Appl Intell (2012) 37:1–11
DOI 10.1007/s10489-011-0309-9

A graph coloring constructive hyper-heuristic for examination timetabling problems

Nasser R. Sabar · Masri Ayob · Rong Qu · Graham Kendall

Published online: 3 August 2011

Abstract In this work we investigate a new graph coloring constructive hyper-heuristic for solving examination timetabling problems. We utilize the hierarchical hybridizations of four low level graph coloring heuristics, these being largest degree, saturation degree, largest colored degree and largest enrollment. These are hybridized to produce four ordered lists. For each list, the difficulty index of scheduling the first exam is calculated by considering its order in all lists to obtain a combined evaluation of its difficulty. The most difficult exam to be scheduled is scheduled first (i.e. the one with the minimum difficulty index). To improve the effectiveness of timeslot selection, a roulette wheel selection mechanism is included in the algorithm to probabilistically select an appropriate timeslot for the chosen exam. We test our proposed approach on the most widely used uncapacitated Carter benchmarks and also on the recently introduced examination timetable dataset from the 2007 International Timetabling Competition. Compared against other methodologies, our results demonstrate that the graph coloring constructive hyper-heuristic produces good results and outperforms other approaches on some of the benchmark instances.

Keywords Examination timetabling · Graph coloring · Hybridization · Hyper-heuristics · Roulette wheel selection

1 Introduction

Educational timetabling is an ongoing challenge that most academic institutions face when scheduling courses or exams. This is due to the large number of constraints that have to be accommodated. Courses are often scheduled by the individual faculties and departments, whereas the examination timetable is usually centrally generated to cover the entire university. Both problems are complex, involving a variety of constraints, and thus present a challenging topic for both researchers and practitioners.

Examination timetabling can be defined as the process of assigning a set of exams into a limited number of timeslots and rooms so as not to violate any hard constraints and to minimize soft constraint violations as much as possible [36]. *Hard constraints* have to be respected in order to have a feasible timetable. For example, no student can sit more than one exam at the same time and there must be a sufficient number of seats to accommodate the exams being scheduled in a given room. A *soft constraint* represents a constraint that, ideally, should be satisfied as far as possible. However, the timetable is still considered feasible even if some of these soft constraints are violated. An example of a soft constraint is that exams should be spread within the timetable for a given student, and the aim is to minimize

N.R. Sabar (✉) · M. Ayob
Data Mining and Optimisation Research Group (DMO), Centre for Artificial Intelligent (CAIT), Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor, Malaysia
e-mail: naserdolayme@yahoo.com

M. Ayob
e-mail: masri@ftsm.ukm.my

R. Qu · G. Kendall
ASAP Research Group, School of Computer Science, The University of Nottingham, Nottingham NG8 1BB, UK

R. Qu
e-mail: rxq@cs.nott.ac.uk

G. Kendall
e-mail: Graham.Kendall@nottingham.ac.uk

G. Kendall
University of Nottingham Malaysia Campus, 43500 Semenyih, Selangor, Malaysia

soft constraint violations. Therefore, the timetabling problem can be considered as a problem of minimizing soft constraint violations, while respecting all the hard constraints.

A large number of different approaches have been developed for solving examination timetabling problems in the last four decades. These include graph based sequential techniques [7, 39], constraint based techniques [28, 29], local search methods including tabu search [14], simulated annealing [12, 17, 18, 40], population based algorithms including genetic algorithms [37], ant colony optimization [15], scatter search [22], pattern recognition based method [21] and hybrid approaches [1, 38], etc. For more details please refer to [36]. Most of these methods aimed to develop problem specific techniques that are able to produce the best results for one or more datasets [6, 9].

Recently, there has been a growing trend toward more general methods. Hyper-heuristics represent one of these approaches [6, 9, 11]. The term hyper-heuristic refers to an approach that focuses on a search space of heuristics rather than a search space of solutions [7, 9, 34]. Low level heuristics (e.g. different neighborhood move structures or different constructive heuristics) are controlled by high level general mechanisms (e.g. meta-heuristics or reinforcement learning) in order to provide solutions to a wider variety of problems, rather than developing tailor-made solutions for each problem encountered [9, 24].

However, Qu and Burke [34] commented that a small change to the heuristic list (especially at the beginning) often results in quite different solutions being generated. Thus, as long as the high level search is diversified, a simple multi-start local search works as well as a fine tuned tabu search when single heuristics are used in heuristic lists in their graph based hyper-heuristics (GHH).

In GHH, as only single heuristics are used in heuristic lists, a lot of ties may occur when ordering (and selecting) exams and assigning them to timeslots during solution construction. By simply randomly choosing an exam from those of the same rank, potentially good solutions may be missed. Also by assigning a chosen exam to the first least-cost timeslot, only a small part of the solution search space can be explored by using the simple heuristic lists.

Therefore, in this work, we introduce a new graph coloring constructive hyper-heuristic (GCCHH) which utilizes the hybridizations of four graph coloring heuristics in constructing four ordered lists of exams in timetable construction. GCCHH employs hierarchical heuristics and probabilistic timeslot selection. The focus is not to design another high level search but to investigate more intelligent criteria of exam ranking and timeslot selection. The latter is seldom studied in timetabling research. More potential solutions, of higher quality, in the solution space can thus be found. Instead of sequentially applying low level heuristics to construct timetables, as has been used in previous research [7, 34], the hybridizations of the four low level heuristics are applied simultaneously. Four heuristic hybridizations, consisting of different graph coloring heuristics, have been developed and tested on the un-capacitated (where the size of the room is disregarded) Carter benchmarks (Toronto *b*, see [36]) and the ITC 2007 [23, 26] examination timetabling datasets.

The paper is organized as follows: Sects. 2 and 3 review related work on hyper-heuristics and the ITC 2007 examination timetabling, respectively. Our proposed GCCHH approach is presented in Sect. 4, followed by our results in Sect. 5. Finally discussions and concluding remarks are presented in Sects. 6 and 7.

2 Related work on hyper-heuristics for examination timetabling problems

Recently, hyper-heuristics have been attracting increasing research attention. Burke et al. [9] define a hyper-heuristic as: "*A search method or learning mechanism for selecting or generating heuristics to solve computational search problems*". The high level mechanism of the hyper-heuristic, at each iteration, selects the appropriate low level heuristic based on certain selection criteria. The high level mechanism can be, for example, any kind of meta-heuristic algorithm; whilst the low level heuristics can be, for example, constructive heuristics, graph heuristics or a local search. One of the goals of a *hyper-heuristic* is to provide solution methodologies which are more general than currently possible [9].

Two types of hyper-heuristics are distinguished in the literature, namely constructive and improvement based hyper-heuristics [9]. Constructive based hyper-heuristics start with an empty timetable, and select low level heuristics to build a solution step by step. Improvement based hyper-heuristics start with an initial solution and, at each iteration, selects appropriate improvement low level heuristics to perturb the solution. These two types of approaches can be further extended to on-line or off-line approaches, based on the learning methods employed. In on-line hyper-heuristics, the learning takes place during the problem solving. In off-line hyper-heuristics, the learning happens during the training phase before solving other problem instances (see [10]). Our work concentrates on on-line constructive hyper-heuristics for solving examination timetabling problems.

Some hyper-heuristic approaches have been studied for examination timetabling problems. These include tabu search [5, 7, 19, 20], case-based reasoning [8, 41], variable neighborhood search [6, 34, 35], graph based methods [2, 5, 7, 34, 35], memetic algorithms [16], heuristic combinations [31] and genetic programming [32, 33]. More details of these hyper-heuristics can be found in a recent survey by Burke et al. [6].