Hyper-Heuristics for Educational Timetabling

Nelishia Pillay

School of Mathematics, Statistics and Computer Science

+27 33 2605644

+27 33 2605648

pillayn32@ukzn.ac.za

Abstract: Hyper-heuristics aim at providing generalized solutions to combinatorial optimization problems. Educational timetabling encompasses university examination timetabling, university course timetabling and school timetabling. This paper provides an overview of the use of hyper-heuristics to solve educational timetabling problems. The paper then proposes future research directions focusing on using hyper-heuristics to provide a generalized solution over the domain of educational timetabling instead of for a specific timetabling problem.

Keywords: hyper-heuristics, educational timetabling, university examination timetabling, university course timetabling, school timetabling

1. Introduction

Whereas research into solving combinatorial optimization problems have generally focused on producing the best results for one or more problems, hyper-heuristics aim at generalizing well over a set of problems (Burke et al. 2003). Based on the classification presented by Burke et al. (2010a) hyper-heuristics can be selective or generative. Selection hyper-heuristics choose low-level heuristics to construct or improve a potential solution timetable while generation hyper-heuristics induce new low-level heuristics for a particular domain. Hyper-heuristics can also be categorized as being constructive or perturbative. Constructive hyper-heuristics either select or generate construction low-level heuristics to create a solution. Perturbative hyper-heuristics either choose or generate low-level heuristics to improve an initial candidate solution. A perturbation hyper-heuristic consists of two components one for heuristic selection and another for move acceptance. Thus, the four main categories of hyper-heuristics can be described as selection constructive, selection perturbative, generation constructive and generation perturbative.

There are three main areas of educational timetabling, namely, university examination timetabling, university course timetabling and school timetabling. All three types of timetabling involve the allocation of events to timetable periods while at the same time satisfying a set of hard constraints and minimizing a set of soft constraints (Qu et al. 2009c; McCollum et al. 2008; Pillay 2010a). These events are exams for exam timetabling, meetings between groups of students and lecturers in a specific venue for university course timetabling and meetings between classes and teachers for school timetabling. Hard constraints of the problem must be met in order to obtain an operable timetable. A timetable meeting the hard constraints is described as feasible. Examples of hard constraints include students not being scheduled to sit for two or more examinations during the same period; classes, teachers and venues not being scheduled more than once in the same period. Soft constraints define characteristics that we would like a timetable to possess, e.g. certain events to be scheduled at a particular time of the day, examinations with large numbers to be scheduled early in the timetable to facilitate marking. The number of soft constraints violated is minimized as these constraints are often contradictory and thus a soft constraint cost of zero is not attainable. There are two types of university course timetabling problems namely, curriculum-based and post enrolment. In the curriculum-based version student enrolment is not known at the time of timetabling construction while in the post enrolment version this is known (McCollum et al.2008).

The paper firstly provides an overview of hyper-heuristics to solve educational timetabling problems. Section 2 focuses on university examination timetabling, section 3 on university course timetabling and section 4 on school timetabling. Section 5 presents an analysis of the use of hyper-heuristics to solve educational timetabling problems and section 6 proposes future research directions.

2. Hyper-heuristics for University Examination Timetabling

A majority of the research into the use of hyper-heuristics for educational timetabling has been for the domain of examination timetabling. The hyper-heuristics employed to solve this problem have been either selection constructive or selection perturbative hyper-heuristics.

2.1 Selection Constructive Hyper-Heuristics

Burke et al. (2002) present a case-based hyper-heuristic to solve examination timetabling problems. The hyper-heuristic maintains a case base of previously solved problems and the low-level construction heuristic that was most appropriate to use at each stage of the timetable construction process. A timetable for a new problem is constructed by using the same low-level construction heuristic as that used in a previous case most similar to the current point of construction. A similarity measure was used for this purpose. Each problem in the case base is defined in terms of the problem characteristics and partial solutions, including the heuristic used at each stage of the timetable construction process. The low-level construction heuristics used include largest degree, largest degree using tournament selection, colour degree and saturation degree. The system was evaluated on generated timetabling problems. Tabu search was employed to determine the best list of problem characteristics for case comparisons. In later work (Burke et al. 2006) an additional low-level heuristic, namely, hill-climber which improves an initial solution created randomly using hill-climbing, was added to the heuristic set.

Yang and Petrovic (2004) present a hybrid approach combining a case-based hyper-heuristic and the great deluge algorithm to solve the examination timetabling problem. The great deluge algorithm improves a candidate solution timetable created using a low-level construction heuristic such as largest degree, largest enrollment, largest colour degree, largest weighted degree and saturation degree. Yang et al. implement a case-based hyper-heuristic to choose which construction heuristic to use to create the initial solution. The case base stores previously solved examination timetabling problems and the construction heuristic used. When solving a new examination timetabling problem the hyper-heuristic uses a fuzzy similarity measure to match the problem to problems in the case base and so identify which construction heuristic to apply to create an initial solution which is then improved by the great deluge algorithm. The case base was created using generated examination timetabling problems. The approach produced feasible good quality timetables for problems from the Carter benchmark set.

Burke et al. (2005) compare the performance of a tabu search and a hybrid hyper-heuristic in solving the examination timetabling problem. The former employs a tabu search to explore a space of combinations of the two low-level construction heuristics, namely, largest degree and saturation degree. The hybrid approach combines case-based reasoning and tabu search. Case-based reasoning is used to determine the percentage of largest degree and saturation degree in each combination. The characteristics of the problem being solved are compared to previous cases. The same hybridization of largest degree and saturation degree is used as that in the case that is the closest match. Tabu search was used to determine the most appropriate list of characteristics to use for comparison to previous cases. Both the hyper-heuristics were used to solve six generated examination timetabling problems and four problems from the Carter benchmark set. The tabu search hyper-heuristic outperformed the hybrid hyper-heuristic.

Burke et al. (2007) investigate the performance of the tabu search hyper-heuristic further by extending the set of low-level heuristics used to include largest colour degree, largest enrollment, largest weighted degree and random ordering. The revised tabu search hyper-heuristic was used to solve eleven of the Carter benchmark problems. In Qu et al. (2009b) the heuristic combinations performing well are studied to identify any patterns with the respect to the positions of the low-level heuristics in the combinations. This revealed that the best performing combination contained the saturation degree and largest weighted degree heuristics however the best percentage of each low-heuristic and the best position of these occurrences in the heuristic combination is problem dependent. Based on this an adaptive mechanism was built into the hyper-heuristic to hybridize the amount of saturation degree and largest weighted degree in a heuristic combination. The hyper-heuristic was used to solve eleven problems from the Carter benchmark set.

Qu and Burke (2005) investigate the use of a selection constructive hyper-heuristic to solve the examination timetabling problem. The hyper-heuristic employs variable neighbourhood search to explore a space of heuristic combinations consisting of two or more graph colouring heuristics, namely, color degree, largest degree, largest enrollment, largest weighted degree, saturation degree or random ordering. Each heuristic is applied in order to allocate an exam to a minimum penalty period. The hyper-heuristic was used to solve the Carter benchmark set of problems.

Pillay (2008, 2010b, 2012) implement an evolutionary algorithm hyper-heuristic to search a space of heuristic combinations of low-level construction heuristics chosen from a set containing the largest degree, largest weighted degree, largest enrollment, saturation degree and highest cost heuristics. The hyper-heuristic was able to produce good quality timetables for both the Carter set of benchmark problems and the benchmark set for the second international timetabling competition (McCollum et al. 2008). This research also examined the effect of the representation used for heuristic combinations on the performance of the evolutionary algorithm hyper-heuristic. Three representations, namely, fixed length, variable length and n-times and a combination of all three representations were tested. The latter option produced the best results.

The hyper-heuristic implemented by Burke et al. (2009b) employed the greedy random adaptive search procedure (GRASP) to hybridize the use of two low-level construction heuristics, namely, saturation degree and largest weighted degree, in choosing the next examination to schedule during the timetable construction process. An improvement phase is also conducted to improve the candidate solution constructed. Steepest descent is used for this purpose. The hyper-heuristic was used to solve problems in the Carter benchmark set.

Qu and Burke (2009a) compare the performance of various hyper-heuristics, each employing a different search to explore the heuristic space, to solve the examination timetabling problem. These hyper-heuristics search a space of heuristics combinations comprised of low-level construction heuristics. The combinations are constructed by selecting heuristics from a set containing the largest degree, largest weighted degree, largest colour degree, largest enrollment, saturation degree and random ordering heuristics. The hyper-heuristics were tested on eleven problems from the Carter benchmark set. The iterated local search hyper-heuristic was found to produce the best results. The performance of the hyper-heuristic was improved by searching the solution space, using iterative local search, at different intervals during timetable construction.

Saber et al. (2011) have used a selection constructive hyper-heuristic to solve this problem. In this study four low-level heuristics are combined to decide which examination to schedule next. The latter three heuristics in the combination are used to deal with ties. Roulette wheel selection is used to decide which period to allocate the examination to. The hyper-heuristic was tested on the benchmark set for the second international timetabling competition.

2.2 Selection perturbative hyper-heuristics

Kendall and Hussin (2004) use a tabu search hyper-heuristic to solve the examination timetabling problem for MARA University. The tabu search hyper-heuristic is used to improve an initial solution created using either the largest degree or saturation degree construction low-level heuristic. Two variations of the standard tabu search hyper-heuristic, namely, tabu search hyper-heuristics with hill-climbing and tabu seach hyper-heuristics with great deluge were also tested. Low-level heuristics include five move heuristics that reschedule examinations, two swap heuristics that swap the periods of two exams, a heuristic that unschedules an examination, and five construction heuristics (largest enrolment, largest degree, largest weighted degree, largest colour degree, and saturation degree) to reschedule unscheduled exams. The timetable produced by the hyper-heuristic was an improvement on the manually created timetable used by the university. In later work Kendall and Hussin (2005) applied the tabu search hyper-heuristic to eight problems from the Carter benchmark set.

Biligin et al. (2006) test seven approaches for heuristic selection and five for move acceptance. The heuristic selection methods include simple random, random descent, random permutation, random permutation descent, choice function, tabu search, and a greedy method. The three move acceptance approaches evaluated are accept all moves, accept improving moves only, great deluge and Monte Carlo. Low-level heuristics used in the study include three hill-climbing operators (next ascent hill-climbing, Davis' bit hill climber, random mutation hill climber) and three mutation operators (swap dimension, dimensional mutation and hypermutation). All six operators are applied to binary operands. The hyper-heuristic was used to solve the Carter benchmark set of problems and the examination timetabling for the Faculty of Architecture and Engineering at Yeditepe University. The hyper-heuristic combining the use of a choice function and Monte Carlo for move acceptance produced the best results.

In Ersoy et al. (2007) a hyper-heuristic is embedded in a memetic algorithm used to solve the examination timetabling problem. The hyper-heuristic is used to select one of three hill-climbers to be used by the memetic algorithm. The memetic algorithm using various hyper-heuristics was tested on six of the Carter benchmark problems. Self-adaptive hyper-heuristics using either a choice function for heuristic selection and great deluge for move acceptance or simple random combined with improving and equal move acceptance, were found to perform well.

Burke et al. (2008) study the use of simulated annealing selection perturbative hyper-heuristics. Simulated annealing is used for move acceptance. Three methods, namely, simple random, a greedy method and a choice function are evaluated for heuristic selection. Four low-level heuristics which reschedule examinations are used. Three of the heuristics attempt to reschedule exams so as to remove constraint violations. The last heuristic attempts to reschedule all the allocated exams. The hyper-heuristic using a choice function for heuristic selection with simulated annealing for move acceptance was found to outperform the other hyper-heuristic combinations.

Ozcan et al. (2009) also implement a perturbation hyper-heuristic to solve the examination timetabling problem. The move acceptance component employs a late acceptance strategy. Instead of comparing the current candidate solution to that obtained on the previous iteration, the move acceptance component compares it to a solution from *n* previous iterations. Heuristic selection methods tested include simple random, greedy, reinforcement learning, reinforcement learning with tabu search, and a choice function. Four low-level heuristics are implemented. The first is a mutation operator which attempts to reschedule all exams. The remaining three heuristics reschedule exams so as to reduce constraint violations. Tournament selection is used to select an exam and to select a slot to reschedule the examination in. The hyper-heuristic using simple random for heuristic selection and late acceptance strategy for move acceptance produced the best results.

Burke et al. (2010b) have implemented a Monte Carlo selection perturbative hyper-heuristic to solve the capacitated version of the Carter benchmark set (Qu et al. 2009) of examination timetabling problems. This set consists of data collected from thirteen different institutions. Methods tested for heuristic selection include simple random, a greedy method, a choice function and reinforcement learning. Similarly, different methods were made available for move acceptance, namely, simulated annealing, simulated annealing with reheating and an exponential

Monte Carlo method. Three low-level perturbative heuristics are used. The first reschedules an exam based on the number of conflicts the examination is involved in. The second reschedules exams so as to meet the capacity constraint while the third reschedules an examination randomly. The hyper-heuristic producing the best results for the benchmark set used the choice function for heuristic selection and simulated annealing for move acceptance.

Burke et al. (2010c) employ a hyper-heuristic to improve the quality of an initial feasible solution created using the largest degree construction heuristic. The hyper-heuristic uses four low-level perturbative heuristics, namely, move exam, swap exam, Kempe chain move and swap timeslot. All four heuristics aim at producing the least penalty timetable. Preliminary studies indicated that Kempe chain in combination with swap timeslot performed the best over problems of differing characteristics. The best hybridization (i.e. percentage occurrence and position) of these two heuristics in an optimal heuristic combination is problem dependent. An adaptive component is built into the hyper-heuristic to perform the hybridization of these two heuristics. The saturation degree is used to choose an examination, causing a soft constraint violation, which the move operator is applied to. The hyper-heuristic was used to find solutions to problems from the Carter benchmark set and the benchmark set for the second international timetabling competition.

Ozcan et al. (2012) have implemented a selection perturbative hyper-heuristic employing reinforcement learning for heuristic selection and great deluge for move acceptance. The hyper-heuristic was used to improve an initial solution. Three types of low-level heuristics were used. The first type aims at rescheduling the examination causing the most constraint violations in a set of *n* examinations. The second reschedules the examination that has the highest impact on the capacity violation for a particular period from a set of *n* periods with capacity violations. The last type of low-level heuristic attempts to reschedule all allocated examinations probabilistically. The hyper-heuristic was used to induce timetables for Yeditepe University and the Carter benchmark set.

In the study conducted by Sin and Kham (2012) reinforcement learning is used for heuristic selection and great deluge for move acceptance. Three variants of great deluge were tested, namely, flex deluge, non-linear great deluge, and extended great deluge. Low-level heuristics focused on changing timeslots of examinations or swapping subsets of examinations between two timeslots. The hyper-heuristic was used to improve an initial solution created using the largest enrollment construction heuristic. Evaluation on the Carter benchmark set revealed that the hyper-heuristic using the extended great deluge for move acceptance was the most effective.

2.3 Selection generative hyper-heuristics

Asmuni et al. (2005; 2007; 2009) combine low-level graph heuristics, namely, largest degree, saturation degree and largest enrollment, using a fuzzy logic function. The fuzzy function combines two to three heuristics and the single value produced is used to sort examinations to be scheduled according to difficulty. The hyper-heuristic was used to solve the Carter benchmark set of problems.

Pillay and Banzhaf (2009b) proposed that low-level heuristics be combined hierarchically allowing them to be applied simultaneously instead of combining them linearly and applying them sequentially. The use of conditional and logically operators have facilitated the hierarchical combination and simultaneous application of low-level construction heuristics chosen from largest degree, largest weighted degree, largest enrollment, saturation degree and highest cost heuristics. Four such combinations were created and tested on the Carter benchmark set of problems. These combinations produced results competitive to other hyper-heuristics tested on the same benchmark set of problems. Pillay (2009a) automates the process of creating the hierarchical heuristic combinations. In this study genetic programming is used to evolve these combinations comprised of conditional and logical operators and the low-level heuristics.

Pais and Burke (2010) use a Choquet integral to combine five low-level construction heuristics, namely, largest degree, colour degree, largest weighted degree, largest enrollment and saturation degree. The single value produced by the Choquet integral estimates the difficulty of scheduling an examination. The examinations are sorted in decreasing order according to this value and allocated accordingly. The performance of the Choquet integral is compared to that of each of the low-level heuristics applied individually to sort the examinations. The low-level heuristics and the Choquet integral were evaluated on the Carter benchmark problems and the benchmark set for the second international timetabling competition. The Choquet integral produced the best results for eleven of the thirteen Carter problems and for five of the eight timetabling competition problems. Burke and Pais (2011) extend this work and evaluate differential evolution to induce fuzzy measures to estimate examination difficulty. This improved the performance of the hyper-heuristic.

2.4 Summary of Hyper-Heuristic Performance

This section summarizes the performance of the different types of hyper-heuristics in solving the examination timetabling problem. There are essentially two benchmark problem sets that these hyper-heuristics have been applied to, namely, the Carter (also known as the Toronto) benchmark set (Qu et al. 2009c) and the benchmark set used for the second international timetabling competition ITC' 2007 (McCollum et al. 2008). A majority of the hyper-heuristics have been evaluated on the Carter benchmark set. The characteristics of the problems included in this benchmark set are listed in Table 1 in Appendix A. This benchmark set has been constructed by collecting data from real-world educational institutions. The density of the clash matrix is a ratio of the number students involved in clashes to the total number of students and is a measure of the difficulty of the problem. The hard constraint for this set of problems is that there must be no clashes, i.e. a student must not be scheduled to write more than one examination at a time. The soft constraint is that the examinations must well-distributed over the examination period for any one student. A distance formula is used to calculate the soft constraint cost (Qu et al. 2009c). Performance of selection constructive, selection perturbative and generation constructive hyper-heuristics applied to the Carter benchmarks are tabulated in Appendix B, Appendix C and Appendix D respectively. Note that only those studies that have reported these results are included.

From the results presented in Appendix B the hybrid approach combining cased-based reasoning with the great deluge algorithm appears to have performed the best. The evolutionary algorithm hyper-heuristic, using a combination of three different representations for individuals, has also produced fairly good results. Selection perturbative hyper-heuristics have only been applied to subsets of Carter benchmark problems and need to be evaluated further. The best performing selection perturbative hyper-heuristic is the adaptive selection perturbative hyper-heuristic implemented by Burke et al. (2012c). However, the results produced by this hyper-heuristics is not as good as that of the selection constructive hyper-heuristics listed in Appendix B. Similarly, the generation constructive hyper-heuristics presented in Appendix D do not perform as well as the selection constructive hyper-heuristics on the Carter benchmark set.

The comparison in this section has been restricted to examination timetabling as there has not been sufficient research into university course timetabling and school timetabling to do the same.

3. Hyper-Heuristics for University Course Timetabling

The use of hyper-heuristics to solve the university course timetabling problem is not as well researched as for the university examination timetabling problem. Most of the research in this area has focused on the use of selection constructive hyper-heuristics to find solutions to this problem.

3.1 Selection constructive hyper-heuristics

Rossi-Doria and Paechter (2003) implement an evolutionary algorithm selection constructive hyper-heuristic. Each chromosome is a comprised of two rows of integers representing heuristics. The first row represents heuristics to choose which event to schedule next and are chosen from largest degree, largest colour degree, least saturation degree, maximum weighted number of event correlations, maximum number of students, maximum number of features by events, minimum number of possible rooms, event with room suitable for most events, least saturation degree with room consideration. The second row represents heuristics used to select room and timeslots, e.g. smallest possible room, least room suitable, least used room, latest or earliest timeslot in the day. The evolutionary algorithm is steady-state and uses binary tournament selection. One point crossover and mutation is used to produce offspring. The hyper-heuristic was tested on five generated problems of medium difficulty and produced competitive results for two of these problems.

A case-based hyper-heuristic is proposed in Burke et al. (2006) to solve the university course timetabling problem. The case base stores previously solved problems in terms of problem features and steps of the construction process and the low-level construction heuristic used to schedule each event. Each new problem is solved by finding a match to stored cases at each stage of the timetable construction process. The hyper-heuristic was used to solve generated university course timetabling problems.

Burke et al. (2007) implement a tabu search to explore a space of heuristic combinations of low-level construction heuristics. These heuristics include, random ordering, largest degree, saturation degree, largest colour degree, largest enrollment and largest weighted degree. The hyper-heuristic was used to solve eleven benchmark course timetabling problems (Socha et al. 2002).

Qu et al. (2009a) evaluate various search methods for use by a selection constructive hyper-heuristic. The hyper-heuristic, using different search techniques, was tested on eleven benchmarks problem made available by Socha et al. (2002). The hyper-heuristic employing variable neighbour search to explore the space of heuristic combinations comprised of low-level construction heuristics produced the best results for the benchmark set. The low-level construction heuristics used include largest degree, largest weighted degree, largest colour degree, largest enrollment, saturation and random ordering heuristics. A variation of the hyper-heuristic employing iterative local search to explore the solution space at various stages during the timetable construction process was found to improve the performance of the hyper-heuristic.

3.2 Selection perturbative hyper-heuristics

The selection perturbative hyper-heuristic implemented by Bai et al. (2007a, 2007b) uses simulated annealing for move acceptance. Heuristic selection is initially random until a heuristic performance history has been developed and is then based on the performance of the low-level heuristics in the previous iterations. Three low-level perturbative heuristics are available for use by the hyper-heuristic. The first reschedules a randomly selected event. The second swaps the periods of two randomly chosen events. The third swaps the events of two randomly selected periods. The hyper-heuristic is used to improve an initial feasible solution. The hyper-heuristic was tested on two benchmark problem sets. The first set contained five small, five medium and one large problem and the second twenty problems. The hyper-heuristic performed better than two other hyper-heuristics and meta-heuristics applied to the problems in the first benchmark set and to one of the problems in the second benchmark set.

3.3 Generation Perturbative Hyper-Heuristic

In the study conducted by Rattadilok (2010) an initial solution is created using a random or greedy approach and is improved using a generation perturbative hyper-heuristic. A choice function is used for heuristic selection. This function selects low-level heuristics based on their previous performance. Swap operators are used as low-level heuristics. Each operator is created by making configuration decisions, namely, a number of candidates involved in the swap, swap candidate sets and acceptance criteria for termination. Sub-controllers are used to formulate configuration decisions. Ten low-level swap heuristics are used. The hyper-heuristic was applied to data sets from the first international university course timetabling competition.

4. School Timetabling

There has not been much research conducted into the use of hyper-heuristics for solving the school timetabling problem. There have basically been two studies, one investigating the use of a selection constructive hyper-heuristic and the second evaluating a generation constructive hyper-heuristic in solving the school timetabling problem.

4.1 Selection constructive hyper-heuristics

Pillay (2010c) implements an evolutionary algorithm hyper-heuristic to solve the school timetabling problem. The evolutionary algorithm explores a space of heuristic combinations of low-level construction heuristics. Construction heuristics used include random ordering, largest degree, saturation degree, class degree, teacher degree and class-teacher degree. The hyper-heuristic was tested with different subsets of low-level heuristics from which the elements of each heuristic combination are chosen. The subset consisting of largest degree and saturation degree produced the best results. The incorporation of hill-climbing in the genetic operators was found to improve the performance of the EA hyper-heuristic. The EA hyper-heuristic produced competitive results in solving a difficult generated problem and outperformed a neural work and greedy search applied to the same problem. Pillay (2011a) applied this EA hyper-heuristic to solving the school timetabling problem for a South African primary school. In this study the low-level construction heuristic set included largest degree, saturation degree, double degree and period preference degree. A Pareto function of the hard and soft constraint costs was found to be the most effective option to evaluate the fitness of each heuristic combination. The EA hyper-heuristic produced a solution of better quality than that currently being used by the school.

4.2. Generation constructive hyper-heuristics

Pillay (2011b) employed genetic programming to evolve heuristics for the school timetabling problem. The function set was composed of arithmetic operators, arithmetic logical operators and conditional operators. The terminal set contains variables to represent the characteristics of the problem, namely, the number of class-teacher meetings a class is involved in and the number of class-teacher meetings a teacher is involved in as well as the heuristics that are traditionally used in solving the school timetabling problem, namely, largest degree and saturation degree. The hyper-heuristic was tested on a difficult generated problem and performed better than the saturation degree and largest degree applied to the same problem. The generation hyper-heuristic also performed better than a tabu search, the evolutionary algorithm hyper-heuristic described in section 4.1, a Hopfield neural network and greedy search in solving the same problem.

5. Discussion and Future Research Directions

As is evident from the above discussion most of the research into using hyper-heuristics for solving educational timetabling problems has been on examination timetabling and on selection constructive hyper-heuristics for this domain. Furthermore selection constructive hyper-heuristics appear to perform the best for examination timetabling. However, this may not be a fair comparison at this stage as the other types of hyper-heuristics have not been as well researched. Hyper-heuristic research for the three different types of educational timetabling have been conducted in isolation of each other. Future research should aim at investigating the effectiveness of the different types of hyper-heuristics for educational timetabling as a whole. The hyper-heuristics also need to be more widely tested. The two benchmark sets available for examination timetabling are the Carter benchmark set (Qu et al. 2009c) and the benchmark set of problems used for the examination timetabling track of the second international timetabling competition (McCollum et al. 2008). There are three benchmark sets available for the university course timetabling problem, namely, that provided by Socha et al. (2002), the problem set used for the first international timetabling competition (Paechter et al. 2003) and the data sets for curriculum-based and post enrolment university course timetabling tracks of the second international timetabling competition. The third international timetabling competition (Post 2011) is focused on school timetabling and various real-world school timetabling data sets have been made available for the competition.

A fair amount of research has been conducted into using selection constructive hyper-heuristics to solve the different educational timetabling problems. In a majority of these studies a metaheuristic has been employed to explore the space of heuristic combinations. These heuristic combinations are comprised of low-level construction heuristics. The low-level heuristics that have been used for this purpose have generally been the graph colouring heuristics, namely, largest degree, largest weighted degree, largest colour degree, largest enrollment and saturation degree. Some studies have introduced other low-level heuristics, namely, highest cost which estimates the soft constraint cost (Pillay and Banzhaf 2009b) and period and room heuristics introduced by Rossi-Doria and Paechter (2003). Metaheuristics used to search such a heuristic space include tabu search, iterated local search, variable neighbourhood search and evolutionary algorithms. Cased-based reasoning has also been used for low-level heuristic selection. The best performing hyper-heuristic was a hybrid combining case-based reasoning and greatest deluge. Processes that automate the hybridization of low-level heuristics that perform well have also been studied for selection constructive hyper-heuristics. The effectiveness of searching both the solution space and heuristic space during the construction of a timetable has also been illustrated. This needs to be investigated further for educational timetabling in general.

The use of selection perturbative hyper-heuristics for solving educational timetabling problems, have also been researched. Low-level heuristics commonly used by these hyper-heuristics include hill-climbing operators, mutation operators, rescheduling events with high constraint violation costs, swapping events or subsets of events, swapping timetable periods, unscheduling and rescheduling events. Techniques commonly used for heuristic selection include simple random, greedy, a choice function, reinforcement learning and tabu search. Methods evaluated for move acceptance are the late acceptance strategy, simulated annealing, Monte Carlo and the great deluge algorithm. Selection perturbative hyper-heuristic heuristic selection and move acceptance pairs that have performed well for different timetabling problems include a choice function with either simulated annealing, Monte Carlo or the great deluge algorithm and simple random with the late acceptance strategy. Further research into the effectiveness of different methods for driving this category of hyper-heuristics as well as a more expansive evaluation of these hyper-heuristics needs to be conducted.

There has not been much research into the use of generation hyper-heuristics in solving educational timetabling problems. Fuzzy logic and genetic programming are the most popular methods used to induce constructive low-level heuristics. There has only been one study into generation perturbative hyper-heuristics for solving timetabling problems, namely, that conducted by Rattadilok (Rattadilok 2010) to configure swap operators. The generation of low-level construction heuristics for timetabling needs to researched further. Generally, graph colouring heuristic have been used as low-level heuristics for timetable construction. The induction of heuristics based on problem characteristics need to be studied. Given its success in other domains (Burke et al. 2009a), genetic programming can be investigated for this purpose.

Most selection constructive hyper-heuristics have focused on constructive heuristics for selecting which event to schedule next. There is a need for investigations into developing selection and generation hyper-heuristics that cater for construction heuristics for choosing timetable periods and rooms in addition to heuristics for event selection.

An area that has not been investigated is that of hybrid hyper-heuristics that combine different types of hyper-heuristics, e.g. combining selection constructive and perturbative hyper-heuristics. Furthermore, should such combinations be sequential, i.e. apply one type of hyper-heuristic followed by another, or should there be an alternating application of the different hyper-heuristics.

7. References

Asmuni, H., Burke, E.K. & Garibaldi, J.M. (2005) Fuzzy Multiple Ordering Criteria for Examination Timetabling. In: Burke, E.K. and Trick, M. (Eds.), selected papers from the *5th International Conference on the Theory and Practice of Automated Timetabling (PATAT*

2004) - The Theory and Practice of Automated Timetabling V, *Lecture Notes in Computer Science*, 3616, 147–160.

Asmuni, H, Burke, E.K., Garibaldi, J.M. & McCollum, B. (2007) Determining Rules in Fuzzy Multiple Heuristic Orderings for Constructing Examination Timetables. In: Bapiste, P., Munier, A. Kendall, G. & Sourd, F. (Eds.), *proceedings of the 3rd Multidisciplinary International Scheduling: Theory and Applications Conference, MISTA 2007* (pp. 59-66).

Asmuni, H., Burke, E.K., Garibaldi, J.M., McCollum, B. & Parkes, A.J. (2009). An Investigation of Fuzzy Multiple Heuristic Orderings in the Construction of University Examination Timetables. *Computers and Operations Research*, 36(4), 981-1001.

Bai, R., Blazewicz, J., Burke, E.K., Kendall, G. & McCollum, B. (2007a) *A Simulated Annealing Hyper-Heuristic Methodology for Flexible Decision Support* (Technical Report No. NOTTCS-TR-2007-8). School of Computer Science and Information Technology, University of Nottingham, Nottingham.

Bai, R., Burke, E.K., Gendreau, M., Kendall, G. & McCollum, B. (2007b) Memory Length Hyper-Heuristics: An Empirical Study. In *proceedings of the 2007 IEEE Symposium on Computational Intelligence in Scheduling, CI-Sched 2007* (pp. 173-178).

Bilgin, B., Ozcan, E. & Korkmaz, E.E. (2006) An Experimental Study on Hyper-Heuristics and Exam Timetabling. In Burke, E.K. & Rudova, H. *proceedings of the international conference on the Practice and Theory of Automated Timetabling, PATAT 2006* (pp. 123-140).

Burke, E.K. , Dror, M. , Petrovic, S. & Qu, R. (2005) Hybrid Graph Heuristics with a Hyper-Heuristic Approach to Exam Timetabling Problems. In: Golden, B., Raghavan, S. & Wasil, E.A. (Eds.), the *Next Wave in Computing, Optimization, and Decision Technologies – Conference Volume of the 9th Informs Computing Society Conference* (pp. 79 -91).

Burke, E., Hart, E., Kendall, G., Newall, J., Ross, P. & Schulenburg, S. (2003) Hyper-Heuristics: An Emerging Direction in Modern Research. In the *Handbook of Metaheuristics*, Chapter 16, 457– 474.

Burke, E.K., Hyde, M., Kendall, G., Ochoa, G., Ozcan, E. &Woodard, J. (2009a) Exploring Hyper-Heuristic Methodologies with Genetic Programming. *Computational Intelligence*, 6, 177-201.

Burke, E.K., Hyde, M., Kendall, G., Ochoa, G., Ozcan, E. &Woodard, J. (2010a) A Classification of Hyper-Heuristic Approaches. In the *Handbook of Metaheuristics*, International Series in Operations Research and Management Science, Volume 146, 449-468.

Burke, E.K., Kendall, G., Misir, M. & Ozcan, E. (2008) A Study of Simulated Annealing Hyper-Heuristics. In the *proceedings of the international conference on the Practice and Theory of Automated Timetabling (PATAT 2008)*, http://www.asap.cs.nott.ac.uk/patat/patat08/Papers/Ozcan-HD3a.pdf . Accessed 12 February 2012.

Burke, E.K., McCollum, B., Meisels, A., Petrovic, S. & Qu, R. (2007) A Graph-Based Hyper-Heuristic for Educational Timetabling Problems. *European Journal of Operational Research*, 176, 177-192.

Burke, E. K., Kendall, G., Misir, M. & Ozcan, E. (2010b) Monte Carlo Hyper-Heuristics for Examination Timetabling. *Annals of Operations Research*, doi 10.1007/s10479-010-0782-2.

Burke, E.K., McCollum, B., Meisels, A., Petrovic, S. & Qu, R. (2007). A Graph-Based Hyper-Heuristic for Educational Timetabling Problems. *European Journal of Operational Research*, 176, 177 – 192.

Burke, E. K. & Pais, T. C. (2011) Using Differential Evolution to Identify Fuzzy Measures for the Exam Timetabling Problem. In proceedings of the Multidisciplinary International Conference on Scheduling: Theory and Applications, MISTA 2011 (pp. 335-351).

Burke, E.K., Petrovic, S. & Qu, R. (2002) Case Based Heuristic Selection for Examination Timetabling. In *proceedings of SEAL '02* (pp. 277-281).

Burke, E.K., Petrovic, S. & Qu, R. (2006) Cased-Based Heuristic Selection for Timetabling Problems. *Journal of Scheduling*, 9(2), 115-132.

Burke, E.K., Qu, R. & Soghier, A. (2009b) Adaptive Selection of Heuristics within a GRASP for Exam Timetabling. In *proceedings of the Multidisciplinary Conference on Scheduling: Theory and Application, MISTA 2009* (pp. 409-423).

Burke, E.K., Qu, R. & Soghier, A. (2010c) Adaptive Selection of Heuristics for Improving Constructed Exam Timetables. In proceedings of the 8th International Conference on the Practice and Theory of Automated Timetabling, PATAT 2010 (pp. 136 -151).

Ersoy, E., Ozcan, E. & Uyar, S. (2007). Memetic algorithms and hillclimbers.In: Baptiste, P., Kendall, G., Kordon, A.M. & Sourd, F. (Eds.), *proceedings of the 3rd Multidisciplinary International Conference on Scheduling: Theory and Applications Conference. MISTA 2007* (pp. 159–166).

Kendall, G. & Hussin, M.H. (2004) Tabu Search Hyper-Heuristic Approach to the Examination Timetabling Problem at University Technology MARA. In the *proceedings of the international conference on the Practice and Theory of Automated Timetabling, PATAT 2004* (pp. 270-295).

Kendall, G. & Hussin, N.M. (2005). An investigation of a tabu search based on hyper-heuristics for examination timetabling. In: Kendall G., Burke E.K. & Petrovic S. (Eds.*) proceedings of the 2nd Multidisciplinary Scheduling: Theory and Applications Conference, MISTA 2005* (pp. 309–328).

McCollum, B., McMullan, P., Paechter, B., Lewis, R., Schaerf, A., DiGapsero, L., Parkes, A.J., Qu, R . & Burke, E.K. (2008). Setting the research agenda in automated timetabling: The second international timetabling competition. *INFORMS Journal of Computing*, 22(1), 120–130.

Ozcan, E., Bykov, Y., Birben, M. & Burke, E.K. (2009) Examination Timetabling Using Late Acceptance Hyper-Heuristics. In *proceedings of the IEEE Congress on Evolutionary Computing, CEC '09* (pp. 997-1004).

Ozcan, E., Misir, M., Ochoa, G. & Burke, E.K. (2012) A Reinforcement Learning – Great-Deluge Hyper-Heuristic for Examination Timetabling. *Modeling, Analysis, and Applications in Metaheuristic Computing*, 34-55.

Pais, T.C. & Burke, E. K. (2010) Choquet Integral for Combining Heuristic Values for Exam Timetabling Problem. In *proceedings of the 8th International Conference on the Practice and Theory of Automated Timetabling, PATAT 2010* (pp. 305 -320).

Paechter, B., Gambardella, L. M., Rossi-Doria, O. (2003) International Timetabling Competition, http://www.idsia.ch/Files/ttcomp2002/oldindex.html. Accessed 1 July 2012.

Pillay, N. (2008) An Analysis of Representations for Hyper-Heuristics for the Uncapacitated Examination Timetabling Problem in a Genetic Programming System. In Cilliers, C., Barnard, L. & Botha R. (Eds.), *proceedings of SAICSIT 2008* (pp. 188-192).

Pillay, N. (2010a) An Overview of School Timetabling Research. In *proceedings of the 8th International Conference on the Practice and Theory of Automated Timetabling, PATAT '10* (pp. 321-335).

Pillay, N. (2010b) Evolving Hyper-Heuristics for a Highly Constrained Examination Timetabling Problem. In *proceedings of the 8th international conference on the Practice and Theory of Automated Timetabling, PATAT 2010* (pp. 336-346).

Pillay, N. (2010c) A Study into the Use of Hyper-Heuristics to Solve the School Timetabling Problem. In *proceedings of SAICSIT 2010* (pp. 258-264).

Pillay, N. (2011a) A Hyper-Heuristic Approach to Solving School Timetabling Problems. In *proceedings of the Multidisciplinary International Conference on Scheduling: Theory and Applications, MISTA 2011* (pp.628-632).

Pillay, N. (2011b) Evolving Heuristics for the School Timetabling Problem. *In proceedings of the 2011 IEEE Conference on Intelligent Computing and Intelligent Systems (ICIS011)*, Vol. 3 (pp. 281-286).

Pillay, N. (2012) Evolving Hyper-Heuristics for the Uncapacitated Examination Timetabling Problem. *Journal of the Operational Research Society*, 63, 47-58.

Pillay, N. (2009a) Evolving Hyper-Heuristics for the Uncapacitated Examination Timetabling Problem. In proceedings of the Multidisciplinary International Conference on Scheduling: Theory and Applications, MISTA 2009 (pp. 409-422).

Pillay, N. & Banzhaf, W. (2009b) A Study of Heuristic Combinations for Hyper-Heuristic Systems for the Uncapacitated Examination Timetabling Problem. *European Journal of Operational Research*, 197, 482-491.

Post, G. (2011) Third International Timetabling Competition (ITC 2011), http://www.utwente.nl/ctit/itc2011/. Accessed 1 July 2012.

Qu, R. & Burke, E.K. (2005). Hybrid Variable Neighbourhood HyperHeuristics for Exam Timetabling Problems. In: *Proceedings of the MIC2005: The Sixth Metaheuristics International Conference*, Vienna, Austria. http://www.cs.nott.ac.uk/~rxq/files/MIC05.pdf, accessed 28 June 2008.

Qu, R. and Burke, E.K. (2009a). Hybridisations within a graph based hyper-heuristic framework for university timetabling problems. *Journal of the Operational Research Society*, 60, 1273–1285.

Qu, R., Burke, E.K. & McCollum, B. (2009b). Adaptive automated construction of hybrid heuristics for exam timetabling and graph colouring problems. *European Journal Operational Research*, 198(2), 392–404.

Qu, R., Burke, E.K., McCollum, B., Merlot, L.T.G. & Lee, S.Y. (2009c) A survey of search methodologies and automated system development for examination timetabling. *Journal of Scheduling*, 12(1), 55–89.

Rattadilok, P. (2010) An Investigation and Extension of a Hyper-Heuristic Framework. *Informatica*, 34, 523-534.

Rossi-Doria, O. & Paechter, B. (2003) A Hyperheuristic Approach to Course Timetabling Problems Using an Evolutionary Algorithm, http://www.metaheuristics.net/media/documents/hyperEA.pdf. Accessed 12 February 2012.

Saber, N.R., Ayob, M., Qu, R. & Kendall, G. (2011) A Graph Colouring Constructive Hyper-Heuristic for Examination Timetabling Problems. *Applied Intelligence*, doi: 10.1007/s10489-011-0309-9.

Sin, E.S. & Kham, N.S.M. (2012) Hyper Heuristic Based on Great Deluge and its Variants for Exam Timetabling Problem. Cornell University Library, http://arxiv.org/abs/1202.1891. Accessed 12 February 2012.

Socha, K., Knowles, J. & Sampels, M. (2002) A Max-Min Ant System for the University Course Timetabling Problem. In *proceedings of the 3rd International Workshop on Ant Algorithms. Lecture Notes in Computer Science*, 2463, 1-13.

Yang, Y. & Petrovic, S. (2004) A Novel Similarity Measure for Heuristic Selection in Examination Timetabling. In the *proceedings of the international conference on the Practice and Theory of Automated Timetabling,PATAT 2004* (pp. 247-269).

Appendix A – Carter Benchmark Data Set

Table 1. Characteristics of problems in the Carter benchmark set

Data	Institution	Periods	No. of Exams	No. of Students	No. Enrolments	Density of Conflict Matrix
car-f-92 I	Carleton University, Ottawa	32	543	18419	55522	0.14
car-s-91 I	Carleton University, Ottawa	35	682	16925	56877	0.13
ear-f-83 I	Earl Haig Collegiate Institute, Toronto	24	190	1125	8109	0.27
hec-s-92 I	Ecole des Hautes Etudes Commerciales, Montreal	18	81	2823	10632	0.42
kfu-s-93	King Fahd University of Petroleum and Minerals, Dharan	20	461	5349	25113	0.06
lse-f-91	London School of Economics	18	381	2726	10918	0.06
pur-s-93 I	Purdue University, Indiana	43	2419	30029	120681	0.03
rye-s-93	Ryerson University, Toronto	23	486	11483	45051	0.08
sta-f-83 I	St Andrew's Junior High School, Toronto	13	139	611	5751	0.14

tre-s-92	Trent University, Peterborough, Ontario	23	261	4360	14901	0.18
uta-s-92 I	Faculty of Arts and Sciences, University of Toronto	35	622	21266	58979	0.13
ute-s-92	Faculty of Engineering, University of Toronto	10	184	2749	11793	0.08
yor-f-83 I	York Mills Collegiate Institute, Toronto	21	181	941	6034	0.29

Appendix B – Performance of Selection Constructive Hyper-Heuristics

Table 2. Comparison of selection constructive hyper-heuristic performance

Data	**Yang & Petrovic (2004)**	**Burke et al. (2007)**	**Burke et al. (2009)**	**Qu & Burke (2005)**	**Pillay (2012)**	**Qu & Burke (2009a)**	**Qu & Burke (2009b)**	**Sabar et al. (2011)**
car-f-92 I	3.93	4.84	4.74	4.7	4.22	4.77	4.32	4.70
car-s-91 I	4.50	5.41	5.48	5.4	4.95	5.3	5.11	5.14
ear-f-83 I	33.71	38.19	37.71	37.29	35.95	38.39	35.56	37.86
hec-s-92 I	10.83	12.72	12.41	12.23	11.27	12.72	11.62	11.90
kfu-s-93	13.82	15.76	16.84	15.11	14.12	15.09	15.18	15.30

lse-f-91	10.35	13.15	12.29	12.27	10.76	12.72	11.32	12.33
pur-s-93 I	-	-	-	-	-	-	-	5.37
rye-s-93	8.53	-	-	-	9.23	-	-	10.71
sta-f-83 I	151.52	141.08	163.63	159.1	157.69	159.2	158.88	160.12
tre-s-92	7.92	8.85	9	8.67	8.43	8.74	8.52	8.32
uta-s-92 I	3.14	3.54	3.62	3.56	3.33	3.32	3.21	3.88
ute-s-92	25.39	32.01	30.01	30.23	26.95	30.32	28	32.67
yor-f-83 I	36.53	40.13	42.54	43	39.63	40.24	40.71	40.53

Appendix C – Performance of Selection Perturbative Hyper-Heuristics

Table 3. Comparison of selection perturbative hyper-heuristic performance

Data	**Kendall & Hussin (2005)**	**Ersoy et al. (2007)**	**Burke et al. (2010c)**
car-f-92 I	4.67	-	4.31
car-s-91 I	5.37	-	5.19
ear-f-83 I	40.18	-	35.79
hec-s-92 I	11.86	11.6	11.19
kfu-s-93	15.84	15.8	14.51

lse-f-91	-	13.2	10.92
pur-s-93 I	-	-	-
rye-s-93	-	-	-
sta-f-83 I	157.38	157.7	157.18
tre-s-92	8.39	-	8.49
uta-s-92 I	-	-	3.44
ute-s-92	27.6	26.3	26.7
yor-f-83 I	-	40.7	39.47

Appendix D – Performance of Generation Constructive Hyper-Heuristics

Table 4. Comparison of generation constructive hyper-heuristic performance

Data	**Asumni et al. (2005)**	**Asumni et al. (2007)**	**Asumni et al. (2009)**	**Pillay & Banzhaf (2009b)**
car-f-92 I	4.56	4.51	4.54	4.28
car-s-91 I	5.29	5.19	5.29	4.97
ear-f-83 I	37.02	36.16	37.02	36.86
hec-s-92 I	11.78	11.61	11.78	11.85
kfu-s-93	15.81	15.34	15.81	14.62

lse-f-91	12.09	11.35	12.09	11.14
pur-s-93 I	-	-	-	4.73
rye-s-93	10.35	10.02	10.38	9.65
sta-f-83 I	160.75	159.09	160.75	158.33
tre-s-92	8.67	8.62	8.67	8.48
uta-s-92 I	3.57	3.52	3.57	3.4
ute-s-92	27.78	27.64	28.07	28.88
yor-f-83 I	40.66	39.25	39.80	40.74