Abstraction in Constraint Optimization Problems[1]

Alexandru Ioan Tomescu
University of Bucharest,
Faculty of Mathematics and Computer Science,
Str. Academiei, 14,
010014 Bucharest, Romania
E-mail: alexandru.tomescu@gmail.com

Many real-life applications such as planning, scheduling and timetabling can be classified as hard combinatorial optimization problems. These are characterized by a search space that grows exponentially with the size of the problem, many complex constraints, a high number of feasible solutions among which we have to choose the best one. The performance of search-based algorithms can be improved by look-back techniques, which extract information from failing search paths, or by look-ahead techniques, which use constraint propagation to avoid such dead-ends altogether.

This paper formalizes the search-based algorithm with abstraction proposed by Jaffar et al. for Constraint Optimization Problems, and gives some propositions regarding its correctness. The technique abstracts a search state so that a future encounter of a state in its abstract class can be pruned. The algorithm can be improved using a branch-and-bound scheme. As a running example, the crew rostering problem is considered, and a proof of correctness for the algorithm is given. The running times and the search space of its implementation in CLP(P) show a significant improvement over naive backtracking. Finally, some guidelines are provided for integrating look-back and look-ahead methods with abstraction for optimization problems.

Keywords: constraints, constraint logic programming, backtracking, abstraction, branch-and-bound, scheduling.

AMS Subject Classification: 65K10, 68T20, 90B35, 90C27.

[1] This paper was written during an Internship Attachment at the School of Computing, National University of Singapore.