SAT-Based Approaches for the General High School Timetabling Problem*

Emir Demirović
Database and Artificial Intelligence Group
Vienna University of Technology (TU Wien), Vienna, Austria
demirovic@dbai.tuwien.ac.at

Abstract

High School Timetabling (HSTT) is a well known and widespread problem. It consists of coordinating resources (e.g. teachers, rooms), times, and events (e.g. lectures) with respect to various constraints. In this paper, I summarize the work I have done towards exploring the relationship between propositional logic and HSTT. This includes various modeling techniques in the form of maxSAT and bitvectors, data structures for local search algorithms, and the combination of maxSAT and metaheuristic algorithms. In addition, I discuss possible directions for future work as a part of a long-term research goal to combine complete and metaheuristic algorithms.

1 Long-Term Research Goal

Combinatorial optimization consists of computing the best solution, defined according to specific criteria, within a finite set of feasible solutions. There are two extremes when solving combinatorial coptimization problems: local search algorithms, which find "good" solutions fast by selectively searching through only a small portion of the search space, and compete algorithms, which provide optimal results by exhaustively enumerating all solutions over longer periods of time. A natural question that arises, given that both approaches have unique (dis)advantages, is whether it is possible to combine these two extremes. This is precisely the long-term goal of my research: tightly couple local search and complete algorithms into powerful *hybrid* techniques for combinatorial optimization problems. This would result in methods that provide best of both worlds: find "good" solutions fast while guaranteeing the optimal solution given enough time. Hybrid algorithms have been studied for decades and are sometimes called *matheuristics* [Maniezzo *et al.*, 2010]. I chose to focus on the particular problem of the general high school timetabling problem (XHSTT) [Post *et al.*, 2014] and maxSAT. The aim of XHSTT is to coordinate resources (e.g. rooms, teachers) with times to fulfill certain goals (e.g. schedule lectures). Timetables directly contribute to the quality of the educational system and satisfaction of students and staff, among other things, making timetabling an extremely important and responsible task. Thus, developing algorithms that generate high quality timetables is of great importance. To the best of my knowledge, maxSAT has not been considered within hybrid approaches prior to my work.

Most of the research for XHSTT focuses on local search techniques and only recently have integer programming (IP) hybrid algorithms been considered. See [Demirović and Musliu, 2017a] for a brief overview. Given that maxSAT was not explored for XHSTT and that there is an innate connection between timetabling and SAT, as the former has many logic characteristics and as such some of its constraints can be naturally expressed in propositional logic (e.g., if assignment X is made, then Y cannot be made), it was chosen as the complete method in my research. Overall, the developed algorithms are specific to XHSTT and maxSAT, but I believe similar techniques can be used for other problems, which is part of my long-term goal: devising hybrid techniques that can be applied to many different combinatorial optimization problems.

2 Contributions

I modeled XHSTT as a propositional logic formula, using Boolean variables and basic logical connectives only [Demirović and Musliu, 2017b]. To account for the soft constraints, the model was extended with the use of partial weighted maxSAT. I evaluated different cardinality constraint encodings, solvers, and special cases to simplify the modeling in practice. Resource assignment constraints have been considered only for special cases, rather than in general. In addition, I investigated a maxSAT-based satisfiability modulo theories (SMT) approach. The experimental results on numerous benchmarks proved that my maxSAT approach provides competitive results, outperforming the state-of-the-art complete approach based on IP [Kristiansen *et al.*, 2015].

I provided a bitvector modeling of XHSTT [Demirović and Musliu, 2016]. A single bitvector is associated with each event (e.g., a lecture). Each bit can take the values zero or one, indicating if the event assigned to it is taking place at a discrete time associated with the bit. By using a series of bitvector operations on the set of event bitvectors, I was able to model all XHSTT constraints, with the exception of resource assignment constraints. One of the key points of

---

[*] The work was supported by the Vienna PhD School of Informatics and the Austrian Science Fund (FWF): P24814-N23.

the bitvector modeling is that modern processors have built-in support for bitvector operations, allowing us to efficiently compute constraint violations. The resulting bitvector model is used as an efficient data structure for representing XHSTT for local search algorithms and SMT solvers.

Lastly, I developed a hybrid algorithm which exploits maxSAT for XHSTT [Demirović and Musliu, 2017a]. I combined local search and large neighborhood search (LNS) to solve XHSTT instances which are modeled as maxSAT. Iteratively, starting from a solution obtained with local search, part of the solution is *destroyed* and is then *repaired* by maxSAT, which amounts to fixing a subset of variables based on the current solution and then solving the remaining problem exhaustively. The approach provides highly quality solutions in limited time (e.g., 20 minutes) and outperforms the state-of-the-art solvers. Furthermore, to the best of my knowledge, it was the first time maxSAT was used within a LNS scheme.

3 Future Work

There is a number of challenges to addressed in future work, both general algorithmic and domain-specific issues.

Solver and cardinality constraint encoding selection. When modeling XHSTT as maxSAT, one has to decide which cardinality constraint encoding and solver to use. Better understanding of these two topics would lead to improved results. Therefore, for future work, I would like to address these issues, possibly by a *portfolio-based* approach.

Resource assignments. Modeling resource assignments is done for a special case, but not in general. If this is done in a straightforward manner, apart from drastically increasing the number of variables in the model, symmetries may easily be introduced, negatively impacting the solution process. Therefore, innovative modeling approaches are to be studied to address these concerns. In addition, I would like to research the possibility of including IP techniques such as *delayed column generation* for maxSAT to handle resource assignments.

Integrating domain-specific knowledge. The maxSAT solver does not incorporate any domain-specific knowledge which could be valuable during the solution process. For example, the *pigeon-hole problem* asks to put *n-1* pigeons in n holes and no hole can be occupied by more than one pigeon. The problem is clearly unsatisfiable and it may appear when solving timetabling problems, but there is no polynomial-sized proof of its unsatisfiability for SAT [Haken, 1995]. Therefore, I would like to research techniques that would allow the user to specify domain-specific knowledge.

New complete algorithm for XHSTT. I believe it would be beneficial to devise a specialized complete algorithm for XHSTT. The main issue with this approach is that it might be impractical: even if one develops a good methodology, implementing such algorithms can be time consuming.

New propagators and conflict driven clause learning techniques. Most SAT solvers include *unit-propagation*, but more complicated propagation schemes might be useful for maxSAT and XHSTT (similar ideas have been successful in constraint programming [de Uña *et al.*, 2010]). Similarly, specialized CDCL m could be devised.

Learning from conflicts. Understanding where most conflicts arise during the search would lead to a better understanding of the main problems for XHSTT and its solution process. This could be used to adapt the algorithm more closely for the instances through e.g. better variable selection and automated neighborhood generation.

Better integration of local search and LNS. The LNS algorithm divides the search space into several (not necessarily disjoint) neighborhoods and solves each neighborhood individually. With time, the neighborhoods increase in size and eventually the complete search space is considered. While state-of-the-art results were obtained with this technique, I believe it can be improved further by keeping more information in between solving each neighborhood apart from learned clauses. In the current state, neighborhoods might overlap, but I do not exploit this. If this could efficiently be done, one could search through the most prominent parts of the search space first, allowing us to find "good" solutions fast, and with time other parts of the search space would be considered with minimum overhead, ideally providing a direct improvement over a standard complete approach.

Resilience. I would like to investigate computing timetables that are require only minor changes after additional constraints are added post-solving.

References

[de Uña *et al.*, 2010] Diego de Uña, Graeme Gange, Peter Schachte, and Peter J. Stuckey. A bounded path propagator on directed graphs. In *Proceedings of CP'16*, pages 189–206, 2010.

[Demirović and Musliu, 2016] Emir Demirović and Nysret Musliu. Modeling High School Timetabling with Bitvectors. *Annals of Operations Research*, 2016.

[Demirović and Musliu, 2017a] Emir Demirović and Nysret Musliu. MaxSAT-Based Large Neighborhood Search for High School Timetabling. *Computers & Operation Research*, 78:172–180, 2017.

[Demirović and Musliu, 2017b] Emir Demirović and Nysret Musliu. Modeling high school timetabling as partial weighted maxSAT. *technical draft*, 2017.

[Haken, 1995] Armin Haken. Counting bottlenecks to show monotone P <=> NP. In *36th Annual Symposium on Foundations of Computer Science*, pages 36–40, 1995.

[Kristiansen *et al.*, 2015] Simon Kristiansen, Matias Sørensen, and Thomas R. Stidsen. Integer programming for the generalized high school timetabling problem. *Journal of Scheduling*, 18(4):377–392, 2015.

[Maniezzo *et al.*, 2010] Vittorio Maniezzo, Thomas Stützle, and Stefan Voß, editors. *Matheuristics - Hybridizing Metaheuristics and Mathematical Programming*, volume 10 of *Annals of Information Systems*. Springer, 2010.

[Post *et al.*, 2014] Gerhard Post, Jeffrey H. Kingston, Samad Ahmadi, Sophia Daskalaki, Christos Gogos, Jari Kyngäs, Cimmo Nurmi, Nysret Musliu, Nelishia Pillay, Haroldo Santos, and Andrea Schaerf. XHSTT: an XML archive for high school timetabling problems in different countries. *Annals of Operations Research*, 218(1):295–301, 2014.