

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Alfred Kobsa

University of California, Irvine, CA, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

TU Dortmund University, Germany

Madhu Sudan

Microsoft Research, Cambridge, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max Planck Institute for Informatics, Saarbruecken, Germany

Sarfraz Khurshid Koushik Sen (Eds.)

Runtime Verification

Second International Conference, RV 2011
San Francisco, CA, USA, September 27-30, 2011
Revised Selected Papers

Volume Editors

Sarfraz Khurshid
The University of Texas at Austin
Electrical and Computer Engineering
1 University Station C5000
Austin, TX 78712-0240, USA
E-mail: khurshid@ece.utexas.edu

Koushik Sen
University of California
Department of Electrical Engineering
and Computer Sciences
581 Soda Hall #1776
Berkeley, CA 94720-1776, USA
E-mail: ksen@cs.berkeley.edu

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-29859-2 e-ISBN 978-3-642-29860-8
DOI 10.1007/978-3-642-29860-8
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2012937468

CR Subject Classification (1998): D.2, F.2, D.2.4, D.1, F.3, D.3, C.2

LNCS Sublibrary: SL 2 – Programming and Software Engineering

© Springer-Verlag Berlin Heidelberg 2012

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

The use of general descriptive names, registered names, trademarks, 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.

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The Second International Conference on Runtime Verification (RV 2011) was held in the historic Fairmont Hotel in San Francisco, California, USA, during September 27–30, 2011. The conference program included invited talks, peer-reviewed presentations and tool demonstrations, as well as tutorials.

RV 2011 was attended by researchers and industrial practitioners from all over the world. It provided a forum to present foundational theories and practical tools for monitoring and analysis of software or hardware system executions, as well as a forum for presenting applications of such tools to real-world problems. The field of runtime verification is often referred to under different names, such as runtime verification, runtime monitoring, runtime checking, runtime reflection, runtime analysis, dynamic analysis, runtime symbolic analysis, trace analysis, log file analysis, etc. Runtime verification can be used for many purposes, such as security or safety policy monitoring, debugging, testing, verification, validation, profiling, fault protection, behavior modification (e.g., recovery), etc. A running system can be abstractly regarded as a generator of execution traces, i.e., sequences of relevant states or events. Traces can be processed in various ways, e.g., checked against formal specifications, analyzed with special algorithms, visualized, etc. Runtime verification now has a number of sub-fields, for example, program instrumentation, specification languages for writing monitors, dynamic concurrency analysis, intrusion detection, dynamic specification mining, and program execution visualization. Additionally, techniques for runtime verification have strong connections to techniques in other related fields such as combined static and dynamic analysis, aspect-oriented programming, and model-based testing. This year's conference included, in addition to papers that advance analyses commonly used for runtime verification, papers on symbolic execution, a well-known program analysis technique, which so far has not seen much use in this field but holds promise in enabling novel approaches to runtime verification.

The Runtime Verification series of events started in 2001, as an annual workshop. The workshop series continued through 2009. Each workshop was organized as a satellite event to an established forum, including CAV (2001–2003, 2005–2006, and 2009), ETAPS (2004 and 2008), and AoSD (2007). The RV 2006 workshop was organized jointly with the Formal Aspects of Testing workshop. The proceedings for RV from 2001 to 2005 were published in *Electronic Notes in Theoretical Computer Science*. Since 2006, the RV proceedings have been published in *Lecture Notes in Computer Science*.

Starting with the year 2010, RV became an international conference to recognize the sense of community that had emerged and the maturity the field had reached over the decade since the inception of the series. Broadening the scope of the event to a conference allowed further enlarging of the community and

increasing the visibility of RV events as well as making submission and participation more attractive to researchers. This was evident in the record number of submissions at RV 2010, which received a total of 74 submissions of which 15 were tutorials and tool demonstrations.

RV 2011 received a slightly smaller number of submissions – a total of 71 submissions of which 10 were tutorial and tool demonstrations. Thus, there was an increase of two research paper (full/short) submissions and a decrease of five tutorial or tool demonstration submissions over the previous year. All research paper and tool demonstration submissions to RV 2011 were reviewed by the Program Committee, with each paper receiving at least three reviews. The Program Committee selected 22 full papers (of 52 submissions), 2 short papers (of 9 submissions), and 4 tool demonstrations (of 5 submissions) for presentation at the conference. Four tutorials (of five submissions) were selected for presentation by the Chairs. Invited talks at RV 2011 were given by Dawson Engler (Stanford University), Cormac Flanagan (UC Santa Cruz), Wolfgang Grieskamp (Google), Sharad Malik (Princeton University), and Steven P. Reiss (Brown University).

RV 2011 gave two awards to peer-reviewed submissions. The “Best Paper Award” was given to “Runtime Verification with State Estimation” by Scott Stoller, Ezio Bartocci, Justin Seyster, Radu Grosu, Klaus Havelund, Scott Smolka and Erez Zadok. The “Best Tool Paper Award” was given to “MONPOLY: Monitoring Usage-Control Policies” by David Basin, Matus Harvan, Felix Klaedtke and Eugen Zalinescu.

The Chairs would like to thank the Program Committee for their high-quality reviews and hard work in making RV 2011 a successful event. Financial support for the conference was provided by Microsoft Research, the ARTIST Network of Excellence on Embedded Systems Design, Intel Corporation, Google Inc., the PRECISE Research Center of University of Pennsylvania, Laboratory for Reliable Software (LaRS) at NASA’s Jet Propulsion Laboratory, and the University of California, Berkeley. We would like to particularly thank the local Organizing Chairs Jacob Burnim and Nicholas Jalbert, and Klaus Havelund for extensive help in making arrangements and organizing the event, and Oleg Sokolsky for handling the finances and accounting. Submission and evaluation of papers, as well as the preparation of this proceedings volume, was handled by the Easy-Chair conference management service.

We hope that the strength of programs at RV conferences will continue to provide a flagship venue for the RV community and to foster new collaborations with researchers in related fields.

November 2011

Sarfraz Khurshid
Koushik Sen

Organization

Program Committee

Howard Barringer	University of Manchester, UK
Eric Bodden	Technical University Darmstadt, Germany
Rance Cleaveland	University of Maryland, USA
Mads Dam	Kungliga Tekniska högskolan, Sweden
Wim De Pauw	IBM T.J. Watson Research Center, USA
Brian Densky	University of California at Irvine, USA
Bernd Finkbeiner	Saarland University, Germany
Cormac Flanagan	University of California at Santa Cruz, USA
Patrice Godefroid	Microsoft Research Redmond, USA
Jean Goubault-Larrecq	ENS Cachan, France
Susanne Graf	Verimag, France
Radu Grosu	State University of New York at Stony Brook, USA
Lars Grunske	University of Kaiserslautern, Germany
Aarti Gupta	NEC Laboratories America, USA
Rajiv Gupta	University of California at Riverside, USA
Klaus Havelund	NASA/JPL, USA
Mats Heimdahl	University of Minnesota, USA
Gerard Holzmann	NASA/JPL, USA
Sarfraz Khurshid	University of Texas at Austin, USA
Viktor Kuncak	École Polytechnique Fédérale De Lausanne, Switzerland
Kim Larsen	Aalborg University, Denmark
Martin Leucker	University of Lübeck, Germany
Rupak Majumdar	Max Planck Institute, Germany, and University of California at Los Angeles USA
Greg Morrisett	Harvard University, USA
Mayur Naik	Intel Berkeley Labs, USA
Brian Nielsen	Aalborg University, Denmark
Klaus Ostermann	University of Marburg, Germany
Corina Pasareanu	NASA Ames, USA
Doron Peled	Bar Ilan University, Israel
Suzette Person	NASA Langley, USA
Gilles Pokam	Intel, Santa Clara, USA
Shaz Qadeer	Microsoft Research Redmond, USA
Derek Rayside	University of Waterloo, Canada
Grigore Rosu	University of Illinois at Urbana-Champaign, USA
Wolfram Schulte	Microsoft Research Redmond, USA
Koushik Sen	University of California, Berkeley, USA

VIII Organization

Peter Sestoft	IT University of Copenhagen, Denmark
Scott Smolka	State University of New York at Stony Brook, USA
Oleg Sokolsky	University of Pennsylvania, USA
Manu Sridharan	IBM T.J. Watson Research Center, USA
Mana Taghdiri	Karlsruhe Institute of Technology, Germany
Serdar Tasiran	Koc University, Turkey
Nikolai Tillmann	Microsoft Research Redmond, USA
Shmuel Ur	Shmuel Ur Innovation, Israel
Willem Visser	University of Stellenbosch, South Africa
Mahesh Viswanathan	University of Illinois at Urbana-Champaign, USA
Xiangyu Zhang	Purdue University, USA

Additional Reviewers

Akhin, Marat	Le Guernic, Gurvan
Ayoub, Anaheed	Lin, Changhui
Bartocci, Ezio	Lundblad, Andreas
Bekar, Can	Meredith, Patrick
Benzina, Hedi	Mikučionis, Marius
Bollig, Benedikt	Pohlmann, Christian
Bulychev, Peter	Reger, Giles
Charan K., Sai	Seyster, Justin
Decker, Normann	Stümpel, Annette
Dimitrova, Rayna	Tan, Li
Ehlers, Rüdiger	Thoma, Daniel
Elmas, Tayfun	Tkachuk, Oksana
Eom, Yong Hun	Wang, Shaohui
Faymonville, Peter	Wang, Yan
Feng, Min	Wies, Thomas
Gerke, Michael	Xiao, Xusheng
Gottschlich, Justin	Zhang, Pengcheng
Jobstmann, Barbara	Zorn, Benjamin
Komuravelli, Anvesh	

Table of Contents

Tutorials

Internal versus External DSLs for Trace Analysis (Extended Abstract)	1
<i>Howard Barringer and Klaus Havelund</i>	
Predicting Concurrency Failures in the Generalized Execution Traces of x86 Executables	4
<i>Chao Wang and Malay Ganai</i>	
Runtime Monitoring of Time-Sensitive Systems [Tutorial Supplement]	19
<i>Borzoo Bonakdarpour and Sebastian Fischmeister</i>	
Teaching Runtime Verification	34
<i>Martin Leucker</i>	

Invited

Runtime Verification: A Computer Architecture Perspective	49
<i>Sharad Malik</i>	

A: Parallelism and Deadlocks

Isolating Determinism in Multi-threaded Programs	63
<i>Lukasz Ziarek, Siddharth Tiwary, and Suresh Jagannathan</i>	
Efficiency Optimizations for Implementations of Deadlock Immunity	78
<i>Horatiu Julia, Silviu Andrica, and George Candea</i>	
Permission Regions for Race-Free Parallelism	94
<i>Edwin Westbrook, Jisheng Zhao, Zoran Budimlić, and Vivek Sarkar</i>	
Dynamic Race Detection with LLVM Compiler: Compile-Time Instrumentation for ThreadSanitizer	110
<i>Konstantin Serebryany, Alexander Potapenko, Timur Iskhodzhanov, and Dmitriy Vyukov</i>	

B: Malware Detection

NORT: Runtime Anomaly-Based Monitoring of Malicious Behavior for Windows	115
<i>Narcisa Andreea Milea, Siau Cheng Khoo, David Lo, and Cristian Pop</i>	

C: Temporal Constraints and Concurrency Bugs

Runtime Verification of LTL-Based Declarative Process Models	131
<i>Fabrizio Maria Maggi, Michael Westergaard, Marco Montali, and Wil M.P. van der Aalst</i>	
Parametric Identification of Temporal Properties	147
<i>Eugene Asarin, Alexandre Donzé, Oded Maler, and Dejan Nickovic</i>	
Marathon: Detecting Atomic-Set Serializability Violations with Conflict Graphs	161
<i>William N. Sumner, Christian Hammer, and Julian Dolby</i>	
Coverage Metrics for Saturation-Based and Search-Based Testing of Concurrent Software	177
<i>Bohuslav Křena, Zdeněk Letko, and Tomáš Vojnar</i>	

D: Sampling and Specification Conformance

Runtime Verification with State Estimation	193
<i>Scott D. Stoller, Ezio Bartocci, Justin Seyster, Radu Grosu, Klaus Havelund, Scott A. Smolka, and Erez Zadok</i>	
Efficient Techniques for Near-Optimal Instrumentation in Time-Triggered Runtime Verification	208
<i>Samaneh Navabpour, Chun Wah Wallace Wu, Borzoo Bonakdarpour, and Sebastian Fischmeister</i>	
CoMA: Conformance Monitoring of Java Programs by Abstract State Machines	223
<i>Paolo Arcaini, Angelo Gargantini, and Elvinia Riccobene</i>	
Automated Test-Trace Inspection for Microcontroller Binary Code	239
<i>Thomas Reinbacher, Jörg Brauer, Daniel Schachinger, Andreas Steininger, and Stefan Kowalewski</i>	

Invited

What Is My Program Doing? Program Dynamics in Programmer's Terms	245
<i>Steven P. Reiss and Alexander Tarvo</i>	

E: Real-Time

Algorithms for Monitoring Real-Time Properties	260
<i>David Basin, Felix Klaedtke, and Eugen Zălinescu</i>	

F: Software and Hardware Systems

Runtime Monitoring of Stochastic Cyber-Physical Systems with Hybrid State	276
<i>A. Prasad Sistla, Miloš Žefran, and Yao Feng</i>	
Combining Time and Frequency Domain Specifications for Periodic Signals	294
<i>Aleksandar Chakarov, Sriram Sankaranarayanan, and Georgios Fainekos</i>	
Runtime Verification for Ultra-Critical Systems	310
<i>Lee Pike, Sebastian Niller, and Nis Wegmann</i>	
Runtime Verification of Data-Centric Properties in Service Based Systems	325
<i>Guoquan Wu, Jun Wei, Chunyang Ye, Xiaozhe Shao, Hua Zhong, and Tao Huang</i>	

Invited

Cooperative Concurrency for a Multicore World (Extended Abstract)...	342
<i>Jaeheon Yi, Caitlin Sadowski, Stephen N. Freund, and Cormac Flanagan</i>	

G: Memory Transactions

Monitoring Data Structures Using Hardware Transactional Memory	345
<i>Shakeel Butt, Vinod Ganapathy, Arati Baliga, and Mihai Christodorescu</i>	

Tools

MONPOLY: Monitoring Usage-Control Policies	360
<i>David Basin, Matúš Harvan, Felix Klaedtke, and Eugen Zălinescu</i>	
MOPBox: A Library Approach to Runtime Verification (Tool Demonstration)	365
<i>Eric Bodden</i>	
Elarva: A Monitoring Tool for Erlang	370
<i>Christian Colombo, Adrian Francalanza, and Rudolph Gatt</i>	

DA-BMC: A Tool Chain Combining Dynamic Analysis and Bounded Model Checking	375
<i>Jan Fiedor, Vendula Hrubá, Bohuslav Křena, and Tomáš Vojnar</i>	

H: Foundational Techniques and Multi-valued Approaches I

Monitoring Finite State Properties: Algorithmic Approaches and Their Relative Strengths	381
<i>Rahul Purandare, Matthew B. Dwyer, and Sebastian Elbaum</i>	
Unbounded Symbolic Execution for Program Verification	396
<i>Joxan Jaffar, Jorge A. Navas, and Andrew E. Santosa</i>	
Execution Trace Exploration and Analysis Using Ontologies	412
<i>Newres Al Haider, Benoit Gaudin, and John Murphy</i>	

I: Foundational Techniques and Multi-valued Approaches II

Monitoring Realizability	427
<i>Rüdiger Ehlers and Bernd Finkbeiner</i>	
Runtime Verification of Traces under Recording Uncertainty	442
<i>Shaohui Wang, Anaheed Ayoub, Oleg Sokolsky, and Insup Lee</i>	
Author Index	457