Timetabling with Answer Set Programming

Marvin Beese, mabeese@uni-potsdam.de, 786300

University of Potsdam

Abstract. Answer Set Programming can be successfully used in educational timetabling. Timetabling, which is generally a difficult problem, is represented here with hard and soft constraints and finds an optimal solution. This summary examines the general problem of educational timetabling and gives an overview of the required and optional constraints.

1 Introduction

Course timetabling in general is a problem where a certain number of lectures is assigned to timeslots and rooms. This is a complex and difficult problem in computer science. This problem needs to satisfy conditions regarding expressiveness, extensibility and flexibility. This means that the encoding of the problem must be human-readable and intuitively, it must have the option to capture new constraints, to switch between hard and soft constraints and it must be able to deal with different formulations. The article "Answer set programming as a modeling language for course timetabling" by Mutsunori Banbara, Takehide Soh, Naoyuki Tamura, Katsumi Inoue and Torsten Schaub [1] deals with the use of Answer Set Programming (ASP) in curriculum-based course timetabling, which is one of many applications of timetabling, like transport timetabling, employee timetabling or sports timetabling. ASP is used, as this meets the needed requirements and is also reliable to current competition. In the following I am going to describe the educational timetabling problem and consider the problem definition of timetabling, including the used constraints.

2 The educational timetabling problem

The goal in the educational timetabling problem and especially in the curriculum-based course time tabling problem (CB-CTT) is to find an assignment of lectures to a limited set of timeslots and rooms, so that a given set of hard constraints is satisfied and only a minimum of a set of soft constraints is violated. The use of soft constraints is eligible, because different institutions may choose different policies wherefore these constraints can become (in-)valid. In the encoding, the hard constraints use integrity constraints and aggregates and the soft constraints use rules, where the head holds the predicate $\mathit{penalty}(S, V, C)$ and the body calculates whether a soft constraint S is violated by violation V, which causes a penalty cost C. Besides the hard and soft constraints, which will be investigated deeper in the next section, there exists an optimization statement, whose objective is to represent a feasible solution where the *penalty* cost is minimal.

3 Problem definition

For the representation of the CB-CTT problem we must consider a timeslot as a pair of a day and a period and a curriculum as a group of courses, that several students attend. The hard constraints are necessary and must be satisfied. These state that lectures must be assigned to distinct timeslots (H_1), that courses of the same curriculum or by the same teacher must be scheduled in different timeslots so that there are no time-dependent lecture conflicts (H_2), that in one room only one lecture can be held at a time (H_3) and that a lecture of a course can only be scheduled at that timeslot, when the teacher is available at that time (H_4).
The soft constraints do not necessarily have to be satisfied, but the penalty cost of their violation has to be minimized. For that either a soft constraint with constant or with calculated cost of the penalty comes into account. Some soft constraints with constant cost are isolated lectures, where one lecture is not adjacent to another one (S_3), instantaneous travel moves, where students have to travel between buildings (S_7), insufficiently equipped rooms (S_8) and double lectures, which should be grouped together (S_9). Soft Constraints with calculated cost are the room capacity, that should not be undercut regarding the number of participants of a lecture (S_1) and a minimum days of work, where the lectures of each course should be spread above a minimum number of days (S_2). Further, a curriculum should not have time periods without teaching between lectures (S_4), the lectures of a course should be held in the same room (S_5) and a number of daily lectures within a curriculum should not exceed a maximum value (S_6).
Further encodings can be found in [1] and are not dealt with in this summary, as this would get too technical.

4 Conclusion

Answer Set Programming has advantages for course timetabling, as the challenge of combining both hard and soft constraints can be met. With the use of minimization statements, the CB-CTT problem becomes an optimization problem for the soft constraints, which must not be strictly satisfied. As the paper [1] further analyzes, the implementation with ASP is able to compete with other implementations, so that in many instances of the problem it improves or produces the same bounds compared to previous best known bounds of other implementations.

References

1. Mutsunori Banbara, Takehide Soh, Naoyuki Tamura, Katsumi Inoue, Torsten Schaub: Answer set programming as a modeling language for course timetabling. TPLP 13(4-5): 783-798 (2013)