Technical University of Denmark

High School Timetabling: Modeling and solving a large number of cases in Denmark

Sørensen, Matias; Stidsen, Thomas Jacob Riis

Published in:
Proceedings of the Ninth International Conference on the Practice and Theory of Automated Timetabling (PATAT 2012)

Publication date:
2012

Document Version
Publisher's PDF, also known as Version of record

High School Timetabling: Modeling and solving a large number of cases in Denmark

Matias Sørensen · Thomas R. Stidsen

Keywords High School Timetabling · Modeling · Adaptive Large Neighborhood Search

1 Introduction

A general model for the timetabling problem of high schools in Denmark is introduced, as seen from the perspective of the commercial system Lectio[1], and an Adaptive Large Neighborhood Search (ALNS) algorithm is proposed for producing solutions. Lectio is a general-purpose cloud-based system for high school administration (available only for Danish high schools), which includes an embedded application for creating a weekly timetable. Currently, 230 high schools are customers of Lectio, and 191 have bought access to the timetabling software. This constitutes the majority of high schools in Denmark.

This large customer base entails a need for a model of the problem which is general enough to suit many different requirements, while still remain tractable by computer aided solution methods. This supports the recent trend of developing general models for timetabling problems (see Burke et al (1998); Asratian and de Werra (2002); Özcan (2005); Causmaecker and Berghe (2010); Bonutti et al (2010); Post et al (2011, 2012)). Furthermore, the timetabling problem of Danish high schools has not been formally described in the literature before. Some recent formulations of related problems from other countries include Wright (1996); Wood and Whitaker (1998); Bufé et al (2001); Melício et al (2005); Avella et al (2007); Nurmi and Kyngas (2007); Boland et al (2008); Santos et al (2010); Minh et al (2010).

Matias Sørensen
MaCom A/S, DK
Tel.: +45 3379 7900
E-mail: ms@macom.dk

Thomas R. Stidsen
Department of Management Engineering
Technical University Of Denmark

[1] http://www.lectio.dk [lectio@macom.dk], developed by MaCom A/S, Vesterbrogade 48 1., DK-1620 Copenhagen V.

2 Model formulation

The following sets are given:

- Timeslots, usually made up of 5 days and 4-8 daily timeslots.
- Entities, the combined set of students and teachers.
- Classes, a subset of entities which is taught/teaching a specific topic.
- Rooms
- Events (corresponding to lectures), the basic timetabling-unit which must be assigned exactly one timeslot in the timetable.

For each class, a number of events is given (usually between 2 and 5). The basic timetabling problem concerns the assigning of each event to a timeslot, and a room to each event, such that no clashes among entities or rooms occur.

Furthermore we introduce the concept of EventChains, which separates this problem from related problems described in the literature. An EventChain consists of a subset of events, and each of these events are assigned an *offset*. At least one event must have offset 0, corresponding to the start of the EventChain. All events of an EventChain with the same offset must be placed in the same timeslot. All events of offset 1 must be assigned the timeslot following the timeslot assigned to events of offset 0, and so forth. See also Fig. 1. No restrictions are posed on which events can be included in the same EventChain, and the offsets of events in an EventChain are completely for the user to decide, providing a lot of flexibility. E.g. a double lecture can be set up by creating an EventChain consisting of two events for the same class, with offsets 0 and 1, respectively. EventChains also allows for parallel double lectures for several classes, triple lectures, grouping of elective classes in the same timeslot, etc. Many of such features have been requested by the users of Lectio, and EventChains are a pretty generic way of solving them.

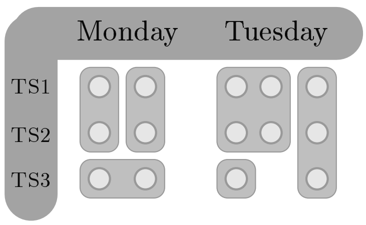

Fig. 1 Six EventChains placed in a partial timetable.

Fig. 2 illustrates the data-model by an example with two classes, each assigned three events, and two EventChains.

Besides the no-clashes constraints, the following hard constraints are included in the model:

- An event cannot be assigned to any of its forbidden timeslots
- Each event must be assigned an admissible room

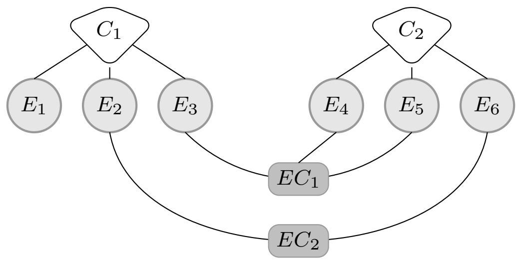

Fig. 2 Two classes and their respective events, and two EventChains.

- Only one event of each class per day (unless otherwise specified in the given EventChain)
- For each teacher, a required number of days-off is given

The following weighted objectives (soft constraints) are used:

- Maximize the number of events assigned to a timeslot (very high priority)
- Maximize the number of events which are assigned a room (high priority)
- Maximize the number of days-off for teachers (low priority)
- Minimize idle slots for entities (medium priority)
- Minimize the amount of different rooms which are assigned to events of the same class (room stability constraint, low priority)
- Minimize the number of occurrences of two events of the same class being assigned two consecutive days (neighbor-day constraint, low priority)

3 Adaptive Large Neighborhood Search

ALNS is a recent extension of the Large Neighborhood Search (LNS) paradigm, often credited to Ropke and Pisinger (2006). As in the LNS framework, first a destruct (ruin/remove) operator is applied to the solution at hand, and then a construct (recreate/insert) operator is used to repair the solution. In an ALNS framework, multiple destruct and construct operators are used, and the adaptive layer keeps track of their individual performance, and increases the probability of selecting operators which have previously performed 'good'. ALNS has mainly been applied to variants of the Vehicle Routing Problem (VRP) (Azi et al (2010); Hemmelmayr et al (2011); Salazar-Aguilar et al (2011); Ribeiro and Laporte (2012)), but lately also other problem-domains (Muller et al (2011); Muller (2010); Kristiansen et al (2011); Kristiansen and Stidsen (2012); Sørensen et al (2012))

We propose here an ALNS heuristic for solving the described timetabling problem. The following remove- and insertion-operators are used:

- InsertGreedy: Insert EventChains in greedy way based on contribution to objective. At each insertion, also attempt to assign rooms to inserted events.

- InsertRegretN: This is similar to the Regret-N neighborhood applied to variants of the VRP (Tillman and Cain (1972); Martello and Toth (1981); Potvin and Rousseau (1993)). The regret-measure is specified in terms of best and second-best insert-move for a given EventChain.

- RemoveRandom: Select N random EventChains and unassign them.
- RemoveRelated: Related to Shaw operator (Shaw (1997, 1998)). Select EventChains to remove based on their similarity between classes, feasible rooms and entities.
- RemoveTime: Select a random timeslot, and remove EventChains assigned to it. Repeat until N EventChains has been removed.
- RemoveClass: Select a random class, and remove EventChains which contains it. Repeat until N EventChains has been removed.

Furthermore we apply a special set of operators, RoomRemove/Insert, which are coupled. In a coupled set of operators, the choice of remove operator implies the choice of repair operator. In RoomRemove, N random room-assignment are removed from the solution. In RoomInsert, rooms are assigned to events in a greedy way.

4 Preliminary results

The Lectio database contains about 4000 datasets from 94 different high schools. Grouping these by school and year entails about 200 'unique' datasets. The authors plan to make at least some of these datasets public, most likely using the XHSTT format (Post et al (2012)).

Table 1 shows statistics and preliminary computational results for six datasets. These results show that the heuristic finds solutions where many events are unassigned, but if they are assigned a timeslot, a suitable room is also assigned. In the full paper, comprehensive computational studies will be made. This will include comparison of the found solutions with a bound provided by an integer programming model.

Name	#E.	#E.C.	#R.	#C.	#Ent.	#T.	#E./w pos.	#E./w room
Sorø2011	424	424	70	132	230	50	360.6 (13.6)	360.6 (13.6)
Skive2010	2955	1749	58	331	304	90	2358.2 (13.9)	2358.2 (13.9)
Fjerritslev2009	530	387	51	152	122	40	487.9 (2.9)	487.9 (2.9)
ViborgT2010	536	447	21	116	85	50	515.2 (1.8)	515.2 (1.8)
Herning2010	1671	1616	86	355	213	60	1545.2 (2.5)	1545.2 (2.5)
Hasseris2011	1343	1080	79	404	578	50	1331.8 (3.2)	1331.8 (3.2)

Table 1 Preliminary results for 10 runs of ALNS heuristic on 6 datasets. Columns 2-9 shows the number of Events, EventChains, Rooms, Classes, Entities, Timeslots, Avg. events assigned to a timeslot, Avg. events assigned to a room, respectively. For columns 8 and 9, also the standard deviation is shown.

References

Asratian A, de Werra D (2002) A generalized class-teacher model for some timetabling problems. European Journal of Operational Research 143(3):531 – 542, DOI 10.1016/S0377-2217(01)00342-3

Avella P, DAuria B, Salerno S, Vasilâev I (2007) A computational study of local search algorithms for italian high-school timetabling. Journal of Heuristics 13:543–556, 10.1007/s10732-007-9025-3

Azi N, Gendreau M, Potvin JY (2010) An Adaptive Large Neighborhood Search for a Vehicle Routing Problem with Multiple Trips. CIRRELT

Boland N, Hughes B, Merlot L, Stuckey P (2008) New integer linear programming approaches for course timetabling. Computers & Operations Research 35(7):2209 – 2233, DOI DOI: 10.1016/j.cor.2006.10.016, part Special Issue: Includes selected papers presented at the ECCO'04 European Conference on combinatorial Optimization

Bonutti A, De Cesco F, Di Gaspero L, Schaerf A (2010) Benchmarking curriculum-based course timetabling: formulations, data formats, instances, validation, visualization, and results. Annals of Operations Research pp 1–12, 10.1007/s10479-010-0707-0

Bufé M, Fischer T, Gubbels H, Häcker C, Hasprich O, Scheibel C, Weicker K, Weicker N, Wenig M, Wolfangel C (2001) Automated solution of a highly constrained school timetabling problem - preliminary results. In: Boers E (ed) Applications of Evolutionary Computing, Lecture Notes in Computer Science, vol 2037, Springer Berlin / Heidelberg, pp 431–440

Burke E, Kingston J, Pepper P (1998) A standard data format for timetabling instances. In: Burke E, Carter M (eds) Practice and Theory of Automated Timetabling II, Lecture Notes in Computer Science, vol 1408, Springer Berlin / Heidelberg, pp 213–222, 10.1007/BFb0055891

Causmaecker PD, Berghe G (2010) Towards a reference model for timetabling and rostering. Annals of Operations Research pp 1–10, 10.1007/s10479-010-0721-2

Hemmelmayr VC, Cordeau JF, Crainic TG (2011) An adaptive large neighborhood search heuristic for two-echelon vehicle routing problems arising in city logistics. Tech. Rep. CIRRELT-2011-42, Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation

Kristiansen S, Stidsen TR (2012) Adaptive large neighborhood search for student sectioning at danish high schools. In: Proceedings of the Ninth International Conference on the Practice and Theory of Automated Timetabling (PATAT 2012)

Kristiansen S, Sørensen M, Stidsen T, Herold M (2011) Adaptive large neighborhood search for the consultation timetabling problem, to appear

Martello S, Toth P (1981) An algorithm for the generalized assignment problem. Operational research 81:589–603

Melício F, Caldeira P, Rosa A (2005) Solving real school timetabling problems with metaheuristics. In: Proceedings of the 4th WSEAS International Conference on Applied Mathematics and Computer Science, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA, pp 4:1–4:8

Minh K, Thanh N, Trang K, Hue N (2010) Using tabu search for solving a high school timetabling problem. In: Nguyen N, Katarzyniak R, Chen SM (eds) Advances in Intelligent Information and Database Systems, Studies in Computational Intelligence, vol 283, Springer Berlin / Heidelberg, pp 305–313

Muller L (2010) An adaptive large neighborhood search algorithm for the multi-mode resource-constrained project scheduling problem. l, Department of Management Engineering, Technical University of Denmark Produktionstorvet, Building 426, DK-2800 Kgs. Lyngby, Denmark

Muller L, Spoorendonk S, Pisinger D (2011) A hybrid adaptive large neighborhood search heuristic for lot-sizing with setup times. European Journal of Operational Research Volume 218(Issue 3):614–623

Nurmi K, Kyngas J (2007) A framework for school timetabling problem. In: Proceedings of the 3rd multidisciplinary international scheduling conference: theory and applications, pp 386–393

Post G, Kingston JH, di Gaspero L, McCollum B, Schaerf A (2011) The third international timetabling competition (itc2011), http://www.utwente.nl/ctit/hstt/itc2011/.

Post G, Ahmadi S, Daskalaki S, Kingston J, Kyngas J, Nurmi C, Ranson D (2012) An xml format for benchmarks in high school timetabling. Annals of Operations Research 194:385–397

Potvin JY, Rousseau JM (1993) A parallel route building algorithm for the vehicle routing and scheduling problem with time windows. European Journal of Operational Research 66(3):331 – 340

Ribeiro GM, Laporte G (2012) An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem. Computers & Operations Research 39(3):728 – 735, DOI 10.1016/j.cor.2011.05.005

Ropke S, Pisinger D (2006) An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transportation Science 40:455–472

Salazar-Aguilar M, Langevin A, Laporte G (2011) An adaptive large neighborhood search heuristic for a snow plowing problem with synchronized routes. In: Pahl J, Reiners T, Voss S (eds) Network Optimization, Lecture Notes in Computer Science, vol 6701, Springer Berlin / Heidelberg, pp 406–411

Santos H, Uchoa E, Ochi L, Maculan N (2010) Strong bounds with cut and column generation for class-teacher timetabling. Annals of Operations Research pp 1–14, 10.1007/s10479-010-0709-y

Shaw P (1997) A new local search algorithm providing high quality solutions to vehicle routing problems

Shaw P (1998) Using constraint programming and local search methods to solve vehicle routing problems. In: Maher M, Puget JF (eds) Principles and Practice of Constraint Programming — CP98, Lecture Notes in Computer Science, vol 1520, Springer Berlin / Heidelberg, pp 417–431

Sørensen M, Kristiansen S, Stidsen TR (2012) International timetabling competition 2011: An adaptive large neighborhood search algorithm. In: Proceedings of the Ninth International Conference on the Practice and Theory of Automated Timetabling (PATAT 2012)

Tillman FA, Cain TM (1972) An upperbound algorithm for the single and multiple terminal delivery problem. Management Science 18(11):664 – 682

Wood J, Whitaker D (1998) Student centred school timetabling. The Journal of the Operational Research Society 49(11):1146–1152

Wright M (1996) School timetabling using heuristic search. The Journal of the Operational Research Society 47(3):347–357

Özcan E (2005) Towards an xml-based standard for timetabling problems: Ttml. In: Kendall G, Burke EK, Petrovic S, Gendreau M (eds) Multidisciplinary Scheduling: Theory and Applications, Springer US, pp 163–185