Real-world university course timetabling at the International Timetabling Competition 2019

Hana Rudová[1], Tomáš Müller[2], and Zuzana Müllerová[3]

[1] Faculty of Informatics, Masaryk University, Brno, Czech Republic
hanka@fi.muni.cz
[2] Purdue University, West Lafayette, Indiana, USA
muller@unitime.org
[3] UniTime, s.r.o., Zlín, Czech Republic
mullerova@unitime.org

1 Introduction

The International Timetabling Competition 2019 (ITC 2019) [12] introduced a variety of real-life university course timetabling problems coming from different parts of the world. A novel model of a complex course timetabling problem allows the specification of problems from many different universities. In the competition, representative problems from ten universities worldwide were considered. However, they represent a fraction of the institutions using UniTime [16], a non-commercial software, from which the instances for the competition were taken. Thirty benchmark problems together with six test instances are available at the competition website [6], which allows for solution validation and provides a repository of existing solutions. This paper will discuss the characteristics of the course timetabling problems considered in the competition. We will demonstrate that the model proposed for the competition allows encapsulating very different features.

The first International Timetabling Competition 2002 considered a simplified course timetabling where post-enrollment problems were solved. For these problems, course enrollments of students are defined, and courses must be assigned in timeslots and rooms without any overlap for students. All benchmark instances were randomly generated. The second competition in 2007 [9] has organized two course timetabling tracks. One of them slightly extended the post-enrollment problem from the first competition [8] and the other introduced curriculum-based timetabling based on problems from the University of Udine in Italy [4]. The curriculum contains a set of courses, which must be assigned into timeslots with no overlaps. All the competitions, including ours, were supported by the PATAT conference together with several other competitions from different domains [13]. A recent survey about educational timetabling benchmarks and competitions is available at [1].

2 Characteristics

Our problems use a complex course structure to model the presence of students in different parts of a course. A course may contain one or more course configurations, each with one or more classes that can be of different types and have an optional parent-child relationship between them. These classes are to be timetabled into rooms and time periods. A class may occupy multiple time periods, possibly spanning multiple days and weeks. This allows us to model classes with multiple meetings at the same time and room, and/or classes that are taught only during certain weeks of the semester. All benchmark problems have five-minutes time periods, which are going from midnight to midnight, have seven days a week, and are running for a given number of weeks (between 6 and 21). This permits a very flexible organization of time and supports various irregularities and other exceptions in class placements. Students are enrolled in courses and are to be assigned to classes based on the defined course structure. A student must get one class of each type from a single course configuration, following the parent-child relationship when defined. For example, each student must get a lecture and a seminar, where only some lecture-seminar combinations are allowed. Finally, there are soft and hard distribution constraints of nineteen types defined on subsets of classes. Most of the constraints such as `SameDays` or `NoOverlap` can be validated on pairs of classes, i.e., each pair that does not satisfy the constraint incurs a penalty. Four types of constraints such as `MaxDays` must be validated on the whole subset of classes.

There are four essential optimization criteria. The goal is to minimize penalties for time and room assignments of classes, penalties for unsatisfied soft distribution constraints, and the number of student conflicts. Minimizing the number of student conflicts is a fundamental part of the problem, which is crucial for university course timetabling. A student conflict exists if the student cannot attend a pair of his/her classes. The conflicts are not only between classes that overlap in time, but they are also between classes that students cannot attend due to travel distances between assigned rooms.

3 Problems from different universities

We will see that the proposed XML model allows specifying very different real-life university course timetabling problems. Timetabling problems may differ even within the same institution.

We have included three different problems from Masaryk University (Czech Republic). The timetable for the Faculty of Informatics can be generated based on pre-enrollments of students into courses. Otherwise, it is a relatively standard mid-size problem with about 500 classes each scheduled weekly or sometimes bi-weekly. There are two different types of problems for the Faculty of Education and Faculty of Sport Studies, representing (1) the common present form of study and (2) the lifelong together with the combined forms of study [11]. This second problem is very specific and complex. Here, a different timetable is needed each week, and each course is taught only several weeks during the semester. On top of that, the Faculty of Sport Studies timetables are significantly influenced by traveling to various sports facilities that are spread over the city. We can see specific curricula patterns for the Faculty of Education, typically composed of a pair of "sub-curricula", each representing one field of study such as Math, Physics, English, or Music. These pairs may result in many student conflicts because it is impossible to satisfy all the possible combinations.

Purdue University (USA) uses automated timetabling for all its departments together [15], which means that we can see the large-scale problem representing all courses of the large public university with about 40,000 students. The construction of the timetable starts with timetabling for the large lecture rooms, which the university shares. We have only a few courses for each student in this problem, but the room utilization is very high since large rooms for hundreds of students represent a scarce and expensive resource. At Purdue, we can also see a typical example of an American university where classes are taught several times a week at the same time and same room, for instance, Monday, Wednesday, Friday at the half-hour (7:30 am, 8:30 am, ... 4:30 pm). Also, courses may be taught using different patterns, e.g., either two times a week for an hour and a half or three times a week for one hour. In contrast to other problems discussed before, there are neither curricula nor pre-enrollments. The timetable is constructed based on last-like semester course enrollments (e.g., timetable construction for Fall 2019 used as an input real course enrollments for Fall 2018).

AGH University of Science and Technology from Poland builds course timetables separately for each faculty. Still, they share some resources, and some of the faculties provide a lot of courses for students outside of their faculty. For instance, in our data sets, the Faculty of Humanities has almost two-thirds of the classes for students of other faculties. The data are structured so that the courses for students from these faculties can be managed and timetabled separately. AGH uses pretty rigid curricula, only containing mandatory and elective courses. In the original (UniTime) problem, students of the same curriculum are kept together and attend the same classes. There are no student conflicts allowed to be created by the solver.

For several other universities from Asia, such as Turkish-German University, İstanbul Kültür University, and Bethlehem University, it may seem that there are no students involved because no students are present in the data set. This is because these universities decided to model student course requirements using the `SameAttendees` or `NoOverlap` distribution costraints.

Many other specifics of the competition problems will be described in the full version of this paper.

4 Conclusion

The competition problems introduce complex real-world problems with many different characteristics, which led to a relatively small number of approaches capable of solving them. Initially, five different teams submitted results to the competition, and two others, including one competition organizer, computed solutions as well. Thanks to the open-source prize, the source codes of three solvers are now publicly available. As of June 2022, thirteen teams have submitted some solutions published on the competition website. The winning team applied a parallelized matheuristic [10] based on the graph-based mixed integer programming formulation [5]. The second team applied mixed integer programming [14], and the third team's solution [3] is representative of a metaheuristic method, which is the modified version of simulated annealing. The fifth team opted for a solution using the MaxSAT solver combined with a local search [7]. The winning team also maintains a website [2], where their lower bounds for all competition instances are published. We can see that five instances are now solved optimally. For many instances, the gap is still significant, promising opportunities for future research. Also, there is a high potential for developing more efficient methods capable of solving the problems in a more reasonable time frame.

References

1. Ceschia, S., Gaspero, L.D., Schaerf, A.: Educational timetabling: Problems, benchmarks, and state-of-the-art results (2022), arXiv:2201.07525
2. DSUM data science for university management, ITC 2019, https://dsumsoftware.com/itc2019/
3. Gashi, E., Sylejmani, K., Ymeri, A.: Simulated annealing with penalization for university course timetabling. In: Proceedings of the 13th International Conference on the Practice and Theory of Automated Timetabling PATAT 2021, Volume II (2021). pp. 361–366 (2021)
4. Gaspero, L.D., McCollum, B., Schaerf, A.: The second International Timetabling Competition (ITC-2007): Curriculum-based course timetabling (track 3). Tech. Rep. QUB/IEEE/Tech/ITC2007/CurriculumCTT/v1.0, Queen's University, Belfast (2007)
5. Holm, D.S., Mikkelsen, R.Ø., Sørensen, M., Stidsen, T.J.R.: A graph-based MIP formulation of the International Timetabling Competition 2019. Journal of Scheduling (2022), published: 11 March 2022
6. ITC 2019: International Timetabling Competition, https://itc2019.org
7. Lemos, A., Monteiro, P.T., Lynce, I.: ITC 2019: University course timetabling with MaxSAT. In: Proceedings of the 13th International Conference on the Practice and Theory of Automated Timetabling PATAT 2021, Volume I (2020). pp. 105–128 (2021)
8. Lewis, R., Paechter, B., McCollum, B.: Post enrolment based course timetabling: A description of the problem model used for track two of the second International Timetabling Competition. Cardiff Working Papers in Accounting and Finance A2007-3, Cardiff Business School, Cardiff University (2007)
9. McCollum, B., Schaerf, A., Paechter, B., McMullan, P., Lewis, R., Parkes, A.J., Gaspero, L.D., Qu, R., Burke, E.K.: Setting the research agenda in automated timetabling: The second International Timetabling Competition. INFORMS Journal on Computing **22**(1), 120–130 (2010)
10. Mikkelsen, R.Ø., Holm, D.S.: A parallelized matheuristic for the International Timetabling Competition 2019. Journal of Scheduling (2022), published: 3 May 2022

11. Müller, T., Rudová, H.: Real-life curriculum-based timetabling with elective courses and course sections. Annals of Operations Research **239**(1), 153–170 (2016)
12. Müller, T., Rudová, H., Müllerová, Z.: University course timetabling and International Timetabling Competition 2019. In: PATAT 2018 — Proceedings of the 12th International Conference on the Practice and Theory of Automated Timetabling. pp. 5–31 (2018)
13. PATAT conferences, http://patatconference.org/communityService.html
14. Rappos, E., Thiémard, E., Robert, S., Hêche, J.F.: A mixed-integer programming approach for solving university course timetabling problems. Journal of Scheduling (2022), published: 15 February 2022
15. Rudová, H., Müller, T., Murray, K.: Complex university course timetabling. Journal of Scheduling **14**(2), 187–207 (2011)
16. UniTime.org: University timetabling, comprehensive academic scheduling solutions, https://www.unitime.org/