SAT in Saturation: A Satisfied Match

Laura Kovács
TU Wien, Austria

Abstract

Saturation is the leading concept behind the proof-search algorithms of state-of-the-art first-order theorem provers [3, 11, 10]. The key idea behind saturation-based proof search is to reduce the problem of proving validity of a first-order formula to the problem of establishing unsatisfiability of the respective formula, by using a sound inference system, such as resolution and superposition [2, 7]. Central to efficient saturation-based proof search is the implementation of redundancy in the form of simplification rules [9, 6]: such rules do not add new formulas to search space, but instead simplify/delete redundant formulas from the search space, while not loosing refutational completeness of superposition. Redundancy in first-order theorem proving is controlled via term/clause ordering and literal selection functions in extension of standard superposition: redundant clauses are logical consequences of smaller clauses with respect to the considered ordering.

While redundancy is essential for efficient proof search, establishing whether an arbitrary first-order formula is redundant is as hard as proving whether it is valid. First-order provers therefore implement sufficient conditions towards proving redundancy, so that these conditions can be efficiently checked, ideally using only syntactic arguments over formulas. One such condition comes with the notion of subsumption, yielding one of the most important simplification rules in automated reasoners [1]. It is common that millions of subsumption checks are performed during a single solver run [8]. However, in contrast to propositional subsumption as used by SAT solvers and implemented using sophisticated polynomial algorithms, first-order subsumption in first-order theorem proving involves NP-complete search queries, turning the efficient use of first-order subsumption into a huge practical burden.

This talks presents a tailored integration of SAT solving for detecting variants of subsumption in superposition. Key to our approach is retrieving clauses from the search space and checking whether subsumption with retrieved clauses can be applied, using multi-literal matching. A solution to our SAT-based encoding gives a concrete application of (variants of) subsumption, allowing the first-order prover to apply that instance of subsumption as a simplification rule during saturation [5, 8, 4]. Our SAT encoding captures subset relations among literals/clauses and formalizes matching of literals between inference premises/conclusions. We show that SAT encodings improve literal matching, and thus subsumption, in first-order theorem proving. In particular, our experimental results using the Vampire prover demonstrate the practical benefits of using SAT solving for variants of first-order subsumption.

2012 ACM Subject Classification Theory of computation → Automated reasoning; Theory of computation → Logic and verification; Computing methodologies → Theorem proving algorithms; Computing methodologies → Boolean algebra algorithms

Keywords and phrases Automated Reasoning, First-Order Theorem Proving, Superposition, Subsumption, Redundancy, SAT Solving, Vampire

Digital Object Identifier 10.4230/LIPIcs.SAT.2026.1

Category Invited Talk

Funding *Laura Kovács*: This work has been partially funded by the the European Research Council (ERC) Consolidator Grant ARTIST 101002685; the ERC Proof of Concept Grant LEARN 10121341; the Austrian Science Fund (FWF) 10.55776/DOC1345324; the FWF SFB project SpyCoDe F8504; the WWTF ICT22-007 grant ForSmart; and the SBA Research COMET Center SBA-K1 NGC.

Acknowledgements This talk is based on joint works with a number of authors, including Armin Biere, Robin Coutelier, Bernhard Gleiss, Jakob Rath, and Michael Rawson.

References

1 Leo Bachmair and Harald Ganzinger. Rewrite-Based Equational Theorem Proving with Selection and Simplification. *J. Log. Comput.*, 4(3):217–247, 1994. doi:10.1093/LOGCOM/4.3.217.

2 Leo Bachmair and Harald Ganzinger. Resolution Theorem Proving. In *Handbook of Automated Reasoning*. North-Holland, 2001. doi:10.1016/B978-044450813-3/50004-7.

3 Filip Bártek, Ahmed Bhayat, Robin Coutelier, Márton Hajdú, Matthias Hetzenberger, Petra Hozzová, Laura Kovács, Jakob Rath, Michael Rawson, Giles Reger, Martin Suda, Johannes Schoisswohl, and Andrei Voronkov. The Vampire Diary. In *CAV*, 2025. doi:10.1007/978-3-031-98682-6_4.

4 Robin Coutelier, Jakob Rath, Michael Rawson, Armin Biere, and Laura Kovács. SAT solving for variants of first-order subsumption. *Formal Methods Syst. Des.*, 67(1):27–70, 2025. doi:10.1007/S10703-024-00454-1.

5 Bernhard Gleiss, Laura Kovács, and Jakob Rath. Subsumption Demodulation in First-Order Theorem Proving. In *IJCAR*, pages 297–315, 2020. doi:10.1007/978-3-030-51074-9_17.

6 Laura Kovács and Andrei Voronkov. First-Order Theorem Proving and Vampire. In *CAV*, 2013. doi:10.1007/978-3-642-39799-8_1.

7 Robert Nieuwenhuis and Albert Rubio. Paramodulation-Based Theorem Proving. In *Handbook of Automated Reasoning (in 2 volumes)*, pages 371–443. Elsevier and MIT Press, 2001. doi:10.1016/B978-044450813-3/50009-6.

8 Jakob Rath, Armin Biere, and Laura Kovács. First-Order Subsumption via SAT Solving. In *FMCAD*, pages 160–169. IEEE, 2022. doi:10.34727/2022/ISBN.978-3-85448-053-2_22.

9 John Alan Robinson. A machine-oriented logic based on the resolution principle. *J. ACM*, 12(1):23–41, 1965. doi:10.1145/321250.321253.

10 Stephan Schulz, Simon Cruanes, and Petar Vukmirovic. Faster, Higher, Stronger: E 2.3. In *CADE*, 2019. doi:10.1007/978-3-030-29436-6_29.

11 Christoph Weidenbach, Dilyana Dimova, Arnaud Fietzke, Rohit Kumar, Martin Suda, and Patrick Wischnewski. SPASS version 3.5. In *CADE*, 2009. doi:10.1007/978-3-642-02959-2_10.