More information about this series at http://www.springer.com/series/7407

VMCAI provides a forum for researchers from the communities of verification, model checking, and abstract interpretation, facilitating interaction, cross-fertilization, and advancement of hybrid methods that combine these and related areas. VMCAI topics include: program verification, model checking, abstract interpretation and abstract domains, program synthesis, static analysis, type systems, deductive methods, program certification, debugging techniques, program transformation, optimization, hybrid and cyber-physical systems.

This year the conference attracted 60 submissions. Each submission was reviewed by at least three Program Committee members. The committee decided to accept 27 papers. The principal selection criteria were relevance, quality, and originality. We are pleased to include in the proceedings the contributions of three invited keynote speakers: Ernie Cohen (Amazon Web Services), Pascal Cuoq (Trust in Soft), and Jasmin Fisher (Microsoft Research). We warmly thank them for their participation and for their contributions.

We would like also to thank the members of the Program Committee and the external reviewers for their excellent work. We also thanks the members of the Steering Committee, and in particular Andreas Podelski and Lenore Zuck, for their helpful advice, assistance, and support. We also thank Laure Gonnord for her invaluable help in all aspects related to the organization of the conference. We thank Annabel Satin for the help in coordinating the events co-located with POPL 2017, and we thank the POPL 2017 Organizing Committee for providing all the logistics for organizing VMCAI. We are also indebted to EasyChair for providing us with an excellent conference management system.

Finally, we would like to thank our generous sponsors: AdaCore, Amazon Web Services, Facebook, and Microsoft Research.

December 2016

Ahmed Bouajjani
David Monniaux
Organization

Program Committee

Erika Abraham             RWTH Aachen University, Germany
Mohamed Faouzi Atig      Uppsala University, Sweden
Roderick Bloem           Graz University of Technology, Austria
Ahmed Bouajjani          IRIF, Paris Diderot University, France
Wei-Ngan Chin            National University of Singapore, Singapore
Deepak D'Souza           Indian Institute of Science, Bangalore, India
Cezara Drăgoi             Inria, ENS, France
Roberto Giacobazzi       University of Verona, Italy
Laure Gonnord            University of Lyon/LIP, France
Oma Grumberg             Technion - Israel Institute of Technology, Israel
Dejan Jovanović          SRI International, USA
Konstantin Korovin       Manchester University, UK
Laura Kovacs             Vienna University of Technology, Austria
Shuvendu Lahiri          Microsoft Research, USA
Akash Lal                Microsoft Research, India
Rupak Majumdar           MPI-SWS, Germany
David Monniaux           VERIMAG, CNRS & Université Grenoble Alpes, France
Madhavan Mukund          Chennai Mathematical Institute, India
Corina Pasareanu         CMU/NASA Ames Research Center, USA
Andreas Podelski         University of Freiburg, Germany
Jean-Francois Raskin     Université Libre de Bruxelles, Belgium
Sriram Sankaranarayanan  University of Colorado, Boulder, USA
Armando Solar-Lezama     MIT, USA
Marielle Stoelinga       University of Twente, The Netherlands
Boris Yakobowski         CEA, LIST, France

Additional Reviewers

Basso-Blandin, Adrien     Costea, Andreea
Ben-Amram, Amir           Coti, Camille
Blom, Stefan              Darabi, Saeed
Bobot, François           Dehnert, Christian
Brain, Martin             Demange, Delphine
Braud-Santoni, Nicolas    Enea, Constantin
Cai, Zhouhong             Feret, Jerome
Castellan, Simon          Forget, Julien
Chakarov, Aleksandar     Frenkel, Hadar
Garg, Pranav
Ghilardi, Silvio
Girault, Alain
Gleiss, Bernhard
Habermehl, Peter
Hadarean, Liana
Halbwachs, Nicolas
He, Shaobo
Heßner, Alexander
Ho, Hsi-Ming
Iusupov, Rinat
Jansen, Nils
Jaroschek, Maximilian
Jecker, Ismaël
Khalimov, Ayrat
Koenighofer, Bettina
Konnov, Igor
Korovina, Margarita
Kremer, Gereon
Kretinsky, Jan
Lange, Tim
Le Roux, Stephane
Le, Quang Loc
Le, Ton Chanh
Lee, Benedict
Mastroeni, Isabella
Matteplackel, Raj Mohan
Merz, Stephan
Mukherjee, Suvam
Muoi, Tran Duc
Narayan Kumar, K.
Navas, Jorge A.
Ngo, Tuan Phong
Niksic, Filip
Petri, Gustavo
Rakamaric, Zvonimir
Rasin, Dan
Rensink, Arend
Rezine, Othmane
Rodriguez, Cesar
Roek, Franz
Rothenberg, Bat-Chen
Sangnier, Arnaud
Scherer, Gabriel
Schilling, Christian
Shi, Jinghao
Sofronie-Stokkermans, Viorica
Suda, Martin
Tiwari, Ashish
Urban, Caterina
van Glabbeek, Rob
Vedrine, Franck
Verdoolaege, Sven
Widder, Josef
Abstracts of Invited Talks
Bringing LTL Model Checking to Biologists

Zara Ahmed\textsuperscript{1}, David Benque\textsuperscript{2}, Sergey Berezin\textsuperscript{3}, Anna Caroline E. Dahl\textsuperscript{4}, Jasmin Fisher\textsuperscript{1,5}, Benjamin A. Hall\textsuperscript{6}, Samin Ishtiaq\textsuperscript{1}, Jay Nanavati\textsuperscript{1}, Nir Piterman\textsuperscript{7}, Maik Riechert\textsuperscript{1}, and Nikita Skoblov\textsuperscript{3}

\textsuperscript{1} Microsoft Research, Cambridge, UK
jasmin.fisher@microsoft.com
\textsuperscript{2} Royal College of Art, London, UK
\textsuperscript{3} Moscow State University, Moscow, Russia
\textsuperscript{4} Center for Technology in Medicine and Health, KTH Royal Institute of Technology, Huddinge, Sweden
\textsuperscript{5} Department of Biochemistry, University of Cambridge, Cambridge, UK
\textsuperscript{6} MRC Cancer Unit, University of Cambridge, Cambridge, UK
\textsuperscript{7} University of Leicester, Leicester, UK

Abstract. The BioModelAnalyzer (BMA) is a web based tool for the development of discrete models of biological systems. Through a graphical user interface, it allows rapid development of complex models of gene and protein interaction networks and stability analysis without requiring users to be proficient computer programmers. Whilst stability is a useful specification for testing many systems, testing temporal specifications in BMA presently requires the user to perform simulations. Here we describe the LTL module, which includes a graphical and natural language interfaces to testing LTL queries. The graphical interface allows for graphical construction of the queries and presents results visually in keeping with the current style of BMA. The Natural language interface complements the graphical interface by allowing a gentler introduction to formal logic and exposing educational resources.
Abstract. Modular code verification, suitably extended with shared atomic objects, supports a number of useful verification idioms and semantic models, without further logical extension.

Keywords: Real-time · Hybrid systems · Probability · Stopping failures · Weak memory · Cryptography · Ownership · Permissions · Simulation · Knowledge · Behavioral polymorphism · Device drivers · Concurrent data structures · Transactions · Linearizability · Deductive verification · VCC
Detecting Strict Aliasing Violations in the Wild

Pascal Cuoq¹, Loïc Runarvot¹, and Alexander Cherepanov²,³

¹ TrustInSoft, Paris, France
cuoq@trust-in-soft.com
² Openwall, Moscow, Russia
³ National Research University Higher School of Economics,
Moscow, Russia

Abstract. Type-based alias analyses allow C compilers to infer that memory locations of distinct types do not alias. Idiomatic reliance on pointers on the one hand, and separate compilation on the other hand, together make it impossible to get this aliasing information any other way. As a consequence, most modern optimizing C compilers implement some sort of type-based alias analysis. Unfortunately, pointer conversions, another pervasive idiom to achieve code reuse in C, can interact badly with type-based alias analyses. This article investigate the fine line between the allowable uses of low-level constructs (pointer conversions, unions) that should never cause the predictions of a standard-compliant type-based alias analysis to be wrong, and the dangerous uses that can result in bugs in the generated binary. A sound and precise analyzer for strict aliasing violations is briefly described.
## Contents

Bringing LTL Model Checking to Biologists .................................. 1  
*Zara Ahmed, David Benque, Sergey Berezin, Anna Caroline E. Dahl,  
Jasmin Fisher, Benjamin A. Hall, Samin Ishtiaq, Jay Nanavati,  
Nir Piterman, Maik Riechert, and Nikita Skoblov*

Detecting Strict Aliasing Violations in the Wild .......................... 14  
*Pascal Cuqo, Loïc Runarvot, and Alexander Cherepanov*

Effective Bug Finding in C Programs with Shape and Effect Abstractions ... 34  
*Iago Abal, Claus Brabrand, and Andrzej Wąsowski*

Synthesizing Non-Vacuous Systems ......................................... 55  
*Roderick Bloem, Hana Chockler, Masoud Ebrahimi, and Ofer Strichman*

Static Analysis of Communicating Processes Using Symbolic Transducers ... 73  
*Vincent Botbol, Emmanuel Chailloux, and Tristan Le Gall*

Reduction of Workflow Nets for Generalised Soundness Verification .... 91  
*Hadrien Bride, Olga Kouchnarenko, and Fabien Peureux*

Structuring Abstract Interpreters Through State and Value Abstractions ... 112  
*Sandrine Blazy, David Bühler, and Boris Yakobowski*

Matching Multiplications in Bit-Vector Formulas .......................... 131  
*Supratik Chakraborty, Ashutosh Gupta, and Rahul Jain*

Independence Abstractions and Models of Concurrency .................. 151  
*Vijay D’Silva, Daniel Kroening, and Marcelo Sousa*

Complete Abstractions and Subclassical Modal Logics .................... 169  
*Vijay D’Silva and Marcelo Sousa*

Using Abstract Interpretation to Correct Synchronization Faults .......... 187  
*Pietro Ferrara, Omer Tripp, Peng Liu, and Eric Koskinen*

Property Directed Reachability for Proving Absence of Concurrent Modification Errors .................................................. 209  
*Asya Frumkin, Yotam M.Y. Feldman, Ondřej Lhoták, Oded Padon,  
Mooly Sagiv, and Sharon Shoham*

Stabilizing Floating-Point Programs Using Provenance Analysis ........ 228  
*Yijia Gu and Thomas Wahl*
Dynamic Reductions for Model Checking Concurrent Software ............... 246
Henning Günther, Alfons Laarman, Ana Sokolova, and Georg Weissenbacher

Synthesising Strategy Improvement and Recursive Algorithms for Solving
2.5 Player Parity Games. ...................................................... 266
Ernst Moritz Hahn, Sven Schewe, Andrea Turrini, and Lijun Zhang

Counterexample Validation and Interpolation-Based Refinement
for Forest Automata .......................................................... 288
Lukáš Holík, Martin Hruška, Ondřej Lengál, Adam Rogalewicz, and Tomáš Vojnar

Block-Wise Abstract Interpretation by Combining Abstract Domains
with SMT ................................................................. 310
Jiahong Jiang, Liqian Chen, Xueguang Wu, and Ji Wang

Solving Nonlinear Integer Arithmetic with MCSAT. ......................... 330
Dejan Jovanović

Accuracy of Message Counting Abstraction in Fault-Tolerant
Distributed Algorithms. ..................................................... 347
Igor Konnov, Josef Widder, Francesco Spegni, and Luca Spalazzi

Efficient Elimination of Redundancies in Polyhedra by Raytracing ......... 367
Alexandre Maréchal and Michaël Péron

Precise Thread-Modular Abstract Interpretation of Concurrent Programs
Using Relational Interference Abstractions .................................. 386
Raphaël Monat and Antoine Miné

Detecting All High-Level Dataraces in an RTOS Kernel ...................... 405
Suvam Mukherjee, Arun Kumar, and Deepak D’Souza

Reachability for Dynamic Parametric Processes. .......................... 424
Anca Muscholl, Helmut Seidl, and Igor Walukiewicz

Conjunctive Abstract Interpretation Using Paramodulation ................. 442
Or Ozeri, Oded Padon, Noam Rinetzky, and Mooly Sagiv

Reasoning in the Bernays-Schönfinkel-Ramsey Fragment
of Separation Logic. ......................................................... 462
Andrew Reynolds, Radu Iosif, and Cristina Serban

Finding Relevant Templates via the Principal Component Analysis .......... 483
Yassamine Seladji

Sound Bit-Precise Numerical Domains. ..................................... 500
Tushar Sharma and Thomas Reps
IC3 - Flipping the E in ICE ......................................................... 521
Yakir Vizel, Arie Gurfinkel, Sharon Shoham, and Sharad Malik

Partitioned Memory Models for Program Analysis .................. 539
Wei Wang, Clark Barrett, and Thomas Wies

Author Index ................................................................. 559