Structure-Based Satisfiability Checking

Analyzing and Harnessing the Potential

Matti Järvisalo
Helsinki University of Technology (TKK)
Dept. Information and Computer Science
P.O. Box 5400, FI-02015 TKK, Finland
E-mail: matti.jarvisalo@tkk.fi

This note summarizes a dissertation that focuses on methods for solving structured real-world problems based on propositional satisfiability (SAT). The work contributes to the analysis and development of both complete search (DPLL and clause learning) and stochastic local search methods for SAT.

Keywords: propositional satisfiability, SAT, clause learning, DPLL, stochastic local search, heuristics, proof complexity, problem structure, backdoors, answer set programming

1. Introduction

Constraint satisfaction deals with developing automated techniques for solving computationally hard problems in a declarative fashion. This note summarizes a dissertation that focuses on search-based methods for the propositional satisfiability problem (SAT).

As methods for propositional satisfiability checking have rapidly progressed during the last 15 years, implementations of decision procedures for SAT, so called SAT solvers [3], have been found to be extremely efficient as back-end search engines in solving large industrial-scale combinatorial problems. Since SAT solvers have become a standard tool for attacking various real-world problem instances of increasing size and difficulty, there is a demand for more and more robust and efficient solvers. For understanding the successes (and failures) of SAT solvers in specific problem domains, it is important to investigate how different types of structural properties of SAT instances are related to the efficiency of solving the instances with different SAT-based constraint satisfaction techniques. This is the underlying motivation for the thesis.

The emphasis of the thesis is on search-based SAT solving techniques for solving structured real-world problems. The work focuses on the analysis and development of both complete search (DPLL and Clause Learning (CL)) and stochastic local search (SLS) methods for SAT.

The thesis, available online as [5], consists of an overview and seven published articles [4,6,7,9,10,11, 12].

2. Contributions of the Thesis

The main contributions divide into three topics.

- Structure-Based Decision Heuristics [6,7,11,12]
- Structure-Exploiting Local Search [9,10]
- Hard Satisfiable SAT Benchmark Families [4]

Structure-Based Decision Heuristics. Branching heuristics, i.e., techniques for deciding on which variable to next set a value during search, play an important role in the efficiency of DPLL-style search. Intuitively, the structure of a problem domain is reflected in individual variables in the SAT encoding, and making decisions on structurally irrelevant variables may have an exponential effect on the running times of SAT solvers.

From the theoretical point of view, we investigate the *best-case* performance of SAT solvers using different structure-based branching heuristics through *proof complexity* [2] by studying the relative power of the inference systems (or *proof systems*) underlying the solvers.

The theoretical results presented in the thesis reveal inherent weaknesses in various structure-based branching heuristics suggested in the literature for DPLL-based SAT solvers. Extending our previous study of structure-based branching in DPLL [8], both static ("input-restricted") [7] and dynamic ("top-down") [6] branching heuristics are considered. For example, we show that by restricting branching in CL to variables encoding input (or independent) variables, the resulting system cannot efficiently simulate DPLL *without clause learning*. This is surprising, since without this restriction the considered variant of CL can efficiently simulate general resolution [1], being thus very pow-

AI Communications
ISSN 0921-7126, IOS Press.

erful compared to DPLL. This implies that all implementations of CL, even with optimal heuristics, have the potential of suffering a notable efficiency decrease if branching is restricted to input variables. Furthermore, considering various dynamic branching heuristics and branching styles [6] for circuit-level DPLL and CL solvers, we show that although the idea of eagerly justifying the values of currently unjustified constraints is an intuitively appealing one, it can lead to dramatic losses in the best-case efficiency of SAT solvers.

Complementing the theoretical results, a detailed experimental evaluation of the effects of structure-based branching on state-of-the-art clause learning SAT solvers was carried out [11]. Since statically restricted branching can be seen as *backdoor-based branching* (using possible non-minimal backdoor sets), the work provides insights into possible limitations on the applicability of backdoor sets for guiding branching heuristics in SAT solvers. As an example, the results confirm that input-restricted branching can cause a notable loss of robustness in a clause learning SAT solver.

In connection to branching in SAT, the work introduces the Extended ASP Tableaux proof system [12] in the context of answer set programming, which is a field closely related to SAT solving. The extension rule of Extended ASP Tableaux allows for adding redundant structure to ASP instances, resulting in the fact that ASP solvers, which are closely related to DPLL-based SAT solvers, may branch on these added substructures during search. In addition to theoretical observations on Extended ASP Tableaux, we experimentally study the effect of redundant structure in ASP instances on the efficiency of ASP solvers.

Structure-Exploiting Local Search. We address the challenge [13] of developing SLS SAT solvers which exploit variable dependencies in structured real-world SAT instances by developing novel structure-based local search techniques [9,10]. The idea is to drive local search top-down in the formula structure. By guiding the search using justification frontiers, we enable exploiting observability don't cares in the context of local search and drive the search to relevant parts of the formula structure [10]. We also analyze approximate completeness of the method, and develop new adaptive noise mechanisms aimed specifically for the method [9]. Experimentally, the method can outperform typical CNF-level SLS methods such as WalkSAT and AdaptNovelty+ up to 3–4 magnitudes of difference on a family of structured real-world instances.

Hard Satisfiable Instances. As the third point of view to structure in SAT, we develop methods for generating extremely hard satisfiable CNF SAT instances by introducing the *regular XORSAT* model [4]. The novelty lies in combining regularity and randomness in the context of XORSAT by employing random regular graphs as the underlying constraint graphs, motivated by their expansion properties. Both DPLL-based and local search state-of-the-art SAT solvers scale exponentially on regular XORSAT. This includes solvers with equivalence reasoning techniques, due to the novel techniques for adding nonlinearity into regular XORSAT developed in the work. Compared to several other families of satisfiable instances, regular XORSAT is among the hardest.

Acknowledgements

The work was supervised by Professor Ilkka Niemelä. The thesis committee consisted of Professors Bart Selman, Paul Beame, and João Marques-Silva. The work was funded by HeCSE graduate school, Academy of Finland (grants #122399 and #211025), and the foundations of Emil Aaltonen, Jenny and Antti Wihuri, Nokia, and TES.

References

[1] P. Beame, H.A. Kautz, and A. Sabharwal. Towards understanding and harnessing the potential of clause learning. *Journal of Artificial Intelligence Research*, 22:319–351, 2004.

[2] S.A. Cook and R.A. Reckhow. The relative efficiency of propositional proof systems. *J. Symbolic Logic*, 44(1):36–50, 1979.

[3] C.P. Gomes, H.A. Kautz, A. Sabharwal, and B. Selman. Satisfiability solvers. In *Handbook of Knowledge Representation*, chapter 2, pages 89–134. Elsevier, 2008.

[4] H. Haanpää, M. Järvisalo, P. Kaski, and I. Niemelä. Hard satisfiable clause sets for benchmarking equivalence reasoning techniques. *Journal on Satisfiability, Boolean Modeling and Computation*, 2(1-4):27–46, 2006.

[5] M. Järvisalo. *Structure-Based Satisfiability Checking: Analyzing and Harnessing the Potential*. Volume D10 of TKK Dissertations in Information and Computer Science. `http://lib.tkk.fi/Diss/2008/isbn9789512296422/`, Helsinki University of Technology, 2008.

[6] M. Järvisalo and T. Junttila. On the power of top-down branching heuristics. In *AAAI*, pages 304–309. AAAI Press, 2008.

[7] M. Järvisalo and T. Junttila. Limitations of restricted branching in clause learning. *Constraints*, to appear (2009). doi:10.1007/s10601-008-9062-z.

[8] M. Järvisalo, T. Junttila, and I. Niemelä. Unrestricted vs restricted cut in a tableau method for Boolean circuits. *Annals of Mathematics and Artificial Intelligence*, 44(4):373–399, 2005.

[9] M. Järvisalo, T. Junttila, and I. Niemelä. Justification-based local search with adaptive noise strategies. In *LPAR*, volume 5330 of *LNCS*, pages 31–46. Springer, 2008.

[10] M. Järvisalo, T. Junttila, and I. Niemelä. Justification-based non-clausal local search for SAT. In *ECAI*, pages 535–539. IOS Press, 2008.

[11] M. Järvisalo and I. Niemelä. The effect of structural branching on the efficiency of clause learning SAT solving: An experimental study. *Journal of Algorithms*, 63(1–3):90–113, 2008.

[12] M. Järvisalo and E. Oikarinen. Extended ASP tableaux and rule redundancy in normal logic programs. *Theory and Practice of Logic Programming*, 8(5–6):691–716, 2008.

[13] H.A. Kautz and B. Selman. The state of SAT. *Discrete Applied Mathematics*, 155(12):1514–1524, 2007.