Crash Reproduction Using Helper Objectives

Derakhshanfar, Pouria; Devroey, Xavier; Zaidman, Andy; van Deursen, Arie; Panichella, Annibale

DOI
10.1145/3377929.3390077

Publication date
2020

Document Version
Accepted author manuscript

Published in
GECCO 2020 Companion - Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion

Citation (APA)

Important note
To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.
Crash Reproduction Using Helper Objectives

Pouria Derakhshanjf, Xavier Devroey, Andy Zaidman, Arie van Deursen, Annibale Panichella
Delft University of Technology
Delft, The Netherlands

ABSTRACT
Evolutionary-based crash reproduction techniques aid developers in their debugging practices by generating a test case that reproduces a crash given its stack trace. In these techniques, the search process is typically guided by a single search objective called Crash Distance. Previous studies have shown that current approaches could only reproduce a limited number of crashes due to a lack of diversity in the population during the search. In this study, we address this issue by applying Multi-Objectivization using Helper-Objectives (MO-HO) on crash reproduction. In particular, we add two helper-objectives to the Crash Distance to improve the diversity of the generated test cases and consequently enhance the guidance of the search process. We assessed MO-HO against the single-objective crash reproduction. Our results show that MO-HO can reproduce two additional crashes that were not previously reproducible by the single-objective approach.

CCS CONCEPTS
• Software and its engineering → Software testing and debugging; Search-based software engineering

KEYWORDS
crash reproduction, search-based software testing, MOEA

ACM Reference Format:

1 INTRODUCTION
When a software application crashes, a report (or issue), including information gathered during the crash, is assigned to developers for debugging. One common practice to identify the root cause of a crash is to write a test case reproducing it [8]. This test case can later be adapted and integrated into the test suite to prevent future regressions. However, depending on the amount of information available in the report, writing this crash reproducing test case can be time-consuming and labor-intensive [7].

Consequently, various approaches in the literature try to automate crash reproduction. The earlier empirical study [7] shows that search-based crash reproduction outperforms other crash reproduction approaches in terms of crash reproduction ratio (percentage of crashes that could be reproduced) and efficiency (time taken to reproduce a given crash successfully). Search-based crash reproduction generates a test case that, when executed, is able to reproduce that crash by modeling the crash reproduction problem as an optimization problem. This approach reformulates crash reproduction as a single search objective (Crash Distance hereafter), which measures how far a generated test is from reproducing the crash, and applies a single-objective evolutionary algorithm (Single-Objective Search hereafter) to generate solutions (i.e., tests).

Although Single-Objective Search performs well compared to other crash reproduction approaches, a more extensive empirical study [6] revealed that it is not successful in reproducing complex crashes (i.e., large stack traces). Hence, further studies to enhance the guidance of the search process are required.

In this study, we investigate a new strategy to Multi-Objectivize crash reproduction based on Helper-Objectives (MO-HO) [5]. More specifically, we add two additional helper-objectives to Crash Distance (first objective): method call diversity (second objective) and test case length minimization (third objective). For the second objective, we re-use a distance function that measures the diversity of the methods called in the test cases. For the third objective, we count the number of statements in the generated test case. Since these three objectives conflict with Crash Distance, we expect an increase in the solutions’ diversity and, hence, an improvement in crash reproduction effectiveness (crash reproduction ratio) and efficiency. We utilize SPEA2 [9], which is a multi-objective evolutionary algorithm (MOEA), to solve this optimization problem.

We compare our approach against Single-Objective Search [7] from the perspectives of crash reproduction. Our results show that MO-HO can reproduce new crashes, which are not reproducible with Single-Objective Search.

2 MULTI-OBJECTIVIZATION WITH HELPER-OBJECTIVES (MO-HO)
As suggested by Jensen et al. [3], adding helper-objectives, which are in conflict with the primary one, to an existing single objective can help search algorithms escape from local optima. Therefore,
We used 30 randomly selected crashes from which are not reproducible by Single-Objective Search. We observed that our approach could reproduce two new crashes: XWIKI-14227 and LANG-19b. Moreover, we observed that all of the crashes reproduced by Single-Objective Search could be reproduced by MO-HO as well.

4 CONCLUSION AND FUTURE WORK

Generating crash reproducing test cases can ease the process of debugging for developers. A promising approach for automating this process is using evolutionary algorithms. This approach defines an optimization objective called Crash Distance and applies a single objective guided evolutionary algorithm (Single-Objective Search). This strategy may end up generating test cases that are not diverse enough because of a low exploration during the search process.

In this initial study, we apply MO-HO to tackle the problem of the former technique. In MO-HO, we define two helper-objectives in addition to Crash Distance to alleviate the lack of exploration. Moreover, the introduced helper-objectives conflict with the main objective Crash Distance.

We assessed the application of MO-HO to SPEA2 (a commonly used MOEA) to solve the crash reproduction problem and compared its results against Single-Objective Search. Results indicate that MO-HO can reproduce two new crashes (6% of selected crashes) not reproduced by Single-Objective Search. Since our early results are encouraging, we seek to perform an empirical study (on more crashes) and characterize the contributing factors in MO-HO in a future study.

ACKNOWLEDGMENTS

This research was partially funded by the EU Horizon 2020 ICT-10-2016-RIA “STAMP” project (No.731529).

REFERENCES


