Software Engineering Seminar

Model-Based Fault Localization

Description

Automated fault localization techniques assist developers with the task of pointing out program elements that are most probable to be responsible for a detected error. Over the years, many different techniques have been developed [5].

Model-based fault localization (MBFL) techniques have originally been developed for locating faulty components in physical systems, but have later been adopted to locate bugs in software systems. The general task in MBFL is to find discrepancies between the behavior of a model and the observed (or expected) behavior of a system. If a model of the system is not provided by the developer, then a model needs to first be generated. There exist various techniques to automatically generate system models with different abstraction levels from static information (e.g. source code, program language semantics) and/or from dynamic information (e.g. execution traces, runtime values).

The goal of this topic is to examine and discuss the current state of the art of MBFL techniques, to evaluate their relevancy and to analyze/estimate their capabilities compared to other fault localization techniques. Special attention should be given to the automatic generation of models that is performed by the techniques.

References

[1] Rui Abreu and Arjan JC Van Gemund. A low-cost approximate minimal hitting set algorithm and its application to model-based diagnosis. In *SARA*, volume 9, pages 2–9, 2009.

[2] Gerhard Friedrich, Markus Stumptner, and Franz Wotawa. Model-based diagnosis of hardware designs. *Artificial Intelligence*, 111(1-2):3–39, 1999.

[3] Wolfgang Mayer and Markus Stumptner. Model-based debugging–state of the art and future challenges. *Electronic Notes in Theoretical Computer Science*, 174(4):61–82, 2007.

[4] Wolfgang Mayer and Markus Stumptner. Evaluating models for model-based debugging. In *2008 23rd IEEE/ACM International Conference on Automated Software Engineering*, pages 128–137. IEEE, 2008.

[5] W. Eric Wong, Ruizhi Gao, Yihao Li, Rui Abreu, and Franz Wotawa. A survey on software fault localization. *IEEE Transactions on Software Engineering*, 42(8):707–740, 2016.

[6] Franz Wotawa, Mihai Nica, and Iulia Moraru. Automated debugging based on a constraint model of the program and a test case. *The journal of logic and algebraic programming*, 81(4):390–407, 2012.

[7] Franz Wotawa, Jörg Weber, Mihai Nica, and Rafael Ceballos. On the complexity of program debugging using constraints for modeling the program's syntax and semantics. In *Conference of the Spanish Association for Artificial Intelligence*, pages 22–31. Springer, 2009.

Contacts

Simon Heiden (`heiden@informatik.hu-berlin.de`)
Software Engineering Group
Institut für Informatik
Humboldt-Universität zu Berlin