Towards
Constraint-Based Grammar School Timetabling

Michael Marte*

Computer Science Department, University of Munich
Oettingenstraße 67, 80538 München, Germany
marte@informatik.uni-muenchen.de

In school timetabling, the goal is twofold. First, lessons must be scheduled such that all resources (classes, pupils, teachers, rooms, and other equipment) required for education are available. Second, timetables are required to satisfy various working-time regulations, organizational and educational constraints. School timetabling with unavailabilities has been shown to be NP-complete [9]. I aim to tackle German grammar school (GGS) timetabling problems by means of constraint programming (CP) [14].

In CP, like in operations research (OR), discrete combinatorial problems are specified declaratively by stating a set of constraints among problem variables. However, the constraint systems provided by CP and OR differ considerably. Typically, while OR constraint systems allow for linear equations and disequations over 0-1 variables, CP constraint systems provide carefully selected abstractions (called *global constraints*) that are intended to facilitate the specification of problems from different fields of applications in a concise and natural way. For example, non-preemptive single-resource scheduling problems may be expressed by means of the global constraint *cumulative* [3]. Increasing the expressiveness of single constraints yields interesting perspectives in problem solving: By exploiting the semantics of global constraints, local consistency may be established efficiently at each search node by dedicated propagation algorithms. Frequently, these algorithms are based on methods from artificial intelligence, graph theory, and OR (e.g. [11,18]). Global constraints play a vital role in the current success of CP.

The GGS comprises nine grades (5-13). In grades 5-11, pupils are grouped to form classes. In grades 12 and 13, pupils must choose from a set of courses resulting in a more college-like education. In grades 9-11, several branches of education, differing in curricula, may be available. Depending on the branch, two or three foreign languages are taught. For each branch, several language curricula may be available. Each pupil must decide for a branch and a language curriculum. Frequently, heterogeneous classes with boys and girls from different branches, with different religious denominations, and different language curricula cannot be avoided. For economical and educational reasons, heterogeneous classes usually imply the need to join pupils from different classes for sports education, religious education, branch-specific lessons, and foreign-language instruction. The resulting need for simultaneous education of pupils from several classes complicates timetabling considerably due to simultaneous resource demands.

During the 1990s, research in automated school timetabling concentrated on stochastic methods such as simulated annealing, tabu search, and genetic algorithms (e.g.

---

* I am a PhD student funded by the German Research Council (DFG).

[5,6,8,10,17,19]). Results have been promising; however, they do not transfer to the general situation in GGS timetabling, which is distinguished by numerous requests for synchronized education due to multiple branches, denominations, and language curricula. CP has been applied to university and college timetabling (e.g. [2,7,12,13,15]) but, to my best knowledge, not to school timetabling.

I aim to model and solve GGS timetabling problems by means of global constraints. In problem solving, I investigate partial branch & bound procedures that are guided by optimality criteria. For experimental performance analysis, a simulation environment is developed that comprises a problem generator[1], a model generator, and various tools for performance debugging. In the long term, I plan to publish my problem sets to facilitate the comparison of different methods.

From a semantic point of view, existing global constraints suffice to model GGS timetabling problems. For example, requirements in resource allocation and significant educational constraints (such as bounds on the number of math lessons taught per day) can be mapped to cumulative constraints directly. By introducing intermediate layers of representation, significant working-time regulations (such as bounds on the number of gaps in timetables and bounds on the number of free days) can be mapped to global cardinality constraints [18]. However, the operational behaviour of this model is insufficient due to certain properties of the functions mediating between the different layers of representation. Therefore, to cope with working-time regulations, I need to define and implement new global constraints.

In branch & bound search, cost-based domain filtering [11] and value orders play a vital role. Filtering prunes values that cannot participate in better solutions. Preferring the values that are expected to cause a minimum decrease in solution quality results in greedy search procedures. While filtering is performed by propagating the upper bounds of the variables that are involved in the objective function, greedy search requires to tighten their lower bounds.

The use of a problem generator is motivated by economical and scientific reasons. First, collecting large quantities of data is too time-consuming. Second, a problem generator facilitates a systematic investigation of problem classes and algorithms. Third, a problem generator allows for case studies and what-if analysis.

To enable case studies and to ensure the practical relevance of results, problem sets are generated on the basis of brief school profiles. A school profile contains key features such as resource capacities, resource availabilities, and resource requirements of lessons; frequencies of and correlations among features of pupils such as gender, denomination, branch, and language curriculum; the number of pupils, class sizes and optimality criteria.

This paper presents a project that aims at applying CP technology to GGS timetabling. Models are based on global constraints and optimality criteria guide search. Empirical evaluation relies on artificial problems that are very similar to real GGS timetabling problems.

---

[1] Unfortunately, the problem generator used by [8] is not suitable for my purposes; it was not designed to produce GGS timetabling problems and it cannot be adapted for this task because important parameters are hard-wired and the source code is not available.

References

1. *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*. AAAI Press / MIT Press, 1996.
2. S. Abdennadher and M. Marte. University course timetabling using Constraint Handling Rules. *Journal of Applied Artificial Intelligence*, 14(4):311–326, 2000.
3. A. Aggoun and N. Beldiceanu. Extending CHIP in order to solve complex scheduling and placement problems. *Mathematical and Computer Modelling*, 17(7):57–73, 1993.
4. E. Burke and P. Ross, editors. *Practice and Theory of Automated Timetabling*, LNCS 1153. Springer, 1996.
5. A. Colorni, M. Dorigo, and V. Maniezzo. Metaheuristics for high-school timetabling. *Computational Optimization and Applications*, 9(3):277–298, 1998.
6. D. Costa. A tabu search algorithm for computing an operational timetable. *European Journal of Operational Research*, 76(1):98–110, 1994.
7. F. P. M. Dignum, W. P. M. Nuijten, and L. M. A. Janssen. Solving a time tabling problem by constraint satisfaction. Technical report, Eindhoven University of Technology, 1995.
8. A. Drexl and F. Salewski. Distribution requirements and compactness constraints in school timetabling. *European Journal of Operational Research*, 102(1):193–214, 1997.
9. S. Even, A. Itai, and A. Shamir. On the complexity of timetable and multicommodity flow problems. *SIAM Journal on Computing*, 5(4):691–703, 1976.
10. C. Fernandes, J. P. Caldeira, F. Melicio, and A. Rosa. High-school weekly timetabling by evolutionary algorithms. In *ACM Symposium on Applied Computing*, pages 344–350, 1999.
11. F. Focacci, A. Lodi, and M. Milano. Cost-based domain filtering. In [16], pages 189–203.
12. H.-J. Goltz and D. Matzke. University timetabling using constraint logic programming. In *Practical Aspects of Declarative Languages*, LNCS 1551, pages 320–334. Springer, 1999.
13. C. Gueret, N. Jussien, P. Boizumault, and C. Prins. Building university timetables using constraint logic programming. In [4], pages 130–145.
14. P. V. Hentenryck and V. A. Saraswat. Constraint programming: Strategic directions. *Constraints Journal*, 2:7–33, 1997.
15. M. Henz and J. Würtz. Using Oz for college timetabling. In [4], pages 283–296.
16. J. Jaffar, editor. *Fifth International Conference on Principles and Practice of Constraint Programming*, LNCS 1713. Springer, 1999.
17. K. Kaneko, M. Yoshikawa, and Y. Nakakuki. Improving a heuristic repair method for large-scale school timetabling problems. In [16], pages 275–288.
18. J.-C. Régin. Generalized arc consistency for global cardinality constraint. In [1], pages 209–215.
19. A. Schaerf. Tabu search techniques for large high-school timetabling problems. In [1], pages 363–368.