Master's Thesis / Praxis der Forschung

Bislicing – Slicing for Relational Verification

Background. The problem whether two programs are equivalent is of great inte-rest in the daily practice of software development—especially in order to support evolving software systems. We developed reve, a tool that proves the equiva-lence of two C programs with the same behaviour on a local function level. This leads to the next challenge: the scalability on full software projects.

Idea. Lightweight analysis techniques provide *Program Dependence Graphs* (PDGs) that capture all dependencies between statements within one program. We can use the well-known theoretical result that two equivalent programs have isomor-phic PDGs in order to rapidly check whether certain parts of the two analyzed programs are equivalent. This would allow reve to focus on the more difficult program parts. We call this process of excluding equivalent parts (this result is taken from the PDG-analysis) of the two programs for the equivalence verification *bi-slicing*.

Task. The focus of this thesis should be a theoretical concept of bi-slicing for case equivalence checking. Implementation and evaluation are also on the agenda, but subordinate.

Your profile. You are a serious and competent student. You are interested in formal systems and/or formal languages. Programming in C++ poses no challenge to you. Knowledge taught in the lecture *Formale Systeme* (or similar) is required.

Kontakt

Mihai Herda   herda@kit.edu   50.34 R227