

Founding Editors

Gerhard Goos

Karlsruhe Institute of Technology, Karlsruhe, Germany

Juris Hartmanis

Cornell University, Ithaca, NY, USA

Editorial Board Members

Elisa Bertino

Purdue University, West Lafayette, IN, USA

Wen Gao

Peking University, Beijing, China

Bernhard Steffen 

TU Dortmund University, Dortmund, Germany

Gerhard Woeginger 

RWTH Aachen, Aachen, Germany

Moti Yung

Columbia University, New York, NY, USA

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

Huaikou Miao · Cong Tian ·
Shaoying Liu · Zhenhua Duan (Eds.)

Structured Object-Oriented Formal Language and Method

9th International Workshop, SOFL+MSVL 2019
Shenzhen, China, November 5, 2019
Revised Selected Papers

Editors

Huaikou Miao
School of Computer Engineering
and Science
Shanghai University
Shanghai, China

Shaoying Liu
Hosei University
Tokyo, Japan

Cong Tian
Institute of Computing Theory
and Technology
Xidian University
Xi'an, China

Zhenhua Duan
Xidian University
Xi'an, Shaanxi, China

ISSN 0302-9743

ISSN 1611-3349 (electronic)

Lecture Notes in Computer Science

ISBN 978-3-030-41417-7

ISBN 978-3-030-41418-4 (eBook)

<https://doi.org/10.1007/978-3-030-41418-4>

LNCS Sublibrary: SL1 – Theoretical Computer Science and General Issues

© Springer Nature Switzerland AG 2020

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, expressed 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.

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

Preface

In spite of extensive research on formal methods and many efforts on transferring the technology to industry over the last three decades, how to enable practitioners to easily and effectively use formal techniques still remains challenging. The Structured Object-Oriented Formal Language (SOFL) has been developed to address this challenge by providing a comprehensive specification language, a practical modeling method, various verification and validation techniques, and tool support through effective integration of formal methods with conventional software engineering techniques. SOFL integrates Data Flow Diagram, Petri Nets, and VDM-SL to offer a visualized and formal notation for constructing specification; a three-step approach to requirements acquisition and system design; specification-based inspection and testing methods for detecting errors in both specifications and programs; and a set of tools to support modeling and verification. The Modeling, Simulation and Verification Language (MSVL) is a parallel programming language. It enables users to model a system as an MSVL program, to execute the program for simulating the behavior of the system, and to verify properties of the system. Its supporting toolkit MSV including MSVL compiler, interpreter, and several types of model checkers has been developed for automatic program verification. In particular, to verify C and Verilog/VHDL programs in large scale, some translators such as C2M and V2M, and a run time verification tool in code level have also been built to facilitate modeling, simulating, and verifying large systems in a formal manner.

Following the success of the previous SOFL+MSVL workshops, the 9th International Workshop on SOFL+MSVL for Reliability and Security (SOFL+MSVL 2019) was jointly organized in Shenzhen with the aim of bringing together industrial, academic, and government experts and practitioners of SOFL or MSVL to communicate and to exchange ideas. The workshop attracted 43 submissions on specification-based testing, specification inspection, model checking, formal verification, formal semantics, and formal analysis. Each submission was rigorously reviewed by two or more Program Committee (PC) members on the basis of technical quality, relevance, significance, and clarity, and 23 papers were accepted for publication in the workshop proceedings. The acceptance rate was 53%.

We would like to thank ICFEM 2019 for supporting the organization of the workshop, all of the PC members for their great efforts and cooperation in reviewing and selecting the papers, and our postgraduate students for all their help. We would also like to thank all of the participants for attending presentation sessions and actively joining discussions at the workshop. Finally, our gratitude goes to the editors at Springer for their continuous support in publishing the workshop proceedings.

December 2019

Huakou Miao
Cong Tian
Shaoying Liu
Zhenhua Duan

Organization

Program Committee

Shaoying Liu (General Chair)	Hosei University, Japan
Zhenhua Duan (General Chair)	Xidian University, China
Huaikou Miao (Program Co-chair)	Shanghai University, China
Cong Tian (Program Co-chair)	Xidian University, China
Yuting Chen	Shanghai Jiao Tong University, China
Busalire Emeka	Hosei University, Japan
Colin Fidge	Queensland University of Technology, Australia
Weikai Miao	East China Normal University, China
Fumiko Nagoya (Program Co-chair)	Nihon University, Japan
Shin Nakajima	National Institute of Informatics, Japan
Kazuhiro Ogata	JAIST, Japan
Shengchao Qin	Teesside University, UK
Wuwei Shen	Western Michigan University, USA
Xinfeng Shu	Xi'an University of Posts and Telecommunications, China
Rong Wang	Hosei University, Japan
Xi Wang	Shanghai University, China
Jinyun Xue	Jiangxi Normal University, China

Contents

Testing and Debugging

Analysis and Remodeling of the DirtyCOW Vulnerability by Debugging and Abstraction	3
<i>YanJun Wen and Ji Wang</i>	
A Formal Technique for Concurrent Generation of Software's Functional and Security Requirements in SOFL Specifications	13
<i>Busalire Emeka and Shaoying Liu</i>	
Distortion and Faults in Machine Learning Software	29
<i>Shin Nakajima</i>	
A Divide & Conquer Approach to Testing Concurrent Java Programs with JPF and Maude	42
<i>Canh Minh Do and Kazuhiro Ogata</i>	

Formal Verification

An Approach to Modeling and Verifying Multi-level Interrupt Systems with TMSVL	61
<i>Jin Cui, Xu Lu, and Buwen Liang</i>	
Towards Formal Verification of Neural Networks: A Temporal Logic Based Framework	73
<i>Xiaobing Wang, Kun Yang, Yanmei Wang, Liang Zhao, and Xinfeng Shu</i>	
UMC4M: A Verification Tool via Program Execution	88
<i>Meng Wang, Junfeng Tian, and Hong Zhang</i>	
Parallel Runtime Verification Approach for Alternate Execution of Multiple Threads.	99
<i>Bin Yu, Jinhui Liu, Ming Lei, Yong Yu, and Hao Chen</i>	

Problem Solving

A Planning Approach Based on APTL	113
<i>Haiyang Wang and Yao Liu</i>	
Solving Constraint Optimization Problems Based on Mathematica and Abstraction.	123
<i>Guoteng Pan, Mengjun Li, and Guodong Ou</i>	

A Forward Chaining Heuristic Search with Spatio-Temporal Control Knowledge.	141
<i>Xu Lu, Jin Cui, Yansong Dong, Wensheng Wang, Runzhe Ma, Yifeng Li, and Qing Feng</i>	
Formal Development and Verification of Reusable Component in PAR Platform.	155
<i>Qimin Hu, Jinyun Xue, Zhen You, Zhuo Cheng, and Zhengkan Zuo</i>	
A New Mutant Generation Algorithm Based on Basic Path Coverage for Mutant Reduction	167
<i>Xu Qin, Shaoying Liu, and Zhang Tao</i>	
Formal Specification and Model Checking of a Ride-sharing System in Maude	187
<i>Eiichi Muramoto, Kazuhiro Ogata, and Yoichi Shinoda</i>	
Model Checking Python Programs with MSVL.	205
<i>Xinfeng Shu, Fengyun Gao, Weiran Gao, Lili Zhang, Xiaobing Wang, and Liang Zhao</i>	
Software Analysis and Evolution	
Prediction of Function Removal Propagation in Linux Evolution.	227
<i>Lei Wang, Guoxiong Chen, and Liang Li</i>	
Regression Models for Performance Ranking of Configurable Systems: A Comparative Study	243
<i>Yuntianyi Chen, Yongfeng Gu, Lulu He, and Jifeng Xuan</i>	
Combining Model Learning and Model Checking to Analyze Java Libraries	259
<i>Shahbaz Ali, Hailong Sun, and Yongwang Zhao</i>	
Data Provenance Based System for Classification and Linear Regression in Distributed Machine Learning.	279
<i>Muhammad Jahanzeb Khan, Ruoyu Wang, Daniel Sun, and Guoqiang Li</i>	
Software Analysis and Testing	
Metamorphic Testing in Fault Localization of Model Transformations	299
<i>Keke Du, Mingyue Jiang, Zuohua Ding, Hongyun Huang, and Ting Shu</i>	
A Fault Localization Method Based on Dynamic Failed Execution Blocks	315
<i>Baoyi Pan, Ting Shu, Jinsong Xia, Zuohua Ding, and Mingyue Jiang</i>	

Adaptive Random Testing by Bisection and Comprehensive Distance 328
Chengying Mao, Mengting Quan, Zhilei Chen, and Tsong Yueh Chen

CMM: A Combination-Based Mutation Method for SQL Injection 345
Jing Zhao, Tianran Dong, Yang Cheng, and Yanbin Wang

Author Index 363