International Journal on Software Tools for Technology Transfer
https://doi.org/10.1007/s10009-023-00699-x

GENERAL

Special Issue: TACAS 2021

Tools and algorithms for the construction and analysis of systems: a special issue on tool papers for TACAS 2021

Peter Gjøl Jensen · Thomas Neele

Abstract This special issue contains six revised and extended versions of tool papers that appeared in the proceedings of TACAS 2021, the 27th International Conference on Tools and Algorithms for the Construction and Analysis of Systems. The issue is dedicated to the realization of algorithms in tools and the studies of the application of these tools for analysing hard- and software systems.

Keywords Software verification, Hardware verification, Theorem proving, Static analysis, Runtime verification, Neural Networks, SAT and SMT solving, Tool environments and tool architectures

1 TACAS & tools

TACAS aims to bring together researchers, developers and users interested in rigorously based tools and algorithms for the construction and analysis of systems. The conference bridges the gaps between different communities with this common interest and aids them in improving the utility, reliability, flexibility and efficiency of these tools and algorithms.

Even though TACAS always has had a strong focus on tools, there has been no mechanism in place to evaluate these tools nor to replicate stated empirical results. TACAS 2018 saw the introduction of an *Artifact Evaluation Committee* (AEC), which runs the tools and replicates results inside a standardised virtual machine, based on instructions provided by the authors. In the first year, artifact submission was optional, but starting from TACAS 2019, authors of *tool papers* (which highlight the design and features of a software tool) are required to submit an artifact along with their paper. Submitting an artifact for a research paper is still optional, and may be done in a second round, after notification of paper acceptance. Papers whose artifact is accepted by the AEC receive an *artifact badge*. In view of *open science*, authors are encouraged to publish their artifacts in a public archive such as *Zenodo* or *figshare*.

2 This special issue

In total TACAS 2021 [3,4] received 141 submissions, out of which 29 *tool papers* (of maximum 16 pages) and 16 *tool demo papers* (max 6 pages). Furthermore, 63 accompanying artifacts were submitted to be reviewed by the AEC. After careful testing and review, the AEC accepted 50 artifacts. Taking these judgments into account, the Program Committee accepted 47 papers, including seven tool papers and six tool demo papers. The AEC furthermore accepted seven out of nine artifacts submitted to the post-acceptance artifact evaluation.

Based on the scores of the paper reviews, as well as the artifact reviews, four tool papers and two tool demo papers were selected and invited for submission to this special issue[1]. This resulted in six extended papers, all of which were accepted after at three reviews. Below, we give a short summary for each of these papers.

2.1 Improving AMulet2 for verifying multiplier circuits using SAT solving and computer algebra

While many hardware verification techniques use Boolean reasoning, current methods for multiplier circuits still rely on algebraic reasoning. Daniela Kaufmann and Armin Biere present their tool AMULET2 [5] which approaches this problem by distinguishing the components within a multiplier

Aalborg University, Aalborg, Denmark
E-mail: pgj@cs.aau.dk
Eindhoven University of Technology, Eindhoven, Netherlands
E-mail: t.s.neele@tue.nl

[1] A selection of research papers was invited for a separate special issue, to appear in *Logical Methods in Computer Science*

circuit and applying either algebra or SAT solving. This decision is made depending on the performance of either method on the type of circuit the component contains. This tool is the successor of AMulet1, improving on the implemented algorithms and modularisation of the program. The paper furthermore presents a new XOR-based slicing approach and optimisations that improve memory efficiency.

2.2 SyReNN: a tool for analyzing deep neural networks

The SyReNN tool focuses on the expressive subclass of Deep Neural Networks (DNNs) in which activation functions are Piecewise Linear [7]. Matthew Sotoudeh, Zhe Tao, and Aditya V. Thakur show that a precise symbolic representation of the input/output relation of a DNN can be obtained and exploited for visualization, analysis and even repair of the DNN. SyReNN is demonstrated on a series of benchmarks on which it outperforms state-of-the-art approximate DNN verification tools in both precision and performance. Lastly, the authors leverage SyReNN to improve the robustness of optical character recognition against adversarial inputs.

2.3 Verified propagation redundancy and compositional UNSAT checking in CakeML

Modern SAT solvers provide proof certificates, and many do so in the state-of-the-art *propagation redundancy* (PR) format. The `cake_lpr` tool by Yong Kiam Tan, Marijn J. H. Heule and Magnus O. Myreen [8] is the first certificate checker that works directly on the succinct PR format, rather than via translation into weaker proof systems. In addition `cake_lpr` comes with formal correctness guarantees as authors provide a verifcation of the machine-code of the tool via the CakeML toolchain. To tackle the problem, the authors provide a translation from the PR format into the proposed Linear PR format (LPR). This format further facilitate compositional proofs, a feature the authors exploit to implement concurrent certificate checking in `cake_lpr`. The full approach is demonstrated on a large benchmark of certificates, demonstrating competitive performance.

2.4 Combining rule- and SMT-Based reasoning for verifying floating-point Java programs in KeY

Rosa Abbasi, Jonas Schiffl, Eva Darulova, Mattias Ulbrich and Wolfgang Ahrendt address the limited support for reasoning on floating-point arithmetics in deductive verification techniques for Java programs. In particular, they extend the KeY verifier [1] with a combination of rule-based reasoning and external calls to SMT-solvers with floating point reasoning. The authors introduce a novel benchmark containing realistic program fragments implementing e.g., transcendental functions. On this benchmark KeY is able to automatically prove certain value constraints and special value absence (Not-A-Number and infinity), and do so within reasonable computation times.

2.5 On the road with RTLola: testing real driving emissions on your phone

The RTLola tool [2] is introduced by Sebastian Biewer, Bernd Finkbeiner, Holger Hermanns, Maximilian A. Köhl, Yannik Schnitzer and Maximilian Schwenger to bring runtime monitoring to the masses – here specifically for monitoring vehicle exhaust emissions. RTLola provides a full platform where data is collected via the On-Board-Diganostics interface of the vehicle, transmitted to the RTLola smart phone application where it is analysed. The authors furthermore formalize parts of the European Real Driving Emission test requirements in the stream-based specification language and demonstrate via RTLola the runtime monitoring in a real environment.

2.6 Algorithm selection for SMT solvers

In their paper [6], Joseph Scott, Aina Niemetz, Mathias Preiner, Saeed Nejati and Vijay Ganesh propose the tool MachSMT for discriminating between the growing family of SMT-solvers for single problem instances. The MachSMT tool utilizes machine learning techniques trained on e.g., the grammatical and syntactical features of the input problem to derive a ranking over a collection of SMT-solvers. The approach demonstrates an overall performance improvement both when used for discriminating across solvers but also when used as a configuration selector for CVC5.

References

1. Abbasi, R., Schiffl, J., Darulova, E., Ulbrich, M., Ahrendt, W.: Combining Rule- and SMT-Based Reasoning for Verifying Floating-Point Java Programs in KeY. International Journal on Software Tools for Technology Transfer (this issue) (2023)
2. Biewer, S., Finkbeiner, B., Hermanns, H., Köhl, M.A., Schnitzer, Y., Schwenger, M.: On the Road with RTLola - Testing Real Driving Emissions on your Phone. International Journal on Software Tools for Technology Transfer (this issue) (2023)
3. Groote, J.F., Larsen, K.G. (eds.): Tools and Algorithms for the Construction and Analysis of Systems - 27th International Conference, TACAS 2021, Proceedings, Part I, *Lecture Notes in Computer Science*, vol. 12651. Springer (2021).
4. Groote, J.F., Larsen, K.G. (eds.): Tools and Algorithms for the Construction and Analysis of Systems - 27th International Conference, TACAS 2021, Proceedings, Part II, *Lecture Notes in Computer Science*, vol. 12652. Springer (2021).
5. Kaufmann, D., Biere, A.: Improving AMulet2 for verifying multiplier circuits using SAT solving and computer algebra. International Journal on Software Tools for Technology Transfer (this issue) (2023)
6. Scott, J., Niemetz, A., Preiner, M., Nejati, S., Ganesh, V.: Algorithm Selection for SMT Solvers . International Journal on Software Tools for Technology Transfer (this issue) (2023)

7. Sotoudeh, M., Tao, Z., Thakur, A.V.: SyReNN: A Tool for Analyzing Deep Neural Networks. International Journal on Software Tools for Technology Transfer (this issue) (2023)
8. Tan, Y.K., Heule, M.J.H., Myreen, M.O.: Verified Propagation Redundancy and Compositional UNSAT Checking in CakeML. International Journal on Software Tools for Technology Transfer (this issue) (2023)