Abstract

In this thesis, we develop reduction techniques for combinatorial optimization and constraint satisfaction problems that can be embedded in a tree search approach. In combinatorial optimization, bound estimates and variable fixing algorithms are commonly used for that purpose, whereas in constraint programming filtering algorithms undertake the task of shrinking the search space by eliminating values from variable domains. The algorithms that we develop are meant to be used as symbolic constraints in constraint programming solvers and also in optimization software that provides high level description constructs like the SCIL-library. Therefore, we consider the work done in this thesis as a contribution toward the development of efficient and easy to use optimization software.

Note that, when solving discrete optimization problems to optimality, there are really two tasks to be considered. First, an optimal solution must be constructed, and second, its optimality must be proven. Optimal or at least near-optimal solutions can often be found quickly by heuristics or by approximation algorithms, both specially tailored for the given problem. In contrast to the construction of a high quality solution, the algorithmic optimality proof requires the investigation of the entire search space, which in general is much harder than to partly explore the most promising regions only. By eliminating parts of the search space that do not contain improving solutions, problem reduction can help with respect to both aspects of discrete optimization.

The thesis is organized in two parts. Part I consists of the Chapters 2–4 and is method oriented. This means that the task of achieving a certain degree of consistency for some special filtering problems is studied theoretically. The efficiency of the algorithms developed is measured in terms of worst case complexity and the degree of consistency that they achieve.

The first type of reduction algorithm that we develop involves a special kind of symbolic constraint: In Chapter 2, our goal is to provide a set of so-called *optimization constraints*. By linking the objective function with the constraint structure of a problem, such constraints can be used for pruning and problem reduction with respect to cost considerations, a process called *cost-based filtering*. That way, optimization constraints naturally combine the optimization abilities of operations research algorithms and the efficient modeling and filtering concepts of constraint programming.

Particularly, we study the problem of achieving different degrees of consistency for optimization constraints. Since achieving a state of hyper-arc-consistency may turn out to be an NP-hard problem itself, we introduce a new type of consistency for optimization constraints, so-called *relaxed consistency*. Based on the two concepts, we develop efficient cost-based filtering algorithms for shortest path constraints (on directed acyclic graphs, undirected graphs with non-negative edge weights, and directed graphs without negative cycles), weighted stable set constraints in interval graphs, weighted all-different constraints, and knapsack constraints.

These constraints are supposed to be used as basic building blocks when modeling real-life discrete optimization problems. By exploiting the knowledge of the given constraint structures, the corresponding reduction algorithms make use of previously developed bounds and the efficient ways known to compute them.

As we shall see, the loose connection of optimization constraints via variable domains results in less effective and thus also less efficient problem reduction. Therefore, in Chapter 3, we present a theory that motivates the linking of optimization constraints via the standard operations research decomposition techniques column generation and Lagrangian relaxation.

Then, a second type of reduction algorithm is developed that bases its decisions on the constraint structure of a problem rather than on cost considerations. Obviously, a search node does not need to be expanded if it represents a previously considered configuration. However, this situation occurs frequently when tackling problems that contain symmetry. In Chapter 4, we present a general symmetry breaking method called SBDD that is based on dominance detection between choice points. An experimental evaluation shows that the method is better suited to tackle highly symmetric problems than previously developed symmetry breaking techniques.

Part II of the thesis covers the Chapters 5–9 and is application oriented. Several combinatorial optimization and constraint satisfaction problems are investigated. The approaches that we develop are based on the algorithms and methods from Part I. This allows an empirical evaluation of the previously developed reduction algorithms on top of the theoretical work done in the first part.

In particular, we consider the Airline Crew Assignment Problem in Chapter 5. The approach presented is based on the concept of CP-based column generation in combination with shortest path constraints. By exploiting CP and OR specific advantages, we are able to speed up the computation of real-world airline crew schedules considerably. The ideas that we present have been integrated in an industrial airline crew assignment software system and have yielded drastic savings in running time.

In Chapter 6, we study the Automatic Recording Problem, that evolves in the context of modern multimedia applications. After giving an approximation scheme for the NP-hard problem, an exact algorithmic approach is presented that links knapsack constraints and weighted stable set constraints on interval graphs following the idea of CP-based Lagrangian relaxation. Numerical results show that our implementation is efficient enough to tackle real-size problem instances in an amount of time that is well affordable in practice.

The Capacitated Network Design Problem is tackled in Chapter 7. Lower bounds can be computed by decomposing the problem. We review previously developed reduction techniques and use CP-based Lagrangian relaxation to link them together. Moreover, a new technique is presented that adds locally valid cuts based on a Lagrangian relaxation of the problem. In our experiments, we show that a heuristic version of our potentially exact solver is able to provide solutions of higher quality in less time than the best known heuristic techniques known so far.

A new approach for the Social Golfer Problem is developed in Chapter 8. Using SBDD for symmetry breaking and the new idea of heuristic constraint propagation, we are able to solve problems that were previously out of reach for solvers based on constraint programming.

Finally, in Chapter 9, we develop a solver for the Graph Bisection Problem. The core of the algorithm is a lower bounding procedure that approximates maximum multicommodity flows with multiple sinks. Comparisons with a previously developed bound based on semi-definite programming show the gains in quality and computation time on sparse, structured graphs. Especially, our implementation is the first to compute the bisection widths of DeBruijn 9, Shuffle-Exchange 9, and Shuffle-Exchange 10.