Special issue on symbolic computation in software science

Citation for the original published paper (version of record):
Bouhoula, A., Buchberger, B., Kovacs, L. et al (2015). Special issue on symbolic computation in software science. Journal of Symbolic Computation, 69(1): 1-2.
http://dx.doi.org/10.1016/j.jsc.2014.09.027

(article starts on next page)

Journal of Symbolic Computation 69 (2015) 1–2

Contents lists available at ScienceDirect

Journal of Symbolic Computation

www.elsevier.com/locate/jsc

Foreword

Special issue on symbolic computation in software science

This special issue of the Journal of Symbolic Computation is related to the Fourth and Fifth International Symposia on Symbolic Computation in Software Science, SCSS 2012 and SCSS 2013, held, respectively, in Gammarth, Tunisia, December 15–17, 2012 and in Hagenberg, Austria, July 5–6, 2013. After an open call for papers, we received 18 submissions, six of which have been selected for publication after rigorous reviewing.

The scope of the SCSS symposium and this special issue covers a wide range of topics related to theoretical and practical aspects of symbolic computation in software science. The accepted papers address problems in algorithm synthesis, termination analysis, program debugging, service composition, computational origami, formalization and computerization of knowledge, and automated reasoning.

The article by *María Alpuente, Demis Ballis, Francisco Frechina,* and *Julia Sapiña* considers the problem of trace exploration for Rewriting Logic computations. Trace exploration is an important technique for dynamic analysis of program behavior. As traces might grow very large, it is a challenge to develop adequate methods and tools. The approach proposed in the paper is based on a generic algorithm that can be tuned in different ways to reduce the size and the complexity of the traces being explored. It is intended for debugging and optimizing Rewriting Logic-based tools that manipulate computations in conditional rewrite theories modulo equations. The algorithm is implemented in the graphical tool called Anima that can be used for the analysis of Maude computations.

Walid Belkhir, Yannick Chevalier, and *Michael Rusinowitch* present an automata-based approach to service composition. The problem under consideration is composition synthesis: Given a client and available services, compute a mediator agent that enables communication between the client and the services in such a way that each client request is forwarded to an appropriate service. The problem reduces to the existence of simulation between the target service (specified on the basis of the client request) and the asynchronous product of available services. Communication actions are parametrized by (possibly constrained) data from an infinite domain, which can be problematic for automata-based approaches to composition synthesis due to certain undecidability results for simulation. The authors deal with this problem by introducing an expressive service specification formalism in the form of an extension of automata, called parametrized automata, prove decidability of simulation preorder there, and use it to develop a mediator synthesis procedure.

Automation of algorithm synthesis and theory exploration are the problems considered in the paper by *Isabela Drămnesc* and *Tudor Jebelean*. The focus is on proving-based automated synthesis of list algorithms: Given an input list, prove constructively the existence of its sorted version. Using different proof techniques, the authors synthesize five sorting algorithms as well as auxiliary functions used in them. During the process of synthesis, the theory of lists is explored by introducing and proving the properties that are used in the synthesis.

The paper by *Tetsuo Ida, Fadoua Ghourabi,* and *Kazuko Takahashi* describes a formalization of polygonal knot origami. Computational origami represents algorithms and methods for modeling the ways

http://dx.doi.org/10.1016/j.jsc.2014.09.027

how materials can be folded, and has interesting applications in engineering. The authors analyze the fold of regular pentagonal and heptagonal knots and show that it is a new method of folding. The knots are first formalized in the language of the e-origami system Eos. Next, the specification is transformed into algebraic expressions. Knot fold represents a constraint solving problem for those expressions, which is done symbolically and numerically for the construction. To verify the construction, the geometry theorem prover of Eos is used that incorporates the methods of Gröbner bases computation.

The work of *Cezary Kaliszyk* and *Josef Urban* is motivated by the need of efficient automated reasoning over large formal knowledge bases. Such mathematical libraries contain a lot of formalized concepts, theorems, proofs, theory developments. Mathematicians reuse a very small part of it in later proofs. The authors propose criteria for estimating usefulness of statements in such a knowledge for proving further conjectures. The idea is to add to the set of main theorems in a large library the most useful lemmas extracted from the proofs in the library. Such lemmas can be used then as premises to automated reasoning systems to attack new proving problems. The proposed approach is experimentally tested on HOL Light and Flyspeck libraries, adding millions of the best lemmas to them. The experiments show that such a method, combined with learning, strengthens automated theorem proving of new conjectures over large formal mathematical libraries.

Harald Zankl, *Sarah Winkler*, and *Aart Middeldorp* present an automated termination proof for a rewrite system whose derivational complexity cannot be bounded by a multiple recursive function. The system represents an encoding of the computation of Goodstein sequences. The method uses a novel implementation of algebras based on ordinal interpretations and can be applied to some other problems that until now were beyond the reach of automated termination proving tools. The authors also show how these ideas can be used to automate elementary interpretations for termination analysis.

We thank the authors for their contributions and the referees for their careful and thorough work. We are grateful to Hoon Hong, the Editor-in-Chief of the Journal of Symbolic Computation, for agreeing to organize this special issue.

Adel Bouhoula
Higher School of Communication of Tunis,
University of Carthage, Tunisia
E-mail address: Adel.Bouhoula@supcom.rnu.tn

Bruno Buchberger
Research Institute for Symbolic Computation,
Johannes Kepler University, Linz, Austria
E-mail address: Bruno.Buchberger@risc.jku.at

Laura Kovács [1]
Department of Computer Science and Engineering,
Chalmers University of Technology, Gothenburg, Sweden
E-mail address: laura.kovacs@chalmers.se

Temur Kutsia
Research Institute for Symbolic Computation,
Johannes Kepler University, Linz, Austria
E-mail address: kutsia@risc.jku.at

3 July 2014
Available online 1 October 2014

[1] This work was partially supported by the Swedish VR grant D0497701 and the Austrian research projects FWF S11410-N23 and WWTF ICT C-050.