Heuristic Strategies to Modify Existing Timetables

Edmund Burke[*], Barry McCollum[†], Paul McMullan[†], Jay Yellen[*#]

[*] School of Computer Science
University of Nottingham
Jubilee Campus
Nottingham, NG8 1BB, UK
{ekb, jzy}@cs.nott.ac.uk

[†] School of Computer Science
Queen's University
Belfast
University Road
N. Ireland
BT7 1NN
{b.mccollum, P.P.McMullan}@qub.ac.uk

[#] Department of Mathematics and Computer Science
Rollins College
Winter Park
Florida, 32789 USA
jyellen@rollins.edu

The development of interactive decision support systems provides a much needed practical framework for the automation of timetabling procedures within organisations. The research literature tends to represent the timetabling process as a fully automated one – a dataset is loaded, one or more algorithms are applied, and a solution is returned [Qu et al (2006)]. In practice, however, user interaction plays a central role in producing a final solution [McCollum (2007)]. One of the strongest arguments for non-automation is that the input data is often incomplete or inaccurate or may contain elements that come to light only after a solution construction has begun. Typically, following the preparation of the data and some initial attempts to construct a solution, the user undertakes a series of improvement steps (modifications or repairs). Here, 'improvement' takes on a quite different and broader meaning than with its traditional use by researchers in the context of local search techniques, metaheuristics, and evolutionary computation. Our research aims to make it easier and more natural for the human timetabler to react and make appropriate changes interactively as the timetable evolves, and, as such, represents an important step towards bridging the gap that currently exists between timetabling research and practice [McCollum (2007)]. The two interrelated components of this work are: (a) the design and implementation of heuristic strategies for modifying an existing timetable, which will become part of the automatic timetabling algorithm and which will also guide the user's interactive decisions; and (b) the development of an intelligent and intuitive graphical user interface, which will allow the user to participate throughout the solution process.

User interaction with the developing solution, which is important for many institutional reasons, takes the form of modification of resources, constraints or evaluation criteria. The extent of these modifications varies depending on the stage of solution development. Many institutions begin their timetabling process with the previous year's timetable, and initial changes are made based on new circumstances, e.g., changes in timeslot, room, staff, or equipment availability. Once an initial timetable has been drafted, further modifications may be requested (or required). For example, a single event might have to be moved, but all other timeslots are either infeasible for it or will cause additional violations of some soft constraint, unless other events are also moved. A reasonable strategy is to fulfill the request while minimising the number of changes to the overall timetable, i.e., construct a minimum perturbation [Müller et al (2005)]. The strategies of incurring further soft penalties, breaking hard constraints (the timetabler may decide this is acceptable) or making multiple 'intelligent' changes (minimising the ripple effect) are all possible options. An effective system would help the user see the trade-offs and impact of as many of these options as possible without overloading the automated or manual components. It should also allow the user to run various 'what if' scenarios in testing new ideas or in evaluating hunches that may lead to a better final timetable. Clearly, the user's experience and intuition is likely to be a valuable asset in the construction of high quality solutions, and any timetabling system that can take advantage of these features is better for it.

The first phase of our research focuses on the design and implementation of a decision-support mechanism for examination timetabling, which will eventually be adapted to a general timetabling system. One of the more common objectives in examination timetabling is to minimize the number of students having little or no time between exams. Accordingly, when rescheduling an exam to another timeslot, it is desirable not to place the exam in close proximity with *neighbouring* exams, i.e., those with which it has overlapping student enrolments. This project builds on the weighted graph model introduced in [Carrington et al (2007)] in which vertices correspond to exams, edges indicate neighbouring exams, and colours represent timeslots. The model quantifies and keeps track of the undesirability of each colour assignment for each vertex as the colouring progresses. In particular, a *conflict penalty* is incurred if neighbouring vertices (exams) are assigned the same colour (timeslot), and a *proximity penalty* is incurred when neighbouring vertices are assigned two colours representing timeslots too near each other. The heuristic strategies for vertex and colour selection introduced in that paper, and revised and extended in [Burke et al (2008)] are largely based on these two quantities, which are updated after each vertex is coloured.

Whereas, those two papers focus on the initial solution, our work here is concerned with the improvement and repair of existing solutions. We adapt those vertex- and colour-selection strategies and the model parameters on which they rely to guide decisions for uncolouring and recolouring certain vertices on demand. This forms the core of both the manual and automated components of our system for modifying an existing timetable. A byproduct of this effort is a backtracking component that can augment the one-pass constructive algorithm introduced in [Carrington et al (2007)].

The graphical user interface, which is still very much under development, builds on the work presented in [Ahmadi et al (2002)] and aims at providing a visual framework within which the user can combine intuition with our algorithmic strategies to confidently and quickly make decisions on how the current solution can be completed and/or improved. Our conference presentation will include a demonstration of some of the features of our system. For instance, we will randomly select an exam and demand that its current timeslot be changed. The user will then be able to view and select from various options that are displayed. We will also present initial experimental results on the effectiveness of our underlying strategies for this modify/repair mechanism.

References

Burke et al (2008), Burke EK, Pham N, Qu R, Yellen J, Some New Heuristic Strategies for Examination Timetabling, to be presented at PATAT 2008, Montreal, Canada.

Carrington et al (2007), Carrington JR, Pham N, Qu R, Yellen J, An Enhanced Weighted Graph Model for Examination/Course Timetabling, Proceedings of 26th Workshop of the UK Planning and Scheduling, 2007, pp. 9-15.

McCollum (2007), McCollum B, A Perspective on Bridging the Gap between Theory and Practice in University Timetabling, Practice and Theory of Automated Timetabling VI (PATAT 2006), Springer Lecture Notes in Computer Science Vol. 3867, 2007, pp 3-23.

Qu et al (2006), Qu R, Burke EK, McCollum B, Merlot LTG, and Lee SY, A Survey of Search Methodologies and Automated Approaches for Examination Timetabling, Technical Report NOTTCS-TR-2006-4, School of Computer Science, University of Nottingham, 2006, accepted for publication in Journal of Scheduling, to appear 2009.

Ahmadi et al (2002) Ahmadi S, Barone SR, Burke EK, Cheng P, Cowling P, McCollum B, Integrating human abilities and automated systems for timetabling: a competition using STARK and HuSSH Representations, Proceedings of the Conference on the Practice and Theory of Automated Timetabling (PATAT 2002), Gent, Belgium, August 2002, pp. 265-273.

Müller et al (2005), Müller T, Rudovà H, Bartak R, Minimal Perturbation Problem in Course Timetabling, Practice and Theory of Automated Timetabling V (PATAT 2004), Springer Lecture Notes in Computer Science Vol. 3616, 2005, pp 126-146.