Lecture Notes in Computer Science **14415**

The series Lecture Notes in Computer Science (LNCS), including its subseries Lecture Notes in Artificial Intelligence (LNAI) and Lecture Notes in Bioinformatics (LNBI), has established itself as a medium for the publication of new developments in computer science and information technology research, teaching, and education.

LNCS enjoys close cooperation with the computer science R & D community, the series counts many renowned academics among its volume editors and paper authors, and collaborates with prestigious societies. Its mission is to serve this international community by providing an invaluable service, mainly focused on the publication of conference and workshop proceedings and postproceedings. LNCS commenced publication in 1973.

Paolo Arcaini · Tao Yue · Erik M. Fredericks
Editors

Search-Based Software Engineering

15th International Symposium, SSBSE 2023 San Francisco, CA, USA, December 8, 2023 Proceedings

Springer

Editors
Paolo Arcaini
National Institute of Informatics
Tokyo, Japan

Tao Yue
Beihang University
Beijing, China

Erik M. Fredericks
Grand Valley State University
Allendale, MI, USA

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-031-48795-8 ISBN 978-3-031-48796-5 (eBook)
https://doi.org/10.1007/978-3-031-48796-5

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

Paper in this product is recyclable.

Preface

Message from the General Chair

We are very excited to bring you the proceedings of the 15th edition of the International Symposium on Search-Based Software Engineering (SSBSE 2023). SSBSE is a premier event for all who are interested in the intersection of search optimization and difficult software engineering problems spanning numerous domains. This year SSBSE was held as a hybrid event; physically located in San Francisco, California, USA and remote for those that were unable to attend in person. We continue to be co-located with ESEC/FSE and are grateful for their support of our symposium.

Thank you to Tao Yue and Paolo Arcaini for putting together an exciting research track program; without your constant feedback and support planning this event would have been extremely difficult! I would also like to thank the track chairs for their hard work supporting SSBSE as well: Rebecca Moussa and Thomas Vogel (Hot Off the Press track), Gregory Gay and Max Hort (RENE/NIER track), and Bobby Bruce, José Miguel Rojas, and Vali Tawosi (Challenge track). I would also like to thank Alexander Lalejini for his work on the SSBSE website and Emeralda Sesari for her work on publicizing our event. Many thanks to our multiple Program Committees as well for their tireless work on reviewing papers and suggesting improvements and to our Steering Committee for continuing to oversee and support this event.

Very special thanks to our sponsor, Grand Valley State University. I would also like to thank Satish Chandra from Google for his support as the general chair of ESEC/FSE 2023, as well as Federica Sarro at University College London as a point of contact in the Steering Committee for continuing support whenever an issue arose that I did not have an answer for.

I enjoyed seeing you all at SSBSE 2023 and hope you enjoyed the program!

December 2023 Erik Fredericks

Message from the Program Chairs

Speaking for the SSBSE 2023 Program Committee, we are delighted to introduce the proceedings of the 15th International Symposium on Search-Based Software Engineering. Search-Based Software Engineering (SBSE) focuses on formulating various optimization problems in software engineering as search problems and then addressing them with search techniques, intending to automate complex software engineering tasks. A wide class of software engineering challenges can be formulated into SBSE problems, including test optimization, design and code refactoring, software/system development process optimization, and many more, which have been addressed with various search techniques. The community continuously advances the field by identifying, formulating, and solving new challenges such as searching for critical driving scenarios for testing autonomous vehicles, optimizing the energy consumption of software applications, and automating hyperparameter tuning of machine learning models.

SSBSE 2023 upheld the longstanding tradition of SSBSE by gathering the SBSE community yearly to share and discuss advancements in the field. This year, we welcomed submissions showcasing innovative contributions across all SBSE domains. Specifically, we received 13 submissions to the main Research track, four to the Replications and Negative Results (RENE)/New Ideas and Emerging Results (NIER) track, six to the SBSE Challenge track, and two to the Hot Off the Press (HOP) track. Each valid submission to the Research track were double-blind reviewed by at least three PC members and, eventually, six papers were accepted. For the RENE/NIER track, three papers were accepted. For the SBSE Challenge track, five papers were accepted. The HOP track accepted two papers.

December 2023

Paolo Arcaini
Tao Yue

Organization

General Chair

Erik Fredericks	Grand Valley State University, USA

Research Track Chairs

Paolo Arcaini	National Institute of Informatics, Japan
Tao Yue	Beihang University, China

Hot Off the Press Track Chairs

Rebecca Moussa	University College London, UK
Thomas Vogel	Humboldt-Universtität zu Berlin, Germany

RENE/NIER Track Chairs

Gregory Gay	Chalmers—University of Gothenburg, Sweden
Max Hort	Simula Research Laboratory, Norway

Challenge Track Chairs

Bobby Bruce	University of California at Davis, USA
José Miguel Rojas	University of Sheffield, UK
Vali Tawosi	J. P. Morgan AI Research, UK

Publicity Chair

Emeralda Sesari	University of Groningen, The Netherlands

Web Chair

Alexander Lalejini	Grand Valley State University, USA

Steering Committee

Shaukat Ali	Simula Research Laboratory and Oslo Metropolitan University, Norway
Andrea Arcuri	Kristiania University College and Oslo Metropolitan University, Norway
Giovani Guizzo	University College London, UK
Gordon Fraser	University of Passau, Germany
Phil McMinn	University of Sheffield, UK
Justyna Petke	University College London, UK
Federica Sarro	University College London, UK
Silva Regina Vergilio	Federal University of Paraná, Brazil
Shin Yoo	KAIST, Republic of Korea

Research Track Committee

Paolo Arcaini (Chair)	National Institute of Informatics, Japan
Tao Yue (Chair)	Beihang University, China
Shaukat Ali	Simula Research Laboratory and Oslo Metropolitan University, Norway
Andrea Arcuri	Kristiania University College and Oslo Metropolitan University, Norway
Aitor Arrieta	Mondragon University, Spain
Marcio Barros	Universidade Federal do Estado do Rio de Janeiro, Brazil
Matteo Biagiola	Università della Svizzera Italiana, Italy
Betty H. C. Cheng	Michigan State University, USA
Pouria Derakhshanfar	JetBrains Research, The Netherlands
Xavier Devroey	University of Namur, Belgium
Thiago Ferreira	University of Michigan - Flint, USA
Gordon Fraser	University of Passau, Germany
Alessio Gambi	IMC University of Applied Sciences Krems, Austria
Gregory Gay	Chalmers—University of Gothenburg, Sweden
Lars Grunske	Humboldt-Universität zu Berlin, Germany
Hadi Hemmati	York University, Canada

Fuyuki Ishikawa	National Institute of Informatics, Japan
Fitsum Kifetew	Fondazione Bruno Kessler, Italy
Thelma Elita	State University of Maringá, Brazil
Colanzi Lopez	.
Phil McMinn	University of Sheffield, UK
Inmaculada Medina-Bulo	Universidad de Cádiz, Spain
Leandro Minku	University of Birmingham, UK
Manuel Núñez	Universidad Complutense de Madrid, Spain
Annibale Panichella	Delft University of Technology, The Netherlands
Vincenzo Riccio	University of Udine, Italy
José Miguel Rojas	University of Sheffield, UK
Federica Sarro	University College London, UK
Jeongju Sohn	University of Luxembourg, Luxembourg
Valerio Terragni	University of Auckland, New Zealand
Silva Regina Vergilio	Federal University of Paraná, Brazil
Thomas Vogel	Humboldt-Universtität zu Berlin, Germany
Man Zhang	Kristiania University College, Norway

RENE/NIER Track Committee

Gregory Gay (Chair)	Chalmers—University of Gothenburg, Sweden
Max Hort (Chair)	Simula Research Laboratory, Norway
Gabin An	KAIST, Republic of Korea
Marcio Barros	UNIRIO, Brazil
Carlos Cetina	San Jorge University, Spain
Michael C. Gerten	Iowa State University, USA
Emanuele Iannone	University of Salerno, Italy
Bruno Lima	University of Porto and INESC TEC, Portugal
Ruchika Malhotra	Delhi Technological University, India
Pasqualina Potena	RISE Research Institutes of Sweden AB, Sweden
José Raúl Romero	University of Cordoba, Spain
Jeongju Sohn	University of Luxembourg, Luxembourg
Andrea Stocco	Technical University of Munich & fortiss, Germany
Fiorella Zampetti	University of Sannio, Italy

Challenge Track Committee

Bobby Bruce (Chair)	University of California at Davis, USA
José Miguel Rojas (Chair)	University of Sheffield, UK

Vali Tawosi (Chair)	J. P. Morgan AI Research, UK
Wesley Assunção	North Carolina State University, USA
Kate M. Bowers	Oakland University, USA
José Campos	University of Porto, Portugal
Tao Chen	University of Birmingham, UK
Thelma Elita Colanzi Lopez	State University of Maringá, Brazil .
Gregory Kapfhammer	Allegheny College, USA
Stephan Lukasczyk	University of Passau, Germany
Inmaculada Medina-Bulo	Universidad de Cádiz, Spain
Rebecca Moussa	University College London, UK
Sebastian Schweikl	University of Passau, Germany

Keynotes

Search-Based Software Engineering for Learning-Enabled Self-Adaptive Systems

Betty H. C. Cheng

Michigan State University, USA

Abstract. Trustworthy artificial intelligence (Trusted AI) is essential when autonomous, safety-critical systems use learning-enabled components (LECs) in uncertain environments. When reliant on deep learning, these learning-enabled systems (LES) must address the reliability, interpretability, and robustness (collectively, the assurance) of learning models. Three types of uncertainty most significantly affect assurance. First, uncertainty about the physical environment can cause suboptimal, and sometimes catastrophic, results as the system struggles to adapt to unanticipated or poorly understood environmental conditions. For example, when lane markings are occluded (either on the camera and/or the physical lanes), lane management functionality can be critically compromised. Second, uncertainty in the cyber environment can create unexpected and adverse consequences, including not only performance impacts (network load, real-time responses, etc.) but also potential threats or overt (cybersecurity) attacks. Third, uncertainty can exist with the components themselves and affect how they interact upon reconfiguration. Left unchecked, this may cause unexpected and unwanted feature interactions. While learning-enabled technologies have made great strides in addressing uncertainty, challenges remain in addressing the assurance of such systems when encountering uncertainty not addressed in training data. Furthermore, we need to consider LESs as first-class software-based systems that should be rigorously developed, verified, and maintained (i.e., software engineered). In addition to developing specific strategies to address these concerns, appropriate software architectures are needed to coordinate LECs and ensure they deliver acceptable behavior even under uncertain conditions. To this end, this presentation overviews a number of our multi-disciplinary research projects involving industrial collaborators, which collectively support a search-based software engineering, model-based approach to address Trusted AI and provide assurance for learning-enabled systems (i.e., SBSE4LES). In addition to sharing lessons learned from more than two decades of research addressing assurance for (learning-enabled) self-adaptive systems operating under a range of uncertainty, near-term and longer-term research challenges for addressing assurance of LESs will be overviewed.

SSBSE Summary of .NET/C# Instrumentation for Search-Based Software Testing

Amid Golmohammadi[1], Man Zhang[1], and Andrea Arcuri[1,2]

[1] Kristiania University College, Norway, Norway
{amid.golmohammadi,man.zhang,andrea.arcuri}@kristiania.no
[2] Oslo Metropolitan University, Norway

Abstract. C# is a widely used programming language. However, to the best of our knowledge, there is no documented work on enabling Search-Based Software Testing methods for C# and .NET applications. This paper introduces a white-box testing approach and an open-source tool for C# applications that collects code coverage information in real-time via .NET bytecode instrumentation. The method improves the search's direction by using *Branch Distance* heuristics. The tool was evaluated on three .NET RESTful APIs after being integrated into the EvoMaster test generator. Results show that our strategy achieves significantly higher code coverage than grey-box testing tools.

Keywords: .NET instrumentation · SBST · REST APIs

Summary

This paper provides a summary of [3].

We have used the `Mono.Cecil` library to implement instrumentation for .NET programs, allowing analysis and modification of CIL code. This works offline with .NET libraries compiled as DLLs. Integration with SBST techniques is essential for generating tests. Our experiments leverage EvoMaster [2], which produces system-level tests for RESTful APIs. EvoMaster has two main components: a core process and a driver, facilitating instrumentation and interaction through RESTful APIs. This instrumentation takes place via a .NET Core console application. To utilize EvoMaster, we created a C# driver mirroring the existing JVM driver's endpoints. The core, written in Kotlin, was adapted to include C# output, enabling the generation of test cases as HTTP call sequences based on xUnit.

.NET programs consist of assemblies with classes containing methods. EvoMaster is designed to optimize test case creation, maximizing coverage of these methods' targets. For tracking coverage, `EnteringStatement` and `CompletedStatement` probes are inserted before and after each statement. These probes, essentially calls to

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 864972).

static methods in a console application developed to support the instrumentation, use parameters like class name and line number to uniquely identify statements. Coverage is measured heuristically between 0 (not covered) and 1 (fully covered). While `EnteringStatement` marks a statement's start, its error-free execution is confirmed by `CompletedStatement`. Inserting these probes presents challenges, especially with control-altering instructions such as jump or exit instructions (i.e., `br`, `throw`, `rethrow`, `endfinally`, `leave` and `ret`).

During the search, calculating branch distances for both numeric and string values are vital as they can provide gradient to the search algorithm to solve constraints. Numeric branch instructions are categorized into three types: *one-arg jump*, *two-arg compare*, and *two-arg jump*. One-arg jumps decide control flow based on a single value, while two-arg compare instructions compare two values and push the result (i.e., either 0 or 1) back on the evaluation stack, and two-arg jumps combine both. Calculating numeric branch distance involves challenges like duplication of values on the stack, using bytecode method replacements, and detecting specific data types. For string branch distances, boolean-returning operators and methods within `System.String` class are identified and replaced with probes to calculate the distance while retaining original functionality. The entire process ensures better test coverage and represents an intricate part of understanding how a system's control flow works, enhancing overall software reliability.

Our team assessed the efficacy of our approach by integrating bytecode instrumentation and branch distance-based heuristics into EvoMaster, calling it EvoMaster.NET. Utilizing default settings, we contrasted it with a grey-box testing approach on three .NET REST APIs (i.e., `rest-ncs`, `rest-scs`, and `menu-api`) from *EMB* repository [1]. The findings revealed substantial improvements in the numerical and string problems (i.e., `rest-ncs` and `rest-scs`, respectively) up to 98% and 86% line coverage respectively compared to the grey-box approach, but equivalent results for the `menu-api`. The enhanced performance illustrates our method's ability to resolve the majority of numeric and string branches. However, for the `menu-api`, which involves working with a database, the performance was not enhanced, signifying the necessity for SQL database-specific adaptations.

References

1. Arcuri, A., et al.: EMB: a curated corpus of web/enterprise applications and library support for software testing research. IEEE (2023)
2. Arcuri, A., Galeotti, J.P., Marculescu, B., Zhang, M.: EvoMaster: a search-based system test generation tool. J. Open Source Softw. **6**(57), 2153 (2021)
3. Golmohammadi, A., Zhang, M., Arcuri, A.: .NET/C# instrumentation for search-based software testing. Softw. Qual. J. 1–27 (2023)

Contents

Challenge Track