An Automatically Configured Modular Algorithm for Post Enrollment Course Timetabling

Chris Fawcett, Holger H. Hoos
Computer Science Department,
University of British Columbia, Vancouver, BC, Canada
{fawcettc,hoos}@cs.ubc.ca

Marco Chiarandini
Department of Mathematics and Computer Science,
University of Southern Denmark, Odense, Denmark
marco@imada.sdu.dk

Abstract

Timetabling tasks form a widely studied type of resource scheduling problem, with important real-world applications in schools, universities and other educational settings. In this work, we focus on post-enrollment course timetabling, the problem that was covered by Track 2 of the recent 2nd International Timetabling Competition (ITC2007). Following an approach that makes strong use of automated exploration of a large design space of modular and highly parameterised stochastic local search algorithms for this problem, we have obtained a solver that achieves consistently better performance than the top-ranked solver from the competition. This represents a substantial improvement in the state of the art for post-enrollment course timetabling.

1 Introduction

Course and examination timetabling is a resource-constrained scheduling problem encountered by universities and other educational institutions, involving scheduling a set of events into given rooms and timeslots. The resulting schedule is subject to resource and feasibility constraints derived from the availability of rooms, student enrollments in the events, precedence relations between events, and student or teacher preferences. While the feasibility constraints must be strictly satisfied, giving rise to a satisfaction problem, preferences should not be violated whenever possible, which gives rise to an optimization problem. The presence of such hard and soft constraints is typical for many real-world constraint optimisation problems.

In this work, we present a new state-of-the-art solver for a particular problem in timetabling, the Post-Enrollment Course Timetabling Problem, as considered in Track 2 of the recent 2nd International Timetabling Competition (ITC2007) [13]. Leveraging our previous work which resulted in a solver that placed third in Track 2 of ITC2007 [5], our main contribution lies in the automated approach for designing our new solver, as well as the resulting solver itself, which represents a substantial improvement over the previous best algorithm for the problem (the solver by Cambazard et al. that won Track 2 of ITC2007 [3]).

2 Automated Design Approach and Algorithm Framework

In recent years there has been a considerable amount of methodological research devoted to the issue of configuring the components and tuning the parameters of optimisation algorithms, and especially of heuristic algorithms. Contrary to the traditional approach of trying to minimise the number of user-configurable algorithm parameters, these methods embrace the idea of parameterising as much algorithm functionality as possible [8]. Out of the available procedures for automated algorithm configuration we selected FocusedILS [11, 10], as it is the only procedure we are aware of that has been demonstrated to be effective in dealing with very large, highly discrete design spaces.

Our approach is heavily based on the use of this powerful automated algorithm configuration method, allowing us to search for a performance-optimised design within a very large space of candidate solvers. The same fundamental automated algorithm design approach has been recently used to obtain substantial improvements in the state of the art for solving various types of SAT instances [9, 12] (where the latter piece of work has been undertaken in parallel and to a large degree independently of the work presented here). The space of algorithms for the problem considered in this work is defined by a modular solver framework that is based on stochastic local search (SLS) methods [7] and builds on several ideas from the timetabling and graph coloring literature, as well as on work done for the first timetabling competition in 2003 [4]. The design space for our solvers involves not only traditional numerical parameters, but also choices between preprocessing options and neighbourhoods as well as the diversification strategies employed. This framework contains 18 configuration parameters and design choices, and allows for a total of approximately 10^{13} possible instantiations. The key idea behind this framework is to first solve the feasibility problem and then the optimisation problem by restricting the search to only feasible timetables. Different solution representations and neighbourhoods are interleaved during the search in the two phases, and randomisation together with tabu search and simulated annealing concepts are combined in a flexible and novel way.

The hard constraint solving phase of our configured solver consists of a constructive phase followed by a stochastic local search phase. The constructive phase generates partial assignments using an approach similar to that of Arntzen and Løkketangen [1], inserting events using a topological ordering of the precedence constraints. The local search phase uses a tabu search procedure based on the PARTIALCOL algorithm [2]. At each iteration, an unscheduled event is inserted into the best possible non-tabu timeslot for it, and all events subsequently breaking hard constraints are moved into the list of unscheduled events. After a number of non-improving iterations have been made, a component of the soft constraint solver is used as a diversification mechanism. If this perturbation and subsequent optimisation fails to produce an improvement, a number of events are chosen at random to be unscheduled and the search proceeds as before. When there are no longer any hard constraint violations, the algorithm proceeds to the soft constraint solving phase.

The soft constraint solving phase begins with a first-improvement local search until a local minima is found for all four of the neighbourhoods available in the solver. Next, a simulated annealing phase is applied using both the 2-exchange and swap-of-time-slots neighbourhoods until a specified time limit is reached or an optimal schedule is located. In addition to the usual geometric cooling schedule, the temperature parameter of the annealing procedure is increased after a number of non-improving iterations.

3 Experimental Design and Results

Based on the multi-phase architecture underlying our solver, we applied FocusedILS to first optimise the parameters controlling the behaviour of the hard constraint solver and then to optimise the parameters for the soft constraint solver. The solver was tuned using a runtime cutoff of 600 CPU seconds, and all tuning was performed using a cluster of identical machines, in order to perform multiple independent runs of FocusedILS in parallel. Overall, 360 CPU hours were used for runs of FocusedILS to produce our final solver.

In the course of this work, we also developed a new metric for FocusedILS called the *p-value performance metric*, based on the idea of using empirical solution quality distributions (SQDs) for an existing solver as a target for the algorithm being tuned. This metric was used when tuning the parameters of the soft constraint solver, using empirical SQDs from the Cambazard et al. solver. In addition, we extended FocusedILS to perform more effectively when there is an expectation that some numeric parameters have a convex or unimodal response when the values of the other parameters are fixed. This new version 2.4 of FocusedILS was used for all parameter configuration performed in this work.

Of the 24 available instances from ITC2007 for this problem, the sixteen "public" instances were used for configuration while the eight "private" instances were reserved for testing purposes. Using 100 runs each 600 CPU seconds in length on each of these 24 instances, our tuned solver configuration achieves better median soft constraint violation values than the solver of Cambazard et al. on all of the public instances as well as for five of the eight private instances that were not used in the configuration process. On four instances, the median quality is better by more than an order of magnitude. In addition, we also beat the solver of Cambazard et al. using the rank-based competition metric of ITC2007 in a 2-way race. These results clearly demonstrate a substantial performance improvement compared to the previous state of the art.

In future work, it would be particularly interesting to use these tools in combination with an appropriately expanded version of our solver framework to tackle the two other timetabling problems used in ITC2007 (an examination and a curriculum-based timetabling problem [14, 6]). We have also recently begun applying our timetabling algorithm to the real-world problem of scheduling exams at the University of British Columbia.

Acknowledgements. The authors would like to thank Hadrien Cambazard for providing source code for the solver presented in [3], as well as valuable feedback via e-mail.

References

[1] H. Arntzen and A. Løkketangen. A tabu search heuristic for a university timetabling problem. In *Proceedings of the Fifth Metaheuristics International Conference*, Kyoto, Japan, August 2003.

[2] I. Blöchliger and N. Zufferey. A graph coloring heuristic using partial solutions and a reactive tabu scheme. *Computers & Operations Research*, 35(3):960–975, 2008.

[3] H. Cambazard, E. Hebrard, B. O'Sullivan, and A. Papadopoulos. Local search and constraint programming for the post enrolment-based course timetabling problem. In E. K. Burke and M. Gendreau, editors, *Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2008)*, Montreal, CA, 2008. Université de Montréal.

[4] M. Chiarandini, M. Birattari, K. Socha, and O. Rossi-Doria. An effective hybrid algorithm for university course timetabling. *Journal of Scheduling*, 9(5):403–432, Oct. 2006.

[5] M. Chiarandini, C. Fawcett, and H. H. Hoos. A modular multiphase heuristic solver for post enrollment course timetabling (extended ab-

stract). In *Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2008)*, 2008.

[6] L. D. Gaspero, B. McCollum, and A. Schaerf. The second international timetabling competition (itc-2007): Curriculum-based course timetabling (track 3). Technical Report QUB/IEEE/Tech/ITC2007/CurriculumCTT/v1.0, Queen's University, Belfast, United Kingdom, 2007.

[7] H. Hoos and T. Stützle. *Stochastic Local Search: Foundations and Applications.* Morgan Kaufmann Publishers, San Francisco, CA, USA, 2004.

[8] H. H. Hoos. Computer-aided design of high-performance algorithms. Technical Report TR-2008-16, University of British Columbia, Department of Computer Science, 2008.

[9] F. Hutter, D. Babić, H. H. Hoos, and A. J. Hu. Boosting Verification by Automatic Tuning of Decision Procedures. In *Proceedings of Formal Methods in Computer Aided Design (FMCAD'07)*, pages 27–34, Washington, DC, USA, 2007. IEEE Computer Society.

[10] F. Hutter, H. Hoos, K. Leyton-Brown, and T. Stützle. ParamILS: An automatic algorithm configuration framework. Technical Report TR-2009-01, University of British Columbia, Department of Computer Science, January 2009.

[11] F. Hutter, H. H. Hoos, and T. Stützle. Automatic algorithm configuration based on local search. In *Proc. of the Twenty-Second Conference on Artificial Intelligence (AAAI '07)*, pages 1152–1157, 2007.

[12] A. KhudaBukhsh, L. Xu, H. H. Hoos, and K. Leyton-Brown. SATenstein: Automatically building local search SAT solvers from components. In *Proc. of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI-09)*, pages 517–524, 2009.

[13] R. Lewis, B. Paechter, and B. McCollum. Post enrolment based course timetabling: A description of the problem model used for track two of the second international timetabling competition. Cardiff Working Papers in Accounting and Finance A2007-3, Cardiff Business School, Cardiff University, August 2007.

[14] B. McCollum, P. McMullan, E. K. Burke, A. J. Parkes, and R. Qu. The second International Timetabling Competition: Examination timetabling track. Technical Report QUB/IEEE/Tech/ITC2007/-Exam/v4.0/17, Queen's University, Belfast, Sep 2007.