SMT and Temporal Logics

Pascal Fontaine and Stephan Merz
{Pascal.Fontaine,Stephan.Merz}@inria.fr

Context

VeriDis is a joint team of Inria Nancy – Grand-Est and of Max-Planck-Institut für Informatik in Saarbrücken whose objective is to develop techniques for the formal verification of distributed algorithms. Such verification problems are often reduced to proving logical formulas. For certain properties of algorithms, it is useful to reason about formulas in temporal logics that express facts about values of variables (and hence formulas) that evolve over time in executions of algorithms. Temporal logics are instances of the larger class of modal logics.

It is important to automate as far as possible the proof of logical formulas. SMT (satisfiability modulo theories) solvers (see [1] for an overview) are highly efficient tools for automatically determining the satisfiability of logical formulas in a first-order language where certain symbols are interpreted, such as the symbols of arithmetic. In general, SMT solvers are decision procedures for quite expressive, but quantifier-free languages; reasoning about quantified formulas is handled by instantiation. Temporal or modal connectives are not handled natively. We recently showed that SMT solvers can quite easily be adapted for obtaining a decision procedure for so-called Basic Modal Logic. Current decision procedures for modal or temporal logics are usually based on tableau or automata-theoretic procedures [3] or specific resolution techniques [4, 5].

Description of the research subject

The objective of the work proposed here is to study the use of SMT solvers first as decision procedures for extended modal logics, and then of temporal logics. In a first step, our results on Basic Modal Logic should be extended to account for particular properties (such as transitivity) of accessibility relations, and to cases where additional operators are used. For this, it is important to understand the standard translations of these logics to classical first-order logics and to define (and prove correct) a framework in which formulas of the studied logics are decided by a combination of an instantiation procedure and of a decision procedure for the quantifier-free fragment. Depending on the interests of the student, the resulting algorithms can be prototypically implemented in the SMT solver veriT [2].

The topic can be adapted and extended in different ways according to the interests of the student and the results obtained in the internship, and can in particular give rise to a future PhD subject.

References

[1] Clark Barrett, Roberto Sebastiani, Sanjit A. Seshia, and Cesare Tinelli. Satisfiability modulo theories. In Armin Biere, Marijn J. H. Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability*, volume 185 of *Frontiers in Artificial Intelligence and Applications*, chapter 26, pages 825–885. IOS Press, February 2009.

[2] Thomas Bouton, Diego Caminha B. de Oliveira, David Déharbe, and Pascal Fontaine. veriT: an open, trustable and efficient SMT-solver. In Renate Schmidt, editor, *Proc. Conference*

on Automated Deduction (CADE), volume 5663 of *Lecture Notes in Computer Science*, pages 151–156, Montreal, Canada, 2009. Springer.

[3] Edmund M. Clarke, Orna Grumberg, and Doron A. Peled. *Model Checking*. The MIT Press, Cambridge, Massachusetts, 1999.

[4] Renate A. Schmidt and Ullrich Hustadt. First-order resolution methods for modal logics. In Andrei Voronkov and Christoph Weidenbach, editors, *Programming Logics - Essays in Memory of Harald Ganzinger*, volume 7797 of *LNCS*, pages 345–391. Springer, 2013.

[5] Martin Suda and Christoph Weidenbach. A PLTL-prover based on labelled superposition with partial model guidance. In Bernhard Gramlich, Dale Miller, and Uli Sattler, editors, *6th Intl. Joint Conf. Automated Reasoning (IJCAR 2012)*, volume 7364 of *LNCS*, pages 537–543, Manchester, UK, 2012. Springer.