Concise Paper of the NACCQ Conference 2000
Wellington, New Zealand

Scheduling of Polytechnic Timetables by Constraint Logic Programming, Genetic Algorithms and a Visual Timetabling Tool.

Si-Eng Ling
Auckland University of Technology
si.ling@aut.ac.nz

ABSTRACT

Three approaches to the scheduling of Polytechnic timetables are described. PROLOG for constraint logic programming was capable of generating 98% of all timetable entries. Genetic Algorithms as a parallel search technique was experimented in a case study for global optimisation. Finally, a Visual Basic application for timetable management, which can be demonstrated, is outlined.

Keywords
Scheduling, Timetabling, Constraint Satisfaction.

1. Introduction

Timetabling is a NP-complete problem; so no deterministic algorithm can provide a solution in polynomial time. Even with exponential time, there is also no guarantee of a feasible solution using a deterministic approach. Hence, practical techniques for timetabling would employ non-deterministic, randomised, stochastic algorithms.

2. Overview

A PROLOG-based assignment program was written to plan for the timetables of a large academic department. The timetables involved 1980 students (in 99 groups), 97 tutors and 54 classrooms. The program has to satisfy numerous constraints, e.g. lunch hour for students and tutors, minimum and maximum number of lessons per day, even spread of early morning and late afternoon classes for tutors, maximum of 2 hour waiting between lessons for students, etc. The most difficult, however, is due to the heavy utilisations of classrooms. This program could generate 98% of all lessons.

Genetic Algorithms (GAs) simulate the biological process of evolution. Beginning with a set of randomly generated pool of candidate solutions (parents), genetic operators (crossover, mutation, etc.) are applied with different probabilities to propagate the offspring through thousands of generations until globally optimal solutions emerge. An objective function is used to evaluate each offspring's fitness by assigning various penalty costs for constraint violations. The more promising candidates are then given higher chances of being selected for propagation while the less fit are gradually eliminated. GAs were used to generate optimised timetables for 10 groups of 200 students for a case study of 100% classroom utilisation. But for much larger student populations with more constraints, GAs became less practical, in terms of processing times and programming efforts.

Finally, a Visual Basic program with Access database (TT Expert) was designed to manage the timetables generated by the above-mentioned PROLOG program. It allowed the last 2% of lessons to be inserted and overall timetables fine-tuned by human intervention. TT Expert is interactive, user-friendly and network-based for easy editing, viewing, checking, printing, etc. This has become commercially available and can be demonstrated by arrangement.

3. Conclusion

Three approaches to timetabling are described and full details are presented in the NACCQ poster.