UWrMaxSat - a new MiniSat+-based Solver in MaxSAT Evaluation 2019

Marek Piotrów
Institute of Computer Science, University of Wrocław
Wrocław, Poland
marek.piotrow@uwr.edu.pl

***Abstract*—UWrMaxSat is a new MiniSat+-based solver participating in MaxSAT Evaluation 2019. It has been created recently at the University of Wrocław. It is a complete solver for partial weighted MaxSAT instances. It incrementally uses COMiniSatPS by Chanseok Oh (2016) as an underlying SAT solver, but may be compiled with other MiniSat-like solvers. It was developed on the top of our PB-solver (called kp-minisatp) that was presented at Pragmatics of SAT 2018 and which is an extension of the well-known MiniSat+ solver. In its main configuration, UWrMaxSat applies an unsatisfiability-core-based OLL procedure and uses the kp-minisatp sorter-based pseudo-Boolean constraint encoding to translate new cardinality constraints into CNF.**

***Index Terms*—MaxSAT-solver, UWrMaxSAT, COMiniSatPS, sorter-based encoding, core-guided, complete solver**

I. Introduction

At Pragmatics of SAT 2018 workshop, Michał Karpiński and Marek Piotrów presented a new pseudo-Boolean constraint solver called kp-minisatp [8] that was created as an extension of MiniSat+ 1.1 solver by Eén and Sörensson (2012) [6]. In the solver we replaced the encoding based on odd-even sorting networks by a new one using our construction of selection networks called 4-Way Merge Selection Networks [9]. We also optimized mixed radix base searching procedure and added a few other optimizations based on literature. Our experiments showed that the solver is competitive to other state-of-art solvers.

Believing that the encoding can be also used in MaxSAT solvers, I have implemented such a solver on the top of kp-minisatp and the result called UWrMaxSat is submitted to MaxSAT Evaluation 2019.

II. Description

The solver is prepared to be compiled with one of a few MiniSat-like SAT solvers: COMiniSatPS by Chanseok Oh (2016), Glucose 4.1 (2016) and 3.0 (2013) by Gilles Audemard and Laurent Simon [3], and original Minisat 2.2 (2010) by Niklas Eén and Niklas Sörensson [4]. The first one was selected as the default SAT solver and it is used incrementally with the help of assumptions [5]. Other MiniSat-like SAT solvers can be also applied, because the interface between UWrMaxSAT and a SAT solver is small and defined by MiniSat.

Three different core-guided searching strategies have been implemented in the solver: a linear unsat-sat one, a linear sat-unsat one and a binary search sat-unsat one. The first one was selected as default and it was optimized much more than the others. In order to process unsatisfiability cores in the default strategy we select the OLL procedure [1] and encode its cardinality constraints by our encoding based on 4-Way Merge Selection Network. The encoding uses also Direct Networks for constraints with a small number of literals [2]. A core is minimized [11] before it is converted into a cardinality constraint.

In case of weighted instances, several other optimization techniques are applied:

- A stratification technique: Soft clauses are sorted and grouped by weights; the groups of relaxed soft clauses are delivered to the SAT solver gradually, starting from ones with the largest weights. We use only one relaxation variable per non-unit soft clause.
- A hardening technique: The algorithm refines both lower and upper bounds on the weight of a solution. Base on their values, the heaviest soft clauses can be transformed into hard ones.
- A BMO technique [10]: Some weighted instances can encode multi-objective problems and the optimality criterion is lexicographic. Such instances are detected and the search procedure is optimized accordingly.
- A preprocessing technique: Soft clauses can be preprocess to detect unit cores and at-most-one cores. Such cores are encoded in a more efficient way (implemented based on the code of RC2, see [7]).
- A mixed strategy technique: If the linear unsat-sat searching is unsuccessful for a predefined time, it can be switched to the binary searching without restarting the SAT solver.

Finally, the solver can deal with unbounded integer weights when it is compiled with the -D BIG_WEIGHTS option.

III. Acknowledgments

I would like to thank Chankseok Oh for allowing me to use COMiniSatPS in the MaxSAT Evaluation. I would like also to thank Niklas Eén and Niklas Sörensson for the development of MiniSat 2.2 and MiniSat+ 1.1.

References

[1] Benjamin Andres, Benjamin Kaufmann, Oliver Matheis, and Torsten Schaub. Unsatisfiability-based optimization in clasp. In *Technical Communications of the Twenty-eighth International Conference on Logic Programming (ICLP'12*, volume 17, pages 212–221, 2012.

[2] Roberto Asín, Robert Nieuwenhuis, Albert Oliveras, and Enric Rodríguez-Carbonell. Cardinality networks: a theoretical and empirical study. *Constraints*, 16(2):195–221, 2011.

[3] Gilles Audemard and Laurent Simon. Predicting learnt clauses quality in modern sat solvers. In *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, IJCAI'09, pages 399–404, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[4] Niklas Eén and Niklas Sörensson. An extensible sat-solver. In Enrico Giunchiglia and Armando Tacchella, editors, *Theory and Applications of Satisfiability Testing*, pages 502–518, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[5] Niklas Eén and Niklas Sörensson. Temporal induction by incremental sat solving. *Electronic Notes in Theoretical Computer Science*, 89(4):543 – 560, 2003. BMC'2003, First International Workshop on Bounded Model Checking.

[6] Niklas Eén and Niklas Sörensson. Translating pseudo-boolean constraints into sat. *Journal on Satisfiability, Boolean Modeling and Computation*, 2:1–26, 2006.

[7] Alexey Ignatiev, Antonio Morgado, and Joao Marques-Silva. PySAT: A Python toolkit for prototyping with SAT oracles. In *SAT*, pages 428–437, 2018.

[8] Michał Karpiński and Marek Piotrów. Competitive sorter-based encoding of pb-constraints into sat. In Daniel Le Berre and Matti Järvisalo, editors, *Proceedings of Pragmatics of SAT 2015 and 2018*, volume 59 of *EPiC Series in Computing*, pages 65–78. EasyChair, 2019.

[9] Michał Karpiński and Marek Piotrów. Encoding cardinality constraints using multiway merge selection networks. *Constraints*, Apr 2019.

[10] Joao Marques-Silva, Josep Argelich, Ana Graça, and Inês Lynce. Boolean lexicographic optimization: algorithms & applications. *Annals of Mathematics and Artificial Intelligence*, 62(3):317–343, Jul 2011.

[11] João P. Marques Silva. Minimal unsatisfiability: Models, algorithms and applications (invited paper). pages 9–14, 01 2010.