Proceedings of IYSW, (2020), vol. **9**, pp 196-212.

Journal homepage: http://journals.sdu.edu.kz/index.php/iysw

International Young Scholars' Workshop

The Automation of Course Scheduling in Higher Education Institutions: Mathematical methods and software products

Dr., Professor B. Aitchanov, PhD O. Baimuratov, MSc B. Rustauletov

Faculty of Engineering and Natural Sciences, Suleyman Demirel University

Kazakhstan, Almaty region, city Kaskelen, street Abylaikhan 1/1

Abstract

The formation and management of schedules is repetitive and troublesome for any class, university and organization. Limited resources in terms of people, time and locations. Institutions needs solutions that will meet all of these constraints of scheduling. That system for scheduling should take in control the scheduling process which takes time by automatically allocating time slots for teaching and creating course schedule. And easily can be integrated with their systems without need to do some extra work like adding data for scheduling or exporting them, reducing the time spend on creating schedules and minimizing errors that made by people.

Today, there are various types of systems and services designed to create schedules, reserve classrooms, assign classrooms for course teachers at specific times. The use of modern technologies, methods and models makes it easy to use open applications and services, as well as services at the local level. This paper presents the results of the analysis of systems. Their structures, algorithms, models, and also functional capabilities. The aim of this work is to determine the types of applications and systems, analysis of various options for the implementation of algorithms for the formation of schedules in educational institutions.

Keywords: *Automation, scheduling, time-tables, university, constraints, algorithms*

Introduction

In higher education institutions, the educational process is a complex process in which the main four subsystems operate: the faculties, the studying students, the administration and education staff of institutions.

A schedule is an important method used by various academic institutes as well as organizations to coordinate activities, time, people, locations, and other factors. However, the process of creating one is tedious-done manually through trial and error. It takes so much time and also leads to scheduling mistakes and conflicts

It may not be beneficial to use resources such as Microsoft Excel to build and arrange class schedules, which quickly leads to more time being spent and difficulties approaching. Class schedule can be released late or even with errors as a result of those constraints. In addition, due to the overlap with other events, more restrictions, such as adjusting the timing of events, contribute to difficulties in planning new schedules. Administrative staff continue to test a correctness of timeslot over and over again, thus wasting even more time and reducing their profitability.

In this paper, we first describe the scheduling model, and also review the methods of solving a problem with a discussion of possible problems using individual these methods in practice and give a comparative analyze of programs that automate scheduling

Scheduling Methods

The course scheduling is formed by description of following sets: groups G, lessons L, rooms R, teachers T, time periods P. Each of the elements of these sets may have additional features. For example, classes can be divided into lectures L_l, practices L_p and laboratory works

L_w: rooms can be divided by capacity and equipment; groups can be divided into subgroups and combined into flows, etc. Sets of plans is defined as X= G ∪L ∪ R ∪ T ∪ P. Each element of this set takes values 1(this room is currently occupied by this group and teacher) or 0. The number of elements in this set depends on number of elements in G, L, R, T, P. If for middle school with40 groups, 50 rooms, 10 teaching hours each day and teaching staff about 50 and 50 different lessons number of elements in X set is 300∗106, then for a university this value is hundreds of times larger, which creates difficulties already at the stage of storing information in computer memory. The size of the task can be reduced by technical means. If there is an excess of classrooms, then the set A can be decomposed into parts (buildings), while classes of certain groups will be conducted in specific buildings (for example, to conduct lessons for students of engineering specialties in one building, and in economics in another). But this leads to an increase in rooms downtime and decrease in schedule flexibility, for example, the creation of new specialties and other negative effects in solving the problem. Similarly, if the teachers of an educational organization spend most of their time conducting similar classes, then one room can be assigned to several. teachers, creating a set A ∪ P. However, such techniques are not always possible.

Many plans(schedules) X have restrictions, like at the same time in the room there can be

$$x_{1lrtp} + x_{2lrtp} + ... + x_{Mlrtp} \leq 1$$

Equation Group Restrictions

no more than one group or one flow. This restriction can be formalized as:

Here x is an element of the lesson plan related to the group g = 1, 2 ... M, the lesson with the number l, the room with the number r, the teacher t and the pair p; a value of x= 1 means that such a lesson in a period of p is held, and a value of x = 0 means that it is not.

$$x_{gl1tp} + x_{gl1tp} + ... + x_{glMtp} \leq 1$$

Similarly, a room cannot be occupied by several group at a same time:

here the values x, g, l, r, t, p are interpreted similarly to *Equation 1 Group Restrictions*. Most restrictions can be presented in a linear form (Babkina, 2008). Moreover, the restrictions are usually imposed on one of the sets (for example, the teacher p_1, who works as the head of the department, cannot conduct classes on Monday during the weekly meeting), which increases the number of restrictions and significantly increases the time it takes to solve the problem.

Among the plans X that satisfy the set limits, it is necessary to choose the optimal plan. For that function F is created, that takes into account the preferences of teachers about the time of classes f_1 : T×P→[0..1],the importance of lesson, etc. Obviously, the considered elements of the preference function are linear in nature and can be combined with other elements of the function using addition, which determines its general form as a linear function. Unfortunately, some of optimal criterias can be nonlinear. For example, wishes to reduce the transition time of teachers and groups between classrooms, obviously, has the character of non-linear dependence f_1:T×R×P×R×P→[0..1], f_2:G×R×P×R×P→[0..1], since it should take into account the position of the group both at a given moment in time and in the previous one. The use of nonlinear functions in the model complicates the solution generation of the problem. A separate question for the research is the possibility of including into function the requirements of the stability of the solution, which can be checked, in particular, experimentally by entering random perturbations and studying the changes in the obtained new optimal plan compared to the original one.

The above-described features of the statement of the problem of course scheduling in a university determine the complexity of the choice of tools for solving. This problem is an NP-complete many-extreme combinatorial problem with many restrictions (Ullman, 1973). In most cases, it can be considered as an integer linear programming problem. There are several approaches to solving such problems, in particular, methods based on the Lagrangian decomposition of the model into a series of one-dimensional problems (Nikisha & Marianthi, 2015). All these methods have obvious advantages and disadvantages, in particular, the requirement for linearity and limitations, and function of optimization. There are made some attempts to find a general solution to this problem using other methods, such as clustering, the use of genetic algorithms, etc. (Yong & Yi, 2011). In addition, the development of modern technology allows to use the brute force method, however, to solve the problem in a reasonable amount of time, it involves parallelizing the brute force process, which creates, among other things, technical difficulties. Methods based on model decomposition (for example, the approach of agent modeling considered by (Babkina, 2008), generally ineffective, since there is a strong synergistic connection between the different parts of the model through the remaining components of the model(for example, optimizing the schedule of one teacher will make worse the schedules of other teachers, because he will take the best time and classrooms). In general, from both theoretical and practical points of view, the development of the task of scheduling is currently ongoing. All the methods discussed above do not guarantee the solution of the best plans in a reasonable time, but many are able to find fairly good solutions. Nevertheless, the information market contains a large number of relatively effective software products that can automate the scheduling process based on the basic requirements for its optimality. Let's look at a few examples of such programs.

Scheduling Systems

UniTime

The computer program in question allows you to automate the time-consuming process of scheduling, to obtain a single electronic repository of the necessary information about it. The interface of the UniTime (Müller & Rudová, 2011), (University Timetabling, 2020) program has 4 sections (tabs) for increasing work efficiency: course, student, examination timetabling and event management *(Figure 3 UniTime Interface)* . In the "Course" section, the user of the program can enter the information necessary for scheduling, edit and print it. This can be data about the departments of the educational institution, existing specialties, groups, taught disciplines, classrooms of various types, types of classes. In this section, you can draw up curricula not only for groups of students, but also for teachers, classrooms, as well as for the entire university as a whole. Also the "Course" tab allows you to work with information about the curriculum (including for each specialty separately), the courses of both teachers and students, the distribution of hours for a weekly period (within one semester), and reports on the courses of the entire university, and the courses of the department and the specific teacher. It also means entering, editing and printing data. The “Student” section, the user can run the sectioning solver to assign classes to students whose course requests have been entered into the system. The assignment is based on the timetable, the student requests and, if needed, on the last-like or curriculum course demands. The “Examination” tab provides interface for running the solver to schedule midterm or final examinations. The “Events” tab is the primary starting place for anybody interested in events and associated meetings - be it a student looking for activities of a student organization or an event manager looking for room reservation requests requiring approval.

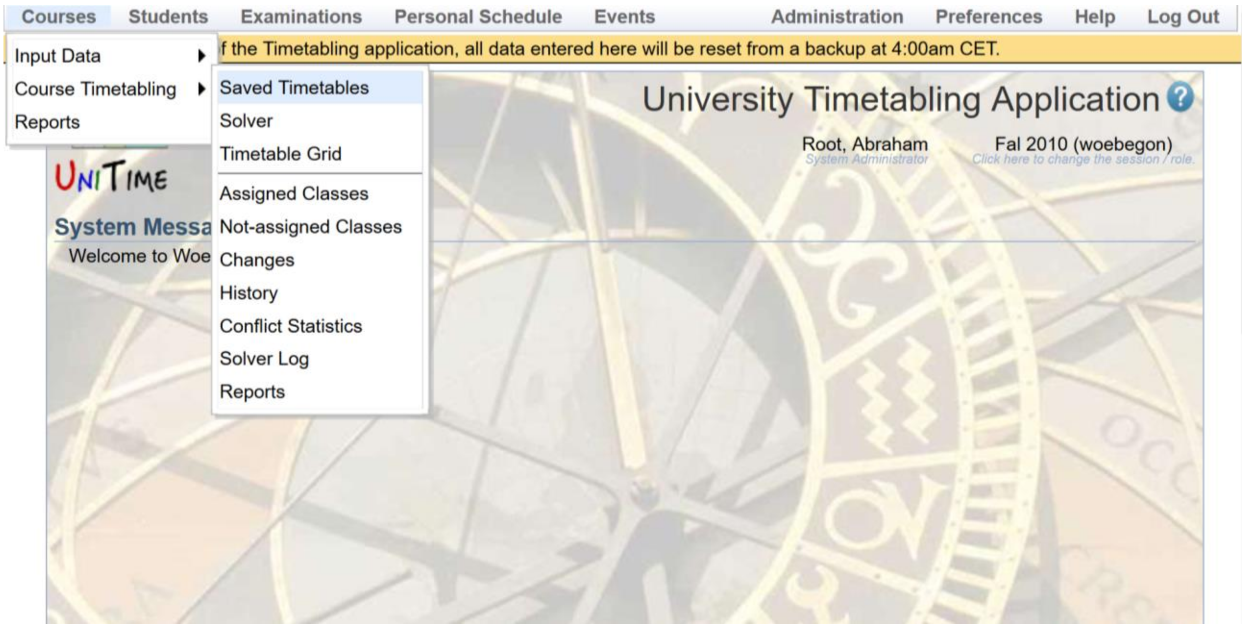

Figure 3 UniTime Interface

The program allows you to work in automatic, manual and combined modes, and the transition between them is possible at any stage of scheduling. Automatic mode is able to take into account various distribution requirements, and manual mode, in turn, has a hint function.

The finished class schedule can be saved in .doc (Microsoft Word), .xls (Microsoft Excel) or html formats.

1C University

The individuality of the institution, additional counts, in addition to the main lesson schedule, the classroom composition and much more are taken into account by the "1C: Automated Schedule program. University" (1C Solutions, 2020). The program for scheduling is designed to solve the problems of automated scheduling and operational management of premises in universities. Using it, you can schedule in automatic, manual and mixed modes, taking into account many restrictions and conditions. At the same time, you can build both an acceptable schedule and an optimized one, in which the number of windows or the number of rooms used is reduced.

This system includes the same features as the programs discussed above. However, its difference lies in the fact that it is more flexible in relation to the characteristics of a particular university. Main functionalities of system:

- scheduling in manual / automatic or mixed mode;
- convenient “Chess Board” form for quick manual modification of the schedule by drag and drop; (*Figure 4 Chess Board)*
- scheduling without reference to the grid of calls with classes of different durations;
- scheduling in terms of semesters / departments / scenarios ("pessimistic", optimistic). Drawing up several schedules and choosing the best;
- taking into account the wishes and possibilities of teachers, groups of students, rooms;
- a comparison of admissibility in scheduling in any mode: type of room / type of occupation, capacity of the room / number of students in the group;
- the choice of an arbitrary periodicity of the schedule (week, two weeks, a fixed period, etc.);
- scheduling a session;
- accounting for parallel classes, subgroups and associations for streaming lectures in scheduling;

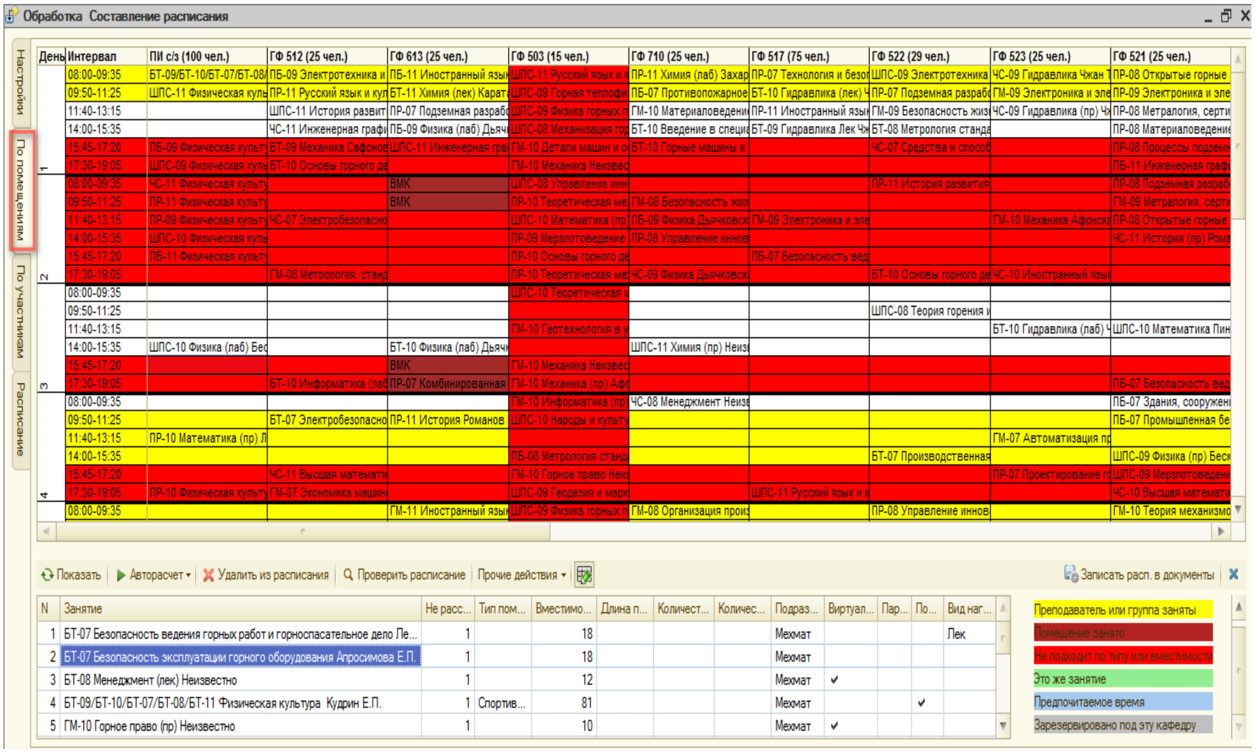

Figure 4 Chess Board

Figure 4 Chess Board shows the main form for scheduling ("chess", lines - days of the week, pairs, columns - rooms) in the program under consideration.

FET (timetabling software)

FET is open source free software for automatically scheduling a school, college or university timetable (Liviu & Volker, 2020). Usually, a complicated timetable can be solved by FET in 5-20 minutes maximum. It can take a shorter time, under 5 minutes (in some cases, a matter of seconds) for simpler timetables. It may take a longer time for extremely difficult timetables, a matter of hours. FET uses text files, xml or html or txt or csv (separate commas-import / export values). The encoding which is used is UTF-8. Completely automatic algorithm generation which also enables semi-automatic or manual allocation. Independent implementation of the application allowing to run on GNU / Linux , Windows, Mac and any system Qt supports.

Flexible structure of students, divided into sets: years, classes, and subgroups. FET allows years and groups to overlap, and subgroups that are not overlapping. You may even define students individually (as separate sets). A wide and versatile collection of time constraints(For teacher, students, an activity or a set of activities/subactivities, rooms).

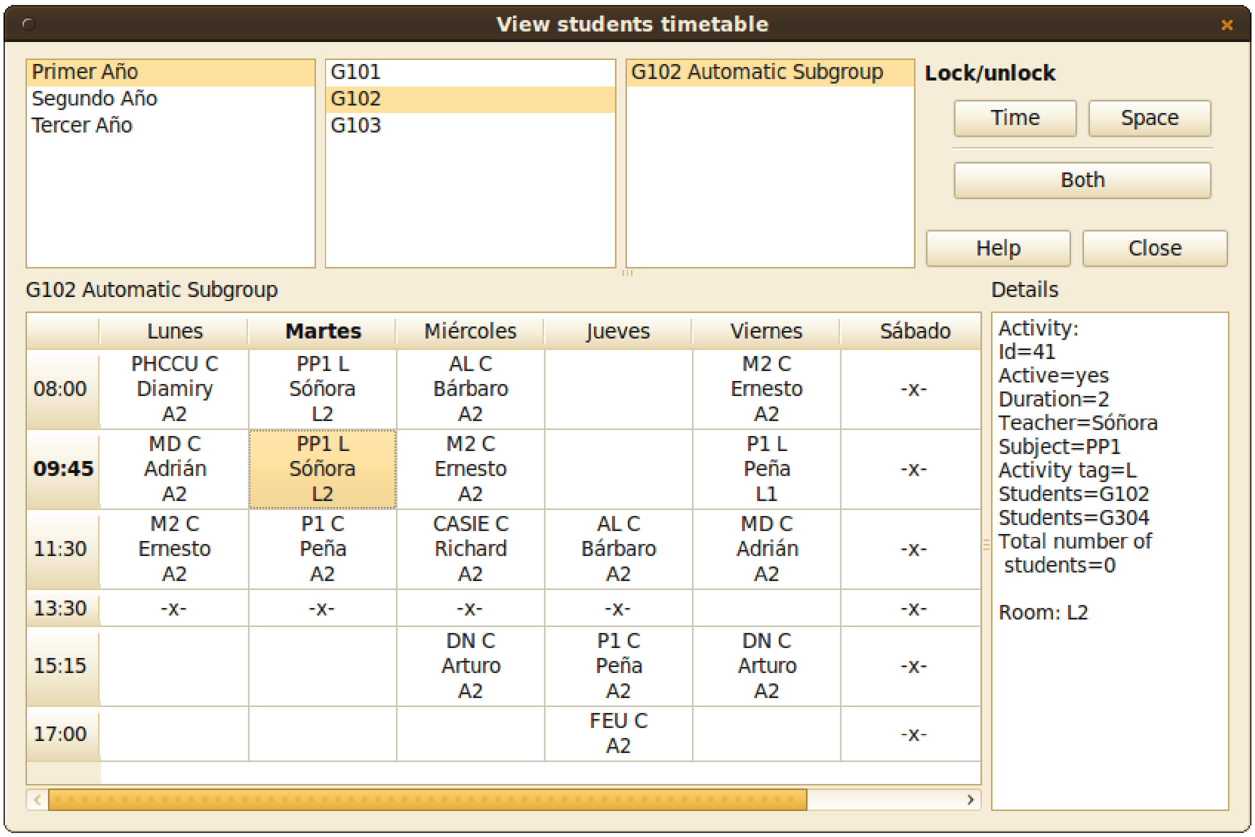

Figure 5 FET Interface

AscTimeTables

ASc TimeTables is the only timetable solution capable of generating school timetables up to the last card.

The curriculum is intended for all types of elementary and secondary schools. It allows you to:

- Respect both psycho-hygienic and pedagogic teaching criteria for individual subjects
- Economically using classrooms, specialist quarters, and other school services
- Reduce timetable-administrative demand
- Eliminate counterproductive behaviors and the human element when designing school schedules;
- Respect the pedagogical team's needs and opportunities; eliminate unwanted free teacher spare lessons (free slots / windows);
- Enhance your school environment and interpersonal relationships.

Compare the considered programs according to certain criteria for their full analysis.

Table 1 Price Comparison

Program	**Price**
UniTime	Free of use. Because it open-source program, it needs license from The Apereo Foundation, who owns project
1C University	License cost: 70,000 rubles. Or ~ 1000$ After the grace period has expired, to receive updates, consultations and services, it is necessary to conclude an agreement on regular support of 1C: Enterprise programs.
FET	FET Software is Free. They consider donating to this project.

AscTimetables	The price is calculated individually after communication with a potential client by the company's specialists. 4 Levels: • Primary School $ 399 + Support $79 • Standard $ 499 + Support $79 • Premium $ 955 + Support $79 • Pro $ 3995 + Support $79

Table 2 Comparison by Criterias

Criteria	UniTime	1C University	FET	AscTimetables
Developer	The Apereo Foundation	1C Company	Volker Dirr	aSc Applied Software Consultants
Client server architecture	Yes	Yes	No	Yes
Software platform	Windows	Windows	Windows	Windows
Work via web-interface	Yes	No	No	No
Intelligent	Yes	Yes	Yes	Yes

support compilation process schedules (modes)				
Consideration of geographic factors of buildings, transition factors from audiences	No	Yes	No	No
Planning exams, consultations, etc.	Yes	Yes	Yes	No
Schedule options in various sections (students, teachers, employment audiences)	Yes	Yes	Yes	Yes
Differentiation of access rights	Yes	Yes	Yes	Yes
Ability to import / export data	Yes	Yes	Yes	Yes
User friendly interface	Yes	Yes	No	Yes

Reporting	Yes	Yes	No	No
Taking into account group, streaming, subgroup classes	No	Yes	Yes	No
Taking into account the load of students and teachers	Yes	Yes	Yes	Yes
Schedule Change Notifications	No	Yes	No	No

Conclusion

Based on the analysis of the available software for scheduling in higher education institutions, we can say that all the programs considered cope with their main task. They allow you to work in automatic, manual and mixed modes, create a schedule of teachers, students, audience occupations, take into account the load of schedule objects, save the schedule in various formats. However, not all programs are able to perform more complex functions of notifications, accounting for various forms of classes, taking into account specific features of the location of the university buildings, its classrooms, etc. Accordingly, the cost of such software depends on the degree of automation of the scheduling process.

The result of the analysis of information sources and official sites of educational institutions of the Republic of Kazakhstan showed that the formation of a schedule - systems, platforms, services do not know direct data, that from this we assume that each institution applies its own third-party application, which is not centralized for all educational institutions.

References

1C Solutions, C. (2020, March 21). *1C : Company*. Retrieved from 1C Solutions: https://solutions.1c.ru/catalog/asp_univer

Babkina, T. (2008). Scheduling: multi-agent approach solution. *Business Informatics*, 23-28.

Liviu, L., & Volker, D. (2020, April 7). *FET*. Retrieved from Free Timetabling Software: https://lalescu.ro/liviu/fet/credits.html

Müller, T., & Rudová, H. (2011). Proceedings of the 5th Multidisciplinary International Scheduling Conference. *Rapid Development of University Course Timetables.* MISTA.

Nikisha, K., & Marianthi, G. (2015). Lagrangian decomposition approach to scheduling large-scale refinery operations. *Computers & Chemical Engineering*, 1-29.

Ullman, J. D. (1973). NP-Complete Scheduling Problems. *COMPUTER AND SYSTEM SCIENCES*, 384-393.

University Timetabling. (2020, March 15). Retrieved from UniTime: https://www.unitime.org

Yong, O., & Yi, C. (2011). The 6th International Conference on Computer Science & Education. *Design of automated Course Scheduling system based on hybrid genetic algorithm* (pp. 256-259). Singapore: SuperStar Virgo.