Lecture Notes in Computer Science 9421

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

Editorial Board

David Hutchison
Lancaster University, Lancaster, UK
Takeo Kanade
Carnegie Mellon University, Pittsburgh, PA, USA
Josef Kittler
University of Surrey, Guildford, UK
Jon M. Kleinberg
Cornell University, Ithaca, NY, USA
Friedemann Mattern
ETH Zurich, Zürich, Switzerland
John C. Mitchell
Stanford University, Stanford, CA, USA
Moni Naor
Weizmann Institute of Science, Rehovot, Israel
C. Pandu Rangan
Indian Institute of Technology, Madras, India
Bernhard Steffen
TU Dortmund University, Dortmund, Germany
Demetri Terzopoulos
University of California, Los Angeles, CA, USA
Doug Tygar
University of California, Berkeley, CA, USA
Gerhard Weikum
Max Planck Institute for Informatics, Saarbrücken, Germany

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

Thomas Hildebrandt · António Ravara
Jan Martijn van der Werf · Matthias Weidlich (Eds.)

Web Services, Formal Methods, and Behavioral Types

11th International Workshop, WS-FM 2014
Eindhoven, The Netherlands, September 11–12, 2014
and 12th International Workshop, WS-FM/BEAT 2015
Madrid, Spain, September 4–5, 2015
Revised Selected Papers

Springer

Editors
Thomas Hildebrandt
IT University of Copenhagen
Copenhagen
Denmark

António Ravara
Universidade NOVA de Lisboa
Caparica
Portugal

Jan Martijn van der Werf
Universiteit Utrecht
Utrecht
The Netherlands

Matthias Weidlich
Humboldt-Universität zu Berlin
Berlin
Germany

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-33611-4 ISBN 978-3-319-33612-1 (eBook)
DOI 10.1007/978-3-319-33612-1

Library of Congress Control Number: 2016936989

LNCS Sublibrary: SL2 – Programming and Software Engineering

© Springer International Publishing Switzerland 2016
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

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

Preface

Large software systems are becoming more and more distributed, collaborative, and communication-centered systems. Services in terms of functional and autonomous building blocks and the interactions between them have been established as fundamental concepts to design, implement, and deploy complex systems. Yet, independent of platforms and programming languages, formal methods play a key role in research on complex, service-based systems. They can help us to define unambiguous semantics for the languages that underpin existing infrastructures, facilitate consistency checking of interactions, empower dynamic discovery, and drive the analysis of security and performance properties of applications.

This volume contains the joint proceedings of two initiatives that have been devoted to the formal foundations of complex systems: the WS-FM:FASOCC 2014 and WS-FM/BEAT 2015 workshops.

The 11th International Workshop on Web Services and Formal Methods: Formal Aspects of Service-Oriented and Cloud Computing (WS-FM:FASOCC 2014) brought together researchers working on service-oriented computing, cloud computing, and formal methods in order to catalyze fruitful collaboration. It was part of the WS-FM workshop series that has a strong tradition of attracting submissions on formal approaches to enterprise systems modelling in general, and business process modelling in particular. Previous editions of the WS-FM workshop series took place in Pisa (2004), Versailles (2005), Vienna (2006), Brisbane (2007), Milan (2008), Bologna (2009), Hoboken (2010), Clermont-Ferrand (2011), Tallinn (2012), and Beijing (2013). WS-FM:FASOCC 2014 was planned to be held in Haifa, Israel, co-located with the 12th International Conference on Business Process Management (BPM 2014). However, the continuous uncertainty regarding the situation in southern Israel and Gaza led to a relocation of BPM 2014 and WS-FM:FASOCC 2014, so that the workshop took place September 11–12, 2014, in Eindhoven, The Netherlands.

In 2015, the WS-FM workshop and the International Workshop on Behavioural Types (BEAT) joined forces, resulting in the International Symposium on Web Services, Formal Methods and Behavioural Types (WS-FM/BEAT 2015). Both, WS-FM and BEAT, target the same research setting, i.e., large-scale behavioral software systems. The aim of this joint workshop event was to bring together researchers and practitioners in all aspects of behavioral software systems and their applications, in order to share results, consolidate the community, and discover opportunities for new collaborations and future directions. Previous editions of the BEAT workshop series took place in Lisbon (2011), Rome (2012 and 2014), and Madrid (2013). The first joint edition of the WS-FM and BEAT workshop series, WS-FM/BEAT 2015, took place September 4–5, 2015, in Madrid, Spain. It was part of the "MADRID MEET 2015 Meeting", which comprised a one-week scientific event with conferences and workshops in the areas of formal and quantitative analysis of systems, performance engineering, computer safety, and industrial critical applications. The program of the

second day of the symposium was shared with that of the 14th International Workshop on Foundations of Coordination Languages and Self-Adaptive Systems (FOCLASA).

The WS-FM:FASOCC 2014 program included keynotes by Giuseppe De Giacomo from the Sapienza Università di Roma, Italy, and Fabrizio Montesi from the University of Southern Denmark, and two sessions with research paper presentations. The WS-FM:FASOCC 2014 workshop attracted a total of 10 submissions, which were each reviewed by at least three members of the Program Committee. Eventually, the committee decided to accept four papers. Further, two of the best papers of the closely related 6th Central European Workshop on Services and Their Composition (ZEUS 2015) were invited to submit extended and revised versions for inclusion in the proceedings. After a review process with the WS-FM:FASOCC 2014 Program Committee, one of these papers was accepted.

The WS-FM/BEAT 2015 program featured keynotes by Cosimo Laneve from the Università di Bologna, Italy, and Javier Esparza from the Technische Universität München, Germany, and four sessions with research paper presentations (including four presentations selected from the submissions to WS-FM/BEAT). The WS-FM/BEAT 2015 workshop attracted a total of six submissions, which were each reviewed by at least three members of the Program Committee. Eventually, the committee decided to accept three papers to include in this volume.

We wish to thank the WS-FM:FASOCC 2014 and WS-FM/BEAT 2015 Program Committees and the external reviewers for their accurate and timely reviewing and acknowledge the support of EasyChair for managing the review process. Finally, we are grateful to the local organization teams of WS-FM:FASOCC 2014 in Haifa (Pnina Soffer, Nilly Schnapp, and others) and Eindhoven (Wil van der Aalst, Ine van der Ligt, and others) and of WS-FM/BEAT 2015 in Madrid (David de Frutos and others). They all did an excellent job in the preparation of the workshops — thanks a lot!

December 2015

Thomas Hildebrandt
Matthias Weidlich
António Ravara
Jan Martijn van der Werf

Organization

WS-FM:FASOCC 2014 Program Co-chairs

Thomas Hildebrandt	IT University of Copenhagen, Denmark
Matthias Weidlich	Imperial College London, UK

WS-FM:FASOCC 2014 Program Committee

Farhad Arbab	CWI and Leiden University, The Netherlands
Ahmed Awad	Cairo University, Egypt
Massimo Bartoletti	Università degli Studi di Cagliari, Italy
Laura Bocchi	Imperial College London, UK
Mario Bravetti	University of Bologna, Italy
Roberto Bruni	Università di Pisa, Italy
Marco Carbone	IT University of Copenhagen, Denmark
Erik De Vink	Technische Universiteit Eindhoven, The Netherlands
Marlon Dumas	University of Tartu, Estonia
Schahram Dustdar	TU Wien, Austria
Dirk Fahland	Technische Universiteit Eindhoven, The Netherlands
José Luiz Fiadeiro	Royal Holloway, University of London, UK
Roberto Guanciale	Università degli Studi di Pisa, Italy
Sylvain Hallé	Université du Québec à Chicoutimi, Canada
Ivan Lanese	University of Bologna/Inria, Italy/France
Niels Lohmann	Universität Rostock, Germany
Marco Montali	Free University of Bozen-Bolzano, Italy
Chun Ouyang	Queensland University of Technology, Australia
Artem Polyvyanyy	Queensland University of Technology, Australia
Alexandra Poulovassilis	Birkbeck College, University of London, UK
Rosario Pugliese	Università degli Studi di Firenze, Italy
António Ravara	Universidade Nova de Lisboa, Portugal
Tijs Slaats	IT University of Copenhagen, Denmark
Jianwen Su	University of California at Santa Barbara, USA
Maurice H. Ter Beek	ISTI-CNR Pisa, Italy
Hugo Torres Vieira	Universidade de Lisboa, Portugal
Emilio Tuosto	University of Leicester, UK
Wil van der Aalst	Technische Universiteit Eindhoven, The Netherlands
Lijie Wen	Tsinghua University, China
Martin Wirsing	Ludwig-Maximilians-Universität München, Germany
Karsten Wolf	Universität Rostock, Germany
Nobuko Yoshida	Imperial College London, UK
Gianluigi Zavattaro	University of Bologna, Italy

WS-FM:FASOCC 2014 Additional Reviewers

Søren Debois
Alexander Knapp
Julien Lange

WS-FM/BEAT 2015 Program Co-chairs

Antonio Ravara	Universidade Nova de Lisboa, Portugal
Jan Martijn van der Werf	Utrecht University, The Netherlands

WS-FM/BEAT 2015 Program Committee

Robin Bergenthum	FernUni Hagen, Germany
Laura Bocchi	University of Kent, UK
Sara Capecchi	Università degli Studi di Torino, Italy
Marlon Dumas	University of Tartu, Estonia
Adrian Francalanza	University of Malta, Malta
Thomas Hildebrandt	IT University of Copenhagen, Denmark
Jeroen Keiren	Open University, The Netherlands
Natallia Kokash	Leiden Institute of Advanced Computer Science, The Netherlands
Hernán Melgratti	Universidad de Buenos Aires, Argentina
Dimitris Mostrous	University of Lisbon, Portugal
Jovanka Pantovic	University of Novi Sad, Serbia
Artem Polyvyanyy	Queensland University of Technology, Australia
Natalia Sidorova	Eindhoven University of Technology, The Netherlands

WS-FM/BEAT 2015 Additional Reviewers

Natallia Kokash
Rumyana Neykova

WS-FM Steering Committee

Mario Bravetti	University of Bologna, Italy
Marlon Dumas	University of Tartu, Estonia
José Luiz Fiadeiro	Royal Holloway, University of London, UK
Wil van der Aalst	Technische Universiteit Eindhoven, The Netherlands
Gianluigi Zavattaro	University of Bologna, Italy

Abstracts of Invited Talks

Verification of Data-Aware Processes

Giuseppe De Giacomo

Sapienza Università di Roma, Rome, Italy
degiacomo@dis.uniromal.it

Keynote Abstract

Information systems are based on two pillars: data, which constitute the information asset of the organization, and business processes, which constitute its modus operandi. Traditionally these two aspects are considered, conceptualized, and formalized more or less in isolation. Such form of separation of concerns has been considered quite fruitful and it led to significant advances in both data and process management fields. However it has recently been questioned by the so called artifact-centric approach that advocates a holistic view of data and processes as a unity. In this talk, we will look at recent progresses in the analysis of processes that live side-by-side with data, both as first-class citizens. These systems are inherently infinite state and pose serious challenges to traditional verification techniques such as model checking.

Kickstarting Choreographic Programming

Fabrizio Montesi✉

University of Southern Denmark, Odense, Denmark
fmontesi@imada.sdu.dk

Abstract. We present an overview of some recent efforts aimed at the development of *Choreographic Programming*, a programming paradigm for the production of concurrent software that is guaranteed to be correct by construction from global descriptions of communication behaviour.

Static Analysis of Unbounded Networks with Behavioral Types

Cosimo Laneve

Department of Computer Science and Engineering,
University of Bologna – INRIA Focus, Bologna, Italy
cosimo.laneve@unibo.it

Keynote Abstract

The analysis of concurrent programs with infinite state models is extremely difficult due to the inability of statically reasoning about unbounded structures. As an example, consider those adaptive systems that, in order to reply to peaks of requests, create networks with arbitrary numbers of servers. In such systems, server interaction becomes complex and is hard to predict or to detect during testing. Additionally, even when possible, it can be tricky to reproduce bugs and find their causes. It turns out that, in these cases, the current analysers either return imprecise answers or do not scale.

This invited talk presents an analysis technique that have been used to verify properties such as deadlock freedom [5, 6], upper bounds of resource usages [3], and upper bounds of the computational cost [4] of programs that do not have a finite model. The proof technique is modular and consists of two parts: a type (inference) system that associates a *behavioural type* to a program and an *algorithm for analysing the behavioural types*.

Behavioural types are simple terms that feature recursion and resource creation – therefore their underlying model is infinite state – and express features of the programs that are relevant to the property one wants to analyze. For example, in case of deadlock analysis, behavioural types highlight resource dependencies; in case of resource analysis, they highlight resource usages; in case of computational cost analysis, they highlight the cost in time of instructions.

The behavioural type system typically performs standard abstractions, such as computing aliases and effects of updates, and its correctness is expressed in a standard way by means of a *subject reduction theorem*. In this setting, the subject-reduction states that: if (*i*) a program P is typable in an environment Γ with behavioural type $\mathbb{b}$ and (*ii*) P reduces to P' then there exists an environment Γ' that types P' with behavioural type $\mathbb{b}'$. It is worth to notice that the types $\mathbb{b}$ and $\mathbb{b}'$ are *different* because, in contrast to standard types, they change during the computation. Nevertheless, these changes are regulated by a relation, called *later-stage relation*, which specifies the correctness of the behavioural type analyzer (see below).

The analysis of behavioural types is performed either by ad-hoc algorithms – this is the case of deadlock analysis [5, 6] – or by automatic off-the-shelf solvers, whenever they are available – this is the case of resource and computational analysis [3, 4].

No matter what property is, analyzer's correctness is demonstrated by verifying that the analysis of behavioural types in the later stage relation, which include those types related by the subject-reduction theorem, always return identical values. That is, if one type has a deadlock (respectively, consumes at most n resources) then the other one in later-stage relation has a deadlock as well (respectively, consumes at most n resources as well).

Our techniques have been prototyped by taking a concurrent object-oriented language as reference language. The prototypes for deadlock analysis and resource analysis are available at [1, 2], respectively, while the prototype for computational cost analysis is under development.

A relevant advantage of the analysis technique presented in the talk derives from modularity. Because of modularity, the technique may be applied to several languages by simply changing the type system and support several behavioural type analysis algorithms.

References

1. Garcia, A., Giachino, E., Laneve, C., Lienhardt, M.: The deadlock framework for ABS (2014). df4abs.nws.cs.unibo.it
2. Garcia, A., Laneve, C.: Static analyzer of resource usage upper bounds (2015). sra.cs.unibo.it
3. Garcia, A., Laneve, C. Lienhardt, M.: Static analysis of cloud elasticity. In: Proceedings of PPDP 2015, pp. 125–136. ACM (2015)
4. Giachino, E., Johnsen, E. B., Laneve, C., Pun, K. I.: Time complexity of concurrent programs. In: Braga, C., Ölveczky, P.C. (eds.) FACS 2015. LNCS, vol. 9539, pp. 199–216. Springer, Berlin (2016)
5. Giachino, E., Kobayashi, N., Laneve, C.: Deadlock analysis of unbounded process networks. In: Baldan,. P., Gorla, D. (eds.) CONCUR 2014, vol. 8704, pp. 63–77. Springer, Berlin (2014)
6. Giachino, E., Laneve, C., Lienhardt, M.: A framework for deadlock detection in core ABS. Software and Systems Modeling, pp. 1–36 (2015)

A Petri-Net-like Model for Multiplayer Distributed Negotiations

Javier Esparza

Fakultät für Informatik, Technische Universität München, Germany
esparza@tum.de

Keynote Abstract

Many modern distributed systems consist of components whose behavior is only partially known. Typical examples include multi-agent systems, business processes, or protocols for conducting elections and auctions. An interaction between these components can be abstractly described as a negotiation in which several parties (the components involved in the negotiation) nondeterministically agree on an outcome, which results in a transformation of the internal states of the parties. In this talk we introduce *negotiations*, a formal model of concurrency close to Petri nets, with multiparty negotiation as primitive.

Negotiations can be ill designed: the parties can reach a deadlock or a livelock (a state from which the termination cannot successfully terminate anymore). In a *sound* negotiation this is not possible: from every reachable marking the final marking of the distributed negotiation can always be reached. A sound distributed negotiation has an equivalent one-step negotiation, called a *summary*. Loosely speaking, an external observer that only sees the initial and final states of the parties cannot distinguish a negotiation from its summary.

In the first part of the talk we study two problems: deciding whether a given negotiation is sound, and computing the summary of a given sound negotiation. We introduce *deterministic* negotiations, in which each participant can always be engaged in at most one next atomic negotiation. We show that, while both problems are untractable for arbitrary negotiations, there are efficient algorithms for the deterministic case. More precisely, we provide a complete set of *reduction rules* for deterministic negotiations. The rules reduce the negotiation to its summary iff the negotiation is sound. Further, the summary is computed after a polynomial number of rule applications.

In the second part of the talk we introduce *negotiation programs*, a global structured modelling language for negotiations, and show that it has the same expressive power as sound and deterministic negotiations: every program can be implemented by an equivalent sound and deterministic negotiation, and every sound and deterministic negotiation is modelled by an equivalent program. Here, a program and a negotiation are equivalent if they have the same Mazurkiewicz traces and thus the same concurrent runs.

The talk is based on joint work with Jörg Desel, published in [1–4].

References

1. Esparza, J., Desel, J.: On negotiation as concurrency primitive. In: D'Argenio, P.R., Melgratti, H.C. (eds.) CONCUR. LNCS, vol. 8052, pp. 440–454. Springer, Berlin (2013). Extended version in CoRR abs/1307.2145
2. Esparza, J., Desel, J.: On negotiation as concurrency primitive II: deterministic cyclic negotiations. In: Muscholl, A. (ed.) FoSSaCS. LNCS, vol. 8412, pp. 258–273. Springer, Berlin (2014). Extended version in CoRR abs/1403.4958
3. Esparza, J., Desel, J.: Negotiation programs. In: Devillers, R.R., Valmari, A. (eds.) PETRI NETS 2015. LNCS, vol. 9115, pp. 157–178. Springer, Berlin (2015)
4. Desel, J., Esparza, J.: Negotiations and Petri nets. In: Moldt, D., Rölke, H., Störrle, H. (eds.) Proceedings of the International Workshop on Petri Nets and Software Engineering. PNSE 2015, vol. 1372, pp. 41–57. CEUR Workshop Proceedings, CEUR-WS.org (2015)

Contents