

Guest editorial for special section on research in search-based software engineering

Claire Le Goues¹ · Shin Yoo²

Published online: 4 April 2017

© Springer Science+Business Media New York 2017

Research in Search-Based Software Engineering (SBSE) adapts and applies meta-heuristic optimization techniques to problems in software engineering. Fundamentally, SBSE involves formulating software engineering tasks as search problems: Representing candidate solutions, defining an objective or fitness function, and either developing new or adapting previous search strategies for traversing the landscape of potential solutions. The area has been established and growing for well more than a decade. Although the phrase “Search Based Software Engineering” was formally coined in 2001 [Harman and Jones 01], many previous efforts had already applied this type of technique to SE problems ranging from project management and testing (still a very common application domain). Recent years have seen a veritable explosion of interest in the field, such that it can now include work that is internally (self-reflective) and externally (ready for adoption) mature.

This special section features three articles from within SBSE that amply substantiate this maturity:

As one of the most well-established and widely-studied subjects in SBSE, test data generation is particularly ripe for studies that move it towards real-world applicability and adoption. To this end, in “A Detailed Investigation of the Effectiveness of Whole Test Suite Generation”, Rojas et al. (DOI 10.1007/S10664-015-9424-2) support our understanding of search-based test data generation via a large scale empirical studies of real world software. Traditional search-based test generation seeks to generate test cases that achieve all goals for a given coverage criterion. By contrast, whole test suite generation optimizes entire test suites towards a set of goals, and initial evidence suggests that doing so can be more effective than the standard approach. The authors corroborate some of these previous findings while

✉ Claire Le Goues
clegoues@cs.cmu.edu

Shin Yoo
shin.yoo@kaist.ac.kr

¹ Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, USA

² Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Guseong-dong, Yuseong-gu, Daejeon, South Korea

identifying nuances in the effectiveness of the whole suite approach when applied to real-world software at scale.

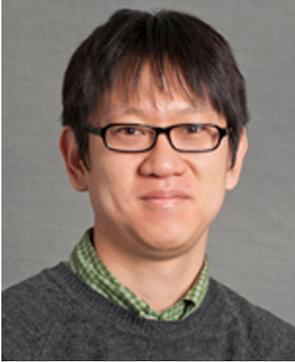
In “A Robust Multi-Objective Approach to Balance Severity and Importance of Refactoring Opportunities”, Mkaouer et al. (DOI 10.1007/S10664-016-9426-8) focus on meta properties of the search problem. They propose a specifically robust multi-objective model for assessing refactoring opportunities in large software systems. Refactoring such systems can be difficult because doing so involves complex tradeoffs between uncertain and competing concerns such as quality, severity, and difficulty; the authors propose a novel technique that substantially outperforms previous work while carrying an acceptable robustness price.

The paper “Generating Valid Grammar-based Test Inputs by means of Genetic

Programming and Annotated Grammars”, by Kifetew et al. (DOI 10.1007/S10664-015-9422-4), demonstrates the potential benefits of a careful integration of domain-specific knowledge into SBSE techniques, fruitful grounds for future work. The authors augment a stochastic grammar-based test generation system with probabilities learned from a real-world corpora, and then with grammar annotations as an alternative to those probabilities. Their results show that both approaches outperform the strictly random approach, and that the annotations provide a constructive alternative in the absence of suitable corpora over which to estimate probabilities.



Claire Le Goues received the BA degree in computer science from Harvard University and the MS and PhD degrees from the University of Virginia. She is an assistant professor in the School of Computer Science at Carnegie Mellon University, primarily affiliated with the Institute for Software Research. She is interested in how to construct and assure high-quality systems in the face of continuous software evolution. More information is available at: <http://www.cs.cmu.edu/~clegoues>.



Shin Yoo is an assistant professor at Korea Advanced Institute of Science and Technology. He received his PhD in King's College London in 2009, and was a lecturer at the Department of Computer Science in University College London from 2012 to 2015. His work focuses on the application of meta-heuristic optimization to the problems in software engineering, especially those of software testing and debugging. He is also interested in the use of information theory and program analysis.