MaxHS in the 2022 MaxSat Evaluation

Fahiem Bacchus
Department of Computer Science
University of Toronto
Toronto, Ontario, Canada
Email: fbacchus@cs.toronto.edu

1. MaxHS

MaxHS originated in the work of Davies [4] who developed the first MaxSat solver based on the Implicit Hitting Set approach (IHS). The core components of MaxHS are described in [5], [6], [7]. The PhD thesis of Saikko [9] also provides an excellent overview of the IHS approach along with a number of additional insights. In addition to various algorithmic and code improvements over the years, MaxHS also employs the techniques of reduced cost fixing [1] and abstract cores [2]. Both of these techniques go beyond the basic IHS approach.

2. 2022

The 2022 version of MaxHS is built on top of the 2021 version. As noted in last years entry, as MaxSat instances have become larger the relative advantages of using a more powerful Sat solver have increased. Hence, MaxHS now uses the Cadical solver for Sat solving [3] (the potentially more efficient Kissat does not fully support all of the Sat solver features, e.g., assumptions, needed by MaxHS). It also uses IBM's commercial Mixed Integer Programming Solver (IBM CPLEX version 20.1.0.0) under IBM's Academic Initiative licencing program.

In 2022 the input parser and option processing subsystems were rewritten, and the preprocessing phase was redesigned and reimplemented. The MaxSat preprocessor MaxPre [8] had been tried before without positive results. This arises from a different processing of soft clause "labels" (also known as the relaxation or blocking variables) done in MaxPre which often produces far more blocking variables than MaxHS. In particular, MaxPre will introduce a new "label" even for unit soft clauses (when the literal of that soft clause appears in both polarities in the rest of the formula). MaxHS on the other hand always reuses the literal of unit softs as the "label", irrespective of its purity in the rest of the formula. Since much of the MaxSat processing is dependent the number of blocking variables, we have found that MaxHS can be more effective without MaxPre. Nevertheless, sometimes MaxPre is able to simplify the problem more profoundly. In the 2022 code, MaxPre is run, but the simplified formula it produces is only used if it contains fewer labels than the original formula. In future

work we plan to identify and integrate only those parts of MaxPre that prove to be useful (which mainly seem to be the technique "generalized subsumed label elimination").

The second change introduced in 2022 was to identify a wider range of special input cases and then configure the solver more specifically for solving these special cases. The special cases identified in the 2022 code are (a) when the input is a hitting set problem, (b) when the input has a small number of soft clauses, (c) when every variable of the formula forms a unit soft clause, and (d) when the input consists of a small number of variables. Cases (a) and (b) are newly identified in 2022. For case (a) we have found that using the MIP solver tended to produce the best results, and for case (b) using a simple time bounded Linear Sat Unsat algorithm which can improve results in the weighted case. Case (c) and (d) were handled in the 2021 code, and are processed by a mixture of first trying the MIP solver for a short period of time, then trying the technique of abstract cores, and then utilizing the information gathered so far for a final round of MIP solving. In future work, we plan on monitoring the progress of these "trial" phases more carefully so that better decisions can be made about when to abort these phases.

References

[1] Fahiem Bacchus, Antti Hyttinen, Matti Järvisalo, and Paul Saikko. Reduced cost fixing in MaxSAT. In J. Christopher Beck, editor, *Principles and Practice of Constraint Programming - 23rd International Conference, CP 2017, Melbourne, VIC, Australia, August 28 - September 1, 2017, Proceedings*, volume 10416 of *Lecture Notes in Computer Science*, pages 641–651. Springer, 2017.

[2] Jeremias Berg, Fahiem Bacchus, and Alex Poole. Abstract cores in implicit hitting set maxsat solving. In Luca Pulina and Martina Seidl, editors, *Theory and Applications of Satisfiability Testing - SAT 2020 - 23rd International Conference, Alghero, Italy, July 3-10, 2020, Proceedings*, volume 12178 of *Lecture Notes in Computer Science*, pages 277–294. Springer, 2020.

[3] A. Biere, K. Fazekas, M. Fleury, and M. Heisinger. Cadical, kissat, paracooba, plingeling and treengeling entering the sat competition 2020. In *Proceedings of the SAT Competition 2020–Solver and Benchmark Descriptions*, volume B-2020-1 of *Department of Computer Science Report Series B*, page 5153. University of Helsinki, 2020.

[4] Jessica Davies. *Solving MAXSAT by Decoupling Optimization and Satisfaction*. PhD thesis, University of Toronto, 2013. https://tspace.library.utoronto.ca/bitstream/1807/43539/6/Davies_Jessica_E_201311_PhD_thesis.pdf.

[5] Jessica Davies and Fahiem Bacchus. Solving MAXSAT by solving a sequence of simpler SAT instances. In Jimmy Ho-Man Lee, editor, *Principles and Practice of Constraint Programming - CP 2011 - 17th International Conference, CP 2011, Perugia, Italy, September 12-16, 2011. Proceedings*, volume 6876 of *Lecture Notes in Computer Science*, pages 225–239. Springer, 2011.

[6] Jessica Davies and Fahiem Bacchus. Exploiting the power of MIP solvers in MaxSAT. In Matti Järvisalo and Allen Van Gelder, editors, *Theory and Applications of Satisfiability Testing - SAT 2013 - 16th International Conference, Helsinki, Finland, July 8-12, 2013. Proceedings*, volume 7962 of *Lecture Notes in Computer Science*, pages 166–181. Springer, 2013.

[7] Jessica Davies and Fahiem Bacchus. Postponing optimization to speed up MAXSAT solving. In Christian Schulte, editor, *Principles and Practice of Constraint Programming - 19th International Conference, CP 2013, Uppsala, Sweden, September 16-20, 2013. Proceedings*, volume 8124 of *Lecture Notes in Computer Science*, pages 247–262. Springer, 2013.

[8] Tuukka Korhonen, Jeremias Berg, Paul Saikko, and Matti Järvisalo. MaxPre: An extended MaxSAT preprocessor. In Serge Gaspers and Toby Walsh, editors, *Theory and Applications of Satisfiability Testing - SAT 2017 - 20th International Conference, Melbourne, VIC, Australia, August 28 - September 1, 2017, Proceedings*, volume 10491 of *Lecture Notes in Computer Science*, pages 449–456. Springer, 2017.

[9] Paul Saikko. *Implicit hitting set algorithms for constraint optimization*. PhD thesis, University of Helsink, 2019. https://helda.helsinki.fi/bitstream/handle/10138/306951/implicit.pdf?sequence=1&isAllowed=y.