

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/7407>

Maurizio Proietti · Hirohisa Seki (Eds.)

Logic-Based Program Synthesis and Transformation

24th International Symposium, LOPSTR 2014
Canterbury, UK, September 9–11, 2014
Revised Selected Papers

Editors
Maurizio Proietti
IASI-CNR
Rome
Italy

Hirohisa Seki
Nagoya Institute of Technology
Nagoya
Japan

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-319-17821-9 ISBN 978-3-319-17822-6 (eBook)
DOI 10.1007/978-3-319-17822-6

Library of Congress Control Number: 2015937958

LNCS Sublibrary: SL1 – Theoretical Computer Science and General Issues

Springer Cham Heidelberg New York Dordrecht London
© Springer International Publishing Switzerland 2015

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

Springer International Publishing AG Switzerland is part of Springer Science+Business Media
(www.springer.com)

Preface

This volume contains a selection of the papers presented at LOPSTR 2014, the 24th International Symposium on Logic-Based Program Synthesis and Transformation held during September 9–11, 2014 at the University of Kent, Canterbury, UK. It was colocated with PPDP 2014, the 16th International ACM SIGPLAN Symposium on Principles and Practice of Declarative Programming.

Previous LOPSTR symposia were held in 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: <http://www.iasi.cnr.it/events/lopstr14/>.

The aim of the LOPSTR series is to stimulate and promote international research and collaboration on logic-based program development. LOPSTR is open to contributions in all aspects of logic-based program development, all stages of the software life cycle, and issues of both programming-in-the-small and programming-in-the-large. LOPSTR traditionally solicits contributions, in any language paradigm, in the areas of synthesis, specification, transformation, analysis and verification, specialization, testing and certification, composition, program/model manipulation, optimization, transformational techniques in software engineering, inversion, applications, and tools. LOPSTR has a reputation for being a lively, friendly forum for presenting and discussing work in progress. Formal proceedings are produced only after the symposium so that authors can incorporate this feedback in the published papers.

In response to the call for papers, 34 contributions were submitted from 21 different countries. The Program Committee accepted 7 full papers for immediate inclusion in the formal proceedings, and 11 more papers presented at the symposium were accepted after a revision and another round of reviewing. Each submission was reviewed by at least 2 and on the average 3.0, Program Committee members or external referees. In addition to the 18 contributed papers, this volume includes the abstracts of the invited talks by two outstanding speakers: Roberto Giacobazzi (University of Verona, Italy), shared with PPDP and Viktor Kuncak (EPFL, Switzerland).

We would like 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. We are very grateful to the LOPSTR 2014 General Co-chairs, Olaf Chitil and Andy King, and the local organizers for the great job they did in managing the symposium. Many thanks also to Olivier Danvy, the Program Committee Chair of PPDP, with whom we often interacted for coordinating the two events. We are grateful to Emanuele De Angelis and Fabrizio Smith, who helped us in maintaining the LOPSTR web site and editing these proceedings. We would also like to thank Andrei Voronkov for his excellent EasyChair system that automates many of the tasks involved in chairing a conference. Special thanks go to all

the authors who submitted and presented their papers at LOPSTR 2014, without whom the symposium would have not been possible. Finally, Maurizio Proietti gratefully acknowledges financial support from the Italian National Group of Computing Science (GNCS-INDAM).

February 2015

Maurizio Proietti
Hirohisa Seki

Organization

Program Committee

Slim Abdennadher	German University of Cairo, Egypt
Étienne André	Université Paris 13, France
Martin Brain	University of Oxford, UK
Wei-Ngan Chin	National University of Singapore, Singapore
Marco Comini	University of Udine, Italy
Włodek Drabent	IPI PAN Warszawa, Poland and Linköping University, Sweden
Fabio Fioravanti	University of Chieti-Pescara, Italy
Jürgen Giesl	RWTH Aachen, Germany
Miguel Gómez-Zamalloa	Complutense University of Madrid, Spain
Arnaud Gotlieb	SIMULA Research Laboratory, Norway
Gopal Gupta	University of Texas at Dallas, USA
Jacob Howe	City University London, UK
Zhenjiang Hu	National Institute of Informatics, Japan
Alexei Lisitsa	University of Liverpool, UK
Yanhong A. Liu	State University of New York at Stony Brook, USA
Jorge A. Navas	NASA Ames Research Center, USA
Naoki Nishida	Nagoya University, Japan
Corneliu Popeea	Technische Universität München, Germany
Maurizio Proietti	IASI-CNR, Rome, Italy (Co-chair)
Tom Schrijvers	Ghent University, Belgium
Hirohisa Seki	Nagoya Institute of Technology, Japan (Co-chair)
Jon Sneyers	Katholieke Universiteit, Leuven, Belgium
Fausto Spoto	University of Verona, Italy
Wim Vanhoof	University of Namur, Belgium
Germán Vidal	Universitat Politècnica de València, Spain

General Co-chairs

Olaf Chitil	University of Kent, UK
Andy King	University of Kent, UK

Organizing Committee

Emanuele De Angelis	IASI-CNR, Rome, Italy
Fabrizio Smith	IASI-CNR, Rome, Italy

Additional Reviewers

Bardin, Sebastien

Bucheli, Samuel

Cai, Zhouhong

Choppy, Christine

Cirstea, Horatiu

Di Gianantonio, Pietro

Emoto, Kento

Englebert, Vincent

Faber, Wolfgang

Fuhs, Carsten

Guo, Hai-Feng

Gutiérrez, Raúl

Haemmerlé, Rémy

Ieva, Carlo

Inuzuka, Nobuhiro

Ismail, Haythem

Kawabe, Yoshinobu

King, Andy

Komendantskaya,

Ekaterina

Lenisa, Marina

Li, Jun

Lovato, Alberto

López-Fraguas, Francisco

Javier

Marple, Kyle

Morihata, Akimasa

Narayanaswamy, Ganesh

Nishimura, Susumu

Pettorossi, Alberto

Salazar, Elmer

Ströder, Thomas

Tan, Tian Huat

Titolo, Laura

Yue, Tao

Zaki, Amira

Obscuring Code

Unveiling and Veiling Information in Programs¹

Roberto Giacobazzi

University of Verona, Verona, Italy
roberto.giacobazzi@univr.it

Abstract. We survey the most recent developments in code obfuscation and protection from a programming languages perspective. Starting from known impossibility results on universal and general purpose code obfuscation, we show that provably secure obfuscation can be achieved by constraining the attack model. This corresponds to associate attacks with suitable forms of interpretation. In this context it is always possible to systematically making code obscure, making this interpretation failing in extracting (attacking) code. The code transformation can itself be specified as the specialization of a distorted interpreter.

¹ An extended version appears in the proceedings of the 16th International Symposium on Principles and Practice of Declarative Programming (PPDP 2014), September 8–10 2014, Canterbury, United Kingdom. ACM Press.

Synthesizing Functions from Relations in Leon

Viktor Kuncak², Etienne Kneuss, and Emmanouil Koukoutos

École Polytechnique Fédérale de Lausanne (EPFL),

Lausanne, Switzerland

`viktor.kuncak@epfl.ch`

Abstract. We present the synthesis functionality of the Leon system (leon.epfl.ch). Leon accepts a purely functional subset of Scala extended with a choice construct. We describe automated and manual synthesis and transformation techniques in Leon, which can eliminate the choice construct and thus transform input/output relation specifications into executable functions from inputs to outputs. The techniques employed include functional synthesis procedures for decidable theories such as term algebras and Presburger arithmetic, synthesis proof rules for decomposing specifications, as well as search-based techniques, such as counterexample-guided synthesis.

² This work is supported in part by the European Research Council (ERC) Project *Implicit Programming*.

Contents

Program Analysis and Transformation

- Analyzing Array Manipulating Programs by Program Transformation 3
*J. Robert M. Cornish, Graeme Gange, Jorge A. Navas,
Peter Schachte, Harald Søndergaard, and Peter J. Stuckey*
- Analysing and Compiling Coroutines with Abstract Conjunctive
Partial Deduction 21
Danny De Schreye, Vincent Nys, and Colin Nicholson

Constraint Handling Rules

- Confluence Modulo Equivalence in Constraint Handling Rules 41
Henning Christiansen and Maja H. Kirkeby
- Exhaustive Execution of CHR Through Source-to-Source Transformation . . . 59
Ahmed Elsayy, Amira Zaki, and Slim Abdennadher
- A Formal Semantics for the Cognitive Architecture ACT-R 74
Daniel Gall and Thom Frühwirth
- CHRAnimation: An Animation Tool for Constraint Handling Rules 92
Nada Sharaf, Slim Abdennadher, and Thom Frühwirth

Termination Analysis

- Extending the 2D Dependency Pair Framework for Conditional
Term Rewriting Systems 113
Salvador Lucas, José Meseguer, and Raúl Gutiérrez

Security

- Partial Evaluation for Java Malware Detection 133
Ranjeet Singh and Andy King
- Access Control and Obligations in the Category-Based Metamodel:
A Rewrite-Based Semantics 148
Sandra Alves, Anatoli Degtyarev, and Maribel Fernández

Program Testing and Verification

Concolic Execution and Test Case Generation in Prolog 167
 Germán Vidal

Liveness Properties in CafeOBJ – A Case Study for Meta-Level Specifications 182
 Norbert Preining, Kazuhiro Ogata, and Kokichi Futatsugi

Program Synthesis

A Hybrid Method for the Verification and Synthesis of Parameterized Self-Stabilizing Protocols 201
 Amer Tahat and Ali Ebneenasir

Drill and Join: A Method for Exact Inductive Program Synthesis 219
 Remis Balaniuk

Program Derivation

Functional Kleene Closures 241
 Nikita Danilenko

Semantic Issues in Logic Programming

On Completeness of Logic Programs. 261
 Włodzimierz Drabent

Polynomial Approximation to Well-Founded Semantics for Logic Programs with Generalized Atoms: Case Studies. 279
 Md. Solimul Chowdhury, Fangfang Liu, Wu Chen, Arash Karimi, and Jia-Huai You

Program Transformation and Optimization

Declarative Compilation for Constraint Logic Programming 299
 Emilio Jesús Gallego Arias, James Lipton, and Julio Mariño

Pre-indexed Terms for Prolog. 317
 J.F. Morales and M. Hermenegildo

Author Index 333