Timetabling at German Secondary Schools: Tabu Search versus Constraint Programming

Frank Jacobsen, Andreas Bortfeldt and Hermann Gehring

University of Hagen, Germany
Frank.Jacobsen@gmx.de

1 Introduction

In recent years many solution approaches for different school timetabling problems (TTP) have been tried, among them tabu search, simulated annealing, genetic algorithms, and constraint programming [8]. However, most of the solution methods developed so far have been tested by means of only few (mostly only a handful) problem instances [6]. Further, until today there are nearly no reports on tests which subject different methods to a comparative analysis on the basis of the same benchmark instances and thus provide empirical evidence for the (relative) suitability of solution approaches.

In this paper, a tabu search algorithm [4] for timetabling at German secondary schools of the Gymnasium type is presented. The TSA is subjected to an extensive test including 1500 problem instances. The instances have been introduced by MARTE [6] and used for the test of his constraint programming [5] method. Therefore, the results obtained with the TSA and the CP method are finally compared here.

2 Problem description

German secondary schools can be compared with the British grammar schools, but they admit extensive choices to pupils [3]. Similar to MARTE [6], it is assumed here that the timetabling is essentially based on the following conditions:

- The lessons are given on the grade levels 5 to 13.
- The sets of teachers, rooms, and classes are fixed.
- The rooms are classified according to certain room types (e.g. gym hall).
- On each grade level one or (from level 7 onward) more teaching programs are offered. A program is fixing a selection of subjects and the number of weekly lessons for each subject. Therefore, there exist one or more pupil groups (PG) per class with the same teaching program for the pupils of a group. A typical feature of a teaching program is a choice of foreign languages and/or a study direction such as Social Sciences or Natural Sciences.
- The complete weekly teaching program is now specified as a set of lesson requirements (LR). A LR is a combination of one teacher, one subject, one or more PG's from one or more classes, and one room type. The teaching of a LR lasts always for a period (of 45 minutes). For each weekday the number of periods which are available for the timetabling is fixed.

The TTP of a secondary school of the Gymnasium type (GYM-TTP) combines the tasks of room and period assignment for pre-defined LR's and can be formulated as

E. K. Burke, H. Rudová (Eds.): PATAT 2006, pp. 439–442. ISBN 80-210-3726-1.

follows. Assign a room of appropriate type and a period to each of the LR's in such a way that the following constraints are met:

- **Clash constraints** A1-A3: The scheduling of teachers, rooms, and classes or pupil groups, respectively, must avoid clashes.
- **Availability constraints** B1-B3: Teachers, rooms, and classes must be scheduled within their availability time windows.
- **Coupling constraints**:
  - C1 Certain LR's are to be scheduled for the same period.
  - C2 Certain pairs of congruent LR's are to be scheduled for two consecutive periods, and the same room is to be assigned to both of the LR's (2-hour lesson).
- **Distribution constraints**:
  - D1 The timetable of each class shouldn't contain idle periods.
  - D2 For each class the lessons should end as early as possible on each day.
  - D3 Certain LR's are to be scheduled for pre-determined periods.
  - D4 For teachers and for pupils the daily minimum and maximum number of lessons should be respected. Further, a lower and an upper limit of working days per week are to be considered for each teacher.
  - D5 For each class the daily minimum and maximum number of lessons on the same subject should be respected.

3 Mathematical model

The GYM-TTP is formulated as a binary optimization model. Except for D1 and D2, all constraints are categorized as hard.

Depending on the type of constraint, either LR's or so-called complex lesson requirements (CLR's) serve as the basis for the modelling. A CLR includes all LR's which, according to C1, are to be held at the same time. Further, both of the LR's of a 2-hour lesson are, according to C2, always assigned to the same CLR. A CLR must comprise either only 2-hour lessons or only (1-hour) LR's.

Two sets of binary decision variables – x_{np} and y_{mr} – are introduced. A variable x_{np} has the value 1 if the CLR n is scheduled for period p, and a variable y_{mr} has the value 1 if the LR m is scheduled for room r. Due to the planning of periods on the level of CLR's, the constraints C1 and C2 are automatically met. The remaining hard constraints are modelled explicitly.

The constraints D1 and D2 are integrated in the objective function f which is defined as $f = f_{ip} + f_{ef}$ and to be minimized. The term f_{ip} is summing up the number of idle periods over all classes, i.e. the unplanned periods which are, however, followed by lessons. The term f_{ef} measures the compactness of a timetable and, roughly expressed, sums up all variables x_{np} which are weighted with the indices of the periods of a day. For evaluation purposes and by means of an obvious lower bound lbf_{ef} , the compactness index is defined as $lbf_{ef} / f_{ef} * 100$.

4 The Tabu Search Algorithm

In the following, the essential properties of the proposed TSA, called TS-Gym, are described.

TS-Gym is a purely deterministic method. In the interest of a high robustness, stochastic components have been omitted.

A generated solution is represented by two vectors, a period vector and a room vector. The period vector is assigning a period to each CLR, while the room vector assigns a room of appropriate type to each LR. The search space contains feasible solutions, which meet all hard constraints (cf. section 3), and infeasible solutions as well.

An initial solution is generated by means of a specific construction heuristic. The heuristic is based on the sorting of the CLR's according to the difficulties arising with their scheduling, is using a graph coloring algorithm [1], and aims primarily at the generation of a feasible solution.

In TS-Gym two types of neighbourhoods are alternatively applied. In the case of the period-neighbourhood, a neighbour s' of a current solution s is derived through the assignment of a deviating period to exactly one CLR. In the case of the room-neighbourhood a neighbour s' of s results from a deviating assignment of a room for exactly one LR. The room-neighbourhood is only applied in situations where the current solution s violates one of the room constraints, A2 or B2. For both neighbourhood types, the best neighbourhood solution is determined by means of a specific evaluation function. The function is attaching a high weight to the violation of hard constraints and a low weight to the value of the objective function, f.

A best neighbourhood solution is accepted as the new best solution only if, in comparison to the current best solution, the number of violated hard constraints is reduced or, for the same number of violated constraints, the value of the objective function is improved.

The tabu list management is designed similar to DESEF et al. [2]. As in the latter case, two tabu lists are kept. The move list contains the (inverse) moves carried out recently. Purpose of the frequency list is to avoid too frequent shifts of individual CLR's. According to the aspiration by objective, the improvement of the best solution in the sense defined above is used as aspiration criterion.

5 Results and comparison with constraint programming

TS-Gym has been tested on a standard notebook (1.6 GHz Pentium-M, 1 GB RAM) using the 1500 test instances from MARTE [6][7] and a fixed parameter setting. The 1500 instances are subdivided into 6 test cases R1 to R6 which correspond to six secondary schools of various kinds and contain 250 instances each. The results obtained with TS-Gym and the CP method from MARTE are shown in Table 1.

For almost all instances both of the methods calculate a feasible solution, i.e., they achieve nearly the same high solution quality with respect to the share of instances solved to feasibility. It should be emphasized, however, that in the case of TS-Gym the consideration of the hard constraints D4 and D5 has not yet been implemented and therefore not been included in the test.

Table 1. Results for the secondary schools R1 to R6 [6][7].

Evaluation criterion	Method	R1	R2	R3	R4	R5	R6
Share of instances solved to feasibility (in %)	MARTE	100	97	100	98	99	92
	TS-Gym	100	96.4	99.6	98	99.2	92.4
Mean no. of idle periods	TS-Gym	11.6	8.4	8.4	7.0	7.3	7.3
Mean compactness	TS-Gym	65.8	73.1	73.3	77.3	76.0	76.7
Mean CPU time (in s)	TS-Gym	62.1	60.1	90.0	61.1	56.6	100.3

For the criteria "number of idle periods" and "compactness" comparative values are not available. Since the teaching program on the upper grade levels is similar to that of a university, where only moderate compactness requirements are to be met, these results and the computing times as well seem to be satisfactory.

Apart from the implementation of the constraints D4 and D5, the improvement of the parameterization and of selected components of TS-Gym will be the subject of further research.

References

1. Coleman, T. F. and Moré, J. J. (1983): Estimation of Sparse Jacobian Matrices and Graph Coloring Problems. In: SIAM Journal of Numerical Analysis, Vol 20, pp 187-209.
2. Desef, T., Bortfeldt, A. and Gehring, H. (2004): A Tabu Search Algorithm for Solving the Timetabling-Problem for German Primary Schools. In: Burke, E. K., Trick, M. (Eds): Proceedings of the 5th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2004), pp. 465-470.
3. Drexl, A. and Salewski, F. (1997): Distribution Requirements and Compactness Constraints in School Timetabling. In: European Journal of Operations Research, Vol 102(1), pp 193-214.
4. Glover, F. and Laguna, M. (1993): Tabu-Search. In: Reeves, C. R. (Ed): Modern Heuristic Techniques for Combinatorial Problems, Blackwell Scientific Publications, Oxford etc.
5. Jaffar, J. and Maher, M. J. (1994): Constraint Logic Programming: A Survey. In: Journal of Logic Programming, Vol 19/20, pp 503-581.
6. Marte, M. (2002): Models and Algorithms for School Timetabling – A Constraint Programming Approach. Doctoral Dissertation at the Faculty of Mathematics, Computer Science and Statistics of the Ludwig-Maximilian-University of Munich, Munich.
7. Marte, M. (2004): Towards Constraint-Based School Timetabling. In: Hnich, B., Walsh, T. (Eds): Proceedings of the Workshop on Modelling and Solving Problems with Constraints, held at ECAI 2004, pp 140-154.
8. Schaerf, A. (1999): A Survey of Automated Timetabling. In: Artificial Intelligence Review, Vol 13, pp 87-127.