

UWrMaxSat: an Efficient Solver in MaxSAT Evaluation 2020

Marek Piotrów

Institute of Computer Science, University of Wrocław

Wrocław, Poland

marek.piotrow@uwr.edu.pl

Abstract—UWrMaxSat 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 KP-MiniSat+ - our PB-solver 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 sorter-based pseudo-Boolean constraint encoding that was first implemented in kp-minisatp to translate cardinality and pseudo-Boolean constraints into CNF. It can switch to a binary search after a given time and then it uses our new encoding of a pseudo-Boolean goal function, where different bounds on its value are set only by assumptions.

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-MiniSat+ [6] that was created as an extension of MiniSat+ 1.1 solver by Eén and Sörensson (2012) [3]. 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 [7]. We also optimized mixed radix base searching procedure and added a few other optimizations based on literature. This year the encoding was extended in such a way that a goal function is encoded only once and then SAT-solver assumptions are used to set different bounds on its value. Our experiments showed that the solver is competitive to other state-of-art pseudo-Boolean solvers.

In 2018 KP-MiniSat+ was extended to deal with MaxSAT instances and called UWrMaxSat. In 2019 the new solver was submitted to MaxSAT Evaluation, where it was ranked second places in both main tracks: Weighted Complete Track and Unweighted Complete Track.

II. DESCRIPTION

In this year version of UWrMaxSat (denoted as 1.1), there are several extension to the version 1.0 submitted to MaxSAT Evaluation 2019. We give a brief description of them below. For the main features of UWrMaxSat 1.0 see [13]. We continue to use incrementally COMiniSatPS by Chanseok Oh (2016) [12] as an underlying SAT solver, but this year it was patched

to better deal with assumptions: all of them are enqueued at once (at level 1) by the SAT solver. The technique was proposed by Hickey and Bacchus in [4].

The default search strategy used by UWrMaxSat is a core-guided linear unsat-sat one, where unsatisfiability cores are processed by the OLL procedure [1], [5], [10] and cardinality constraints generated by it are encoded with the help of 4-Way Merge Selection Networks [7] and Direct Networks [2]. The general description of search strategies used by MaxSat solvers can be found, for example, in [11].

If the linear unsat-sat searching is unsuccessful for a pre-defined time, it can be switched to a binary search similar to that of MiniSat+ [3] without restarting the SAT solver. In such a case, a pseudo-Boolean goal function is created for the relaxation variables of the remaining soft clauses and encoded by our new technique [8], where the function is translated into SAT clauses only once and the different bounds on its value are set and submitted to the SAT solver by assumptions. The lower and upper bounds found before the switching define an initial binary-search interval.

Due to a new “top-k” track and the corresponding requirements of MaxSat Evaluation 2020, UWrMaxSat can now output the top k models of an MaxSat instance (starting from the optimal one). To use this, an argument `-top=k` should be given to UWrMaxSat. In the solver, the hardening technique was changed in the following way: a decreasing sequence of upper bounds is recorded and if there are at least k of them, then the last k -th one is used in the hardening procedure. After each solution found, a blocking clause consisting of relaxation variables is added to the instance, the value of k is decreased and the search is continued. In addition, the models can be output as a binary 0-1 string (with the help of `-bm`).

Finally, the solver can deal with unbounded integer weights, when it is compiled with the `-D BIG_WEIGHTS` option. Moreover, it can be linked with MaxPre, an extended open-source preprocessor for weighted partial MaxSAT problems, which was created at University of Helsinki a few years ago [9]. To this end, it should be compiled with the `-D MAXPRE` option. Note that none of these two options was used in the competition version of UWrMaxSat 1.1, which is submitted to both complete tracks and both top-k tracks (weighted and unweighted). The switching of the search techniques is only used in the complete-weighted track.

ACKNOWLEDGMENTS

I would like to thank Chankseok Oh for his agreement 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 An Int. J.*, 16(2):195–221, 2011.
- [3] Niklas Eén and Niklas Sörensson. Translating pseudo-boolean constraints into sat. *Journal on Satisfiability, Boolean Modeling and Computation*, 2:1–26, 2006.
- [4] Randy Hickey and Fahiem Bacchus. Speeding up assumption-based sat. In Mikoláš Janota and Inês Lynce, editors, *Theory and Applications of Satisfiability Testing – SAT 2019*, pages 164–182, Cham, 2019. Springer International Publishing.
- [5] Alexey Ignatiev, Antonio Morgado, and Joao Marques-Silva. Rc2: an efficient maxsat solver. *Journal on Satisfiability, Boolean Modeling and Computation*, pages 53–64, 2019.
- [6] 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.
- [7] Michał Karpiński and Marek Piotrów. Encoding cardinality constraints using multiway merge selection networks. *Constraints An Int. J.*, Apr 2019.
- [8] Michał Karpiński and Marek Piotrów. Incremental encoding of pseudo-boolean goal functions based on comparator networks. In *Proceedings of the 23th International Conference on Theory and Applications of Satisfiability Testing (SAT 2020)*, to appear in *Lecture Notes in Computer Science*. Springer, 2020.
- [9] Tuukka Korhonen, Jeremias Berg, Paul Saikko, and Matti Järvisalo. MaxPre: An extended MaxSAT preprocessor. In Serge Gaspers and Toby Walsh, editors, *Proceedings of the 20th International Conference on Theory and Applications of Satisfiability Testing (SAT 2017)*, volume 10491 of *Lecture Notes in Computer Science*, pages 449–456. Springer, 2017.
- [10] Antonio Morgado, Carmine Dodaro, and Joao Marques-Silva. Core-guided maxsat with soft cardinality constraints. In Barry O’Sullivan, editor, *Principles and Practice of Constraint Programming*, *Lecture Notes in Computer Science*, pages 564–573, Cham, 2014. Springer International Publishing.
- [11] António Morgado, Federico Heras, Mark H. Liffiton, Jordi Planes, and João Marques-Silva. Iterative and core-guided maxsat solving: A survey and assessment. *Constraints An Int. J.*, 18(4):478–534, 2013.
- [12] Chankseok Oh. *Improving SAT Solvers by Exploiting Empirical Characteristics of CDCL*. PhD thesis, New York University, 2016.
- [13] Marek Piotrów. Uwrmaxsat - a new minisat+-based solver in maxsat evaluation 2019. pages 11–12, 2019.