Lecture Notes in Computer Science 9884

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/7407

Martin Fränzle · Nicolas Markey (Eds.)

Formal Modeling and Analysis of Timed Systems

14th International Conference, FORMATS 2016
Quebec, QC, Canada, August 24–26, 2016
Proceedings

Springer

Editors
Martin Fränzle
Carl von Ossietzky Universität
Oldenburg
Germany

Nicolas Markey
LSV, ENS de Cachan
Cachan Cedex
France

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-44877-0 ISBN 978-3-319-44878-7 (eBook)
DOI 10.1007/978-3-319-44878-7

Library of Congress Control Number: 2016948265

LNCS Sublibrary: SL1 – Theoretical Computer Science and General Issues

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG Switzerland

Preface

This volume of the *Lecture Notes in Computer Science* contains the papers presented at FORMATS 2016, the 14th edition of the International Conference on Formal Modeling and Analysis of Timed Systems, held during August 24–26, 2016, in Quebec City, Canada.

Control and analysis of the timing of computations are crucial to many domains of system engineering, be it, e.g., for ensuring a timely response to stimuli originating in an uncooperative environment or for synchronizing components in VLSI. Reflecting this broad scope, timing aspects of systems from a variety of domains have been treated independently by different communities in computer science and control. Researchers interested in semantics, verification, and performance analysis study models such as timed automata and timed Petri nets, the digital design community focuses on propagation and switching delays, while designers of embedded controllers have to take account of the time taken by controllers to compute their responses after sampling the environment, as well as of the dynamics of the controlled process during this span.

Timing-related questions in these separate disciplines have their particularities. However, there is growing awareness that there are basic problems that are common to all of them. In particular, all these subdisciplines treat systems whose behavior depends upon combinations of logical and temporal constraints; namely, constraints on the temporal distances between occurrences of events. Often, these constraints cannot be separated, as intrinsic dynamics of processes couples them, necessitating models, methods, and tools facilitating their combined analysis.

Reflecting this fact, the aim of FORMATS is to promote the study of fundamental and practical aspects of timed systems, and to bring together researchers from different disciplines that share interests in modeling and analysis of timed systems and, as a generalization, hybrid systems. Typical topics include (but are not limited to):

- Foundations and Semantics: Theoretical foundations of timed systems and languages; comparison between different models (such as timed automata, timed Petri nets, hybrid automata, timed process algebra, max-plus algebra, probabilistic models)
- Methods and Tools: Techniques, algorithms, data structures, and software tools for analyzing or synthesizing timed or hybrid systems and for resolving temporal constraints (e.g., scheduling, worst-case execution time analysis, optimization, model checking, testing, constraint solving)
- Applications: Adaptation and specialization of timing technology in application domains in which timing plays an important role (real-time software, embedded control, hardware circuits, and problems of scheduling in manufacturing and telecommunication, etc.)

FORMATS 2016 continued the tradition of the events previously held in Madrid (2015), Florence (2014), Buenos Aires (2013), London (2012), Aalborg (2011), Klosterneuburg (2010), Budapest (2009), St. Malo (2008), Salzburg (2007), Paris (2006), Uppsala (2005),

Grenoble (2004), and Marseille (2003). It was co-located with the 27th International Conference on Concurrency Theory (CONCUR 2016) and the 13th International Conference on Quantitative Evaluation of Systems (QEST 2016), sharing invited speakers and social events among these conferences, and with the workshops EXPRESS/SOS and TRENDS.

This year FORMATS received 32 full submissions by authors coming from 26 different countries. Each submission had full reviews from three Program Committee (PC) members and their sub-reviewers, plus additional comments from further PC members during an intense discussion phase. The committee finally selected 14 submissions for publication and presentation at the conference, which amounts to a 44 % acceptance rate. In addition, the conference included invited talks by:

- Scott A. Smolka, State University of New York, Stony Brook: "V-Formation as Optimal Control" (joint with Concur and QEST; abstract presented in the proceedings of CONCUR)
- Ufuk Topcu, University of Texas at Austin: "Adaptable yet Provably Correct Autonomous Systems" (joint with QEST, which also includes the abstract in its proceedings)
- Oleg Sokolsky, University of Pennsylvania: "Platform-Specific Code Generation from Platform-Independent Timed Models"

We sincerely thank the invited speakers for accepting our invitation and for providing abstracts of their talks for inclusion in the different proceedings volumes. We are grateful to the 27 PC members and their 33 associated reviewers for their competent and timely reviews of submissions, which were instrumental in securing the scientific standards of FORMATS. The EasyChair conference management system again provided reliable support in the submission phase, during the selection process, and while preparing this volume. We would also like to thank the Steering Committee of FORMATS for giving us the opportunity to put together this exciting event and for their support throughout this process, and Josée Desharnais of the University of Laval, Canada, for the competent and reliable local organization.

Last but not least, we are deeply grateful to all the authors for entrusting us with their papers. Thanks to their contributions we were able to put together the inspiring program reflected in these proceedings.

July 2016

Martin Fränzle
Nicolas Markey

Organization

Program Committee

S. Akshay	IIT Bombay, India
Hanifa Boucheneb	École Polytechnique de Montréal, Canada
Béatrice Bérard	LIP6, Université Pierre et Marie Curie, France
Laura Carnevali	University of Florence, Italy
Franck Cassez	Macquarie University, Australia
Martin Fränzle	University of Oldenburg, Germany
Gilles Geeraerts	Université libre de Bruxelles, Belgium
Michael R. Hansen	Technical University of Denmark
Boudewijn Haverkort	University of Twente, The Netherlands
Franjo Ivancic	Google, USA
Oded Maler	VERIMAG, CNRS, France
Nicolas Markey	LSV, CNRS, ENS Cachan, France
Chris Myers	University of Utah, USA
Jens Oehlerking	Robert Bosch GmbH, Germany
David Parker	University of Birmingham, UK
Karin Quaas	University of Leipzig, Germany
Stefan Ratschan	Czech Academy of Sciences, Czech Republic
Cesar Sanchez	IMDEA Software Institute, Spain
Sriram Sankaranarayanan	University of Colorado, Boulder, USA
Jeremy Sproston	University of Turin, Italy
Jirí Srba	Aalborg University, Denmark
Lothar Thiele	ETH Zurich, Switzerland
Enrico Vicario	University of Florence, Italy
Mahesh Viswanathan	University of Illinois, Urbana-Champaign, USA
James Worrell	University of Oxford, UK
Sergio Yovine	CONICET-UBA, Argentina
Huibiao Zhu	East China Normal University, China

Additional Reviewers

Basset, Nicolas
Bauer, Matthew S.
Berthomieu, Bernard
Bollig, Benedikt
Brenguier, Romain
Chen, Xin
Dang, Thao
Dubikhin, Vladimir
Frehse, Goran
Genest, Blaise
Gray, Travis
Ho, Hsi-Ming
Jensen, Peter Gjøl
Laroussinie, François

Mathur, Umang S.
Mikučionis, Marius
Muniz, Marco
Paolieri, Marco
Phawade, Ramchandra
Poplavko, Peter
Poulsen, Danny Bøgsted
Roohi, Nima
Ruijters, Enno
S., Krishna
Santinelli, Luca
Su, Wen
Sznajder, Nathalie
Ulus, Dogan
Van-Anh, Nguyen
Watanabe, Leandro
Wunderlich, Sascha
Xie, Wanling
Zhang, Zhen

Platform-Specific Code Generation from Platform-Independent Timed Models (Invited Keynote)

Oleg Sokolsky

Department of Computer and Information Science, University of Pennsylvania
sokolsky@cis.upenn.edu

Model-based implementation has emerged as an effective approach to systematically develop embedded software for real-time systems. Functional and timing behavior of the software is modeled using modeling languages with formal semantics. We then use formal verification techniques to demonstrate conformance of the model to the timing requirements for the system. Code generation then automatically generates source code from the verified model. The goal of this process is to guarantee that the final implemented system, running on an embedded platform, also conforms to the timing requirements. Several code generation frameworks have emerged, but they rely on restrictive assumptions regarding code execution by the underlying platform or require manual effort to integrate platform-independent code onto the platform. Both may undermine formal guarantees obtained in the course of model-based development.

In this talk, we consider the well-known four-variable model of system execution introduced by Parnas. The four-variable model makes a clear distinction between the external boundary of the system and internal boundary of the software. Timing requirements are typically verified on the external boundary, while generated code operates at the internal boundary of the system. This distinction can lead to a semantic gap between the verified model and generate code. We explore several complementary approaches to account for the distinction between the two boundaries. One approach composes the platform-independent model with a platform execution model for verification, but applies code generation to the platform-independent model only. Another approach uses integer linear programming to calculate a transformation of timing constants in the platform-independent model that keeps effects of platform delays on the occurrence of observable events in the generated code as small as possible.

This talk presents results of a collaboration with my colleagues BaekGyu Kim (currently at Toyota ITC), Insup Lee, Linh T.X. Phan, and Lu Feng. Material covered in this talk relies on ideas published in [1, 2].

References

1. Kim, B., Feng, L., Phan, L., Sokolsky, O., Lee, I.: Platform-specific timing verification framework in model-based implementation. In: Design, Automation and Test in Europe Conference and Exhibition (DATE 2015), pp. 235–240 (2015)
2. Kim, B., Feng, L., Sokolsky, O., Lee, I.: Platform-specific code generation from platform-independent timed models. In: IEEE Real-Time Systems Symposium (RTSS 2015), pp. 75–86 (2015)

Contents

Workload Analysis