Software Engineering Seminar

Automated Test Case Generation

Description

Testing is essential for developing high-quality software but creating test cases manually is costly. Therefore, automated techniques using heuristics, especially genetic algorithms, have been developed to generate test cases. Over time different techniques have been proposed, such as the *whole-suite* approach [1], *whole-suite with archive* approach [6], *MOSA* [3], and *DynaMOSA* [4] while the different techniques are usually compared empirically [2, 5].

The student should explore and discuss the current state of the art in automated test case generation by investigating, discussing, and comparing different automated techniques.

References

[1] Gordon Fraser and Andrea Arcuri. Whole test suite generation. *IEEE Transactions on Software Engineering*, 39(2):276–291, 2013.

[2] Gordon Fraser and Andrea Arcuri. A large-scale evaluation of automated unit test generation using evosuite. *ACM Trans. Softw. Eng. Methodol.*, 24(2):8:1–8:42, December 2014.

[3] Annibale Panichella, Fitsum Meshesha Kifetew, and Paolo Tonella. Reformulating branch coverage as a many-objective optimization problem. In *2015 IEEE 8th International Conference on Software Testing, Verification and Validation (ICST)*, pages 1–10, 2015.

[4] Annibale Panichella, Fitsum Meshesha Kifetew, and Paolo Tonella. Automated test case generation as a many-objective optimisation problem with dynamic selection of the targets. *IEEE Transactions on Software Engineering*, 44(02):122–158, 2018.

[5] Annibale Panichella, Fitsum Meshesha Kifetew, and Paolo Tonella. A large scale empirical comparison of state-of-the-art search-based test case generators. *Information and Software Technology*, 104:236–256, 2018.

[6] José Miguel Rojas, Mattia Vivanti, Andrea Arcuri, and Gordon Fraser. A detailed investigation of the effectiveness of whole test suite generation. *Empirical Software Engineering*, 22(2):852–893, Apr 2017.

Contacts

Thomas Vogel (`thomas.vogel@informatik.hu-berlin.de`)
Software Engineering Group
Institut für Informatik
Humboldt-Universität zu Berlin