. Therefore heuristic term-based methods are be used. This approach can be extended to algorithms. A common database of mathematical algorithms and software makes sense for different reasons:

- An embedding of swMATH entries in their mathematical context (algorithms). This grants the user the option to easily find adequate methods for solving the problem.
- A more complete overlapping of mathematical software. The database zbMATH contains 383,251 documents (2019-04-08) with the term "algorithm*" inside. That is much more extensive than the number of swMATH entries. Algorithms are a central aspect in at least 94,069 documents (the title of the documents contains often the term "algorithm*").
- A first separate database of algorithms and more complete information about mathematical software. This results because zbMATH is mainly designed to general aspects of content analysis of mathematical publications and imprecise references to software.

In a first step of our approach to the alsoMATH database we look for all zbMATH documents and related sources (zbMATH+) containing the term "algorithm*" (class A) and the zbMATH documents which explicitly cite software (class S). All publications in class S are listed in the database swMATH. To identify zbMATH entries in class A all fields of the database, especially the abstract, will be included in the analysis. Then we typify all zbMATH entries in class A and in class S as follow:

- class A documents containing neither software citation nor indirect hints to software (type I)
- class S documents which are not in class A (type II)
- the common documents in class A and S (type III)
- documents which belong to A but not to S and contain implicit hints to software such as "numerical experiments demonstrate the efficacy of the algorithm" (type IV). This means type IV defines documents which could refer to new special software projects and are therefore interesting for researchers and software developers.

The classification defines a scheme of four disjunct sets which is illustrated in figure 1.

This allows the user to conduct a differentiated search for publications which provide information about both algorithms and software, or publications which provide information only about algorithms or software. It is easy to identify all documents of type I and type II. For this aim we have to compare the new data set of class A with S (the swMATH database). The intersection between A and S is the set III. The set of type II is given as the difference set S minus type III documents. Then we have to calculate the set I and IV. The sets I and IV are subsets of the class S documents minus type III documents. The set I contains all elements of the class S minus set III without indirect links to software, the set IV contains all elements of the set S minus type III

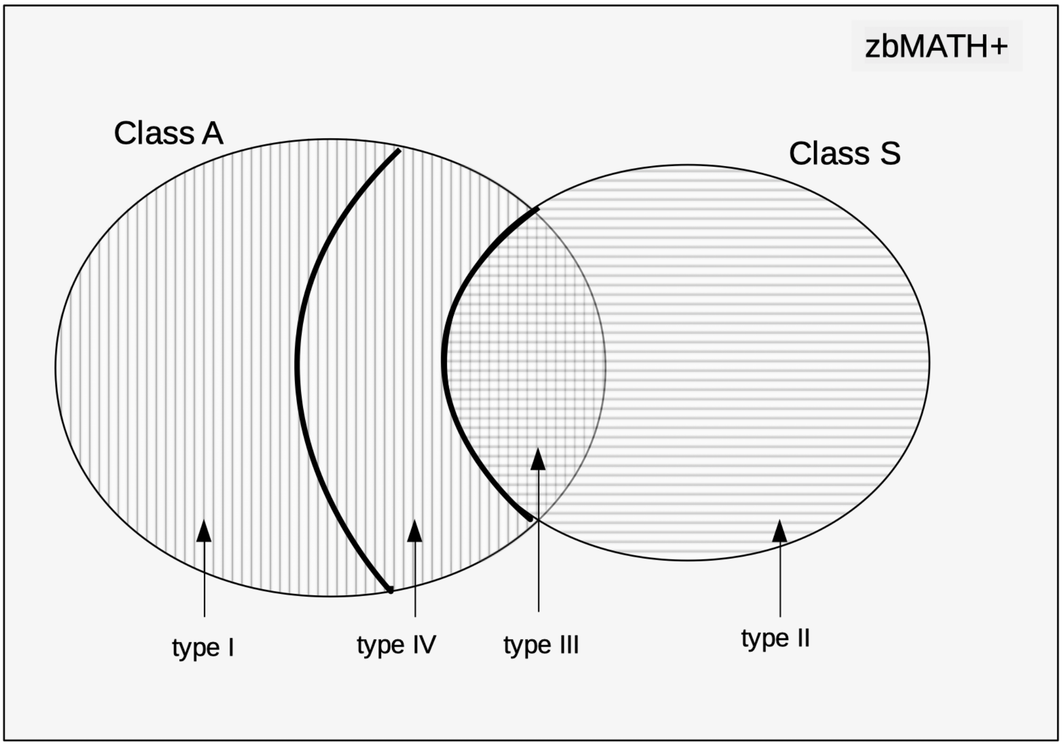

Figure 1: Four disjunct sets of relevant types

with indirect links to software. To detect indirect links to software, we plan to analyze textual phrases in the zbMATH entries. The linking between software and algorithms in publications will be discussed in detail below.

Unfortunately indirect links do not allow to connect the software references of the set of type IV with software names. This leads to an essential problem of the approach: algorithms and software references without identifiers. Often the software names are used as identifiers for the software product (set of all artifacts of a software) and version for a concrete artifact of the software. This data is also suitable for searching software.

The situation for algorithms is more unclear:

- There are some classes of algorithms which have a name. e.g., Newton methods which are often cited as Newton-type methods, but most classes of algorithms don't have a name.
- There are also names for single algorithms, e.g., the "Lickteig-Roy sub resultant algorithm", but most names for single algorithms are missing.
- A lot of references refer to algorithms without a name.

If an algorithm has a name we try to detect the name by heuristic means. For this aim we analyze the textual neighborhood of the term "algorithm". If we can detect a name, we use the MSC classification of the publication to assign an algorithm to its mathematical subjects and application areas. Of course this allows only a rough sorting of algorithms (type I) and the software references of type IV but also a basic search for both algorithms and software (by name and alternatively by the mathematical subjects and application areas).

As a result we get a first database for mathematical algorithms and software and a valuable extension of the information of the swMATH entries (by adding the algorithm which underlies the software). This allows also a better clustering of the software, e.g., by looking for different software which bases on the same algorithm.

3 Linking between algorithms and software

Publications of type III and type IV must be analyzed in more detail. The relations between the software and algorithms can be different, in general we have a m:n relation between mathematical problems, algorithms, and software. But all additional information about the context of an algorithm or software would be helpful for the user. A user gets information about the mathematical background and the method which is implemented by software. Otherwise the retrieval functionalities in swMATH can be substantially increased.

But linking between algorithms and software may happen in a variety of relations.

The 1:1 case

Here we have a direct relation between an algorithm and software. In other words the software is an implementation of an algorithm, e.g., the software presented in TOMS refers directly to the algorithm which is implemented by the software. Also other journals which are specialized to mathematical software describe both the algorithm and its implementation. Moreover, a direct linking to algorithms can be also found in the description of the software, for example "SoPlex is a Linear Programming (LP) solver based on the revised simplex algorithm" or in the so-called "standard publications" in swMATH. This term is reserved in swMATH for publications which describe a software more in details.

But there is a plenty of other relations between algorithms and software. Moreover, the relationships between algorithms and software can change dynamically.

The m:n case

In general, we have relationships between a set of algorithms and a set of different software implementations, e.g., the SCIP software suite provides different solvers for some classes of optimization problems and algorithms.

The 1:n case

An algorithm can be implemented on different way, e.g., the parallelization framework UG in SCIP.

Indirect relationships

As said above some publications give only hints to software. But this is also true for some publications of type III. The swMATH database distinguishes between the standard publications and the user publications which describe use cases of a software. These publications often do not directly link algorithms and software, e.g., a new algorithm is developed and the cited software is used for the solution of a sub problem.

We will start with the analysis of the publications in the mathematical software journals, the descriptions and the standard publications. In other words we try to identify direct relations between software and algorithms. The underlying algorithm will be explicitly presented in the swMATH database, see the following screenshot for the software SCIP (figure 2).

4 Bootstrapping with Linked Open Data from Wikidata

One approach to crowdsource links between publications, algorithms, and software is the knowledge graph Wikidata [Wikidata]. The knowledge base is organized into items with statements and external identifiers [Vrandecic]. In particular, there are external identifiers to specify the zbMATH work ID (property 894, see https://w.wiki/52f for examples) and the swMATH work ID (property 6830, see https://w.wiki/52g for examples) which we introduced recently. With the help of these external identifiers, we can uniquely identify items in swMATH and zbMATH and to associate additional crowdsourced data from Wikidata. For example, the Wikidata item for the computer algebra system Maple Q139380as 13 different statements about the software. Of particular interest is the "interest of" (P31) property which specifies that Maple is a "computer algebra system" (Q830340, a "data analysis softwar" (Q28050159, and more. The linked item "computer algebra system" (Q830340 has specified that computer algebra systems are a "subclass of" (P279) of "mathematical software" (Q1639024 which confirms that the Wikidata community sees Maple, a mathematical software. With state of the art knowledge discovery (KDD) mining methods, such as openRefine, we plan to bootstrap an initial collection of algorithms and associated software [Delpeuch]. In a second step, we will implement a bot that suggests adding statements about the software and the related algorithm. The community feedback on those suggestions will help us to refine the formally presented methods that were based on swMATH and zbMATH data alone [Scharpf, Schubotz].

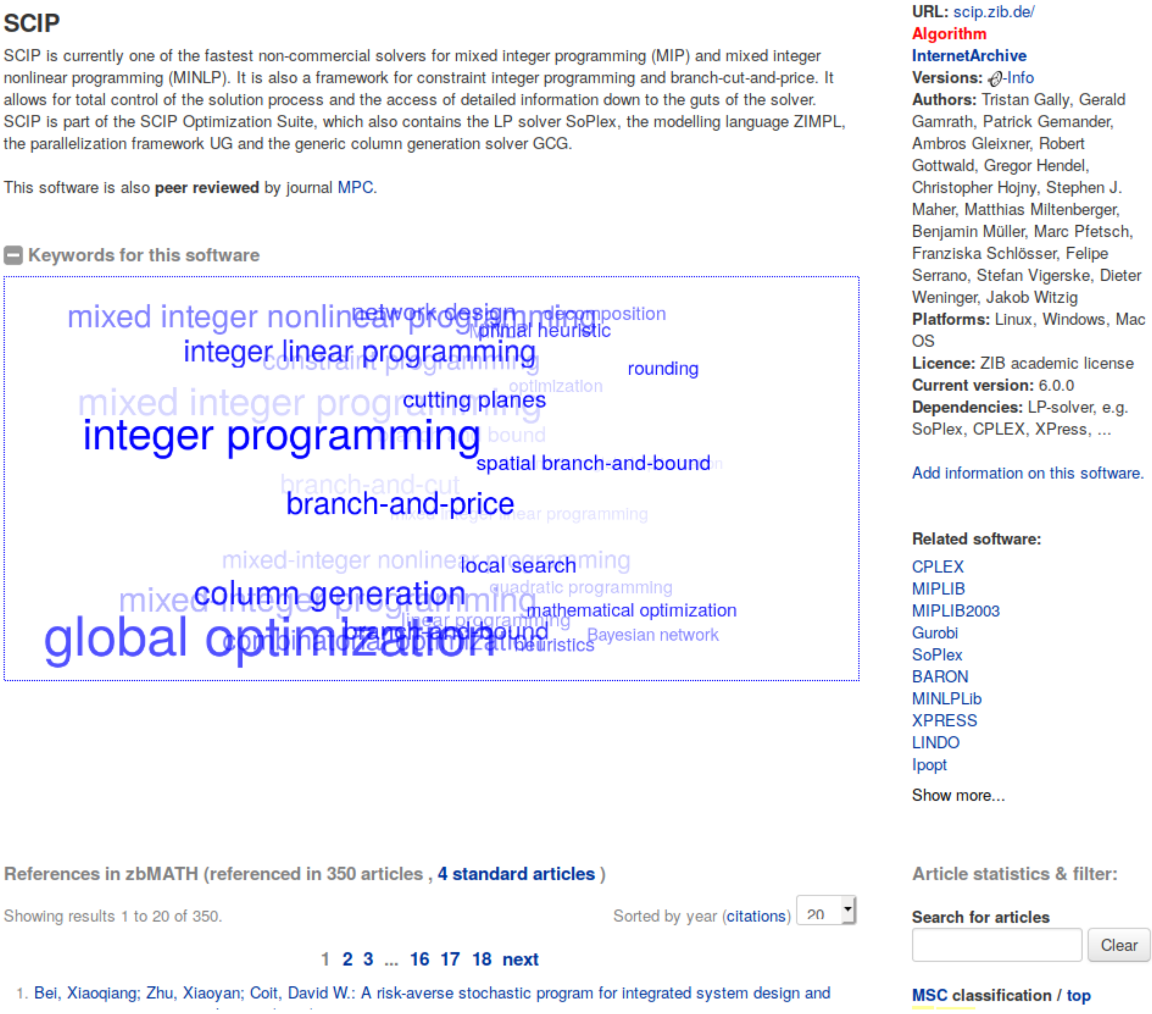

Figure 2: Linking Software and Algorithm

5 Summary

From our point of view a common database about algorithms and software is a natural but non-trivial step for the extension of swMATH. It could provide an overview about the universe of mathematical algorithms and software.

It is necessary to develop extended heuristic methods for the textual analysis of publications especially for the relations between algorithms and software. Up to now no concept for searching algorithms exists. Software but only a small number of algorithms have a name. In a first step we will assign the algorithms to the MSC classes where they are cited and extend this by the keywords of the publication. This allows a first but rough classification of mathematical algorithms and their mathematical subjects.

Acknowledgment

The work for this article has been conducted within the Research Campus MODAL funded by the German Federal Ministry of Education and Research (BMBF grant number 05M14ZAM).

References

[swMATH] The database swMATH (2014 -), *http://www.swmath.org*

[Greul] Gert-Martin Greuel, Wolfram Sperber. swMATH – an information service for mathematical software, in: Hong, Hoon (ed.) et al., Mathematical software – ICMS 2014. 4th International Congress, Seoul, South Korea, ICMS2014. August 5–9 2014. Proceedings. Springer. *Lecture Notes in Computer Science* 8592 (2014), 691-701,

[TOMS] ACM Transactions on Mathematical Software (TOMS), *https://toms.acm.org/*

[zbMATH] The database zbMATH (1868 -), *https://www.zbmath.org*

[Wikidata] Knowlegde Graph Wikidata, *https://www.wikidata.org*

[Delpeuch] Antonin Delpeuch. A survey of openrefine reconciliation services. *CoRR*, abs/1906.08092, 2019.

[Scharpf] Philipp Scharpf, Moritz Schubotz, and Bela Gipp. Representing mathematical formulae in content mathml using wikidata. In *BIRNDL@SIGIR*, volume 2132 of *CEUR Workshop Proceedings*, pages 46–59. CEUR-WS.org, 2018.

[Schubotz] Moritz Schubotz, Philipp Scharpf, Kaushal Dudhat, Yash Nagar, Felix Hamborg, and Bela Gipp. Introducing mathqa - a math-aware question answering system. In *Proceedings of the ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL), Workshop on Knowledge Discovery*, Fort Worth, USA, 6 2018.

[Vrandecic] Denny Vrandecic. Wikidata: a new platform for collaborative data collection. In *WWW (Companion Volume)*, pages 1063–1064. ACM, 2012.