

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison, UK

Josef Kittler, UK

Friedemann Mattern, Switzerland

Moni Naor, Israel

Bernhard Steffen, Germany

Doug Tygar, USA

Takeo Kanade, USA

Jon M. Kleinberg, USA

John C. Mitchell, USA

C. Pandu Rangan, India

Demetri Terzopoulos, USA

Gerhard Weikum, Germany

Advanced Research in Computing and Software Science

Subline of Lecture Notes in Computer Science

Subline Series Editors

Giorgio Ausiello, *University of Rome 'La Sapienza', Italy*

Vladimiro Sassone, *University of Southampton, UK*

Subline Advisory Board

Susanne Albers, *TU Munich, Germany*

Benjamin C. Pierce, *University of Pennsylvania, USA*

Bernhard Steffen, *University of Dortmund, Germany*

Deng Xiaotie, *City University of Hong Kong*


Jeannette M. Wing, *Microsoft Research, Redmond, WA, USA*

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

Amal Ahmed (Ed.)

Programming Languages and Systems

27th European Symposium on Programming, ESOP 2018
Held as Part of the European Joint Conferences
on Theory and Practice of Software, ETAPS 2018
Thessaloniki, Greece, April 14–20, 2018
Proceedings

Editor
Amal Ahmed 
Northeastern University
Boston, MA
USA



ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-89883-4 ISBN 978-3-319-89884-1 (eBook)
<https://doi.org/10.1007/978-3-319-89884-1>

Library of Congress Control Number: 2018940640

LNCS Sublibrary: SL1 – Theoretical Computer Science and General Issues

© The Editor(s) (if applicable) and The Author(s) 2018. This book is an open access publication.

Open Access This book is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this book are included in the book's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the book's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

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. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

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

ETAPS Foreword

Welcome to the proceedings of ETAPS 2018! After a somewhat coldish ETAPS 2017 in Uppsala in the north, ETAPS this year took place in Thessaloniki, Greece. I am happy to announce that this is the first ETAPS with gold open access proceedings. This means that all papers are accessible by anyone for free.

ETAPS 2018 was the 21st instance of the European Joint Conferences on Theory and Practice of Software. ETAPS is an annual federated conference established in 1998, and consists of five conferences: ESOP, FASE, FoSSaCS, TACAS, and POST. Each conference has its own Program Committee (PC) and its own Steering Committee. The conferences cover various aspects of software systems, ranging from theoretical computer science to foundations to programming language developments, analysis tools, formal approaches to software engineering, and security. Organizing these conferences in a coherent, highly synchronized conference program facilitates participation in an exciting event, offering attendees the possibility to meet many researchers working in different directions in the field, and to easily attend talks of different conferences. Before and after the main conference, numerous satellite workshops take place and attract many researchers from all over the globe.

ETAPS 2018 received 479 submissions in total, 144 of which were accepted, yielding an overall acceptance rate of 30%. I thank all the authors for their interest in ETAPS, all the reviewers for their peer reviewing efforts, the PC members for their contributions, and in particular the PC (co-)chairs for their hard work in running this entire intensive process. Last but not least, my congratulations to all authors of the accepted papers!

ETAPS 2018 was enriched by the unifying invited speaker Martin Abadi (Google Brain, USA) and the conference-specific invited speakers (FASE) Pamela Zave (AT & T Labs, USA), (POST) Benjamin C. Pierce (University of Pennsylvania, USA), and (ESOP) Derek Dreyer (Max Planck Institute for Software Systems, Germany). Invited tutorials were provided by Armin Biere (Johannes Kepler University, Linz, Austria) on modern SAT solving and Fabio Somenzi (University of Colorado, Boulder, USA) on hardware verification. My sincere thanks to all these speakers for their inspiring and interesting talks!

ETAPS 2018 took place in Thessaloniki, Greece, and was organised by the Department of Informatics of the Aristotle University of Thessaloniki. The university was founded in 1925 and currently has around 75,000 students; it is the largest university in Greece. ETAPS 2018 was further supported by the following associations and societies: ETAPS e.V., EATCS (European Association for Theoretical Computer Science), EAPLS (European Association for Programming Languages and Systems), and EASST (European Association of Software Science and Technology). The local organization team consisted of Panagiotis Katsaros (general chair), Ioannis Stamelos,

Lefteris Angelis, George Rahonis, Nick Bassiliades, Alexander Chatzigeorgiou, Ezio Bartocci, Simon Bliudze, Emmanouela Stachtari, Kyriakos Georgiadis, and Petros Stratis (EasyConferences).

The overall planning for ETAPS is the main responsibility of the Steering Committee, and in particular of its Executive Board. The ETAPS Steering Committee consists of an Executive Board and representatives of the individual ETAPS conferences, as well as representatives of EATCS, EAPLS, and EASST. The Executive Board consists of Gilles Barthe (Madrid), Holger Hermanns (Saarbrücken), Joost-Pieter Katoen (chair, Aachen and Twente), Gerald Lüttgen (Bamberg), Vladimiro Sassone (Southampton), Tarmo Uustalu (Tallinn), and Lenore Zuck (Chicago). Other members of the Steering Committee are: Wil van der Aalst (Aachen), Parosh Abdulla (Uppsala), Amal Ahmed (Boston), Christel Baier (Dresden), Lujo Bauer (Pittsburgh), Dirk Beyer (Munich), Mikolaj Bojanczyk (Warsaw), Luis Caires (Lisbon), Jurriaan Hage (Utrecht), Rainer Hähnle (Darmstadt), Reiko Heckel (Leicester), Marieke Huisman (Twente), Panagiotis Katsaros (Thessaloniki), Ralf Küsters (Stuttgart), Ugo Dal Lago (Bologna), Kim G. Larsen (Aalborg), Matteo Maffei (Vienna), Tiziana Margaria (Limerick), Flemming Nielson (Copenhagen), Catuscia Palamidessi (Palaiseau), Andrew M. Pitts (Cambridge), Alessandra Russo (London), Dave Sands (Göteborg), Don Sannella (Edinburgh), Andy Schürr (Darmstadt), Alex Simpson (Ljubljana), Gabriele Taentzer (Marburg), Peter Thiemann (Freiburg), Jan Vitek (Prague), Tomas Vojnar (Brno), and Lijun Zhang (Beijing).

I would like to take this opportunity to thank all speakers, attendees, organizers of the satellite workshops, and Springer for their support. I hope you all enjoy the proceedings of ETAPS 2018. Finally, a big thanks to Panagiotis and his local organization team for all their enormous efforts that led to a fantastic ETAPS in Thessaloniki!

February 2018

Joost-Pieter Katoen

Preface

This volume contains the papers presented at the 27th European Symposium on Programming (ESOP 2018) held April 16–19, 2018, in Thessaloniki, Greece. ESOP is one of the European Joint Conferences on Theory and Practice of Software (ETAPS). It is devoted to fundamental issues in the specification, design, analysis, and implementation of programming languages and systems.

The 36 papers in this volume were selected from 114 submissions based on originality and quality. Each submission was reviewed by three to six Program Committee (PC) members and external reviewers, with an average of 3.3 reviews per paper. Authors were given a chance to respond to these reviews during the rebuttal period from December 6 to 8, 2017. All submissions, reviews, and author responses were considered during the online discussion, which identified 74 submissions to be discussed further at the physical PC meeting held at Inria Paris, December 13–14, 2017. Each paper was assigned a guardian, who was responsible for making sure that external reviews were solicited if there was not enough non-conflicted expertise among the PC, and for presenting a summary of the reviews and author responses at the PC meeting. All non-conflicted PC members participated in the discussion of a paper’s merits. PC members wrote reactions to author responses, including summaries of online discussions and discussions during the physical PC meeting, so as to help the authors understand decisions. Papers co-authored by members of the PC were held to a higher standard and discussed toward the end of the physical PC meeting. There were ten such submissions and five were accepted. Papers for which the program chair had a conflict of interest were kindly handled by Fritz Henglein.

My sincere thanks to all who contributed to the success of the conference. This includes the authors who submitted papers for consideration; the external reviewers, who provided timely expert reviews, sometimes on short notice; and the PC, who worked hard to provide extensive reviews, engaged in high-quality discussions about the submissions, and added detailed comments to help authors understand the PC discussion and decisions. I am grateful to the past ESOP PC chairs, particularly Jan Vitek and Hongseok Yang, and to the ESOP SC chairs, Giuseppe Castagna and Peter Thiemann, who helped with numerous procedural matters. I would like to thank the ETAPS SC chair, Joost-Pieter Katoen, for his amazing work and his responsiveness. HotCRP was used to handle submissions and online discussion, and helped smoothly run the physical PC meeting. Finally, I would like to thank Cătălin Hrițcu for sponsoring the physical PC meeting through ERC grant SECOMP, Mathieu Mourey and the Inria Paris staff for their help organizing the meeting, and William Bowman for assisting with the PC meeting.

Organization

Program Committee

Amal Ahmed	Northeastern University, USA and Inria, France
Nick Benton	Facebook, UK
Josh Berdine	Facebook, UK
Viviana Bono	Università di Torino, Italy
Dominique Devriese	KU Leuven, Belgium
Marco Gaboardi	University at Buffalo, SUNY, USA
Roberto Giacobazzi	Università di Verona, Italy and IMDEA Software Institute, Spain
Philipp Haller	KTH Royal Institute of Technology, Sweden
Matthew Hammer	University of Colorado Boulder, USA
Fritz Henglein	University of Copenhagen, Denmark
Jan Hoffmann	Carnegie Mellon University, USA
Cătălin Hrițcu	Inria Paris, France
Suresh Jagannathan	Purdue University, USA
Limin Jia	Carnegie Mellon University, USA
Naoki Kobayashi	University of Tokyo, Japan
Xavier Leroy	Inria Paris, France
Aleksandar Nanevski	IMDEA Software Institute, Spain
Michael Norrish	Data61 and CSIRO, Australia
Andreas Rossberg	Google, Germany
Davide Sangiorgi	Università di Bologna, Italy and Inria, France
Peter Sewell	University of Cambridge, UK
Éric Tanter	University of Chile, Chile
Niki Vazou	University of Maryland, USA
Steve Zdancewic	University of Pennsylvania, USA

Additional Reviewers

Danel Ahman	Mariangiola Dezani
S. Akshay	Derek Dreyer
Aws Albarghouthi	Ronald Garcia
Jade Alglave	Deepak Garg
Vincenzo Arceri	Samir Genaim
Samik Basu	Victor Gomes
Gavin Bierman	Peter Habermehl
Filippo Bonchi	Matthew Hague
Thierry Coquand	Justin Hsu

Zhenjiang Hu
Peter Jipsen
Shin-ya Katsumata
Andrew Kennedy
Heidy Khlaaf
Neelakantan Krishnaswami
César Kunz
Ugo Dal Lago
Paul Levy
Kenji Maillard
Roman Manevich
Paulo Mateus
Antoine Miné
Stefan Monnier
Andrzej Murawski
Anders Møller
Vivek Notani

Andreas Nuyts
Paulo Oliva
Dominic Orchard
Luca Padovani
Brigitte Pientka
Benjamin C. Pierce
Andreas Podelski
Chris Poskitt
Francesco Ranzato
Andrey Rybalchenko
Sriram Sankaranarayanan
Tetsuya Sato
Sandro Stucki
Zachary Tatlock
Bernardo Toninho
Viktor Vafeiadis

RustBelt: Logical Foundations for the Future of Safe Systems Programming

Derek Dreyer

Max Planck Institute for Software Systems (MPI-SWS), Germany
dreyer@mpi-sws.org

Abstract. Rust is a new systems programming language, developed at Mozilla, that promises to overcome the seemingly fundamental tradeoff in language design between high-level safety guarantees and low-level control over resource management. Unfortunately, none of Rust’s safety claims have been formally proven, and there is good reason to question whether they actually hold. Specifically, Rust employs a strong, ownership-based type system, but then extends the expressive power of this core type system through libraries that internally use unsafe features.

In this talk, I will present RustBelt (<http://plv.mpi-sws.org/rustbelt>), the first formal (and machine-checked) safety proof for a language representing a realistic subset of Rust. Our proof is extensible in the sense that, for each new Rust library that uses unsafe features, we can say what verification condition it must satisfy in order for it to be deemed a safe extension to the language. We have carried out this verification for some of the most important libraries that are used throughout the Rust ecosystem.

After reviewing some essential features of the Rust language, I will describe the high-level structure of the RustBelt verification and then delve into detail about the secret weapon that makes RustBelt possible: the Iris framework for higher-order concurrent separation logic in Coq (<http://iris-project.org>). I will explain by example how Iris generalizes the expressive power of O’Hearn’s original concurrent separation logic in ways that are essential for verifying the safety of Rust libraries. I will not assume any prior familiarity with concurrent separation logic or Rust.

This is joint work with Ralf Jung, Jacques-Henri Jourdan, Robbert Krebbers, and the rest of the Iris team.

Contents

Language Design

Consistent Subtyping for All	3
<i>Ningning Xie, Xuan Bi, and Bruno C. d. S. Oliveira</i>	
HOBiT: Programming Lenses Without Using Lens Combinators	31
<i>Kazutaka Matsuda and Meng Wang</i>	
Dualizing Generalized Algebraic Data Types by Matrix Transposition	60
<i>Klaus Ostermann and Julian Jabs</i>	
Deterministic Concurrency: A Clock-Synchronised Shared Memory Approach	86
<i>Joaquín Aguado, Michael Mendler, Marc Pouzet, Partha Roop, and Reinhard von Hanxleden</i>	

Probabilistic Programming

An Assertion-Based Program Logic for Probabilistic Programs	117
<i>Gilles Barthe, Thomas Espitau, Marco Gaboardi, Benjamin Grégoire, Justin Hsu, and Pierre-Yves Strub</i>	
Fine-Grained Semantics for Probabilistic Programs	145
<i>Benjamin Bichsel, Timon Gehr, and Martin Vechev</i>	
How long, O Bayesian network, will I sample thee? A program analysis perspective on expected sampling times	186
<i>Kevin Batz, Benjamin Lucien Kaminski, Joost-Pieter Katoen, and Christoph Matheja</i>	
Relational Reasoning for Markov Chains in a Probabilistic Guarded Lambda Calculus	214
<i>Alejandro Aguirre, Gilles Barthe, Lars Birkedal, Aleš Bizjak, Marco Gaboardi, and Deepak Garg</i>	

Types and Effects

Failure is Not an Option: An Exceptional Type Theory	245
<i>Pierre-Marie Pédro and Nicolas Tabareau</i>	
Let Arguments Go First	272
<i>Ningning Xie and Bruno C. d. S. Oliveira</i>	

Behavioural Equivalence via Modalities for Algebraic Effects.	300
<i>Alex Simpson and Niels Voorneveld</i>	

Explicit Effect Subtyping.	327
<i>Amr Hany Saleh, Georgios Karachalias, Matija Pretnar, and Tom Schrijvers</i>	

Concurrency

A Separation Logic for a Promising Semantics.	357
<i>Kasper Svendsen, Jean Pichon-Pharabod, Marko Doko, Ori Lahav, and Viktor Vafeiadis</i>	

Logical Reasoning for Disjoint Permissions.	385
<i>Xuan-Bach Le and Aquinas Hobor</i>	

Deadlock-Free Monitors.	415
<i>Jafar Hamin and Bart Jacobs</i>	

Fragment Abstraction for Concurrent Shape Analysis.	442
<i>Parosh Aziz Abdulla, Bengt Jonsson, and Cong Quu Trinh</i>	

Security

Reasoning About a Machine with Local Capabilities: Provably Safe Stack and Return Pointer Management.	475
<i>Lau Skorstengaard, Dominique Devriese, and Lars Birkedal</i>	

Modular Product Programs.	502
<i>Marco Eilers, Peter Müller, and Samuel Hitz</i>	

Program Verification

A Fistful of Dollars: Formalizing Asymptotic Complexity Claims via Deductive Program Verification.	533
<i>Armaël Guéneau, Arthur Charguéraud, and François Pottier</i>	

Verified Learning Without Regret: From Algorithmic Game Theory to Distributed Systems with Mechanized Complexity Guarantees.	561
<i>Samuel Merten, Alexander Bagnall, and Gordon Stewart</i>	

Program Verification by Coinduction.	589
<i>Brandon Moore, Lucas Peña, and Grigore Rosu</i>	

Velisarios: Byzantine Fault-Tolerant Protocols Powered by Coq.	619
<i>Vincent Rahli, Ivana Vukotic, Marcus Völpl, and Paulo Esteves-Verissimo</i>	

Program Analysis and Automated Verification

Evaluating Design Tradeoffs in Numeric Static Analysis for Java	653
<i>Shiyi Wei, Piotr Mardziel, Andrew Ruef, Jeffrey S. Foster, and Michael Hicks</i>	
An Abstract Interpretation Framework for Input Data Usage.	683
<i>Caterina Urban and Peter Müller</i>	
Higher-Order Program Verification via HFL Model Checking.	711
<i>Naoki Kobayashi, Takeshi Tsukada, and Keiichi Watanabe</i>	
Quantitative Analysis of Smart Contracts	739
<i>Krishnendu Chatterjee, Amir Kafshdar Goharshady, and Yaron Velner</i>	

Session Types and Concurrency

Session-Typed Concurrent Contracts	771
<i>Hannah Gommerstadt, Limin Jia, and Frank Pfenning</i>	
A Typing Discipline for Statically Verified Crash Failure Handling in Distributed Systems.	799
<i>Malte Viering, Tzu-Chun Chen, Patrick Eugster, Raymond Hu, and Lukasz Ziarek</i>	
On Polymorphic Sessions and Functions: A Tale of Two (Fully Abstract) Encodings.	827
<i>Bernardo Toninho and Nobuko Yoshida</i>	
Concurrent Kleene Algebra: Free Model and Completeness	856
<i>Tobias Kappé, Paul Brunet, Alexandra Silva, and Fabio Zanasi</i>	

Concurrency and Distribution

Correctness of a Concurrent Object Collector for Actor Languages	885
<i>Juliana Franco, Sylvan Clebsch, Sophia Drossopoulou, Jan Vitek, and Tobias Wrigstad</i>	
Paxos Consensus, Deconstructed and Abstracted.	912
<i>Álvaro García-Pérez, Alexey Gotsman, Yuri Meshman, and Ilya Sergey</i>	
On Parallel Snapshot Isolation and Release/Acquire Consistency.	940
<i>Azalea Raad, Ori Lahav, and Viktor Vafeiadis</i>	
Eventual Consistency for CRDTs	968
<i>Radha Jagadeesan and James Riely</i>	

Compiler Verification

A Verified Compiler from Isabelle/HOL to CakeML	999
<i>Lars Hupel and Tobias Nipkow</i>	
Compositional Verification of Compiler Optimisations on Relaxed Memory	1027
<i>Mike Dodds, Mark Batty, and Alexey Gotsman</i>	
Author Index	1057