An Overview of School Timetabling Research

Nelishia Pillay

School of Computer Science, University of KwaZulu-Natal

+27 2605644

+27 2605648

pillayn32@ukzn.ac.za

Abstract. Although there has been a fair amount of research in the area of school timetabling, this domain has not grown as well as other fields of educational timetabling such as university course and examination timetable. This can possibly be attributed to the fact that the studies in this domain have generally been conducted in isolation of each other and have addressed different school timetabling problems. Furthermore, there have been no comparative studies on the success of different methodologies on a variety of school timetabling problems. As a way forward this paper provides an overview of the research conducted in this domain, details of problems sets which are publically available and proposes areas for further research in school timetabling.

Keywords: school timetabling, educational timetabling

1. Introduction

Educational timetabling encompasses university course timetabling, examination timetabling and school timetabling. A lot of progress has been made in university course and examination timetabling research. This can be attributed to the variety of problems publicly available which has enabled a comparative study of different methodologies for these domains. Research in school timetabling has not advanced as rapidly as in the other two areas of educational timetabling (Kingston 2006). This is possibly due to studies being done in isolation of each other for specific schools (Santos et al. 2008) and the lack of a variety of problems which can be accessed publicly (Schaerf 1991; Smith et al. 2003; Jacobsen et al. 2006; Post et al. 2008). According to Nurmi et al. (2008) and Post et al. (2008) the school timetabling problem has not been studied as extensively as the university course and examination timetabling problems. Furthermore, methods implemented have not been widely tested on a variety of school timetabling problems (Jacobsen et al. 2006; Santos et al. 2005) which is essential in order to ascertain how well the methodology can generalize. Due to the unavailability of a general set of benchmarks, there have not been much comparative analyses of different methodologies in solving the school timetabling problem. In cases where such studies have been performed, different methods are compared for a single school and not on a variety of problems.

This paper serves as a starting point for further development in the field. The definition of the school timetabling problem and the hard and soft constraints associated with each problem instance differs from one study to the next. The following section attempts to provide a standardized definition of the problem. Section 3 presents an overview of research in the field.

Problems sets that are publicly available for research are described in section 4. Section 5 proposes a way forward by proposing future directions of research in school timetabling.

2. The School Timetabling Problem (STP)

The school timetabling problem and the terminology used in defining the problem differs drastically from one study to the next. For consistency the following terms are defined.
A *class* refers to a group of students that will be taught a particular subject, e.g. Mathematics. A *lesson* refers to a particular subject being taught to a class by a teacher. The teaching period refers to the duration of the timetable and is usually a week.

Carter et al. (Carter et al. 1997) define the school timetabling problem as a subtype of course timetabling. Solving the school timetabling problem essentially involves allocating class, teacher and room tuples to timetable slots so as to satisfy the hard and soft constraints of the problem (Abramson et al. 1991; Beligiannis et al. 2008; Post et al. 2008). Students are usually grouped into classes prior to the timetable construction process (Abramson 1991). The school timetabling problem differs for each country due to the characteristics and regulations specific to the particular education system (Alvarez-Valdes et al. 1996; Post et al. 2008). In some cases the room does not form part of the tuple and each class-teacher pair has to be allocated to a period and a venue (Wilke et al. 2008; Post et al. 2008). The number and duration of lessons can also differ for grades or class levels in a school (Post et al. 2008). Furthermore, some schools extend over more than one site (Schaerf 1991).

The hard and soft constraints also differ drastically from one problem to the next. Hard constraints are constraints that must be met by a timetable in order for it to be operable. A timetable meeting all the hard constraints of a problem is called a feasible timetable. Soft constraints define the quality of a timetable. Soft constraints are usually contradictory, and as such it is impossible to satisfy all the soft constraints. Thus, we attempt to minimize the soft constraint cost.

The STP can be defined in terms of the requirements of the problem, the hard constraints and soft constraints. Section 2.1 presents a model for specifying the requirements of the STP. Sections 2.2 and 2.3 provide a comprehensive set of hard and soft constraints, respectively, for the school timetabling problem that should be catered for in a standardized definition of the problem. Some constraints are treated as hard constraints in one STP and soft constraints in another. These are described in section 2.4. Objective functions generally used to evaluate school timetables are discussed in section 2.5.

2.1 Requirements of the Problem

The requirements of the STP differ from problem to problem. This section attempts to describe the requirements as generally as possible so as to accommodate most variations of the STP. The requirements of the STP can be defined in terms of the following:

- General statistics – Number of courses, number of teachers, number of periods or number days and number of periods per day.
- Available periods – The number and duration of available periods. An indication of which session the period falls in, e.g. morning or afternoon or the time of each period.
- Lesson requirements – These requirements specify the number of times a teacher has to meet a class over the teaching period. In most problem instances the requirements are specified in terms of class, teacher, room tuples which must be assigned to periods.
- Room requirements - A variation of this problem does not include rooms in the tuple, and room allocation forms part of the timetable construction process.

  In this case class sizes must be provided. In addition to this a list of rooms and their corresponding capacities must be specified. Certain lessons may require specialized rooms such as laboratories or a gymnasium. These need to be indicated. In some versions of the problem classes have fixed rooms and teachers move from one class to another. In this case classes will only move to specialized rooms or in the case of splits or mergers for lessons.
- Teacher requirements - Each teacher's workload is defined in terms of the maximum and minimum number of periods or hours the teacher teaches for the week. Teachers may also be given a set number of free periods over the teaching period. These must be specified. It a teacher is required to commute between two different school sites in a day this must be indicated.
- Teacher unavailabilities – Teachers are usually unavailable for certain periods. This may be due to other administrative duties. Alternatively, teachers may be allocated certain free periods or days off and thus may not be available. Some schools employ teachers on a part-time basis. In this case teachers will teach at different schools on different days of the week and thus may not be available.
- Teacher preferences – Teachers may have preferences to teach in certain periods and not in others.
- Class requirements - In some cases classes are split into subgroups and each subgroup is taught a different subject simultaneously in different venues. Alternatively, classes are merged together for certain subjects and taught in one large venue. The merger can occur over more than one grade. Another scenario involves splitting one or more classes in a form and rejoining the classes into different subgroups. Each subgroup is taught a different subject in the same period in different venues. These requirements must be specified.

  It may also be necessary for classes to have double or triple (i.e. two or three consecutive periods) lessons for certain subjects. Double and triple period requirements must be specified.

  If a lunch break is not built into the timetable, and the corresponding period is not included in the available periods, a specification indicating the need for a lunch break and the duration and range of periods during which it should be scheduled must be provided.

- Class preferences – There may be certain preferences as to when lessons for particular classes should be held, e.g. Mathematics for lower grades in morning sessions. This must be specified.

2.2 Hard Constraints

The hard constraints for the STP can be described in terms of the hard constraints for classes, teachers, and rooms.

Hard Constraints for Classes

- Every class must be allocated.
- Classes must be scheduled for the required number of meetings for each subject over the teaching period (usually a week).
- Classes must not be scheduled more than once during a period, i.e. there must not be any class clashes.
- Splitting and merging of classes (Beligiannis et al. 2008; Jacobsen et al. 2006; Kingston 2004; Kwok et al. 1997; Marte 2006; Post et al. 2008; Wilke et al. 2008) – Classes may be merged together for a lesson. In some cases classes may have to split into subgroups with each subgroup being taught a different subject simultaneously. The split subgroups may also need to be merged differently from the original configuration. The splitting and merging may take place for the same grade or across grades.
- Sequence of lessons (Melicio et al. 2006) – Certain subjects may have to be taught before or after other subjects.
- One period, in a specific range, should be allocated as a lunch break for pupils (Colorni et al. 1998; Wilke et al. 2008).

Hard Constraints for Teachers

- Teachers must not be scheduled more than once during a period.
- Teachers must be scheduled for the required number of meetings with each class over the teaching period.
- Teachers must only be scheduled when available (Beligiannis et al. 2008; Birbas et al. 1997; Post et al. 2008; Santos et al. 2008; Valouxis et al. 2003; Wilke et al. 2008) – Teachers may be unavailable during certain periods due to administration tasks, teaching in another school, allocated free periods or days off. Teachers must not be scheduled to teach during these periods.
- Teacher workload must be adhered to – The teacher workload is defined in terms of a minimum and maximum number of teaching lessons or hours per week (Birbas et al. 2009; de Haan et al. 2007; Nurmi et al. 2008; Santos et al. 2005; Wilke et al. 2008).
- Time permitted for commutation between schools (Schaerf 1991) – Some schools are located over more than one site. Thus, it may be necessary for teachers to commute

between schools. Thus, time for commutation must be allowed, for example a teacher cannot be scheduled to teach two consecutive periods, each on a different site.

Hard Constraints for Rooms

In some versions of the STP rooms have to be allocated as part of the timetable construction process and are not pre-assigned. The following constraints generally have to be met with respect to venues:

- Room capacities must not be exceeded (Wilke et al. 2008).
- All rooms must be used (Groebner et al. 2003).
- Certain lessons require specialized rooms, e.g. science labs, computer lab, the gymnasium. These requirements must be met. In some problems the specialized rooms are highly utilized, making the STP more difficult (Wilke et al. 2008; Wright 1996; Wood et al. 1998).
- Some schools are located over more than one site. In this case a room at the same location must be allocated for all class-teacher meetings of a subject (Post et al. 2008).

2.3 Soft Constraints

The soft constraints for the STP can be described in terms of the soft constraints for classes and teachers.

Soft Constraints for Classes

- Lesson session preferences – This refers to period preferences for lessons (Melicio et al. 2006; Wright 1996). There may be preferences for some subjects to be taught in morning sessions, e.g. mathematics or afternoon sessions (Colorni et al. 1998), a particular subject should not be taught in the first period of a day (Nurmi et al. 2008).

Soft Constraints for Teachers

- Teacher preferences - A teacher may prefer to teach in certain periods and not in others (Schaerf 1991; Nurmi et al. 2008; Valouxis et al. 2003).

2.4 Hard or Soft Constraints

The following constraints have been treated as hard constraints in some versions of the STP and soft constraints in others:

Constraints for Classes

- Idle or free periods – This constraint differs from one STP to the other. In some cases free periods are not allowed at all (Alvarez-Valdes et al. 1996; Jacobsen et al. 2006; Melicio et al. 2006; Wilke et al. 2008). In other problems only some grades or levels, usually higher grades, can have free periods (Filho et al. 2001). Some problems also stipulate when the free periods are permitted, e.g. last two periods of the day (Schaerf et al. 1991; Valouxis et al. 2003).

- Lesson spread – Different STPs have different spread requirements for lessons. For example, there may be a restriction of at most one lesson for a subject per day (Alvarez-Valdez et al. 1996; Filho et al. 2001; Melicio et al. 2006; Wright 1996). Alternatively, lessons must not be taught on consecutive days for *n* days (Alvarez-Valdez et al. 1996). Lessons for each subject must be distributed uniformly throughout the week (Alvarez-Valdez et al. 1996; Birbas et al. 2009; Colorni et al. 1998; Wright 1996).
  There must not be more than two daily lessons with the same teacher (Birbas et al. 1997; Santos et al. 2005). The same subject must not be taught in the last period of one day and the first period of the following day (Wright 1996).
- Double or triple lessons (DeHaan et al. 2007; Filho et al. 2001; Jacobsen et al. 2006; Kingston 2004; Melicio et al. 2006; Santos et al. 2008; Schaerf 1991) – It may be necessary to schedule a double (two consecutive lessons) or triple lesson (three consecutive lessons) with a class for a particular subject.

Constraints for Teachers

- Lesson spread – The lessons taught by a teacher should be well-spaced throughout the week (Valouxis et al. 2003). Alternatively, the lessons taught by a teacher should be concentrated over a limited number of days (Birbas et al. 1997; Filho et al. 2001; Santos et al. 2008).
- Idle/Free periods - Teachers are generally allowed some free periods (Wilke et al. 2008; Wright 1996), however the number of free periods for each teacher should be minimized when constructing the timetable (Birbas et al. 2009; de Haan et al. 2007; Post et al. 2008; Wilke et al. 2008).

2.5 Objective Function

One of two objective functions has been used to calculate the cost of the timetable. The first function is basically the sum of the hard and soft constraint violations (Abramson et al. 1991; Valouxis et al. 2003; Wood et al. 1998). The second function used is the weighted sum of the hard and soft constraint violations which allows for some constraints to have higher priority than others (Schaerf 1991; Wright 1996).

3. Solving the School Timetabling Problem

This section examines some of the methodologies that have been used to solve the STP. This survey of techniques is a work-in-progress and is by no means exhaustive. Methods used to solve the school timetabling problem include simulated annealing, evolutionary algorithms, tabu search, integer programming, constraint programming, GRASP (Greedy Randomized Search Procedure), and tiling algorithms. In some cases hybrid approaches, combining the use of two or more methodologies are implemented. Comparative studies, comparing the performance of two or more techniques in solving a particular STP have also been conducted. This section provides an overview of the different methods and studies.

3.1 Simulated Annealing

Abramson (1991) applies simulated annealing to the school timetabling problem. The atoms correspond to elements of the timetable and the energy to the cost of the timetable.
In order to allow for scheduling to be more flexible, assignments are made to room groups instead of individual rooms. If a group of classes must always take place at the same time, the classes should be scheduled as a group instead of individually. The system was tested on randomly generated problems and data from an Australian school.

Melicio et al. (2006) developed the THOR school timetabling tool to solve the STP for Portuguese schools. THOR firstly creates an initial solution using a heuristic constructive algorithm. This solution is then improved using fast simulated annealing.

3.2 Evolutionary Algorithms

Abramson et al. (1991) use a genetic algorithm to solve the school timetabling problem. A parallel algorithm is applied to speed up the process. Each chromosome consists of *n* periods and each period contains *m* tuples. The mutation operator changes the period of a tuple. Crossover is also applied to two chromosomes by choosing a crossover point in each chromosome and swapping the fragments. One child is returned which contains the first fragment of the first parent and the second fragment of the second parent. Crossover may result in the "label replacement problem", i.e. the child may contain some duplicated and/or missing genes. A label replacement algorithm is used to rectify this problem. The GA was used to solve nine highly constrained school timetabling problems.

Beligiannis et al. (2008) use an adaptive evolutionary algorithm to solve the school timetabling problem. Each element of the population is a matrix with the rows corresponding to the classes and the columns to the periods. Each cell in the matrix stores the teacher that will teach the class in the particular period. Initial studies indicated that crossover was not effective and time consuming and hence it was not used. The period mutation operator swaps the teachers between two time periods for a class. The periods chosen for swapping are randomly selected. The bad period mutation operator does not randomly choose periods, instead the two "most costly" periods in the corresponding teacher timetable are selected. Linear ranking selection is used to choose parents. The best chromosome of each generation is copied into the next generation. This algorithm successfully generated solutions to the Greek high school timetabling problem.

Caldiera et al. (1997) evaluate the use of genetic algorithms (GAs) to solve the STP by applying a GA to a small randomly generated school timetabling problem. An initialization procedure is used to create an initial population of feasible timetables. A GA is used to improve the quality of the initial population. Roulette-wheel selection and an ultra-elitism method are used for selection. Reproduction, mutation and crossover are applied to parents to create the offspring of the next generation. A repair algorithm is applied to offspring to ensure that they are feasible.

Filho et al. (2001) use a constructive genetic algorithm to solve the school timetabling problem for two Brazilian high schools.

Nurmi et al. (2008) convert the curriculum-based university course timetabling problem for the 2nd International Timetabling (ITC ’07) into one for school timetabling and use a genetic algorithm to solve this problem. The GA uses a greedy hill-climbing mutation operator to solve problem.

Raghavjee et al. (2008) apply a genetic algorithm to five highly constrained school timetabling problems (Beasley 2010). The algorithm firstly creates an initial population of timetables using a sequential construction method employing the largest degree heuristic. The mutation operator is used to iteratively refine initial population. A variation of tournament selection is used to choose the parents of each generation. The algorithm found solutions for all five problems and produced better results than other methodologies applied to the same set of problems.

Wilke et al. (2002) use a genetic algorithm to solve the German school timetabling problem. The initial population is comprised of potential timetable solutions, i.e. timetables are directly represented. Each chromosome contains the class timetables for the school. Roulette-wheel selection is used to choose parents. An elitist strategy copying the best two individuals into the next generation is also employed. In addition to crossover and mutation operators, a number of hybrid operators are applied. If there is no improvement in the fitness of offspring, a reconfiguration step is performed during which the parameters of the GA are reset.

3.3 Tabu Search

Bello et al. (2008) treat the school timetabling problem as a graph coloring problem. An adjunct graph is created and colored using an adaptation of the Tabu search algorithm for graph coloring (Tabucol), namely, Modified Tabucol (MT). The system was applied to five instances from Brazilian high schools.

In the approach taken by Jacobsen et al. (2006), an initial solution is firstly created using a construction heuristic with a graph coloring algorithm. The initial solution is then improved using Tabu search. The system was tested on data from German high schools.

Santos et al. (2005) firstly apply a constructive algorithm to create an initial solution. A tabu search using an informed diversification strategy is applied to the initial solution to improve the quality of the timetable. The diversification strategy was tested with transition based long term memory and residence based long term memory. The study showed that the use of a diversification strategy improved the quality of the timetable produced by the tabu search. The algorithm was used to solve the STP for Brazilian high schools.

3.4 Integer Programming

Earlier work by Birbas et al. (1997) use integer programming to solve the school timetabling problem for Greek high schools. This work is extended further in Birbas et al. (2009) which takes a hybrid approach to solving the problem. The first phase solves the shift assignment problem in which teachers are allocated to shifts. The second phase solves the school timetabling problem. Integer programming is used in both phases. The approach was successfully applied to a secondary Hellenic school.

Santos et al. (2008) use mixed integer programming to solve the STP for Brazilian high schools. A cut and column generation algorithm is implemented. The algorithm uses Fenchel cuts.

3.5 Constraint Programming

Valouxis et al. (2003) use constraint programming (CP) in combination with local search to solve the school timetabling problem for Greek high schools. CP is used to find a feasible timetable. The quality of the timetable is then improved using local search until further improvement is not possible. The stopping criterion is a runtime of one hour.

3.6 GRASP

Moura et al. (2010) use GRASP with path-relinking to solve the STP for three Brazilian high schools. GRASP takes a three stage approach to the problem. The first phase ranks lessons. During the second phase the ranking is improved using local search. In the third phase a path-relinking strategy is used to identify optimal solutions. These three phases are repeated a number of times.

3.7 Tiling Algorithms

Kingston (Kingston 2004; Kingston 2006) uses a tiling algorithm in combination with hill-climbing to allocate meetings (teacher and class tuples) and an alternating path algorithm for assigning recourses to meetings after times are fixed. Meetings are firstly placed onto tiles and then the tiles are timetabled. Resources are then allocated to meetings. Later research conducted by Kingston (2008) investigates the use of a bipartite matching model, namely, global tixel matching, to assign resources such as teachers and rooms to meetings. These algorithms have been applied to Australian high schools.

3.8 Hybrid Approaches

Alvarez-Valdes et al.(1996) takes a three phase approach to the school timetabling problem. In the first phase a parallel heuristic algorithm with priority rules is used to create an initial timetable which is not usually feasible. Phase 2 applies a variation of the standard tabu search to the initial timetable created in phase 1 to produce a feasible timetable. Phase 3 improves the quality of the feasible timetable developed in phase 2. A graph theory approach, using the Floyd-Warshall algorithm, is taken in this phase. This approach was tested on randomly generated problems and data sets from 14 Spanish schools.

De Haan et al. (2007) take a four-phase approach to solving the STP. A preprocessing phase is conducted to cluster events into clusters schemes using a branch-and-bound algorithm. The second and third phases focus on constructing feasible timetables. The second phase assigns lessons to day-parts using a dynamic priority rule. The cluster with the lowest availability is scheduled first. If this leads to unscheduled lessons the heuristic value is recalculated. During the third phase day-parts are allocated to timeslots. A graph coloring first-fit heuristic is used for this. The fourth phase uses a Tabu search to improve the feasible timetable. The system was successfully applied to a data set from a Netherlands high school.

The method employed by Schaerf (1991) firstly constructs an initial timetable by randomly assigning teacher-class pairs according to the requirements matrix. The RNA (Randomized Non-Ascendant) search is then applied to improve the initial timetable until no further improvement is possible. At this point the tabu search is applied until there is no more improvement. During the RNA phase the hard constraint cost is weighed higher than the soft constraint cost. During the tabu search this weight is "adjusted dynamically". This is referred to as adaptive relaxation. Adaptive relaxation was found to be essential for finding feasible solutions. The RNA and tabu phases are repeated sequentially until there is no further improvement in the quality of the timetable. This hybrid system was used to solve the STP for a randomly generated data set and data sets obtained from two Italian schools.

3.9 Comparative Studies

Colorni et al. (1998) compare the performance of simulated annealing, tabu search with local search and genetic algorithms in solving the school timetabling problem for two Italian high schools. The GA uses reproduction and crossover and applies mutation iteratively. Mutation swaps a set of contiguous genes in the same row. Day mutation swaps two days in the same row. A filtering algorithm is used to convert an infeasible offspring to a feasible one. Instead of creating timetables from scratch the previous year's handmade solution was used as a starting point. Tabu search produced the best results followed by genetic algorithms and simulated annealing.

Smith et al. (2003) use a Hopfield neural network to solve the school timetabling problem. The neural network is used to solve the problem for nine highly constrained school timetabling problems made available by Abramson (Beasley 2010). The performance of the Hopfield neural network on this data set is compared to that of greedy search, simulated annealing and tabu search. The neural network performed better than the other methods, followed by simulated annealing.

Wilke et al. (2008) compare the performance of tabu search, simulated annealing, genetic algorithms and branch and bound in solving the school timetabling problem for a German high school. The comparison is performed with respect to computation time and solution quality. Tabu search had the best runtimes but was unable to find feasible solutions. Simulated annealing found feasible solutions. The GA was not able to find feasible solutions. Branch and bound had the highest runtimes and also did not produce valid solutions.

4. Benchmark Data Sets for the STP

For further advancement in school timetabling research, it is essential that there is a variety of school timetabling data sets publicly available to test and compare the performance of different methodologies in solving the STP. This section describes school timetabling data sets that are publicly available.

Five of the data sets used in the study conducted by Smith et al. (2003) are available from the OR-Library maintained by Beasley(2010) at http://people.brunel.ac.uk/~mastjjb/jeb/orlib/tableinfo.html.

These problems have been described as “hard” timetabling problems and are more highly constrained than real-world school timetabling problems. The cost of the timetable is the number of class, teacher and venue clashes.

The seven data sets from Greek high schools tested by Beligiannis et al. (2008) are available at http://prlab.ceid.upatras.gr/timetabling/ . Each data set contains a requirements matrix specifying how many times each teacher must meet each class. In addition to this each data set also includes co-teaching and sub-classing (i.e. splitting and merging of classes) requirements.

Post et al. (2008) have initiated a project to facilitate the easy exchange of benchmark school timetabling data sets and so promote research in this domain. Post et al. propose an XML format to describe school timetabling problems and have setup a website for the submission of problems, namely, http://wwwhome.math.utwente.nl/~postgf/BenchmarkSchoolTimetabling/. There are 19 data sets from 7 different countries, namely, Australia (3), Brazil (6), England (1), Finland (4), Italy (1), Greece (1) and the Netherlands (3), available from the website.

5. Future Research Directions for School Timetabling

It is evident from the previous section that a variety of school timetabling data sets are now available. This will facilitate a comparison of different methodologies in solving the school timetabling problem and promote further development of the school timetabling domain.

A majority of the methods described in this paper firstly implement a construction phase during which a heuristic is used to sort class-teacher (or class-teacher-room) tuples in order of difficulty to schedule and allocate each tuple in sequence. An area which has not been investigated as thoroughly as in university examination timetabling (Carter et al. 1996) is different heuristics that can be used to estimate the difficulty of scheduling a tuple. The heuristic commonly used in the domain of school timetabling is a largest degree heuristic which gives priority to the most constrained class-teacher tuples (Raghavjee et al. 2008; Valouxis et al. 2003). The derivation and evaluation of other heuristics for this domain needs to be examined.

The aim of hyper-heuristics is to generalize well over the problems in a particular domain, rather than producing the best result for one or more problem sets (Burke et al. 2003). Hyper-heuristics search a heuristic space rather than a solution space. The heuristic space is usually comprised of combinations of low-level heuristics which can be constructive or perturbative (Pillay et al. 2009). While there has been research into the effectiveness of hyper-heuristics for university course and examination timetabling (Burke et al. 2007), this has not been studied for school timetabling.

A way of stimulating research in a particular field is to arrange competitions for the particular domain. This is evident from the 2nd International Timetabling Competition (McCollum et al. 2009) which has promoted research in examination, post enrollment and curriculum based university course timetabling. Arranging a track for school timetabling in future competitions will help develop the field more rapidly.

Building school timetabling systems that can be deployed in schools and are not just research tools is important to the development of the field. Such a system must allow for timetable reconstruction without much effort. The user must be able to easily make minor changes to the constraints, change the weighting of constraints, make manual changes and request a new timetable taking these into consideration. When methodologies are evaluated in this domain, the evaluation must also take into consideration reconstruction ability.

The application of some search techniques, e.g. evolutionary algorithms, can be time consuming. There have been earlier studies investigating the use of parallel processing to decrease the runtime of timetabling systems (Abramson 1991; Abramson et al. 1991). Given the emergence of multi-core processors the effectiveness of parallel processing in improving runtimes of school timetabling systems needs to be examined.

6. Conclusion

Research in the domain of school timetabling has not advanced as rapidly as other spheres of educational timetabling. This has been attributed to most studies in this domain being done in isolation of each other and the lack of a variety of benchmark problems to perform comparative studies on. The definition of the school timetabling problem varies drastically from one study to the next. This paper has attempted to provide a standardized definition of the problem in terms of problem requirements, hard constraints and soft constraints. The paper provides an overview of methodologies employed to solve the school timetabling problem. In addition to this the paper provides details of publicly available school timetabling data sets. Finally, the paper describes future directions of research in this field, namely, the derivation of new heuristics, an evaluation of hyper-heuristics in this domain, arranging competitions for school timetabling, developing usable systems that promote timetable reconstruction and the use of parallel processing to improve the runtimes of school timetabling systems.

7. References

1. Abramson D (1991) Constructing School Timetables Using Simulated Annealing: Sequential and Parallel Algorithms. Management Science, Vol. 37, No. 1, 98-113.
2. Abramson D, Abela J (1991) A Parallel Genetic Algorithm for the Solving the School Timetabling Problem. In proceedings of the Fifteenth Australian Conference: Division of Information Technology, C.S.I.R.O, 1-11.
3. Alvarez-Valdes R, Martin G, Tamarit JM (1996) Constructing Good Solutions for the Spanish School Timetabling Problem, in the Journal of the Operational Research Society, Vol. 47, No. 10, 1203-1215.

4. Beasley JF (2010) OR Library, http://people.brunel.ac.uk/mastjjb/jeb/orlib/tableinfo.html, last accessed 1 February 2010.
5. Beligiannis GN, Moschopoulos CN, Kaperonis GP, Likothanassis SD (2008) Applying Evolutionary Computation to the School Timetabling Problem: The Greek Case. Computers and Operations Research, Vol. 35, 1265-1280, Elsevier.
6. Bello GS, Rangel MC, Boeres MCS (2008) An Approach for the Class /Teacher Timetabling Problem. In the proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT2008), http://w1.cirrelt.ca/~patat2008/ PATAT_7_ PROCEEDINGS/Papers/Boeres-WA2b.pdf. Last accessed 05/02/10.
7. Birbas T, Daskalaki S, Housos E (1997) Timetabling for Greek High Schools. Journal of the Operational Research Society, Vol. 48, No. 2, 1191-1200, December 1997.
8. Birbas T, Daskalaki S, Housos E (2009) School Timetabling for Quality Student and Teacher Schedules. Journal of Scheduling, Vol. 12, Issue 2, 177-197, April 2009, Kluwer Academic Publishers.
9. Burke E, Hart E, Kendall G, Newall J, Ross P, Schulenburg S (2003) Hyper-Heuristics: An Emerging Direction in Modern Research. Handbook of Metaheuristics, chapter 16. Kluwer Academic Publishers, 457– 474.
10. Burke EK, McCollum B, Meisels A, Petrovic S, Qu R (2007) A Graph-Based Hyper-Heuristic for Educational Timetabling Problems. European Journal of Operational Research (EJOR), 176, 177 – 192.
11. Calderia JP, Ross AC (1997) School Timetabling Using Genetic Search. In the proceedings of the International Conference on the Practice and Theory of Automated Timetabling (PATAT ’97), 115-122.
12. Carter MW, Laporte G, Lee SY (1996) Examination Timetabling: Algorithmic Strategies and Applications. The Journal of the Operational Research Society, 47(3), 373-383.
13. Carter MW, Laporte G (1997) Recent Developments in the Practical Course Timetabling. In Burke E, Carter M (eds.) Practice and Theory of Automated Timetabling II, Lecture Notes in Computer Science, Vol. 1408, 3-19, Springer.
14. Colorni A, Dorigo M, Maniezzo V (1998) Metaheuristics for High School Timetabling. Computational Optimization and Applications, Vol. 9, 275-298, Kluwer Academic Publishers.
15. de Haan P, Landman R, Post G, Ruizenaar H (2007) A Four-Phase Approach to a Timetabling Problem for Secondary Schools. In Burke EK, Rudova H (eds.) The Practice and Theory of Automated Timetabling VI, Lecture Notes in Computer Science, Vol. 3867, 267-279, Springer-Verlag.
16. Filho GR, Lorena LAN (2001) A Constructive Evolutionary Approach to School Timetabling. In Proceedings of the EvoWorkshops on Applications of Evolutionary Computing, Lecture Notes in Computer Science, Vol. 2037, 130-139, Spring-Verlag.

17. Grobner M, Wilke P, Buttcher S (2003) A Standard Framework for Timetabling Problems. In Burke EK, De Causmaecker P (eds.): Proceedings of the International Conference on the Practice and Theory of Automated Timetabling (PATAT 2002), LNCS 2740, 24-38, Springer-Verlag Berlin Heidelberg.
18. Jacobsen F, Bortfeldt A, Gehring H (2006) Timetabling at German Secondary Schools: Tabu Search versus Constraint Programming. In Burke EK, Rudova H (Eds.): proceedings of the International Conference on the Practice and Theory of Automated Timetabling (PATAT 2006), 439-442, ISBN 80-210-3726-1.
19. Kingston JH (2004) A Tiling Algorithm for High School Timetabling. In Practice and Theory of Automated Timetabling V, Lecture Notes in Computer Science, Vol. 3616/2005, 208-225, Springer Berlin/Heidelberg.
20. Kingston JH (2006) The KTS High School Timetabling Systems. In Burke EK, Rudova H (Eds.): proceedings of the International Conference on the Practice and Theory of Automated Timetabling (PATAT 2006), 181-195, ISBN 80-210-3726-1.
21. Kingston JH (2008) Resource Assignment in High School Timetabling. In the proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT2008), http://w1.cirrelt.ca/~patat2008/PATAT7 PROCEEDINGS/ Papers/Kingston-WA2b.pdf. Last accessed 05/02/10.
22. Kwok L, Kong S, Kam Y (1997) Timetabling in Hong Kong Secondary Schools. Computer Education, Vol. 28, No. 3, 173-183, Elsevier Science Ltd.
23. Marte M (2006) Towards Constraint-Based Grammar School Timetabling. http://www3.deis.unibo.it/Events/Deis/Workshops/PapersCPAIOR99/21final.ps, Last accessed 12 February 2010.
24. McCollum B, McMullan P, Paechter B, Lewis R, Schaerf A, Di Gaspero L, Parkes AJ, Qu R, Burke EK (2009a) Setting the Research Agenda in Automated Timetabling: The Second International Timetabling Competition. INFORMS Journal on Computing, doi: 101287/ijoc.1090.0320.
25. Melicio F, Calderia JP, Rosa A (2006) THOR: A Tool for School Timetabling. In Burke EK, Rudova H (eds.): Proceedings of the 6th International Conference on the Practice and Teaching of Automated Timetabling (PATAT 2006), 532-535, ISBN 80-210-3726-1.
26. Moura AV, Scaraficci RA (2010) A GRASP Strategy for a More Constrained School Timetabling Problem. International Journal of Operational Research, Vol. 7, No. 2,152-170, 2010.
27. Nurmi K, Kyngas J (2008) A Conversion Scheme for Turning a Curriculum-Based Timetabling Problem into a School Timetabling Problem. In the proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2008), Montreal, http://w1.cirrelt.ca/~patat2008/PATAT_7_PROCEEDINGS/Papers/ Nurmi-TC1.pdf, last accessed 12 February 2010.
28. Pillay N, Banzhaf W (2009) A Study of Heuristic Combinations for Hyper-Heuristic Systems for the Uncapacitated Examination Timetabling Problem. European Journal of Operational Research(EJOR), 197, 482-491.

29. Post G, Ahmadi S, Daskalaki S, Kingston JH, Kyngas J, Nurmi C, Ranson D, Ruizenaar H (2008) An XML Format for Benchmarks in High School Timetabling. In the proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2008), Montreal, http://w1.cirrelt.ca/~patat2008/ PATAT_7_ PROCEEDINGS/Papers/Post-WD2a.pdf, last accessed 12 February 2010.
30. Raghavjee R, Pillay N (2008) An Application of Genetic Algorithms to the School Timetabling Problem. In Cilliers C, Barnard L, Botha RA (Eds.) Proceedings of SAICSIT 2008, 193-199, ACM Press.
31. Santos HG, Ochi LS, Souza MJF (2005) A Tabu Search Heuristic with Efficient Diversification Strategies for the Class/Teacher Timetabling Problem. Journal of Experimental Algorithms, Vol. 10, 2.9, ACM.
32. Santos HG, Uchoa E, Ochi LS, Maculan N (2008) Strong Bounds with Cut and Column Generation for Class-Teacher Timetabling. In the proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2008), Montreal, http://w1.cirrelt.ca/~patat2008/ PATAT_7_ PROCEEDINGS/Papers/Santos-WD2b.pdf, last accessed 12 February 2010.
33. Schaerf A (1991) Tabu Search Techniques for Large High-School Timetabling Problems. Technical Report CS-R9611 1996, Computer Science/Department of Interactive Systems, Centrum Voor Wiskunder en Informatica (CWI), ISSN 0169-118X.
34. Smith KA, Abramson D, Duke D (2003) Hopfield Neural Networks for Timetabling: Formulations, Methods, and Comparative Results, in Computers and Industrial Engineering, Vol. 44, 283-305, Pergamon.
35. Valouxis C, Housos E (2003) Constraint Programming Approach for School Timetabling. In Computers and Operations Research, Vol. 30, 1555-1572, Pergamon.
36. Wilke P, Grobner M, Oster N (2002) A Hybrid Genetic Algorithm for School Timetabling. AI 2002: Advances in Artificial Intelligence, Lecture Notes in Computer Science, Vol. 2557/2002, 455-464, Springer Berlin.
37. Wilke P, Ostler J (2008) Solving the School Timetabling Problem Using Tabu Search, Simulated Annealing, Genetic and Branch & Bound Algorithms. In the proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2008), Montreal, http://w1.cirrelt.ca/~patat2008/PATAT_7_PROCEEDINGS/ Papers/Wilke-WD2c.pdf, last accessed 12 February 2010.
38. Wright M (1996) School Timetabling Using Heuristic Search. Journal of the Operational Research Society, Vol. 47, No. 3, 347-357.
39. Wood J, Whitaker D (1998) Student Central School Timetabling. Journal of the Operational Research Society, Vol. 49, No. 11, 1146-1152.