A Matheuristic Approach for an Examination Scheduling Problem

R. Bargetto · F. Della Croce · F. Salassa

1 Introduction

Timetabling problems are a specific type of scheduling problem and are mainly concerned with the assignment of events to timeslots (time periods) subject to constraints. Timetabling problems arise in various forms including educational timetabling, nurse scheduling (e.g. [6]), sports timetabling (e.g. [5,4]) and transportation timetabling (e.g. [2]). They represent a challenging and important problem area for researchers across both Operational Research and Artificial Intelligence since the 1960s.

University timetabling problems, mainly examination and course timetabling, are difficult tasks faced by educational institutions. Solving a real world university timetabling problem manually often requires a large amount of time and expensive resources. The research in this area has been very active over the years. A wide variety of papers, in the fields of operational research and artificial intelligence, have addressed the broad spectrum of university timetabling problems. Timetabling within a university context has long been recognized as difficult from both a theoretical and practical point of view (see [9] and [8]).

Many universities are seeing an increasing number of student enrollments into a wider variety of courses and an increasing number of combined degree courses. This is contributing to the growing challenge of developing examination timetabling solutions for the various constraints and demands that are required by different educational institutions across the world.

R. Bargetto
D.A.I. Politecnico di Torino E-mail: s189487@studenti.polito.it

F. Della Croce
D.A.I. Politecnico di Torino E-mail: federico.dellacroce@.polito.it

F. Salassa
D.A.I. Politecnico di Torino E-mail: fabio.salassa@.polito.it

Proceedings of the 11^{th} International Confenference on Practice and Theory of Automated Timetabling (PATAT-2016) – Udine, Italy, August 23–26, 2016

2 Problem Definition

The problem has been provided by the Computer Science College of Politecnico di Torino where the academic year is split into two teaching periods, each of approximately four months. The first period usually starts at the end of September and finishes in January; the second period starts in March and finishes in June. A course has its lectures in one of these two periods.

At the end of the teaching periods, there are the examination sessions. The examination session at the end of the first teaching period is referred to as "winter" session and lasts four weeks; the examination session at the end of the second teaching period is referred to as "summer" session and lasts six weeks. Before a new academic year starts and, consequently, before the first teaching period of the new academic year starts, there is an additional examination session, which usually starts at the end of August and lasts four weeks. This third examination session is referred to as "autumn" session.

Within an examination session ("winter", "summer" or "autumn") there is one examination for each course but if the course has its lectures in the teaching period preceding the examination session, there are two examinations for that course (this applies for "winter" and "summer"). Thus, for each course there are four examinations during the whole academic year. For example, if a given course has its lectures in the first teaching period (September-January) there are two examinations in the "winter" session, one examination in the "summer" session and one examination in the "autumn" session.

For each week of an examination session, only the standard working days from Monday to Friday are available, while the weekend is excluded. Each day of the examination session is split into four timeslots each of two hours and half (150 minutes), from 8:30 AM to 6:30 PM. The three examination sessions of an academic year are mutually independent, hence, to provide an examinations calendar for the whole academic year it is necessary to solve a different problem for each session.

3 Solution Approach and Preliminary Results

For the considered problem a MILP formulation (omitted here, we refer to [1]) can be derived by typically introducing binary variables $x_{i,j,k}$ that indicate whether a given examination i is scheduled on timeslot j in a given classroom k. However, for real instances of this problem, MILP commercial solvers are not able to find the optimal solution in reasonable time. Besides, if only a small part of the general problem is considered, then the MILP solver easily detects the optimum in short time. We propose then a matheuristic solution approach [3] that integrates a search phase realized by an exact algorithm (in our case the MILP solver) on a subset of the original problem into a well-known metaheuristic procedure. As metaheuristic, we applied *Large Neighborhood Search* (LNS) (see [7]) with multiple neighborhoods. The use of multiple neighborhoods typically improves the performances of standard local search approaches and has been widely applied to timetabling problems (e,g., [6]).

Hence, for each considered neighborhood, the search can be performed by solving the related MILP formulation by means of a commercial tool. Our matheuristic approach requires in input a feasible solution that is computed by running for a limited CPU time the MILP solver applied to a general MILP formulation of the problem.

3.1 Neighborhood structures

The definition of the neighborhood structures, that is the way we choose the subset of variables in the local search sub–problems and the corresponding number of variables in the subset, directly affects the performances of the algorithm. A crucial issue is to select neighborhood structures that are strongly different one to another: the aim is that a local optimum delivered by a neighborhood is not (too close to) a local optimum for the other ones. Four types of neighborhood structures are considered:

1. *Single Time Window Neighborhood*: a certain number of consecutive days (the time window) is selected for a re-optimization phase while for all the other days the current solution is kept and the related variables are set to their current value. Then, the MILP formulation of the corresponding subproblem is solved by CPLEX.
2. *Double Time Window Neighborhood*: Similar to the Single Time Window Neighborhood but here two disjoint subsets of consecutive days and related variables are selected for a re-optimization phase while for all the other days the current solution is kept and the related variables are set to their current value. The two subsets of days may or may not be consecutive.
3. *Data Driven Examination Cluster Neighborhood*: all the examinations pertaining to an internal structure (M.D, B.D, college, orientation, etc...) are selected for a re-optimization phase while for all the other examinations the current solution is kept and the related variables are set to their current value.
4. *Randomly Extracted Examination Cluster Neighborhood*: all the examinations in a set (randomly determined) and related variables are selected for a re-optimization phase while for all the examinations the current solution is kept and the related variables are set to their current value.

3.2 Computational Results

The proposed approach has been applied to real world instances constituted by a number of courses ranging approximately between 200 and 300, 30 classrooms and examination sessions periods of either 4 or 6 consecutive weeks corresponding to 80 or 120 timeslots, respectively. In order to test our approach we benchmarked the proposed matheuristic procedure with the commercial solver CPLEX where 1800 seconds were allotted to all runs.

The tests were conducted on an Intel Core i5 CPU @ 3.40GHz with 8GB of RAM. As MIP solver we used CPLEX 12.6. Note that CPLEX always failed to solve the considered instances to optimality within the time limit, hence we considered the solver best feasible solution obtained after 1800 seconds. In Table 1 the relative gaps of both CPLEX and our approach (MATHEUR) are reported. The relative gap is calculated using the best bound obtained by CPLEX at the expiration of the time limit.

Instance	**# Examinations**	**Timeslots**	**MATHEUR Gap**	**CPLEX Gap**
winter1	310	120	25.01%	86.38%
winter2	310	120	31.66%	87.59%
summer1	288	120	24.99%	95.84%
summer2	288	120	12.66%	92.47%
autumm1	209	80	9.31%	11.55%
autumn2	209	80	25.93%	42.45%

Table 1 Preliminary results on real–world instances

The results indicate that the proposed procedure clearly outperforms CPLEX solver. We note also that, though not presented here, the proposed schedules are clearly superior also to the ones currently proposed by the Politecnico staff in charge of the delivery of the examinations timetabling. Future work will be devoted to analyze different strategies for neighborhood structures (larger vs. smaller size) and to study the applicability of our approach to the literature examination timetabling instances.

References

1. Bargetto, R., A Matheuristic Approach for the University Timetabling Problem: Scheduling of Examinations, Master Thesis in Computers Engineering, Politecnico di Torino, Italy, March 2016.
2. Caprara, A., Fischetti, M., Toth, P., Modeling and Solving the Train Timetabling Problem, *Operations Research*, 50(5), 851–861 (2002).
3. Della Croce F., Grosso A., Salassa F., Matheuristics: embedding MILP solvers into heuristic algorithms for combinatorial optimization problems. In: Heuristics : Theory and Applications. NOVA Publisher, Hauppauge NY, USA, 31-52 (2013).
4. Della Croce, F., Tadei, R., Asioli, P.S., Scheduling a Round Robin Tennis Tournament Under Courts and Players Availability Constraints, *Annals of Operations Research*, 92, 349–361 (1999).
5. Della Croce, F., Oliveri, D., Scheduling the Italian Football League: An ILP-Based Approach, *Computers & Operations Research*, 33(7), 1963–1974 (2006).
6. Della Croce, F., Salassa, F., A Variable Neighborhood Search Based Matheuristic for Nurse Rostering Problems, *Annals of Operations Research*, 218, 185–199 (2014).

7. Pisinger, D., Ropke, S., Large Neighborhood Search, in *Handbook of metaheuristics*, 399–419, Springer US 2010.
8. Qu, R., Burke, E.K., McCollum, B., Merlot, L.T.G., Lee, S.Y., A Survey of Search Methodologies and Automated System Development for Examination Timetabling, *Journal of Scheduling*, (12), 55–89 (2009).
9. Schaerf, A., A Survey of Automated Timetabling, *Artificial Intelligence Review*, 13(2), 87–127 (1999).