Lecture Notes in Computer Science 13474

More information about this series at https://link.springer.com/bookseries/558

Alicia Villanueva (Ed.)

Logic-Based Program Synthesis and Transformation

32nd International Symposium, LOPSTR 2022 Tbilisi, Georgia, September 21–23, 2022 Proceedings

Springer

Editor
Alicia Villanueva
Universitat Politècnica de València
Valencia, Spain

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-031-16766-9 ISBN 978-3-031-16767-6 (eBook)
https://doi.org/10.1007/978-3-031-16767-6

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

Preface

The International Symposium on Logic-based Program Synthesis and Transformation (LOPSTR) gathers researchers interested in logic-based program development. The LOPSTR series stimulates and promotes international research and collaboration in all aspects of this area, including all stages of the software life cycle and dealing with issues related to both programming-in-the-small and programming-in-the-large.

The 32nd edition of LOPSTR was held in the city of Tbilisi in Georgia as a hybrid (blended) meeting during September 21–23, 2022. It was organized in the context of the Computational Logic Autumn Summit (CLAS 2022), and co-located with the 24th International ACM SIGPLAN Symposium on Principles and Practice of Declarative Programming (PPDP 2022). Previous LOPSTR symposia were held in Tallin (2021, as a hybrid event), Bologna (2020, as a virtual meeting), Porto (2019), Frankfurt am Main (2018), Namur (2017), Edinburgh (2016), Siena (2015), Canterbury (2014), Madrid (2013 and 2002), Leuven (2012 and 1997), Odense (2011), Hagenberg (2010), Coimbra (2009), Valencia (2008), Lyngby (2007), Venice (2006 and 1999), London (2005 and 2000), Verona (2004), Uppsala (2003), Paphos (2001), Manchester (1998, 1992 and 1991), Stockholm (1996), Arnhem (1995), Pisa (1994), and Louvain-la-Neuve (1993). More information about the symposium can be found at https://lopstr2022.webs.upv.es/.

LOPSTR traditionally solicits contributions, in any programming language paradigm, in the areas of synthesis, specification, transformation, analysis and verification, specialization, testing and certification, composition, program and model manipulation, optimization, transformational techniques in software engineering, inversion, artificial intelligence methods for program development, verification and testing of AI-based systems, applications, and tools. LOPSTR has a reputation for being a lively, friendly forum that allows for the presentation and discussion of both finished work and work in progress.

In response to the call for papers, 18 contributions were submitted from authors in 17 different countries. One of the submissions was withdrawn by the authors, and each of the remaining submissions was reviewed by at least three Program Committee members or external referees. After two rounds of reviewing, eight papers were selected for inclusion in the formal proceedings and presented at the conference. Four additional papers were accepted for presentation at the symposium, leading to an attractive scientific program. In addition, the symposium program included invited talks by three outstanding speakers: Robert Hierons (University of Sheffield, UK) and two joint PPDP-LOPSTR speakers, Florian Zuleger (Technische Universität Wien, Austria), and Niki Vazou (IMDEA Software Institute, Spain). In addition to the eight accepted papers, this volume includes the abstracts of the invited talks.

I want to thank the Program Committee members, who worked diligently to produce high-quality reviews for the submitted papers, as well as all the external reviewers involved in the paper selection. LOPSTR 2022 was hosted and sponsored by the Ivane Javakhishvili Tbilisi State University and the Kurt Gödel Society. I am very grateful to the local organizer, Besik Dundua, and his team for the great job they did

in managing the CLAS 2022 event. Many thanks also to Manuel Hermenegildo and Beniamino Accattoli, the Program Committee chairs of PPDP, with whom I interacted for coordinating the events, and to the Steering Committee, chaired by Maurizio Proietti, for their support. I would also like to thank Andrei Voronkov for his excellent EasyChair conference management system that automates many of the tasks involved in chairing a conference. Special thanks go to the invited speakers and to all the authors who submitted and presented their papers at LOPSTR 2022. I also thank our sponsor, Springer, for the LOPSTR 2022 award and the cooperation and support in the organization of the symposium.

August 2022 Alicia Villanueva

Organization

Steering Committee

Emanuele De Angelis	National Research Council, Italy
Maribel Fernández	King's College London, UK
Fabio Fioravanti	Università degli Studi G. d'Annunzio, Italy
Maurizio Gabbrielli	University of Bologna, Italy
John Gallagher	Roskilde University, Denmark
Manuel Hermenegildo	IMDEA, Spain
Pedro López	IMDEA, Spain
Fred Mesnard	Université de la Réunion, France
Maurizio Proietti (Chair)	National Research Council, Italy
Peter Stuckey	Monash University, Australia
Wim Vanhoof	University of Namur, Belgium
Germán Vidal	Universitat Politècnica de València, Spain

Organizing Committee

Sedat Akleylek	Ondokuz Mayis University, Turkey
Sandra Alves	University of Porto, Portugal
Matthias Baaz	TU Wien, Austria
Besik Dundua (Chair)	Tbilisi State University/Kutaisi International University, Georgia
Santiago Escobar	Universitat Politècnica de València, Spain
Temur Kutsia	Johannes Kepler University Linz, Austria
Mircea Marin	West University of Timisoara, Romania
Konstantine Pkhakadze	Georgian Technical University, Georgia
Mikheil Rukhaia	Tbilisi State University, Georgia

Program Committee

Elvira Albert	Complutense University of Madrid, Spain
Roberto Amadini	University of Bologna, Italy
Emanuele De Angelis	National Research Council, Italy
Włodzimierz Drabent	IPI PAN, Poland/Linköping University, Sweden
Catherine Dubois	ENSIIE-Samovar, France
Fabio Fioravanti	Università degli Studi G. d'Annunzio, Italy
Gopal Gupta	University of Texas at Dallas, USA

Geoff Hamilton	Dublin City University, Ireland
Michael Hanus	Kiel University, Germany
Maja Kirkeby	Roskilde University, Denmark
Ekaterina Komendantskaya	Heriot-Watt University, UK
Temur Kutsia	Johannes Kepler University of Linz, Austria
Maria Chiara Meo	Università degli Studi G. d'Annunzio, Italy
Fred Mesnard	Université de la Réunion, France
Alberto Momigliano	University of Milan, Italy
Naoki Nishida	Nagoya University, Japan
Laura Panizo	University of Málaga, Spain
Laura Titolo	National Institute of Aerospace, USA
Wim Vanhoof	University of Namur, Belgium
Alicia Villanueva (Chair)	Universitat Politècnica de València, Spain

Additional Reviewers

Serdar Erbatur
Marco Antonio Feliú Gabaldón
Samir Genaim
Kenji Hashimoto
Abdel-Rahman Hedar
Alejandro Hernández-Cerezo
Michał Knapik
Clara Rodríguez-Núñez

Keynotes

Systematic Testing for Robotic Systems

Robert Hierons

The University of Sheffield, UK
r.hierons@sheffield.ac.uk

Robotic systems form the basis for advances in areas such as manufacturing, healthcare, and transport. A number of areas in which robotic systems are being used are safety-critical and so there is a need for software development processes that lead to robotic systems that are safe, reliable and trusted. Testing will inevitably be an important component.

We describe recent work on automated testing of robotic systems. The work is model-based: it takes as input a state-based model that describes the required behaviour of the system under test. Models are written in either RoboChart, a state-based language for robotics, or RoboSim, a simulation language for robotics. These languages have been given a formal semantics, making it possible to reason about models in a sound manner. We describe how the development of robotic software can be formalised based on such languages and how this can lead to the potential to automate the generation of sound test cases. Such test cases can be used for testing within a simulation and possibly also for testing the deployed system. Testing is systematic since test cases target potential faults.

Automated Termination and Complexity Analysis

Florian Zuleger

Technische Universität Wien, Austria
florian.zuleger@tuwien.ac.at

We overview two techniques that are suitable for automated termination and computational complexity analysis. 1) We are interested in abstract program models for which we are able to obtain decidability and expressivity results. Such program models can be used as backends in automated analyzers. Results will be presented on the size-change abstraction (SCA), which maintains only inequalities between sizes on the program state, and on vector addition systems with states (VASS), which are an equivalent representation of Petri nets with finite state. 2) Building on a line of previous work, we present an approach based on potential function templates with unknown coefficients. The analysis is stated as a type-and-effect system where the typing rules generate constraints over the unknown coefficients. Our work targets the performance analysis of self-adjusting data structures such as (randomized) splay trees, which requires sophisticated potential functions that include logarithmic expressions.

Contents