

Open-WBO in MaxSAT Evaluation 2017

Ruben Martins[†], Miguel Terra-Neves^{*}, Saurabh Joshi[‡], Mikoláš Janota^{*}, Vasco Manquinho^{*}, Inês Lynce^{*}

[†]University of Texas at Austin / Carnegie Mellon University, USA

^{*}INESC-ID / Instituto Superior Técnico, Universidade de Lisboa, Portugal

[‡]Indian Institute of Technology, Hyderabad, India

I. INTRODUCTION

Open-WBO is an open source MaxSAT solver that started as a spin-off of WBO [1]. Open-WBO implements a variety of algorithms for solving Maximum Satisfiability (MaxSAT) and Pseudo-Boolean (PB) formulas. The algorithms used in Open-WBO are based on a sequence of calls to a SAT solver. Even though Open-WBO can use any MiniSAT-like solver [2], for the purpose of this evaluation we are currently using Glucose 4.1 [3]. The key novelties of Open-WBO are: (i) incremental MaxSAT solving [4] and (ii) partitioning-based MaxSAT solving [5], [6], [7]. Open-WBO is particularly efficient for partial MaxSAT and has been one of the best solvers in the MaxSAT Evaluations of 2014, 2015 and 2016. Two versions of Open-WBO were submitted to the MaxSAT Evaluation 2017: LSU and RES. The remainder of this document describes the algorithms and encodings used in each version.

II. OPEN-WBO 2017: LSU VERSION

The LSU version is based on a linear search algorithm SAT-UNSAT [8] with lexicographical optimization for weighted problems [9]. This algorithm works by performing a sequence of calls to a SAT solver and refining an upper bound μ on the number of unsatisfied soft clauses. To restrict μ at each iteration, we need to encode a cardinality constraint (pseudo-Boolean constraint) for unweighted (weighted) problems into CNF. The LSU version uses the Modulo Totalizer encoding [10] for cardinality constraints and the Generalized Totalizer encoding (GTE) [11] for pseudo-Boolean constraints.

III. OPEN-WBO 2017: RES VERSION

The RES version is based on the unsatisfiability-based algorithms MSU3 [12] and OLL [13]. These algorithms work by iteratively refining a lower bound λ on the number of unsatisfied soft clauses until an optimum solution is found. Both MSU3 and OLL use the Totalizer encoding for incremental MaxSAT solving [4]. For unweighted MaxSAT, we extended the incremental MSU3 algorithm [4] with resolution-based partitioning techniques [7]. We represent a MaxSAT formula using a resolution-based graph representation and iteratively join partitions by using a proximity measure extracted from the graph representation of the formula. The algorithm ends when only one partition remains and the optimal solution is found. Since the partitioning of some MaxSAT formulas may be unfeasible or not significant, we heuristically choose to run MSU3 with or without partitions. In particular, we do not use partition-based techniques when one of the following criteria

is met: (i) the formula is too large ($> 1,000,000$ clauses), (ii) the ratio between the number of partitions and soft clauses is too high (> 0.8), or (iii) the sparsity of the graph is too small (< 0.04). Currently, Open-WBO only supports partition-based techniques for unweighted problems. For weighted MaxSAT, we use the OLL MaxSAT algorithm [13].

IV. AVAILABILITY

The first release of Open-WBO is available under a MIT license at <http://sat.inesc-id.pt/open-wbo/>. The second release of Open-WBO is available under a MIT license in Github at <https://github.com/sat-group/open-wbo>. This version includes the partitioning techniques that made Open-WBO one of the best solvers for partial MaxSAT in the MaxSAT Evaluations of 2015 and 2016. To contact the authors please send an email to: open-wbo@sat.inesc-id.pt.

ACKNOWLEDGMENTS

We would like to thank Laurent Simon and Gilles Audemard for allowing us to use Glucose in the MaxSAT Evaluation.

REFERENCES

- [1] V. Manquinho, J. Marques-Silva, and J. Planes, "Algorithms for Weighted Boolean Optimization," in *SAT*. Springer, 2009, pp. 495–508.
- [2] N. Eén and N. Sörensson, "An Extensible SAT-solver," in *SAT*. Springer, 2003, pp. 502–518.
- [3] G. Audemard and L. Simon, "Predicting Learnt Clauses Quality in Modern SAT Solvers," in *IJCAI*, 2009, pp. 399–404.
- [4] R. Martins, S. Joshi, V. Manquinho, and I. Lynce, "Incremental Cardinality Constraints for MaxSAT," in *CP*. Springer, 2014, pp. 531–548.
- [5] R. Martins, V. Manquinho, and I. Lynce, "On Partitioning for Maximum Satisfiability," in *ECAI*. IOS Press, 2012, pp. 913–914.
- [6] R. Martins, V. M. Manquinho, and I. Lynce, "Community-based partitioning for maxsat solving," in *SAT*. Springer, 2013, pp. 182–191.
- [7] M. Neves, R. Martins, M. Janota, I. Lynce, and V. M. Manquinho, "Exploiting Resolution-Based Representations for MaxSAT Solving," in *SAT*. Springer, 2015, pp. 272–286.
- [8] D. Le Berre and A. Parrain, "The Sat4j library, release 2.2," *Journal on Satisfiability, Boolean Modeling and Computation*, vol. 7, no. 2-3, pp. 59–6, 2010.
- [9] J. Marques-Silva, J. Argelich, A. Graça, and I. Lynce, "Boolean lexicographic optimization: algorithms & applications," *Annals of Mathematics and Artificial Intelligence*, vol. 62, no. 3-4, pp. 317–343, 2011.
- [10] T. Ogawa, Y. Liu, R. Hasegawa, M. Koshimura, and H. Fujita, "Modulo Based CNF Encoding of Cardinality Constraints and Its Application to MaxSAT Solvers," in *ICTAI*. IEEE, 2013, pp. 9 – 17.
- [11] S. Joshi, R. Martins, and V. M. Manquinho, "Generalized Totalizer Encoding for Pseudo-Boolean Constraints," in *CP*. Springer, 2015, pp. 200–209.
- [12] J. Marques-Silva and J. Planes, "On Using Unsatisfiability for Solving Maximum Satisfiability," *CoRR*, 2007.
- [13] A. Morgado, C. Dodaro, and J. Marques-Silva, "Core-Guided MaxSAT with Soft Cardinality Constraints," in *CP*. Springer, 2014, pp. 564–573.