Lecture Notes in Computer Science **11138**

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

More information about this series at http://www.springer.com/series/7408

Shuvendu K. Lahiri · Chao Wang (Eds.)

Automated Technology for Verification and Analysis

16th International Symposium, ATVA 2018
Los Angeles, CA, USA, October 7–10, 2018
Proceedings

Springer

Editors
Shuvendu K. Lahiri
Microsoft Research
Redmond, WA
USA

Chao Wang
University of Southern California
Los Angeles, CA
USA

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-030-01089-8 ISBN 978-3-030-01090-4 (eBook)
https://doi.org/10.1007/978-3-030-01090-4

Library of Congress Control Number: 2018955278

LNCS Sublibrary: SL2 – Programming and Software Engineering

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume contains the papers presented at the 16th International Symposium on Automated Technology for Verification and Analysis (ATVA 2018) held during October 7–10, 2018, in Los Angeles, California, USA.

ATVA is a series of symposia dedicated to the promotion of research on theoretical and practical aspects of automated analysis, verification, and synthesis by providing a forum for interaction between the regional and the international research communities and industry in the field. Previous events were held in Taipei (2003–2005), Beijing (2006), Tokyo (2007), Seoul (2008), Macao (2009), Singapore (2010), Taipei (2011), Thiruvananthapuram (2012), Hanoi (2013), Sydney (2014), Shanghai (2015), Chiba (2016), and Pune (2017).

ATVA 2018 received 82 high-quality paper submissions, each of which received three reviews on average. After careful review, the Program Committee accepted 27 regular papers and six tool papers. The evaluation and selection process involved thorough discussions among members of the Program Committee through the EasyChair conference management system, before reaching a consensus on the final decisions.

To complement the contributed papers, we included in the program three keynote talks and tutorials given by Nikolaj Bjørner (Microsoft Research, USA), Corina Păsăreanu (NASA Ames Research Center, USA), and Sanjit Seshia (University of California, Berkeley, USA), resulting in an exceptionally strong technical program.

We would like to acknowledge the contributions that made ATVA 2018 a successful event. First, we thank the authors of all submitted papers and we hope that they continute to submit their high-quality work to ATVA in future. Second, we thank the Program Committee members and external reviewers for their rigorous evaluation of the submitted papers. Third, we thank the keynote speakers for enriching the program by presenting their distinguished research. Finally, we thank Microsoft and Springer for sponsoring ATVA 2018.

We sincerely hope that the readers find the ATVA 2018 proceedings informative and rewarding.

October 2018

Shuvendu K. Lahiri
Chao Wang

Organization

Steering Committee

E. Allen Emerson	University of Texas, Austin, USA
Teruo Higashino	Osaka University, Japan
Oscar H. Ibarra	University of California, Santa Barbara, USA
Insup Lee	University of Pennsylvania, USA
Doron A. Peled	Bar Ilan University, Israel
Farn Wang	National Taiwan University, Taiwan
Hsu-Chun Yen	National Taiwan University, Taiwan

Program Committee

Aws Albarghouthi	University of Wisconsin-Madison, USA
Cyrille Valentin Artho	KTH Royal Institute of Technology, Sweden
Gogul Balakrishnan	Google, USA
Roderick Bloem	Graz University of Technology, Austria
Tevfik Bultan	University of California, Santa Barbara, USA
Pavol Cerny	University of Colorado at Boulder, USA
Sagar Chaki	Mentor Graphics, USA
Deepak D'Souza	Indian Institute of Science, Bangalore, India
Jyotirmoy Deshmukh	University of Southern California, USA
Constantin Enea	University Paris Diderot, France
Grigory Fedyukovich	Princeton University, USA
Masahiro Fujita	The University of Tokyo, Japan
Sicun Gao	University of California, San Diego, USA
Arie Gurfinkel	University of Waterloo, Canada
Fei He	Tsinghua University, China
Alan J. Hu	The University of British Columbia, Canada
Joxan Jaffar	National University of Singapore, Singapore
Akash Lal	Microsoft, India
Axel Legay	IRISA/Inria, Rennes, France
Yang Liu	Nanyang Technological University, Singapore
Zhiming Liu	Southwest University, China
K. Narayan Kumar	Chennai Mathematical Institute, India
Doron Peled	Bar Ilan University, Israel
Xiaokang Qiu	Purdue University, USA
Giles Reger	The University of Manchester, UK
Sandeep Shukla	Indian Institute of Technology Kanpur, India
Oleg Sokolsky	University of Pennsylvania, USA
Armando Solar-Lezama	MIT, USA
Neeraj Suri	TU Darmstadt, Germany

Aditya Thakur	University of California, Davis, USA
Willem Visser	Stellenbosch University, South Africa
Bow-Yaw Wang	Academia Sinica, Taiwan
Farn Wang	National Taiwan University, Taiwan
Georg Weissenbacher	Vienna University of Technology, Austria
Naijun Zhan	Chinese Academy of Sciences, China

Additional Reviewers

Akshay, S.
Alt, Leonardo
Bastani, Osbert
Bouajjani, Ahmed
Chen, Mingshuai
Chen, Wei
Chen, Zhenbang
Ebrahim, Masoud
Ge, Ning
Habermehl, Peter
Jansen, Nils
Karl, Anja
Katelaan, Jens
Katis, Andreas
Khalimov, Ayrat
Koenighofer, Bettina
Lewchenko, Nicholas V.
Li, Yangjia
Lin, Wang
Liu, Wanwei
Luo, Chen
Maghareh, Rasool
Matteplackel, Raj Mohan
McClurg, Jedidiah
Mordvinov, Dmitry
Pick, Lauren
Poulsen, Danny
Poulsen, Dany
Praveen, M.
Qamar, Nafees
Quilbeuf, Jean
Riener, Martin
Sangnier, Arnaud
Sankaranarayanan, Sriram
Song, Fu
Srivathsan, B.
Suresh, S. P.
Ting, Gan
Tizpaz Niari, Saeid
Trung, Ta Quang
Wang, Chao
Wang, Lingtai
Xue, Bai
Yang, Zhibin
Yin, Liangze
Zhang, Xin

$Z3^{Azure}$ and $Azure^{Z3}$ (Abstract)

Nikolaj Bjørner[1], Marijn Heule[2], Karthick Jayaraman[3], and Rahul Kumar[1]

[1] Microsoft Research
{nbjorner,rahulku}@microsoft.com
[2] UT Austin
marijn@heule.nl
[3] Microsoft Azure
karjay@microsoft.com

Azure to the power of Z3: Cloud providers are increasingly embracing network verification for managing complex datacenter network infrastructure. Microsoft's Azure cloud infrastructure integrates the SecGuru tool, which leverages the Z3 SMT solver, for assuring that the network is configured to preserve desired intent. SecGuru statically validates correctness of access-control policies and routing tables of hundreds of thousands of network devices. For a structured network such as for Microsoft Azure data-centers, the intent for routing tables and access-control policies can be automatically derived from network architecture and metadata about address ranges hosted in the datacenter. We leverage this aspect to perform local checks on a per router basis. These local checks together assure reachability invariants for availability and performance. To make the service truly scalable, while using modest resources, SecGuru integrates a set of domain-specific optimizations that exploit properties of the configurations it needs to handle. Our experiences exemplify integration of general purpose verification technologies, in this case bit-vector solving, for a specific domain: the overal methodology available through SMT formalisms lowers the barrier of entry for capturing and checking contracts. They also alleviate initial needs for writing custom solvers. However, each domain reveals specific structure that produces new insights in the quest for scalable verification: for checking reachability properties in networks, methods for capturing symmetries in networks and header spaces can speed up verification by several orders of magnitude; for local checks on data-center routers we exploit common patterns in configurations to take the time it takes to check a contract from a few seconds to a few milliseconds.

Z3 to the power of Azure: Applications that rely on constraint solving may be in a fortunate situation where efficient solving technologies are adequately available. Several tools building on Z3 rely on this being the common case scenario. Then useful feedback can be produced within the attention span of a person performing program verification, and then the available compute time on a machine or a cluster is sufficient to complete thousands of small checks. As long as this fortunate situation holds, search techniques for SMT is a solved problem. Yet, in spite of rumors to the contrary, SAT

and SMT is by no means a solved problem. When current algorithmic techniques, in particular modern SAT solving search methods based on conflict-driven clause learning, are insufficient to quickly find solutions, a next remedy is to harness computational resources at problems. In the context of SAT solving, the method of Cube & Conquer, has been used with significant success to solve hard combinatorial problems from mathematical conjectures. These are relatively small formulas, but require a substantial search space for analysis. Formulas from applications, such as scheduling and timetabling, are significantly larger and have wildly different structural properties. In spite of these differences, we found that the Cube & Conquer methodology can make a substantial difference as fixing even a limited number of variables can drastically reduce the overhead solving subformulas. We describe a distributed version of Z3 that scales with Azure's elastic cloud. It integrates recent advances in lookahead and distributed SAT solving for Z3's engines for SMT. A multi-threaded version of the Cube & Conquer solver is also available parallelizing SAT and SMT queries.

Contents