Editorial

Parallel and Distributed Methods in Verification

Over the last three decades, we have observed increasing dependency on computer and communication systems—and their hardware and software—in all aspects of daily life. As an example, we mention the use of online banking services, the use of computer-controlled car safety systems and the use of fully software-supported high-definition television sets. All these types of systems are characterized by their complexity as regards their hardware and—first and foremost—their software. If these systems do not behave as expected (or as specified), the effect is that they become less easy to use—to say the least. However, more often than not such situations lead to monetary losses or they endanger lives.

To avoid all of the above, it is paramount that these types of systems are designed in such a way that their desirable characteristics are certified at the design stage, and not as some afterthought. To accomplish this, a wide variety of formal verification techniques have been developed over the last decades. The so-called model checking technique develops a formal model of the design and certifies that this model obeys a certain set of behavioural characteristics. It is noteworthy that the founding fathers of this technique—Edmund M. Clarke, E. Allen Emerson and Joseph Sifakis—received the 2007 ACM Turing award for their achievements in exactly this area.

Model checking is a very attractive approach to certifying the correctness of complex systems. This is due to the fact that this technique is completely automatic and that it provides examples of incorrect behaviour serving as debugging information. One of the main technical challenges in developing widely accessible model checking tools is to deal with the complexity of it, i.e. the time and space resources needed to verify real size systems. Over the last years, the model checking community has developed several methods of dealing with complexity issues. One of them is the use of parallelization and/or distribution to manage the space and time requirements needed to check large systems.

The key idea behind parallelization and distribution is to exploit the power and the joint memory of computer clusters, parallel computers and/or multi-core machines. This requires advanced algorithms to deal with shared memory, communication mechanisms to quickly share and distribute locally computed but globally needed information and to efficiently distribute the model checking problem itself over many computing cores. Recent technological advances make computing clusters (of multi-core processors) available at a reasonable price; it is up to the research community to transfer this raw computing power into model checking algorithms that work in practice.

This special issue is specifically related to the above aim: how can modern workstation clusters and multi-core machines be used as efficient platforms for model checking algorithms? This issue is the result of a number of workshops devoted to this theme. Back in 2002, the first international workshop on Parallel and Distributed Model Checking (PDMC) [3] took place in Brno (as a satellite of Concur 2002). Since then, the PDMC workshop has been hosted annually, resulting in a number of Electronic Notes in Theoretical Computer Science volumes [numbers 68(4), 89(1) 128(3), 135(2), 198(1) and 220(2)], a Lecture Notes in Computer Science volume (number 4346, jointly with International Workshop on Formal Methods for Industrial Critical Systems 2006) and special issues of the journals Software Tools for Technology Transfer [2] and Formal Methods in System Design [1].

Vol. 21 No. 1, © The Author, 2011. Published by Oxford University Press. All rights reserved.
For Permissions, please email: journals.permissions@oxfordjournals.org
doi:10.1093/logcom/exp001

The current special issue brings together a selection of papers that are extensions and/or combined papers from the three most recent PDMC workshops. An initial selection of the best eight papers from the last 3 years was made by the involved Programme Committee chairs and the PDMC steering committee. The authors of these papers were invited to extend their work, and to resubmit the papers for this special issue. The papers submitted in this way were carefully reviewed and revised and, eventually, five high-quality papers were selected to be presented in this special issue. These papers cover the important basic techniques that the verification process builds upon: symbolic as well as explicit representations of the state space of the system to be verified; and a direct search as well as a reduction of the search to satisfiability problem.

The paper *Parallel SAT Solving in Bounded Model Checking* by Erika Ábrahám, Tobias Schubert, Bernd Becker, Martin Fränzle and Christian Herde proposes a novel parallel algorithm for Bounded Model Checking (BMC) based on satisfiability (SAT) solving. The novelty of the approach is running SAT solvers for different counterexample lengths in parallel, rather than parallelizing the satisfiability check of a single formula. Solvers prune the search space by learning and adapting conflict clauses from each other while solving. The experiments show very good speed-up behaviour (linear, or even super-linear) of the method on a suite of test models.

Distributed Algorithms for SCC Decomposition by Jiří Barnat, Jakub Chaloupka and Jaco van de Pol surveys parallel algorithms for the decomposition of a graph into strongly connected components. The authors identify basic procedures the algorithms are assembled from and introduce a new technique based on a recursive application of the Owcty-Backward-Forward (OBF) technique. In an extensive experimental study, the new algorithm outperforms all the other algorithms in most cases.

Stefan Blom, Bert Lisser, Jaco van de Pol and Michael Weber introduce in the paper *A Database Approach to Distributed State Space Generation* a new solution to the problem. The basic scheme of distributed state space generation is enhanced with a global database, in order to provide a globally unique representation of values from recursive data types. A database replication together with a message piggybacking scheme is employed to alleviate the cost of synchronization.

The paper *Speculative Image Computation for Distributed Symbolic Reachability Analysis* by Ming Ying Chung and Gianfranco Ciardo is concerned with symbolic reachability analysis using horizontal slicing of multi-value decision diagrams. The paper explores four methods of the best possible utilization of the idle time of a network of workstations during the state space generation process. The idle time is used to speculatively perform image computation and cache the result which can be potentially used by the algorithm in the future to improve overall efficiency.

The paper *To Parallelise or To Optimise?* by Jonathan Ezekiel, Gerald Lűttgen and Radu Siminiceanu discusses parallelization issues of symbolic model checking and the Saturation algorithm in particular. The authors address the problem of parallelization for the sake of time efficiency instead of the more widely applied parallelization for obtaining more memory resources. Practical results on a benchmark set are given with the outcome that the performance of a parallel symbolic state space generation algorithm is highly dependent on the model. The paper prefers optimization of sequential symbolic model checking algorithms to their parallelization.

Ivana Černá
Masaryk University, Brno, Czech Republic
e-mail: cerna@fi.muni.cz

Boudewijn R. Haverkort
University of Twente, Enschede, The Netherlands.
e-mail: brh@cs.utwente.nl

References

[1] L. Brim and M. Leucker (Guest eds). *Formal Methods in System Design*, vol. 29, (Special issue on parallel and distributed databases), Springer, pp.115–214, 2006.

[2] *International Journal on Software Tools for Technology Transfer*, vol. 7, (Special section on parallel and distributed model checking), Springer, pp. 1–86, 2005.

[3] PDMC workshop series. Available at `http://pdmc.informatik.tu-muenchen.de/` (last accessed on 3 February, 2009)