UDC 007

METHODS OF AUTOMATIC GENERATION OF THE UNIVERSITY TIMETABLE

ULADZISLAU KUKSINSKI, TATSIANA GLUHOVA
Polotsk State University, Belarus

The problem of scheduling classes is a usual task at schools, and is fairly trivial, until you want to automate the process of generation. For this task, classical algorithms, genetic algorithms, and neural networks are used.

The class timetabling problem is a scheduling algorithm of great interest, which is connected with the fields of operational research and artificial intelligence. The problem was first studied by Gotlieb, who formulated a class-teacher timetabling problem taking into account that at every lecture there was a group of students and one teacher, so that the combination of teacher and students can be chosen easily. Because of the size of the problem, almost all effective solutions are heuristic in nature, and do not guarantee optimality. Among well-known results there are those that deal with various cases of problem settings. While setting a timetable, importance is given to effective utilization of resources such as classrooms, teachers, etc. This becomes a very tedious task which needs to be performed at least once a year by every academic institution. Most institutions deal with this problem manually, i.e. a trial and error method is used to set a timetable.

Due to the combinatorially explosive nature of the problem, enumeration and other deterministic methods fail and heuristics is preferable. Since the Timetabling problem is a common problem faced in different walks of life, its correlation with other operational research problems cannot be overlooked. For e.g., in scheduling sports timetables, consideration is not only given to scheduling a sport event, but also to reducing the cost factor, distance traveled by teams, etc. The Traveling tournament, where the total distance traveled by the team is minimized, further increased the academic interest in sports scheduling [3].

Timetabling is known to be a nondeterministic polynomial time complete problem i.e. there is no known efficient way to find a solution. Also, the most striking characteristic of NP-complete problems is that no best solution is known. Hence, in order to find a solution to a timetabling problem, a heuristic approach is chosen. This heuristic approach, therein, leads to a set of good solutions (but not necessarily the best solution). In a general educational timetabling problem, a set of events (e.g. courses and exams, etc.) are assigned into a certain number of timeslots (time periods) subject to a set of constraints, which often makes the problem very difficult to solve in real-world circumstances. In fact, large-scale timetables such as university timetables may need many hours of work spent by qualified people or team in order to produce high quality timetables with optimal constraint satisfaction and optimization of timetable's objectives at the same time. These constraints are of two types Hard and Soft constraints. Hard constraints include those constraints that cannot be violated while a timetable is being computed. For example, for a teacher to be scheduled for a timeslot, the teacher must be available for that time slot. A solution is acceptable only when no hard constraint is violated. On the other hand soft constraints can be changed while arriving at a solution. For example, though importance is given to a teacher's scheduling, focus is also made on setting a valid timetable and this can lead to free periods for teachers. Thus, while addressing the timetabling problem, hard constraints have to be adhered, at the same time effort is made to satisfy as many soft constraints as possible. Due to complexity of the problem, most of the work is concentrated on heuristic algorithms which help to find good approximate solutions.

Let's note the requirements on the preparation of the schedule.

When creating a timetable, the problem of optimal resource management arises: the teaching staff and the classroom fund. In the process of solving the problem, it is necessary to take into account mandatory restrictions, as well as additional requirements that may be violated in some cases [1].

Mandatory restrictions include such restrictions as:

- classroom capacity must be sufficient for the groups who are engaged in it, with the possible option when a few groups of students study in the same classroom simultaneously;
- the requirements on the equipment in the classrooms, which is necessary for this or that class;
- teachers from different universities can hold classes only on certain days and hours.

Additional requirements include such requirements as:

- lectures should be delivered at the beginning of the day, practical training is usually at the end of it;

- study load of each group should be uniform, in order to avoid overwork of students, that is, on those days when a there is a lecture on a subject that is considered hard, the remaining classes should be relatively simple;
- students should not have free periods in their timetable, at the same time there may be a free period for the teacher;
- if possible, teachers at the university should be provided with days free of classes;
- on Friday, the number of classes should be less than on other days of the week;
- the first lesson on Monday should be relatively simple, otherwise the student's progress may be significantly reduced.

It is important to note that the presented requirements are not common for all educational institutions. This list can be changed or supplemented.

Heuristic optimization methods are explicitly aimed at good feasible solutions that may not be optimal in cases where complexity of problem or limited time available does not allow exact solution. Generally, two questions arise: how fast the solution is computed and how close the solution is to the optimal one? Tradeoff is often required between time and quality which is reached by running simpler algorithms more than once, comparing results obtained from more complicated ones and effectiveness of comparing different heuristics. The empirical evaluation of heuristic method is based on analytical difficulty involved in the problem's worst case result. In its simplest form the scheduling task consists of class mapping, teacher and room combinations (which have already been pre-allocated) onto time slots. [3]

The problem of scheduling in educational institution can be divided into three main blocks: data preparation, scheduling, schedule approval. Input of the first block "data preparation" includes: disciplines in the curriculum, groups, classrooms, information on teachers and types of activities. Also at this step, the wishes of the teachers should be taken into account. The "data preparation" block supposes processing and structuring of the entered data. After that the data are transferred to the block "scheduling".

The scheduling unit can be implemented in various ways.

Consider the classic algorithm for scheduling on the basis of the data entered. The following steps should be performed:

- making the list of classes;
- finding free time slot for the lesson in the timetable;
- sorting the list of classes to increase the ease of their placement in the timetable;
- search for the most suitable time and place for the lesson;
- assessment of the quality of the timetable.

When estimating the ease of the lesson placement in the schedule:

- the number of classrooms suitable for the lesson, based on the equipment required and the number of workplaces;
- the number of lessons, conducted by a teacher per week;
- the number of lessons per week for a given group of students.

While sorting, first of all, the lessons with the least ease of placement are included in the schedule. When you add classes to the schedule, you search for the most suitable classroom and time for it. To do this, you need a full range of options, including space (classrooms) and time (time of the class, day of the week). In the process of possible variants estimation, hard constraints are checked first. If mandatory requirements are met, additional criteria are taken into account. Each of the criteria for assessing the quality of the lesson placement in the timetable should be considered as a separate specialized function with a weighting factor that reflects how important this criterion is. Flexibility and adaptability of the algorithm is achieved due to the ability to supplement the list of quality criteria without significant changes in the code. After evaluating the quality of all possible variants of the lesson placement in the schedule, the option with the maximum value is selected.

To improve the quality of the obtained schedule, the main algorithm for scheduling can be improved. In view of the availability of a set of tuning coefficients of the algorithm presented in the form of weights, it is possible to use a genetic algorithm. Thus, the tuning coefficients of the scheduling algorithm should be used as genes. As a result of the mutations, new sets of tuning coefficients will be obtained and as a consequence, the results of the algorithm will be different. The goal of the algorithm will be to achieve the maximum value of the quality of the schedule quality, which can be determined as summarized quality of each individual position.

The presented algorithms are not the only solution to this problem. At the moment there are many fundamentally different algorithms for scheduling classes, both classical and genetic. Also, the self-organizing feature map is used for scheduling, which is a method of projecting a multidimensional space into a space with a lower dimension (most often, two-dimensional), it is also used to solve modeling problems, forecasting, identifying sets of independent features, searching for regularities in large data sets.

Despite the possible reduction of human efforts spent on scheduling, these algorithms are rather cumbersome, and do not always provide sufficiently effective results. Thus, it is most advantageous to use algorithms to approximate the distribution of classes at the beginning of the work, and then to edit the obtained schedule manually to achieve the desired result and to take into account any additional criteria, or to use algorithms to complete the schedule partially compiled by hand.

REFERENCES

1. Береговых, Ю.В. Алгоритм составления расписания занятий / Береговых Ю.В., Васильев Б.А., Володин Н.А. ; Государственный университет информатики и искусственного интеллекта. – Донецк : Штучний інтелект, 2009 – 7 с.
2. Астахова, И.Ф. Составление расписания учебных занятий на основе генетического алгоритма / И.Ф. Астахова, А.М. Фирас ; Воронежский государственный университет. – Воронеж : Вестник ВГУ, 2013. – 7 с.
3. Anirudha, N. An Algorithm to Automatically Generate Schedule for School Lectures Using a Heuristic Approach / N. Anirudha, P.P. Manisha, G. Abhijeet // International Journal of Machine Learning and Computing. – 2012. – Vol. 2, №. 4. –4 p.