Lecture Notes in Computer Science 8559

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

Armin Biere Roderick Bloem (Eds.)

Computer Aided Verification

26th International Conference, CAV 2014
Held as Part of the Vienna Summer of Logic, VSL 2014
Vienna, Austria, July 18-22, 2014
Proceedings

Springer

Volume Editors

Armin Biere
Johannes Kepler University Linz
Altenbergerstr. 69, 4040 Linz, Austria
E-mail: biere@jku.at

Roderick Bloem
IAIK, Graz University of Technology
Inffeldgasse 16a, 8010 Graz, Austria
E-mail: roderick.bloem@iaik.tugraz.at

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-319-08866-2 e-ISBN 978-3-319-08867-9
DOI 10.1007/978-3-319-08867-9
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014942534

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

logic n. **1** the science of reasoning.
– ORIGIN from Greek *logikē tekhnē* ‘art of reason’.

Foreword

In the summer of 2014, Vienna hosted the largest scientific conference in the history of logic. The Vienna Summer of Logic (VSL, http://vsl2014.at) consisted of twelve large conferences and 82 workshops, attracting more than 2000 researchers from all over the world. This unique event was organized by the Kurt Gödel Society and took place at Vienna University of Technology during July 9 to 24, 2014, under the auspices of the Federal President of the Republic of Austria, Dr. Heinz Fischer.

The conferences and workshops dealt with the main theme, logic, from three important angles: logic in computer science, mathematical logic, and logic in artificial intelligence. They naturally gave rise to respective streams gathering the following meetings:

Logic in Computer Science / Federated Logic Conference (FLoC)

- 26th International Conference on Computer Aided Verification (CAV)
- 27th IEEE Computer Security Foundations Symposium (CSF)
- 30th International Conference on Logic Programming (ICLP)
- 7th International Joint Conference on Automated Reasoning (IJCAR)
- 5th Conference on Interactive Theorem Proving (ITP)
- Joint meeting of the 23rd EACSL Annual Conference on Computer Science Logic (CSL) and the 29th ACM/IEEE Symposium on Logic in Computer Science (LICS)
- 25th International Conference on Rewriting Techniques and Applications (RTA) joint with the 12th International Conference on Typed Lambda Calculi and Applications (TLCA)
- 17th International Conference on Theory and Applications of Satisfiability Testing (SAT)
- 76 FLoC Workshops
- FLoC Olympic Games (System Competitions)

Mathematical Logic

- Logic Colloquium 2014 (LC)
- Logic, Algebra and Truth Degrees 2014 (LATD)
- Compositional Meaning in Logic (GeTFun 2.0)
- The Infinity Workshop (INFINITY)
- Workshop on Logic and Games (LG)
- Kurt Gödel Fellowship Competition

Logic in Artificial Intelligence

- 14th International Conference on Principles of Knowledge Representation and Reasoning (KR)
- 27th International Workshop on Description Logics (DL)
- 15th International Workshop on Non-Monotonic Reasoning (NMR)
- 6th International Workshop on Knowledge Representation for Health Care 2014 (KR4HC)

The VSL keynote talks which were directed to all participants were given by Franz Baader (Technische Universität Dresden), Edmund Clarke (Carnegie Mellon University), Christos Papadimitriou (University of California, Berkeley) and Alex Wilkie (University of Manchester); Dana Scott (Carnegie Mellon University) spoke in the opening session. Since the Vienna Summer of Logic contained more than a hundred invited talks, it would not be feasible to list them here.

The program of the Vienna Summer of Logic was very rich, including not only scientific talks, poster sessions and panels, but also two distinctive events. One was the award ceremony of the Kurt Gödel Research Prize Fellowship Competition, in which the Kurt Gödel Society awarded three research fellowship prizes endowed with 100.000 Euro each to the winners. This was the third edition of the competition, themed Logical Mind: Connecting Foundations and Technology this year.

The 1st FLoC Olympic Games formed the other event and were hosted by the Federated Logic Conference (FLoC) 2014. Intended as a new FLoC element, the Games brought together 12 established logic solver competitions by different research communities. In addition to the competitions, the Olympic Games facilitated the exchange of expertise between communities, and increased the visibility and impact of state-of-the-art solver technology. The winners in the competition categories were honored with Kurt Gödel medals at the FLoC Olympic Games award ceremonies.

Organizing an event like the Vienna Summer of Logic was a challenge. We are indebted to numerous people whose enormous efforts were essential in making this vision become reality. With so many colleagues and friends working with us, we are unable to list them individually here. Nevertheless, as representatives of the three streams of VSL, we would like to particularly express our gratitude to all people who helped to make this event a success: the sponsors and the Honorary Committee; the Organization Committee and

the local organizers; the conference and workshop chairs and Program Committee members; the reviewers and authors; and of course all speakers and participants of the many conferences, workshops and competitions.

The Vienna Summer of Logic continues a great legacy of scientific thought that started in Ancient Greece and flourished in the city of Gödel, Wittgenstein and the Vienna Circle. The heroes of our intellectual past shaped the scientific world-view and changed our understanding of science. Owing to their achievements, logic has permeated a wide range of disciplines, including computer science, mathematics, artificial intelligence, philosophy, linguistics, and many more. Logic is everywhere – or in the language of Aristotle, πάντα πλήρη λογικῆς τέχνης.

July 2014

Matthias Baaz
Thomas Eiter
Helmut Veith

Preface

This volume contains the papers presented at CAV 2014: International Conference on Computer Aided Verification held during July 18–22, 2014 in Vienna, Austria.

CAV 2014 was the 26th in a series dedicated to the advancement of the theory and practice of computer-aided formal analysis methods for hardware and software systems.

As part of the Federated Logic Conference (FLoC) and the Vienna Summer of Logic, CAV 2014 was collocated with many other conferences in logic. CAV considers it vital to continue spurring advances in hardware and software verification while expanding to new domains such as biological systems and computer security.

The conference covered the spectrum from theoretical results to concrete applications, with an emphasis on practical verification tools and the algorithms and techniques that are needed for their implementation. The proceedings of the conference will have been the Springer-Verlag Lecture Notes in Computer Science series. A selection of papers was invited to a special issue of Formal Methods in System Design and the Journal of the ACM.

There were 229 paper submissions, 175 regular papers and 54 short papers. Each submission was reviewed by at least three, and on average 4 Program Committee members. The Program Committee decided to accept 57 papers which is an acceptance rate of 25%, consisting of 46 regular papers (26%) and 11 short papers (20%).

There were 293 abstract submissions originally and a couple of papers, not included in the number of 229 papers above, immediately rejected due to excessively exceeding the page limit. Among the 172 rejected papers considered for computing the acceptance rate, there were 7 regular and one short paper withdrawn on behalf of the authors after first versions of reviews had been sent out during the rebuttal phase.

Regarding paper format, CAV 2014 saw some important changes compared to previous versions. Beside long papers, e.g., regular papers, there were also short papers, but short papers were not restricted to be tool papers anymore. It was also encouraged to submit high quality tool papers and empirical evaluations as long papers. These regular papers with mostly only empirical results, and not necessarily new theory, produced some reservations on the side of the reviewers and was an import topic during the discussion of the Program Committee. Further, references did not count towards the page limit.

Beside the presentations of the accepted papers, and shared FLoC sessions, the program of CAV 2014 featured two tutorials, two CAV invited talks, three competition presentations and last but not least the presentation of the CAV award.

The first tutorial was given by David Monniaux, Verimag, Grenoble, France, on "How Do We Get Inductive Invariants?" and the second tutorial by Fabio Somenzi, University of Colorado at Boulder, USA, on "Hardware Model Checking".

The first invited talk by Erik Winfree, Caltech, Pasadena, California, USA, had "Designing and Verifying Molecular Circuits and Systems Made of DNA" as the title. The second invited talk by Rance Cleaveland, University of Maryland and Fraunhofer, USA, discussed "Automated Testing".

The CAV 2014 affiliated competitions consisted of the first "Syntax-Guided Synthesis Competition", organized by Rajeev Alur, Dana Fisman, Rishabh Singh and Armando Solar-Lezama. Then there was the presentation of the results of the first Synthesis Competition for Reactive Systems "SYNTCOMP" organized by Swen Jacobs, Roderick Bloem, Rüdiger Ehlers and the 7th incarnation of the "Hardware Model Checking Competition", which was organized by Armin Biere and Keijo Heljanko.

The CAV award was presented by the CAV Award Committee, which consisted of Moshe Vardi, Ahmed Bouajjani, Tom Ball, and headed by Marta Kwiatkowska.

The FLoC 2014 Interconference Topics on Security and SAT/SMT/QBF are a FLoC 2014 initiative by CAV, CSF, and IJCAR to foster exchange and discussion between conferences. The Interconference Topics consist of sessions from the participating conferences with a joint thematic focus. They provide a special opportunity for FLoC participants with particular interest in these topics.

We would like to thank our workshop and competition chair Martina Seidl, for caring about 21 CAV workshops, including 4 workshops affiliated with other FLoC conferences too. As publication chair Swen Jacobs did an excellent job setting up the web-pages and producing the proceedings.

Of course without the tremendous effort put in the reviewing process by our Program Committee members this conference would not have been possible. We would further thank the Steering Committee for support and guidance during the whole conference process as well as Andrei Voronkov for providing the EasyChair service in general and excellent support during using EasyChair for CAV 2014.

May 2014

Armin Biere
Roderick Bloem

Organization

Program Chairs

Armin Biere	Johannes Kepler University of Linz, Austria
Roderick Bloem	Graz University of Technology, Austria

Program Committee

Rajeev Alur	University of Pennsylvania, USA
Domagoj Babic	Google, USA
Gogul Balakrishnan	University of Wisconsin, USA
Nikolaj Bjorner	Microsoft Research, USA
Ahmed Bouajjani	LIAFA, University Paris Diderot, France
Aaron Bradley	University of Colorado Boulder, USA
Pavol Cerny	University of Colorado Boulder, USA
Koen Claessen	Chalmers University of Technology, Sweden
Byron Cook	Microsoft Research, UK
Azadeh Farzan	University of Toronto, Canada
Bernd Finkbeiner	Saarland University, Germany
Jasmin Fisher	Microsoft Research, UK
Mike Gordon	University of Cambridge, UK
Orna Grumberg	Technion - Israel Institute of Technology, Israel
Leopold Haller	University of Oxford, UK
Keijo Heljanko	Aalto University, Finland
William Hung	Synopsys Inc, USA
Somesh Jha	University of Wisconsin, USA
Susmit Jha	Intel, USA
Barbara Jobstmann	EPFL, Jasper DA, CNRS-Verimag, Switzerland/France
Bengt Jonsson	Uppsala University, Sweden
Laura Kovacs	Chalmers University of Technology, Sweden
Daniel Kroening	University of Oxford, UK
Marta Kwiatkowska	University of Oxford, UK
Kim Guldstrand Larsen	Aalborg University, Denmark
Joao Marques-Silva	University College Dublin, Ireland
Kedar Namjoshi	Bell Labs, USA
Corina Pasareanu	CMU and NASA Ames Research Center, USA
Doron Peled	Bar Ilan University, Israel
Pavithra Prabhakar	IMDEA Software Institute, Spain
Jean-Francois Raskin	Université Libre de Bruxelles, Belgium

Koushik Sen	University of California Berkeley, USA
Natasha Sharygina	Università della Svizzera Italiana, Switzerland
Nishant Sinha	IBM Research, India
Anna Slobodova	Centaur Technology, USA
Fabio Somenzi	University of Colorado Boulder, USA
Cesare Tinelli	University of Iowa, USA
Thomas Wahl	Northeastern University, USA
Georg Weissenbacher	Vienna University of Technology, Austria
Eran Yahav	Technion - Israel Institute of Technology, Israel

Organization Committee

Swen Jacobs	Graz University of Technology, Austria
Martina Seidl	Johannes Kepler University Linz, Austria

Steering Committee

Michael Gordon	University of Cambridge, UK
Orna Grumberg	Technion - Israel Institute of Technology, Israel
Aarti Gupta	NEC Laboratories, USA
Kenneth McMillan	Microsoft Research, USA

Additional Reviewers

Abate, Alessandro
Abd Elkader, Karam
Adzkiya, Dieky
Albarghouthi, Aws
Alberti, Francesco
Alglave, Jade
Alt, Leonardo
Althoff, Matthias
Aronis, Stavros
Atig, Mohamed Faouzi
Avigad, Jeremy
Bacci, Giorgio
Bacci, Giovanni
Bandhakavi, Sruthi
Bansal, Kshitij
Basler, Gerard
Basset, Nicolas
Bastani, Osbert
Basu, Samik
Batty, Mark
Belov, Anton
Ben Sassi, Mohamed Amin
Bingham, Jesse
Birgmeier, Johannes
Blackshear, Sam
Bogomolov, Sergiy
Boker, Udi
Boldo, Sylvie
Botincan, Matko
Boudjadar, A. Jalil
Boulmé, Sylvain
Bozianu, Rodica
Brain, Martin
Brenguier, Romain
Brockschmidt, Marc
Bucheli, Samuel
Bultan, Tevfik
Cabodi, Gianpiero

Cassel, Sofia
Cassez, Franck
Ceska, Milan
Chadha, Rohit
Chakarov, Aleksandar
Chen, Hong-Yi
Chen, Hongyi
Chen, Xin
Chen, Yu-Fang
Chockler, Hana
Choi, Wontae
Chowdhury, Omar
Christodorescu, Mihai
Ciardo, Gianfranco
Clemente, Lorenzo
Cordeiro, Lucas
D'Amorim, Marcelo
D'Antoni, Loris
D'Silva, Vijay
D'Souza, Deepak
Daca, Przemyslaw
Dalsgaard, Andreas Engelbredt
Dang, Thao
David, Alexandre
David, Cristina
Davidson, Drew
Davis, Jared
De Carli, Lorenzo
De Moura, Leonardo
Dehnert, Christian
Delahaye, Benoit
Delaune, Stephanie
Delzanno, Giorgio
Dhawan, Mohan
Dillig, Isil
Dodds, Mike
Donaldson, Alastair
Doyen, Laurent
Drachsler, Dana
Dragoi, Cezara
Duret-Lutz, Alexandre
Een, Niklas
Emmi, Michael
Enea, Constantin
Falcone, Ylies
Faymonville, Peter
Fedyukovich, Grigory
Feng, Lu
Ferrara, Pietro
Filieri, Antonio
Filiot, Emmanuel
Fisman, Dana
Forejt, Vojtech
Fredrikson, Matt
Fränzle, Martin
Fu, Hongfei
Fuhs, Carsten
Furia, Carlo A.
Galenson, Joel
Ganai, Malay
Ganty, Pierre
Gerke, Michael
Girard, Antoine
Gopan, Denis
Greaves, David
Greenstreet, Mark
Griesmayer, Andreas
Griggio, Alberto
Groce, Alex
Grosu, Radu
Grundy, Jim
Gupta, Ashutosh
Gurfinkel, Arie
Haase, Christoph
Hadarean, Liana
Haddad, Axel
Hahn, Ernst Moritz
Hall, Ben
Hao, Kecheng
Harris, William
Hassan, Zyad
He, Fei
Hendriks, Martijn
Herbreteau, Frédéric
Hermanns, Holger
Hoare, Tony
Hochreiter, Sepp
Holik, Lukas
Holzer, Andreas
Hunter, Paul

Hyvärinen, Antti
Ignatiev, Alexey
Iosif, Radu
Itzhaky, Shachar
Ivancic, Franjo
Jacobs, Swen
Jaeger, Manfred
Janota, Mikolas
Jansen, Nils
Jha, Sumit Kumar
Jin, Hoon Sang
Johansson, Moa
Joshi, Saurabh
Jovanovic, Aleksandra
Kahlon, Vineet
Kahsai, Temesghen
Kannan, Jayanthkumar
Katz, Omer
Khlaaf, Heidy
Kiefer, Stefan
Kim, Chang Hwan Peter
Kim, Hyondeuk
Kincaid, Zachary
Klein, Felix
Komuravelli, Anvesh
Konev, Boris
Konnov, Igor
Kotek, Tomer
Kuismin, Tuomas
Kuncak, Viktor
Kupferman, Orna
Kupriyanov, Andrey
Kähkönen, Kari
Lal, Akash
Landsberg, David
Lee, Wonchan
Leino, Rustan
Leonardsson, Carl
Leue, Stefan
Liang, Tianyi
Liu, Jun
Liu, Lingyi
Liu, Peizun
Liu, Wanwei
Loginov, Alexey
Logozzo, Francesco
Lopes, Nuno
Lopes, Nuno P.
Luchaup, Daniel
Luckow, Kasper
Malik, Sharad
Mangal, Ravi
Manquinho, Vasco
Markey, Nicolas
Mauborgne, Laurent
McCamant, Stephen
McClurg, Jedidiah
Mereacre, Alexandru
Meshman, Yuri
Meyer, Roland
Mikucionis, Marius
Minea, Marius
Mitra, Sayan
Moarref, Salar
Monmege, Benjamin
Morgado, Antonio
Moses, Yoram
Mukherjee, Rajdeep
Mukund, Madhavan
Myers, Andrew
Nadel, Alexander
Natraj, Ashutosh
Navarro Perez, Juan Antonio
Navas, Jorge
Nghiem, Truong
Nickovic, Dejan
Niebert, Peter
Nimkar, Kaustubh
Norman, Gethin
Nyman, Ulrik
Olesen, Mads Chr.
Omari, Adi
Ozay, Necmiye
Pajic, Miroslav
Palikareva, Hristina
Paoletti, Nicola
Papavasileiou, Vasilis
Parker, David
Parkinson, Matthew
Parlato, Gennaro

Partush, Nimrod
Passmore, Grant
Pavlogiannis, Andreas
Payer, Mathias
Perez, Guillermo
Peter, Isabelle
Pichon, Jean
Piskac, Ruzica
Podelski, Andreas
Poetzl, Daniel
Prabhu, Prakash
Pradel, M.
Pradel, Michael
Putot, Sylvie
Qian, Kairong
Rabe, Markus N.
Radhakrishna, Arjun
Raghothaman, Mukund
Ramachandran, Jaideep
Raman, Vishwanath
Ranise, Silvio
Ratschan, Stefan
Ravanbakhsh, Hadi
Ray, Sandip
Rezine, Ahmed
Rezine, Othmane
Rinetzky, Noam
Rodriguez, Cesar
Rogalewicz, Adam
Rollini, Simone Fulvio
Rubio-Gonzalez, Cindy
Ruemmer, Philipp
Rybalchenko, Andrey
Saarikivi, Olli
Šafránek, David
Sagonas, Konstantinos
Saha, Indranil
Sanchez, Cesar
Sangnier, Arnaud
Sankaranarayanan, Sriram
Sarkar, Susmit
Schrammel, Peter
Sezgin, Ali
Shacham, Ohad
Sharma, Subodh
Sheinvald, Sarai
Shoham, Sharon
Siirtola, Antti Tapani
Singhania, Nimit
Sinha, Rohit
Sinha, Saurabh
Sorrentino, Francesco
Sosnovich, Adi
Srba, Jiri
Stainer, Amelie
Stenman, Jari
Stergiou, Christos
Sticksel, Christoph
Strichman, Ofer
Sturm, Thomas
Su, Kaile
Sznajder, Nathalie
Tarrach, Thorsten
Tasiran, Serdar
Tautschnig, Michael
Tentrup, Leander
Terauchi, Tachio
Thachuk, Chris
Tiwari, Ashish
Tonetta, Stefano
Topcu, Ufuk
Torfah, Hazem
Totla, Nishant
Trefler, Richard
Trinh, Cong Quy
Tripakis, Stavros
Trivedi, Ashutosh
Tschantz, Michael Carl
Tsiskaridze, Nestan
Tsitovich, Aliaksei
Turrini, Andrea
Udupa, Abhishek
Ujma, Mateusz
Urban, Caterina
Vafeiadis, Viktor
Vardi, Moshe
Veanes, Margus
Vechev, Martin
Villard, Jules
Viswanathan, Mahesh

Viswanathan, Ramesh
Vizel, Yakir
Von Essen, Christian
Wachter, Björn
Widder, Josef
Wieringa, Siert
Wiltsche, Clemens
Wintersteiger, Christoph
Wintersteiger, Christoph M.
Worrell, James
Wrigstad, Tobias
Xia, Bican
Xue, Bingtian
Yi, Wang
Zheng, Feijun
Zhou, Min
Zhu, Yunyun
Zielonka, Wieslaw
Zimmermann, Martin
Zwirchmayr, Jakob

Invited Tutorials and Talks

How Do We Get Inductive Invariants?

David Monniaux

CNRS, Verimag, Grenoble, France

Verifying the correctness of loop-free programs (or of general programs, up to bounded depth) is difficult: the state space explodes exponentially as the depth increases. Yet, the difficulty increases as we allow unboundedly many execution steps; proof approaches then generally rely on finding inductive invariants (properties shown to hold initially, then to remain true by induction).

Abstract interpretation attempts finding inductive invariants within a given domain, e.g. conjunctions of linear inequalities. The classical approach iterates a transformer until the property becomes inductive. In general, this approach may not terminate; thus termination is often enforced with a "widening" operator, which attempts at generalizing the iterates into an inductive property. Unfortunately, widening operators are brittle, with non-monotonic behaviors (supplying more information about a system may result in worse analysis outcomes!). Therefore, other approaches have been developed (policy iteration,...), which avoid this pitfall.

Finally, we shall discuss possible combinations of abstract interpretation and SMT-solving.

Hardware Model Checking

Fabio Somenzi

University of Colorado at Boulder, USA

This tutorial described the state-of-the art in Hardware-Model Checking using SAT and BDD-based techniques, including a discussion of the overall architecture of modern, multi-engine model checkers.

Designing and Verifying Molecular Circuits and Systems Made of DNA

Erik Winfree

California Institute of Technology, Pasadena, CA, USA

Inspired by the information processing core of biological organisms and its ability to fabricate intricate machinery from the molecular scale up to the macroscopic scale, research in synthetic biology, molecular programming, and nucleic acid nanotechnology aims to create information-based chemical systems that carry out human-defined molecular programs that input, output, and manipulate molecules and molecular structures. For chemistry to become the next information technology substrate, we will need improved tools for designing, simulating, and analyzing complex molecular circuits and systems. Using DNA nanotechnology as a model system, I will discuss how programming languages can be devised for specifying molecular systems at a high level, how compilers can translate such specifications into concrete molecular implementations, how both high-level and low-level specifications can be simulated and verified according to behavioral logic and the underlying biophysics of molecular interactions, and how Bayesian analysis techniques can be used to understand and predict the behavior of experimental systems that, at this point, still inevitably contain many ill-characterized components and interactions.

Automated Testing

Rance Cleaveland

University of Maryland, USA

In model-based testing, (semi-)formal models of systems are used to drive the derivation of test cases to be applied to the system-under-test (SUT). The technology has long been a part of the traditional hardware-design workflows, and it is beginning to find application in embedded-software development processes also. In automotive and land-vehicle control-system design in particular, models in languages such as MATLAB(r) / Simulink(r) / Stateflow(r) are used to drive the testing of the software used to control vehicle behavior, with tools like Reactis(r), developed by a team including the speaker, providing automated test-case generation support for this endeavor.

This talk will discuss how test-case generation capabilities may also be used to help verify that models meet formal specifications of their behavior. The method we advocate, Instrumentation-Based Verification (IBV), involves the formalizaton of behavior specifications as models that are used to instrument the model to be verified, and the use of coverage testing of the instrumented model to search for specification violations. The presentation will discuss the foundations of IBV, the test-generation approach and other features in Reactis that are used to support IBV, and the results of several case studies involving the use of the methods.

Competition Presentations

The First Syntax-Guided Synthesis Competition (SyGuS-COMP 2014)

Rajeev Alur[1], Dana Fisman[1], Rishabh Singh[2], and Armando Solar-Lezama[2,⋆]

[1] University of Pennsylvania
[2] Massachusetts Institute of Technology

Abstract. *Syntax-Guided Synthesis (SyGuS)* is the computational problem of finding an implementation f that meets both a semantic constraint given by a logical formula φ in a background theory T, and a syntactic constraint given by a grammar G, which specifies the allowed set of candidate implementations [1]. Such a synthesis problem can be formally defined in SyGuS-IF [2], a language that is built on top of SMT-LIB.

The *Syntax-Guided Synthesis Competition (SyGuS-COMP)* is an effort to facilitate, bring together and accelerate research and development of efficient solvers for SyGuS by providing a platform for evaluating different synthesis techniques on a comprehensive set of benchmarks. The benchmarks for the first competition are restricted to the theories of bitvector and integer linear arithmetic, yet their origin spans a variety of domains including bitvector algorithms, concurrency, robotics, and invariant generation. The solvers are scored primarily on the number of benchmark solved and the solving time, and secondarily on the succinctness of the synthesized solution.

References

1. Alur, R., Bodík, R., Juniwal, G., Martin, M.M.K., Raghothaman, M., Seshia, S.A., Singh, R., Solar-Lezama, A., Torlak, E., Udupa, A.: Syntax-Guided Synthesis. In: FMCAD, pp. 1–17. IEEE (2013)
2. Raghothaman, M., Udupa, A.: Language to Specify Syntax-Guided Synthesis Problems (May 2014), `http://arxiv.org/abs/1405.5590`

⋆ This research was supported by NSF Expeditions in Computing award CCF-1138996 and the competition awards were sponsored by Microsoft Research and FLoC.

Hardware Model Checking Competition CAV 2014 Edition

Armin Biere[1] and Keijo Heljanko[2]

[1] Johannes Kepler University Linz, Austria
[2] Aalto University, Finland

The results of the 7th International Hardware Model Checking Competition were presented at CAV 2014. Model checkers were required to produce witnesses for single safety properties. The traces were checked by the AIGSIM tool, which is part of the AIGER tools. Otherwise, the competition was run in almost the same way as in the previous two years. The competition was run on a cluster at Aalto University with exclusive access to 32 nodes of 2x Six-Core AMD Opteron 2435 2.6GHz with at least 16 GB of RAM. This meant 12 cores for each solver per benchmark, memory limit of 15 GB and time limit of 900 seconds. As fall back we had the cluster at JKU with the same characteristics as in previous years. Beside the requirement to produces witnesses, rules, input and output format did not change.

During the FLoC Olympic Games ceremony in the second week, three real silver medals were handed out to the winners of the three tracks of the competition. These three tracks in the CAV 2014 edition of the competitHardwaion consisted of: the single safety, the liveness and the deep bound track. There was no multiple property track for the CAV edition. It was considered to be the technically most challenging track, particularly while moving to new hardware, and further, only three medals were available. The winner of the deep bound track received both a medal and check, sponsored again by Oski technology. The competition further relied on support by the national research network on Rigorous System Engineering (RiSE) funded by the Austrian Science Fund (FWF) and also used resources made available through the Science-IT project at Aalto University.

SYNTCOMP - Synthesis Competition for Reactive Systems

Roderick Bloem[1], Rüdiger Ehlers[2,3], and Swen Jacobs[1]

[1] Graz University of Technology
Austria
[2] University of Bremen
Germany
[3] DFKI GmbH
Bremen, Germany

We present results of the first competition for reactive synthesis tools. For this first iteration, we focused on safety specifications, which are given as sequential circuits in an extension of the AIGER format for and-inverter graphs [1]. In the extended format, input signals of the circuit can be declared as controllable or uncontrollable. The synthesized implementation can read the uncontrollable input signals, and uses this information to drive the controllable signals such that the circuit never emits a **true** value at its error output signal.

The setting allows to encode a wide range of synthesis problems, and benchmarks for the first competition ranged from machine and robot controllers to hardware components like on-chip bus arbiters, and translations of LTL properties in general. Liveness properties are encodable in the form of *bounded* liveness properties, and the competition featured both benchmarks that make and that do not make use of this approach.

Tools were ranked with respect to the time needed for realizability checks and the size of circuits produced. Particular focus has been put on the verification of the synthesized implementations. Tools had to output the resulting controller in a format that is suitable as input to tools from the hardware model checking competition (HWMCC), i.e., as another and-inverter-graph circuit. Solutions had to be verifiable by current model checking tools in order to count for the competition. The competition also featured a track in which only the realizability of a specification needed to be checked, i.e., whether some controller exists or not.

SYNTCOMP was part of the FLoC 2014 Olympic Games and relied on support by the Austrian national research network on Rigorous Systems Engineering (RiSE), funded by the Austrian Science Fund (FWF).

Reference

1. Jacobs, S.: Extended AIGER Format for Synthesis. arXiv:1405.5793 (May 2014)

Table of Contents

Software Verification

Security

Automata

Model Checking and Testing

Biology and Hybrid Systems

Games and Synthesis

Concurrency

SMT and Theorem Proving

Bounds and Termination

Abstraction