Bachelor Thesis: Join Block Contracts for a Dynamic Logic Calculus

TECHNISCHE UNIVERSITÄT DARMSTADT

Software Engineering Group – Dominic Scheurer
scheurer@cs.tu-darmstadt.de

Context

KeY is a theorem prover for first-order Java Dynamic Logic that can be employed in proving properties of Java programs using a sequent calculus. In the course of such proofs, the effects of the program of interest are evaluated by *Symbolic Execution*. This precise analysis technique, which treats input variables as symbols, transforms modalities containing Java programs into so-called *updates* capturing changes to program variables.

The data structure which the execution works upon is a tree of symbolic execution states. Whenever symbolic execution depends on the value of a program variable which has not yet been set to a concrete value, it splits up into sub branches. This may happen, for instance, during the evaluation of an if-statement or a loop. As a result, the number of branches in the final execution tree is up to exponential in the number of possible branching points. This problem is often referred to as the "*State Explosion Problem*".

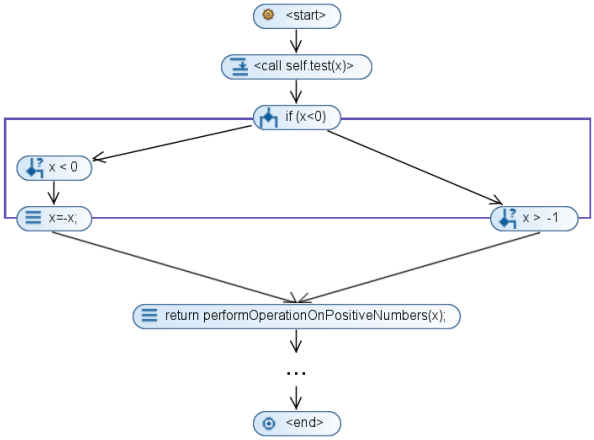
To address the State Explosion Problem, the Software Engineering Group developed techniques for joining states in the symbolic execution tree, for example after the execution of the two branches of an if-statement. So far, joins have to be performed manually in the proof process; only simpler cases are covered by simple strategies (macros) that aim to automatize the joining. A better solution would consist in a dedicated "Join Block Contract Rule", in the spirit of the already existing "Operation Contract Rule", taking into account a JML description of the join procedure to be applied, and ensuring that the join actually takes place at a later point in time during the execution which conforms to the specification.

Thesis

The goal of the thesis is the development and formal specification of a "Join Block Contract Rule" (or, if necessary, a set of rules) as outlined in the previous section, and an according extension of the implementation of KeY. Further possible objectives include the integration of the rule into KeY's automatic strategies and a formal soundness proof for the new rule.

Approximate Work Distribution

Analysis
Programming
Literature



Contact

Dominic Scheurer
Software Engineering Group
scheurer@cs.tu-darmstadt.de
Office: S2|02/A226, Phone +49 6151 16-21366