UWrMaxSat in MaxSAT Evaluation 2021

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

***Abstract*—UWrMaxSat is a complete solver that can be used to solve not only partial weighted MaxSAT instances but also pseudo-Boolean ones. It can be also characterized as anytime solver, because it outputs the best known solution, when its run is interrupted. It incrementally uses COMiniSatPS by Chanseok Oh (2016) as an underlying SAT solver, but may be compiled with a few other solvers as well. In its main configuration, UWrMaxSat applies a core-guided OLL procedure, where it uses own sorter-based pseudo-Boolean constraint encoding to translate cardinality constraints into CNF. It can switch to a binary search strategy after a fixed number of conflicts and then it uses an improved 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

A short history of the solver is as follows: In 2018 Michał Karpiński and Marek Piotrów created a new pseudo-Boolean (PB) constraint solver called KP-MiniSat+ [7] as an extension of MiniSat+ 1.1 solver by Eén and Sörensson (2012) [5]. 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 [8]. We also optimized mixed-radix base searching procedure and added a few other optimizations based on literature. In 2020 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 PB solvers.

At the end of 2018, KP-MiniSat+ was extended to deal with MaxSAT instances and renamed to UWrMaxSat. Three different solving techniques for MaxSAT problems were added to UWrMaxSat together with a translation of PB instances to MaxSAT ones, and vice-versa. In 2019 and 2020 the solver was submitted to MaxSAT Evaluations, where it won the complete-weighted track in 2020 and was ranked second places in both complete-weighted and complete-unweighted tracks in 2019.

This year a new version is submitted to the evaluation with two main additional features: (1) a greedy algorithm is added to the encoder module to check if some of previously encoded sorting networks can be reused in a new encoding, and (2) an algorithm is implemented in the solver module to detect, so called, generalized Boolean multilevel optimization (GBMO) splitting points, defined in [12].

II. Description

This year version of UWrMaxSat is denoted as 1.2. For the main features of versions 1.0 and 1.1 see [13], [14]. A more detailed description of UWrMaxSat ver. 1.1 can be found in [15]. In the current version, we continue to use incrementally COMiniSatPS by Chanseok Oh (2016) [11] as an underlying SAT solver. The default search strategy for the optimal solution is also the same as in previous years, that is, a core-guided linear unsat-sat one, where unsatisfiability cores are processed by the OLL procedure [1], [6], [9] and cardinality constraints generated by it are encoded with the help of 4-Way Merge Selection Networks [8] and Direct Networks [4]. If the linear unsat-sat searching is unsuccessful after a predefined number of conflicts, it can be switched to a binary search technique similar to that of the original MiniSat+ [5], without restarting the SAT solver. Note that previously it was done after a predefined number of seconds. We change this to make the solver less hardware dependent. The general description of search strategies used by MaxSAT solvers can be found, for example, in [10].

It is well known, that a sorter-based encoder gets an input sequence of literals $(x_1, \ldots, x_n)$ and its task is to produce a new sequence of literals $(y_1, \ldots, y_n)$ and a set of clauses such that, for any $k \in \{1, \ldots, n\}$, whenever any subset of inputs of size k is set to 1 by the SAT solver, its unit propagation procedure forces all $y_1, \ldots, y_k$ to be also 1. Note that the order of literals in $(x_1, \ldots, x_n)$ is not important in this task, so the encoder can change it. In the process of solving, the encoder is called many times with different input sequences and it can happen that subsets of the current input have already been encoded. Such a situation is detected in our improved encoder and the corresponding clauses are reused. A greedy strategy with respect to the size of a subsets is implemented to reduce the complexity of the algorithm. The impact of this improvement was mainly observed in solving PB instances.

GBMO was first defined and explored by Paxian, Raiola and Becker in [12], as a preprocessing technique for weighted MaxSAT instances. We have implemented a detection procedure of GBMO splitting points in the MaxSAT solving module, but the detected ones are not fully explored yet in our solving algorithm (due to time limitations). We just used them in the stratification heuristic [2], [3] to harden a corresponding soft constraint. We are planning to use them in all solving strategies, in particular, in the binary-search one.

Finally, the parser of UWrMaxSat was modified to accept

the new proposed input format of partial weighted MaxSAT instances. Recall that there is no p-line in it and hard clauses are preceded by the letter 'h' instead of the "very big" weight given in a p-line.

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] Carlos Ansótegui, Maria Luisa Bonet, Joel Gabàs, and Jordi Levy. Improving sat-based weighted maxsat solvers. In Michela Milano, editor, *Proc. CP*, pages 86–101. Springer, 2012.

[3] Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. Sat-based maxsat algorithms. *Artificial Intelligence*, 196:77 – 105, 2013.

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

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

[6] Alexey Ignatiev, Antonio Morgado, and Joao Marques-Silva. Rc2: an efficient maxsat solver. *Journal on Satisfiability, Boolean Modeling and Computation*, pages 53–64, 2019.

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

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

[9] Antonio Morgado, Carmine Dodaro, and Joao Marques-Silva. Core-guided maxsat with soft cardinality constraints. In Barry O'Sullivan, editor, *Proc. CP*, LNCS, pages 564–573. Springer, 2014.

[10] António Morgado, Federico Heras, Mark H. Liffiton, Jordi Planes, and João Marques-Silva. Iterative and core-guided maxsat solving: A survey and assessment. *Constraints*, 18(4):478–534, 2013.

[11] Chanseok Oh. *Improving SAT Solvers by Exploiting Empirical Characteristics of CDCL*. PhD thesis, New York University, 2016.

[12] Tobias Paxian, Pascal Raiola, and Bernd Becker. On preprocessing for weighted maxsat. In Fritz Henglein, Sharon Shoham, and Yakir Vizel, editors, *Verification, Model Checking, and Abstract Interpretation*, pages 556–577, Cham, 2021. Springer International Publishing.

[13] Marek Piotrów. Uwrmaxsat - a new minisat+-based solver in maxsat evaluation 2019. In Fahiem Bacchus, Matti Järvisalo, and Ruben Martins, editors, *MaxSAT Evaluation 2019 : Solver and Benchmark Descriptions*, Department of Computer Science Report Series B-2019-2, pages 11–12, 2019.

[14] Marek Piotrów. Uwrmaxsat - an efficient solver in maxsat evaluation 2020. In Fahiem Bacchus, Jeremias Berg, Matti Järvisalo, and Ruben Martins, editors, *MaxSAT Evaluation 2020 : Solver and Benchmark Descriptions*, Department of Computer Science Report Series B-2020-2, pages 34–35, 2020.

[15] Marek Piotrów. Uwrmaxsat: Efficient solver for maxsat and pseudo-boolean problems. In *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 132–136, 2020.