

***UTTSE*Exam: A University Examination Timetable Scheduler**

Andrew Lim, Juay-Chin Ang, Wee-Kit Ho, Wee-Chong Oon

School of Computing, National University of Singapore
3 Science Drive 2
Singapore 117543, Singapore
{alim, angjc, howk, oonwc}@comp.nus.edu.sg

Abstract

*UTTSE*Exam is a university examination timetable-scheduling program that was successfully employed to create the examination timetable for semester 1 of the 2001/2002 academic year in the National University of Singapore. This demonstration provides insight on the various components of the system, including the hybrid centralized cum de-centralized scheduling strategy, the Combined Method scheduling algorithm and the overall process required to create the final timetable.

Introduction

The scheduling of the examination timetable for a major university is a complicated and often expensive affair. While most universities still schedule their examination timetables manually, this process can take weeks, is error-prone and is unable to easily cater for the inevitable last-minute changes. Coupled with the increasing popularity of cross-faculty modules (modules that may be taken by students from different faculties), much can be gained by automating the examination timetabling process.

*UTTSE*Exam (Lim et al. 2000, Lim et al. 2002) is one such automated university examination timetabling program. It began development in 1999 in the National University of Singapore, and was recently deployed to schedule the timetable for semester 1 of the 2001/2002 academic year for NUS. The program is split into two versions: the Registrar Version allows unlimited access to all portions of the program, while the Faculty Version allows only access to the information pertaining to the relevant faculty. A divide-and-conquer methodology is employed, whereby the individual faculties first schedule their own timetables using the Faculty Version. These faculty timetables are then uploaded to the Registrar Version to be merged into the collated timetable.

This demonstration shows *UTTSE*Exam in action, including the data input modules, the scheduling process and the final timetable generated. In order to allow full appreciation of the system, we also explain the hybrid

centralized cum de-centralized scheduling strategy and the Combined Method scheduling algorithm.

Demonstration

Both the full-fledged Registrar Version and the stripped-down Faculty Version of *UTTSE*Exam will be on display. The actual, real-life data and constraints from semester 1 of the 2001/2002 academic year in NUS will be loaded into the system to show the magnitude of an actual timetabling problem of a major university.

Scheduling Strategy

One of the most important aspects of university timetabling is the scheduling strategy. In general, there are two possible strategies, each with their own advantages and disadvantages. The *centralized* strategy has a central authority perform the entire scheduling task. While this allows the greatest global view of the problem, the sheer amount of data that a real-life examination timetabling problem of a large university makes it difficult to produce a good solution. The *de-centralized* strategy lets each faculty schedule their own timetables. While this is ideal if all the faculty timetables are entirely disjoint, modern universities offer several cross-faculty modules. The increasing necessity of communication between faculties makes the de-centralized strategy clumsy and infeasible.

As a result, a hybrid approach is adopted (Ho, Lim and Oon 2002). Before student registration, each faculty is assigned a number of seats in each session (called a *venue partition*) based on their examinations' enrolment estimates. Each faculty would then schedule their exams according to their venue partitions, using the Faculty Version, which are then merged into a campus-wide tentative timetable. After student registration, the tentative timetable is updated with the finalized information. The exams are then allocated to their actual venues, verified by the individual faculties and published.

Scheduling Algorithm

The scheduling algorithm used by *UTTSE*Exam is the Combined Method (Ho and Lim 2001). This method makes use of a stochastic search technique to first find a solution that may not be entirely feasible (i.e. some hard

constraints may be violated) but is of high quality in terms of a weighted sum of the different constraints satisfied. This solution is then used to guide a selection algorithm with consistency checking to create a feasible timetable that still retains most of the quality of the initial solution.

UTTSEExam employs the Genetic Algorithm (Marin 1998) as its stochastic search technique, with Tabu Search (Rayward-Smith et al. 1996) post-optimization. This guides the Variable Ordering Method with AC-3 (Mackworth 1977) consistency checking.

Application Modules

The *UTTSEExam* program is made up of several parts. Several modules allow the input of information, including examination, student, venue and session information. Of particular interest is the input on constraint information, which is a crucial part of any timetabling system. The system allows:

- Unary constraints (e.g. paper *X* is to be held in venue *V*),
- Binary constraints (e.g. paper *X* must be at least 3 sessions before paper *Y*),
- Inter-group constraints (e.g. all papers in group *A* must be before all papers in group *B*), and
- Intra-group constraints (e.g. the papers in group *A* must be at least 2 sessions apart).

Additionally, *UTTSEExam* allows output of all the internal information in the form of html files for ready display on the university web pages. There are also tools for defining the venue layouts and the assignment of seat numbers to each candidate.

Benefits

When compared to the only alternative of manual scheduling, automated timetable scheduling offers tremendous benefits.

- Manual scheduling usually takes several weeks to complete. *UTTSEExam* takes only a few minutes.
- The timetable generated by *UTTSEExam* required fewer days and sessions than those created using the previous manual system. Since some of the examinations are held in commercial venues that require rental, this translates to a substantial monetary saving.
- Previous manual timetables often contain overlooked constraint conflicts. The automated timetable is conflict-free.
- *UTTSEExam* can easily handle last-minute changes to the data.
- In the future, administrators can use the system to test the feasibility of possible policy changes, like the shortening of the examination period.

Conclusion

UTTSEExam is a university examination timetable scheduler that has been successfully used to create the

timetable for a major university. This demonstration seeks to reveal how the various techniques employed have been incorporated into a coherent whole. Admittedly, university examination timetabling is an immensely complex and tedious process to automate, but the potential benefits are similarly tremendous. We hope to show that the automation of university examination timetabling is well worth the effort.

References

- Ho, W. K. and Lim, A. *A Hybrid-Based Framework for Constraint Satisfaction Optimization Problems*, in International Conference on Information Systems (ICIS) 2001, pg. 65-76.
- Ho, W. K.; Lim, A.; and Oon, W. C. *UTTSEExam: A University Examination Timetabling System*, submitted to IEEE Intelligent Systems 2002.
- Lim, A.; Ang, J. C.; Ho, W. K.; and Oon, W. C. *A Campus-Wide University Examination Timetabling Application*, in Innovative Applications in Artificial Intelligence (AAAI/IAAI) 2000, pg. 1020-1025
- Lim, A.; Ang, J. C.; Ho, W. K.; and Oon, W. C. *UTTSEExam: A Campus-Wide University Examination-Timetabling System*, submitted to Innovative Applications in Artificial Intelligence (AAAI/IAAI) 2002
- Mackworth, A. K. *Consistency in Networks of Relations*, in Artificial Intelligence 8 (1977): 88-119
- Marin, H. T. "Combinations of GA and CSP Strategies for Solving the Examination Timetabling Problem", *Ph.D. thesis, Instituto Tecnológico y de Estudios Superiores de Monterrey*, 1998.
- Rayward-Smith, V. J.; Osman, I. H.; Reeves, C. R.; and Smith, G. D. *Modern Heuristic Search Methods*, 1996.