Lecture Notes in Computer Science 10107

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/7409

Christian Berger · Mohammad Reza Mousavi
Rafael Wisniewski (Eds.)

Cyber Physical Systems

Design, Modeling, and Evaluation

6th International Workshop, CyPhy 2016
Pittsburgh, PA, USA, October 6, 2016
Revised Selected Papers

Springer

Editors
Christian Berger
University of Gothenburg
Gothenburg
Sweden

Rafael Wisniewski
Aalborg University
Aalborg
Denmark

Mohammad Reza Mousavi
Halmstad University
Halmstad
Sweden

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-51737-7 ISBN 978-3-319-51738-4 (eBook)
DOI 10.1007/978-3-319-51738-4

Library of Congress Control Number: 2017930211

LNCS Sublibrary: SL3 – Information Systems and Applications, incl. Internet/Web, and HCI

Printed on acid-free paper

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

Preface

Welcome to the proceedings of CyPhy 2016: the 6th International Workshop on Design, Modeling and Evaluation of Cyber Physical Systems, which was held on October 5, 2016, in Pittsburgh. This edition of CyPhy was held in conjunction with the Embedded Systems Week, which was organized during October 2–7, 2016, in Pittsburgh, USA.

For this edition, we received 14 submissions. All submission underwent a rigorous review process and each submission was reviewed by at least three, and on average more than four, Program Committee members. The committee decided to accept nine papers, which were presented in the workshop, and of which the revised versions appear in this proceedings volume.

In addition to the contributed papers and presentations, the program featured a keynote presentation by Dr. Jyotirmoy Deshmukh from Toyota. The keynote presentation, of which an abstract is included in this volume, skillfully integrated the scientific rigor of formal methods with the industrial complexity of cyber-physical systems in the automotive domain.

This was the sixth edition of CyPhy and we are glad to see that it has an established tradition and has found a stable place in the landscape of cyber-physical systems research venues.

We would like to gratefully acknowledge the effort of our distinguished Program Committee members for their extensive effort in reviewing papers and for helping us compose a high-quality program. We thank the additional reviewers for their review reports. We would like to thank the Steering Committee of CyPhy and its general chair, Walid Taha, for their help, support, and confidence.

We express our best thanks to Ferenc Bartha and Scott Hissam for having chaired the CyPhy 2016 sessions. We appreciate the valuable contribution of EasyChair and Springer in the seemless organization of the submission, review, and publication processes.

November 2016

Christian Berger
Mohammad Reza Mousavi
Rafael Wisniewski

Organization

Program Committee

Christian Berger	University of Gothenburg, Sweden
Manuela Bujorianu	University of Strathclyde, UK
Thao Dang	CNRS/VERIMAG, France
Scott Hissam	Software Engineering Institute, USA
Daisuke Ishii	University of Fukui, Japan
Mehdi Kargahi	University of Tehran, Iran
Zhiyun Lin	Zhejiang University, China
Mohammad Reza Mousavi	Halmstad University, Sweden
Enrico Pagello	University of Padua, Italy
Mihaly Petreczky	École des Mines de Douai, France
Michel Reniers	Eindhoven University of Technology, The Netherlands
Bernhard Rumpe	RWTH Aachen University, Germany
Maytham Safar	Kuwait University, Kuwait
Christoph Seidl	Technische Universität Braunschweig, Germany
Christoffer Sloth	Aalborg University, Denmark
Jonathan Sprinkle	University of Arizona, USA
Martin Steffen	University of Oslo, Norway
Frits Vaandrager	Radboud University Nijmegen, The Netherlands
Rafael Wisniewski	Aalborg University, Denmark

Additional Reviewers

Carraro, Marco
Chaki, Sagar
Eikermann, Robert
Lachmann, Remo
Schroeder, Jan
Schulze, Christoph
von Wenckstern, Michael

Formal Methods for Cyber-Physical Systems in the Automotive Domain (Extended Abstract)

Jyotirmoy Deshmukh

Toyota Technical Center, Gardena, CA, USA
jyotirmoy.deshmukh@toyota.com

Introduction

Systems where the behavior of a *physical* aspect of the system, such as that of a mechanical component is controlled using embedded software (*i.e.*, the "*cyber*" component) are called *cyber-physical systems*. A modern vehicle is an example of a complex cyber-physical system with burgeoning software size and complexity [2]. There are many exciting things on the horizon for the automotive domain, including advanced driver assist systems, self-driving cars, intelligent transportation systems, and alternative fuel sources. These advances can only further increase the complexity of embedded automotive software. Thus, it is imperative for the embedded software design process to recognize the challenges posed by increasing software complexity.

The problem of checking if all behaviors of a general cyber-physical system satisfy a behavioral property, for even a simple class of such properties is a very hard problem [3]. The *de facto* standard in industrial design, especially when faced with models such as those in [8], is to rely on rigorous testing, either at the level of system models or on the physical implementation of the system. However, a key challenge in such testing is that test scenarios and expected outcomes are often described (formally or informally) in natural language. Thus, engineers often rely on insight and experience to visually inspect test results to judge the performance of their designs. In what follows, we introduce a formal testing methodology that seeks to replace manual knowledge with machine-checkable requirements.

Requirement-Based Testing

Engineers often specify a scenario or setting for performing a test. These "conditions" are often specifications of allowable ranges for environmental factors (*e.g.*, ambient temperature, pressure, *etc.*), or patterns of driving behavior (*e.g.*, how often and how long a driver applies the brake). Then the engineers stimulate the system using an input signal satisfying the scenario specification and make a "judgement" about the output signal observed in relation to the applied input. This is analogous to the practice of specifying pre- and post-conditions on program behavior in the traditional literature on program verification. The key difference is that the pre- and post-conditions here can specify temporal behaviors of entire time-varying signals. Finding input signals satisfying arbitrary pre-conditions is generally challenging, but this problem can be mitigated by defining a parameterized input signal generator that produces a set of distinct input signals, all satisfying the given pre-condition. One approach to generate such signals is used by tools such as S-TaLiRo [1] and Breach [6], that use control points and a user-specified interpolation scheme to generate time-varying signals.

Post-conditions can often be reduced to designers looking for certain patterns in the output signals. Control engineers typically look for properties such as rise times, settling times, overshoots, undershoots, spikes/glitches, oscillatory behavior, and timed causal relations between signals. Several of these patterns can be elegantly expressed using Signal Temporal Logic (STL). Recently, we proposed a library called ST-Lib (Signal Template Library) that represents a subset of STL (and mild extensions) that can capture some of these signal patterns. Using STL or a similar real-time temporal logic has the advantage that it is often possible to define quantitative semantics for such logics. Such semantics map a given post-condition requirement and a trace to a real number. Without loss of generality, the semantics can be defined such that a positive number indicates that the trace satisfies the requirement, while a negative number indicates that the trace violates the requirement, and the spectrum of numbers from positive to negative indicate the degree of satisfaction or violation. This enables the use of global optimization-based techniques or other heuristic search techniques to be employed for automatic test generation and falsification of given system models [1, 3, 6, 7], as well as techniques to mine requirements from models [9, 10].

Conformance Testing

In the model-based development (MBD) paradigm, designers can have a variety of models differing in the level of detail, but representing the same underlying system. In such a setting, it is useful to have a technique to compare different models; model conformance is such a technique that seeks to provide quantitative notions of model similarity. Given a bound δ and a distance metric d on the space of signals, we say that two models are δ-conformant under the distance metric d, if for each input signal, stimulating the two models with this signal results in output signals less than δ distance apart (using the distance metric d to define distance). While several distance metrics have been defined in the literature, we consider the Skorokhod metric. This metric allows comparing signals both in time and value space [4], has efficient computational algorithms, and preserves the order of events in signals when comparing them. Recently, we presented a falsification-based algorithm that seeks to maximize the Skorokhod distance between two model outputs, and thus test models for conformance [4, 5].

Research Challenges

Below we enumerate some of the grand challenges for formal methods for cyber-physical systems in general, and for automotive systems in particular:

1. Modeling physical phenomena using high-fidelity models that can be efficiently simulated is a challenge. Physics-based parametric models have the disadvantage that they need careful tuning to match actual data. An alternative is to use data-driven models, but accuracy and interpretability continues to remain a concern.
2. Though specifying formal requirements with temporal logic has allowed us to make some strides in requirement elicitation, the general problem of specifying requirements continues to be a challenge. A key issue is that control designers often are not trained in temporal logic and prefer formalisms such as frequency-domain properties or statistical metrics. An ongoing challenge is to design a suitable language that allows designers to express all their desired requirements in an intuitive fashion, while being expressive enough.
3. Cyber-physical system designers are faced with a data deluge problem due to copious amounts of monitoring information available. A challenge is to provide tools that can expose intrinsic structure in massive amounts of time-series data, perform supervised learning and clustering, and algorithms for anomaly detection. A bigger challenge is to learn artifacts that are logically interpretable by designers, rather than black-box classifiers (that are typical in standard machine learning algorithms).

Conclusion. In this extended abstract, we present a few in-roads that techniques based on formal methods have been able to make in the domain of automotive cyber-physical systems. We suggest that a testing framework based on formalizing requirements using temporal logic has a higher degree of automation compared to traditional testing practices. We introduce the problem of conformance testing and conclude with some grand challenges.

Acknowledgements. The author would like to acknowledge his colleagues at Toyota including James Kapinski, Xiaoqing Jin, Hisahiro Ito, Jared Farnsworth, and Ken Butts, and co-authors on the papers cited in this paper.

References

1. Annapureddy, Y.S.R., Liu, C., Fainekos, G.E., Sankaranarayanan, S.: S-TaLiRo: a tool for temporal logic falsification for hybrid systems. In: Abdulla, P.A.A., Leino, K.R.M. (eds.) TACS 2011. LNCS 6605, pp. 254–257 (2011)
2. Charette, R.N.: This car runs on code. IEEE Spect. **46**(3), 3 (2009)
3. Deshmukh, J.V., Jin, X., Kapinski, J., Maler, O.: Stochastic local search for falsification of hybrid systems. In: Finkbeiner, B., Pu, G., Zhang, L. (eds.) ATVA 2015. LNCS 9364, pp. 500–517 (2015)
4. Deshmukh, J.V., Majumdar, R., Prabhu, V.: Quantifying conformance using the skorokhod metric. In: Kroening, D., Păsăreanu, C.S. (eds.) CAV 2015. LNCS 9207, pp. 234–250 (2015)
5. Deshmukh, J.V., Majumdar, R., Prabhu, V.: Quantifying conformance using the skorokhod metric. Formal Methods in System Design (accepted for publication) (2016)
6. Donzé, A.: Breach, a toolbox for verification and parameter synthesis of hybrid systems. In: Touili, T., Cook, B., Jackson, P. (eds.) CAV 2010. LNCS 6174, pp. 167–170 (2010)
7. Dreossi, T., Dang, T., Donzé, A., Kapinski, J., Jin, X., Deshmukh, J.V.: Efficient guiding strategies for testing of temporal properties of hybrid systems. In: Havelund, K., Holzmann, G., Joshi, R. (eds.) NFM 2015. LNCS 9058, pp. 127–142 (2015)

8. Jin, X., Deshmukh, J.V., Kapinski, J., Ueda, K., Butts, K.: Powertrain control verification benchmark. In: Proceedings of Hybrid Systems: Computation and Control, pp. 253–262 (2014)
9. Jin, X., Donzé, A., Deshmukh, J.V., Seshia, S.A.: Mining requirements from closed-loopcontrol models. In: Proceedings of Hybrid Systems: Computation and Control (2013)
10. Jin, X., Donzé, A., Deshmukh, J.V., Seshia, S.A.: Mining requirements from closed-loop control models. IEEE Trans. Comput. Aided Des. Integr. Circ. Syst. **34**(11), 1704–1717 (2015)

Contents