Course Timetabling via Constraint Programming

Nikolaos Pothitos, Foivos Theocharis, and Panagiotis Stamatopoulos

Department of Informatics and Telecommunications, University of Athens
Panepistimiopolis, 157 84 Athens, Greece
{pothitos,f.theoharis,takis}@di.uoa.gr

Automated timetabling is a well known problem that has been studied for decades by researchers in the areas of Artificial Intelligence and Operations Research. In educational institutes, this problem comes to front in different ways. In this work, we face the case of *course timetabling* that is formulated according to the 2nd International Timetabling Competition Curriculum-based Course Timetabling Track specifications.

Our approach to this problem was to solve it via Constraint Programming (CP), a popular Artificial Intelligence methodology.Initially, we defined the variables of the problem, the constraints that connect them, and the objective function, in a declarative way that was supported by an object-oriented Constraint Satisfaction Problem (CSP) solver, implemented in C++. The philosophy of CP, which allows the separation of the problem statement and the procedure to solve it, gives the opportunity to easily experiment with many search methods. Thus, we took advantage of an independent library of search methods for the CSP solver, and compared many methods to each other. The direct method 'Depth-Bounded Discrepancy Search' (DDS) [1] was finally chosen. Search methods are guided by heuristics specially designed for the problem. So, the application is able to find feasible solutions to all the problem instances currently published by the competition, in few seconds; while the solver keeps searching, the solutions are optimized.

More specifically, for each lecture of the problem we define two constrained variables: the one represents a teaching period (its domain contains all the allowed periods for the course), and the other represents a classroom (its domain initially contains all the classrooms). Then, the CSP solver uses the DBS to find solutions (i.e. valid assignments to the constrained variables). DBS interchanges with Local Search, that re-schedules the lectures of a specific day, to improve the cost of the timetable. The solver maintains *arc-consistency* in every search node, using the AC-5 algorithm [2] to remove inconsistent values from the domains of the variables, thus pruning the 'no-goods' search paths.

References

1. Walsh, T.: Depth-bounded discrepancy search. In: Proc. 15th Int. Joint Conf. on Artificial Intelligence (IJCAI-97). (1997) 1388–1395
2. van Hentenryck, P., Deville, Y., Teng, C.M.: A generic arc-consistency algorithm and its specializations. Artificial Intelligence **57**(2-3) (1992) 291–321