DECENTRALISED COURSE TIMETABLING IN A LARGE HIERARCHICAL ORGANISATION

Mieke Adriaen, Patrick De Causmaecker*, Greet Vanden Berghe
KaHo Sint-Lieven, Information Technology
Gebr. Desmetstraat 1, 9000 Gent, Belgium
{mieke.adriaen,greet.vandenberghe}@kahosl.be
* *School of Computer Science and IT, University of Nottingham*
Jubilee Campus, Nottingham, NG8 1BB, UK
pxc@cs.nott.ac.uk

Abstract

Keywords: Timetabling, Agent Based Scheduling, Meta-heuristic Search

Introduction

We present a practical, distributed approach for university timetabling that is not restricted to one specific school organisation. The proposed system is called TITAN (Transparent Interactive Timetabling based on Agent Negotiation). It is the main purpose of TITAN to assign, subject to constraints, lecture sessions to rooms and time slots, such that all students can attend their entire program.
The model allows for decomposition into variable subproblems between which agent based negotiation assists in assigning common resources. Constraints exist on different levels in the school hierarchy. They are set by means of a symbolic constraint language. Each subproblem has meta-heuristics at its disposal for searching local improvements.

1. Problem

TITAN tackles timetabling problems for schools with different organisational levels and different teaching locations. KaHo Sint-Lieven, a school with 6 campuses and 4 departments, is an interesting test case. Departments are subdivided in educational sections, to which the students belong.

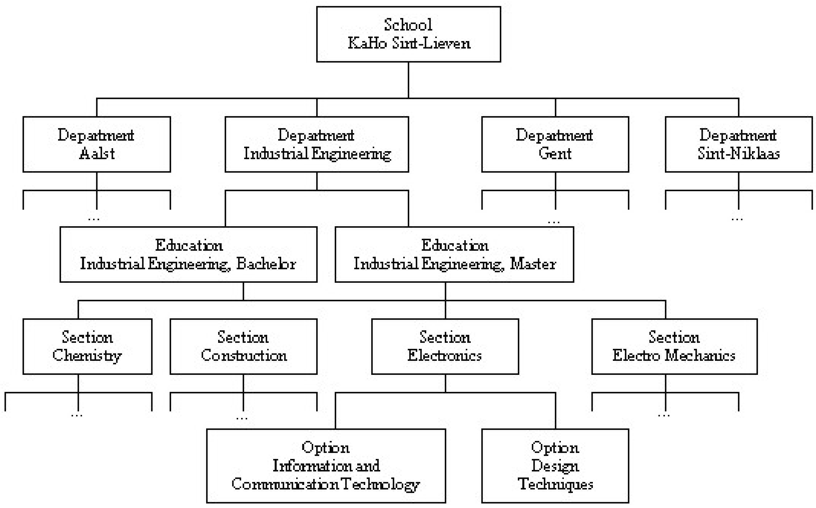

Figure 1. Organisation of the school

The educational organisation of the school is presented in Fig. 1. Students following the same education can take completely different subjects within an academic year. Timetabling aims at constructing schedules (without serious conflicts), that suit lecturers, student groups but also individual students [Burke et al., 2000; Burke et al., 1995; Carter, 2000; Paechter et al., 1995].

Unlike in secondary schools, university timetables differ from week to week. Certain courses are organised weekly, others every fortnight, but there are also modules that are not scheduled in regular sequences. Rooms can be used by different departments, and certain lecturers teach on different campuses. Moreover, the lecture and exam periods can differ for different departments. The school organisation invokes a large set of constraints that should preferably be satisfied. In practice, some constraint violations are unavoidable and they determine the quality of timetables.
It is important for practical use to have a clear and informative user interface [Ahmadi et al., 2002; McCollum et al., 2002]. We have paid special attention to visualising actions and constraints, and we provide expressive interfaces for different aspects of the timetabling problem.

2. Model

Distributed Timetabling

TITAN allows splitting the problem into subproblems that relate to different departments or other organisational units. The reduced dimensions, compared to the global problem, render the search for solutions per subproblem easier [Paechter et al., 1997]. During the planning, negotiation is required between the separate departmental problems, in order not to violate constraints for resources that are applied in more than one department.

Once a timetable is fixed, little changes are required from time to time due to temporary unavailability of resources. We propose a model in which different authorities are given to actors in the timetabling problem. It is not necessary to contact the timetabling manager for simple shifts, such as switching two lectures for the same group of students.

Sessions

The basic concepts in TITAN are called sessions. We define a session as a time-tabling block with a subject, one or more lecturers, one or more student groups, and a duration. Sessions can be shifted around in time and place.

Although there is often a high degree of freedom for setting start and end times for sessions, there are certain customs in every educational section. The common times and session durations for a subproblem are combined in a virtual roster. An example of such virtual roster is presented in Fig. 2. It allows theoretical sessions of 90' and practical sessions of 220' (in which a 10' break is provided). Possible start and end times for the sessions are set by users. These times determine the lines of the virtual roster.

Timetabling becomes less complex by only allowing assignments of sessions if they exactly match an element of the virtual roster. The lower the tackled problem is in the school's hierarchy (see Fig. 1), the smaller the number of blocks in the virtual roster. For the meta-heuristic search, this involves a more restricted neighbourhood for moving the sessions [Di Gaspero and Schaerf, 2002].

Constraints

In order to easily define, set, and interpret constraints on different levels of the organisation, we apply a simple symbolic language. Details of this language will be explained in the full paper.

TITAN contains a number of default constraints (e.g. no assignment for

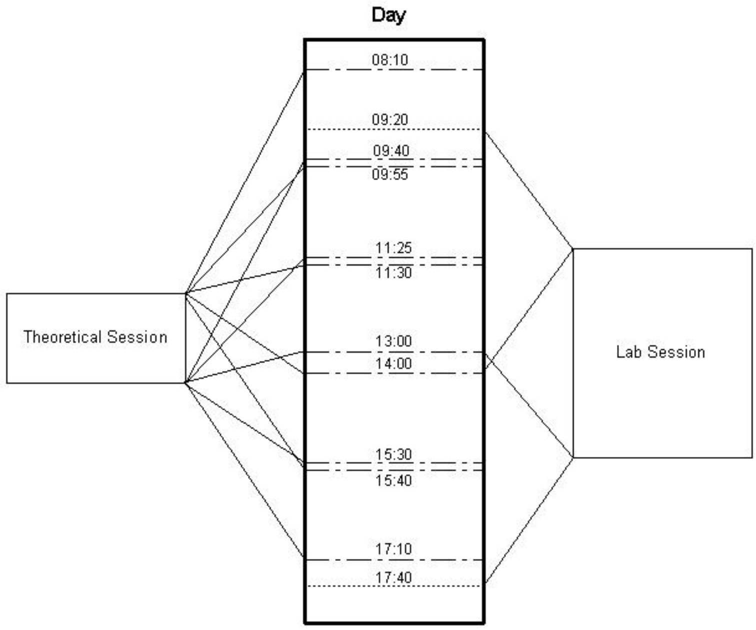

Figure 2. Example of a virtual roster for one day

people to different rooms at the same time) but users need not necessarily activate them. The constraints are organised in different levels, corresponding to the authority levels in the school structure.
A number of constraints are imposed by the general school system. These constraints are hard, they need to be satisfied on every sublevel. Some of these hard constraints can however locally be redefined on a sublevel. Examples of general hard constraints are:

- lecturing activities cannot be organised outside the academic year calendar
- for every student, the entire program should be scheduled
- regular students should not have overlapping courses
- the number of students in a session should not exceed the room capacity
- ...

A set of constraints can be set per department:

- each student's timetable should at least contain a number of free half days every week
- certain lectures should be scheduled after other lectures are finished

- lectures should be scheduled in rooms with suitable equipment
- lectures belonging to the same course, and given to the same group of students, should preferably be scheduled in the same time slot every week
- ...

Authorised people can set a list of new constraints that only hold in the subproblem they are responsible for. For the lowest timetabling level, constraints can be:

- sessions for certain subjects must not exceed an adjustable number of hours
- sessions for certain subjects must not be scheduled at certain time slots.

Multi Agent Approach

The authority of users determines at which level they are allowed to manipulate the timetable. All users have an individual timetabling agent at their disposal.

Meta-heuristics. Each timetabling agent has control of meta-heuristics to create or improve solutions for its own problem. We define an objective function that is related to the number of violations of constraints and to the corresponding cost parameters. The goal for local search is the reduction of the value of the objective function, in which, per local subproblem, different constraints are evaluated.

Setting up new timetables is organised in two steps.
An initialisation procedure first creates sessions for all the lacking activities. These are activities in the educational program that do not yet appear in the timetable. A simple assignment heuristic is responsible for inserting these new sessions into the timetable (i.e. connecting them to a time slot and a room).
In the next step, the agents apply a tabu search algorithm. It enables reducing the value of the objective function by moving sessions around the rooms and time slots. Within this process, it is not possible for a timetabling agent to relocate sessions that are not under its responsibility. Since rooms, lecturers, etc have to be available for more than one department, this is often a very strong restriction.
We therefore allow the algorithms to look at parts of the timetable that are not under the user's responsibility, and to search for better solutions by reconsidering these 'external' timetables. In the next section we explain briefly how the negotiation is organised.

Agent based Negotiation. When different timetables require the same resources, a negotiation procedure will sort out the best common solution. After the search for a solution, agents communicate with other agents that represent different parts of the problem [Meisels and Kaplansky, 2002]. The process mainly involves negotiating about constraints, such that agents can improve the quality of their local solution. We initially model the agent negotiations upon human negotiations for manual timetabling [De Causmaecker et al., 2000]. The details of the negotiations will be explained in the full paper.

3. Conclusions

TITAN enables modelling large and distributed timetabling problems. Local agents can handle subproblems by means of meta-heuristic search. They negotiate, if necessary, about constraints affecting common resources.

The benefits of such a distributed approach are numerous:

- local customs can be modelled without affecting the generality
- timetabling problems become less complex by dividing them into organisational subproblems
- conflicts are avoided on the global level by allowing the local agents to consider the global constraints and resources
- small temporary changes to local timetables can be put through locally.

References

S. Ahmadi, R. Barone, E.K. Burke, P. Cheng, B. McCollum: Integrating human abilities and automated systems for timetabling: A competition using STARK and HuSSH Representations at the PATAT 2002 Conference, *E.K. Burke, P. De Causmaecker (Eds.): Proceedings of Practice and Theory of Automated Timetabling, Gent*, 265-273, 2002

E.K. Burke, Y. Bykov, S. Petrovic: A Multicriteria Approach to Examination Timetabling, *E.K. Burke, W. Erben (Eds.): Practice and Theory of Automated Timetabling, Third International Conference, Konstanz*, Springer, 118-131, 2000

E.K. Burke, J.P. Newall, R.F. Weare: A Memetic Algorithm for University Timetabling, *E.K. Burke, P. Ross (Eds.): Practice and Theory of Automated Timetabling, First International Conference Edinburgh*, Springer, 241-250, 1995

M.W. Carter: A Comprehensive Course Timetabling and Student Scheduling System at the University of Waterloo, *E.K. Burke, W. Erben (Eds.): Practice and Theory of Automated Timetabling, Third International Conference, Konstanz*, Springer, 64-82, 2000

P. De Causmaecker, P. Demeester, P. De Pauw-Waterschoot, G. Vanden Berghe: Agents in a Route Planning Application, *JORBEL Special Issue Orbel-14: Emerging Challenges in Operations Research*, 40(1-2), 105-116, 2000

L. Di Gaspero, A. Schaerf: Multi-Neighbourhood Local Search for Course Timetabling, *E.K. Burke, P. De Causmaecker (Eds.): Proceedings of Practice and Theory of Automated Timetabling, Gent*, 128-132, 2002

B. McCollum, S. Ahmadi, E.K. Burke, R. Barone, P. Cheng, P. Cowling: A Review of Existing Interfaces of Automated Examination and Lecture Scheduling Systems, *E.K. Burke, P. De Causmaecker (Eds.): Proceedings of Practice and Theory of Automated Timetabling, Gent*, 262-264, 2002

A. Meisels, E. Kaplansky: Scheduling Agents - Distributed Employee Timetabling (DETP), E.K. Burke, P. De Causmaecker (Eds.): Proceedings of Practice and Theory of Automated Timetabling, Gent, 182-184, 2002

B. Paechter, A. Cumming, H. Luchian: The use of local seach suggestion lists for improving the solution of timetabling problems with evolutionary algorithms, *T. Fogerty (Ed.): Evolutionary Computation, Springer Lecture Notes in Computer Science*, 993, 86-93, 1995

B. Paechter, R.C. Rankin, A. Cumming: Improving a Lecture Timetabling System for University-Wide Use, *E.K. Burke, M. Carter (Eds.): Practice and Theory of Automated Timetabling II, Second International Conference, Toronto*, Springer, 156-165, 1997