

Improving Test Suite Generation by Testing Google Play's Top 1000 Apps III

Thomas Vogel¹, Chinh Tran², Irene Moser³, and Lars Grunskel¹

¹ Software Engineering Group, Humboldt-Universität zu Berlin, Germany

² Technische Universität Berlin, Germany

³ Swinburne University of Technology, Australia

Search-Based Software Testing

Search-based software engineering

Formulate a software engineering problem as a search/optimization problem

Find a representation of the problem

Define a fitness function to guide the search

Solve the problem with search-based optimization methods

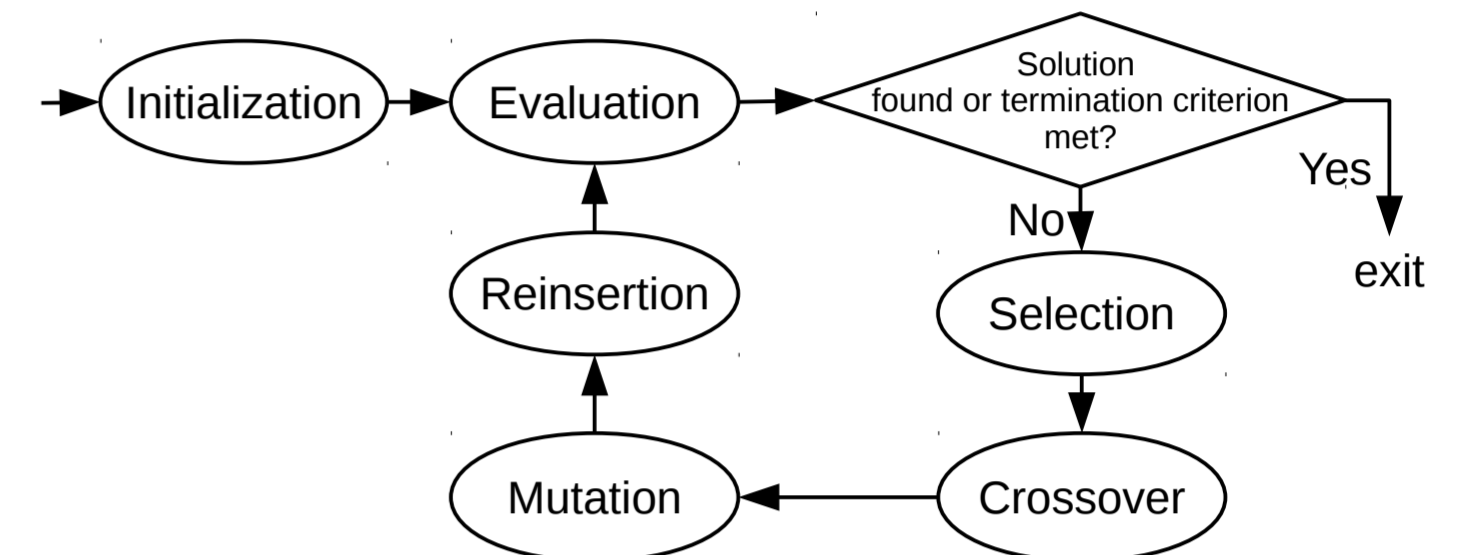
Search-based software testing (example)

Find "optimal" tests

⇒ Sequence of input events

⇒ Coverage, fault revelation, sequence length

⇒ Genetic algorithm (GA) such as NSGA-II



M. Harman and P. McMinn. 2010. A Theoretical and Empirical Study of Search-Based Testing: Local, Global, and Hybrid Search. IEEE Trans. Softw. Eng. 36, 2, 226–247.

Research Problem

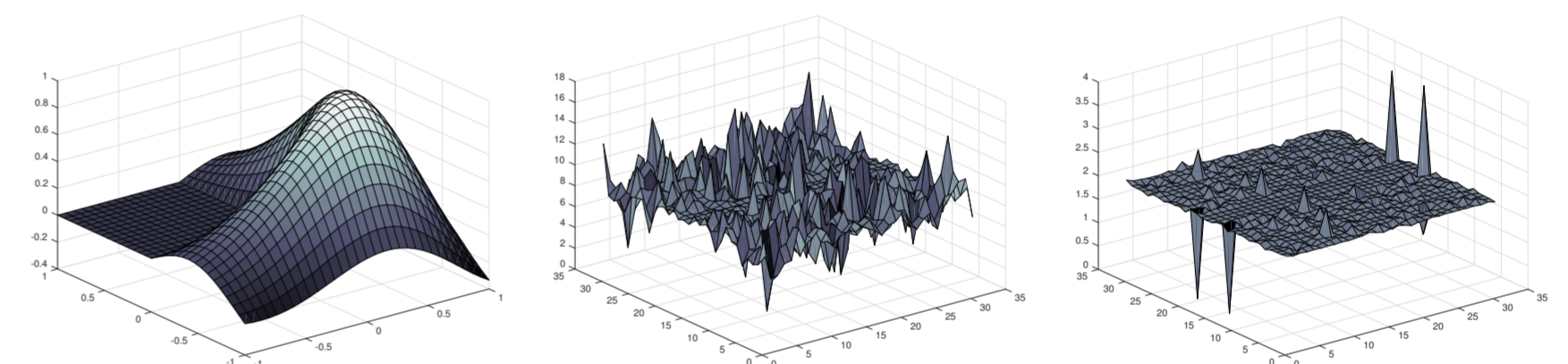
Which optimization method should be selected and how should the selected method be configured?

Typically missing knowledge about the search problem

- ⇒ Test different methods/configurations and select the best (costly)
- ⇒ Use a popular method with its default configuration (no free lunch)

Research Idea

Fitness Landscape Analysis to provide feedback for selecting and configuring search algorithms



Automated Testing of Apps with Sapienz (Deployed at Facebook)



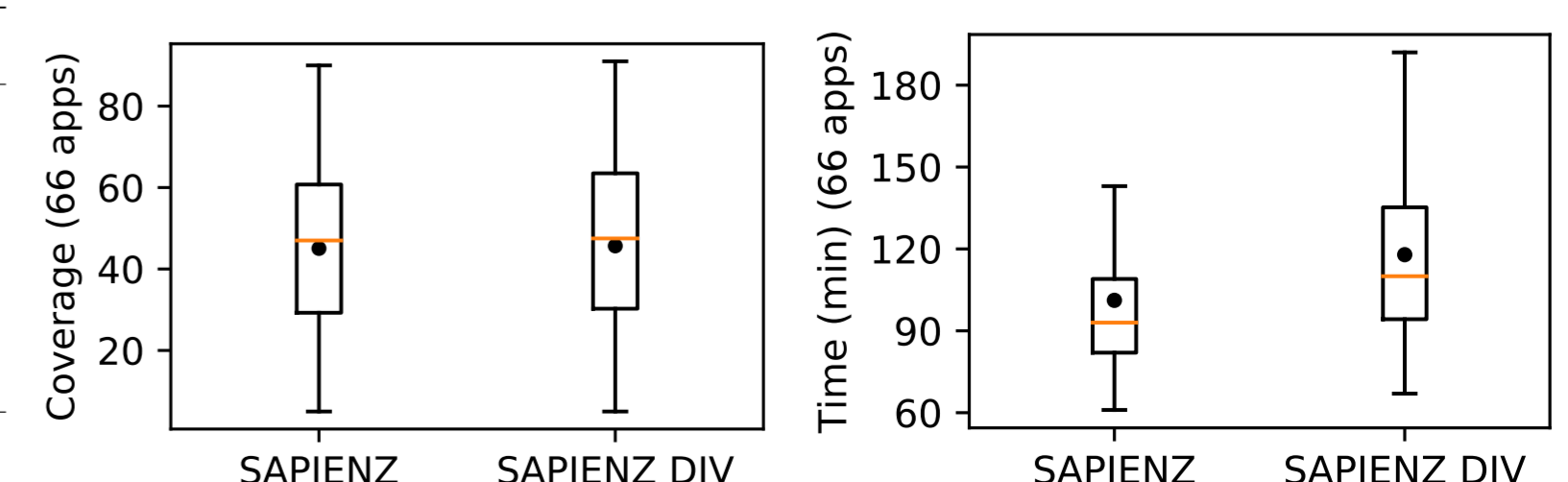
Ke Mao, Mark Harman, and Yue Jia. 2016. Sapienz: multi-objective automated testing for Android applications. In Proc. of the 25th International Symposium on Software Testing and Analysis (ISSTA 2016). ACM, 94-105.

Experiments and Results

- Decreasing diversity of test suites during the search in Sapienz
- Adapt Sapienz with diversity-preserving mechanisms ⇒ Sapienz^{div}
- Head-to-head comparison: Sapienz vs. Sapienz^{div}
- Study 1: run Sapienz/Sapienz^{div} once on 66 apps
- Study 2: run Sapienz/Sapienz^{div} 20x on 10 apps

Study 1:

	Sapienz	Sapienz ^{div}
# App Crashed	43	46
# Total Crashes	5974	6941
# Unique Crashes	119	141
# Disjoint Crashes	29	51
# Intersecting Crashes	90	90
Mean sequence length	209	244



Study 2: Inconclusive results based on inferential statistical analysis