Decision Engines for Software Analysis using Satisfiability Modulo Theories Solvers

Nikolaj Bjørner
Microsoft Research
Redmond, Washington, USA

Abstract

The area of software analysis, testing and verification is now undergoing a revolution thanks to the use of automated and scalable support for logical methods. A well-recognized premise is that at the core of software analysis engines is invariably a component using logical formulas for describing states and transformations between system states. The process of using this information for discovering and checking program properties (including such important properties as safety and security) amounts to automatic theorem proving. In particular, theorem provers that directly support common software constructs offer a compelling basis. Such provers are commonly called satisfiability modulo theories (SMT) solvers.

Z3 is a state-of-the-art SMT solver. It is developed at Microsoft Research. It can be used to check the satisfiability of logical formulas over one or more theories such as arithmetic, bit-vectors, lists, records and arrays. The talk describes some of the technology behind modern SMT solvers, including the solver Z3. Z3 is currently mainly targeted at solving problems that arise in software analysis and verification. It has been applied to various contexts, such as systems for dynamic symbolic simulation (Pex, SAGE, Vigilante), for program verification and extended static checking (Spec#/Boggie, VCC, HAVOC), for software model checking (Yogi, SLAM), model-based design (FORMULA), security protocol code (F7), program run-time analysis and invariant generation (VS3). We will describe how it integrates support for a variety of theories that arise naturally in the context of the applications. There are several new promising avenues and the talk will touch on some of these and the challenges related to SMT solvers.