Reoptimization Techniques for MIP Solvers

Gerald Gamrath, Benjamin Hiller, and Jakob Witzig

Zuse Institute Berlin, Takustraße 7, 14195 Berlin, Germany
{gamrath,hiller,witzig}@zib.de

Abstract. Recently, there have been many successful applications of optimization algorithms that solve a sequence of quite similar mixed-integer programs (MIPs) as subproblems. Traditionally, each problem in the sequence is solved from scratch. In this paper we consider reoptimization techniques that try to benefit from information obtained by solving previous problems of the sequence. We focus on the case that subsequent MIPs differ only in the objective function or that the feasible region is reduced. We propose extensions of the very complex branch-and-bound algorithms employed by general MIP solvers based on the idea to “warmstart” using the final search frontier of the preceding solver run. We extend the academic MIP solver SCIP by these techniques to obtain a reoptimizing branch-and-bound solver and report computational results which show the effectiveness of the approach.

1 Introduction

In the last decades many powerful decomposition- and reformulation-based techniques for solving hard optimization problems were developed, e.g., column generation and Lagrangian relaxation. These methods decompose a problem into a master problem and several subproblems which are repeatedly solved to update the master problem. Frequently, the subproblems solved in successive iterations differ only in the cost vector, reflecting updated information from the master problem. It is a natural idea to exploit this property in order to improve the running time of the overall algorithm for solving the master problem. Methods to achieve this are known as reoptimization techniques. They have been investigated in the context of decomposition methods, e.g., in the context of Lagrangian relaxation [20], column generation [8], and for generic branch-and-bound [17].

In the literature, reoptimization techniques have been investigated for polynomial solvable problems mainly, e.g., the shortest path problem [21] or the min cost flow problem [12]. This is partly due to the fact that traditionally, decomposition methods have been applied such that the resulting subproblems are (pseudo)polynomially solvable. More recently, Mixed Integer Programs (MIPs) have been used as subproblems, e.g., for cut generation [10,11] or in generic decomposition schemes [14,22] and corresponding solvers [9,15]. The resulting subproblems are solved by standard MIP solvers, which are very sophisticated branch-and-bound algorithms. Thus there is a need for reoptimization techniques in MIP solvers to benefit from the knowledge obtained in previous iterations.

One of the first investigations on reoptimizing MIPs was done by Güzelsoy and Ralphs [16,23]. They addressed sequences of MIPs that differ only in the
right-hand side. Their approach is mainly based on duality theory, which they employed to develop techniques for “warm starting” using dual information obtained through primal algorithms. Our approach to reoptimizing MIPs is similar to the One-Tree algorithm [6] for generating multiple solutions of a single MIP. Similar techniques have also been used in [18] to benefit from a preliminary restricted branching phase when solving a single MIP instance.

In this paper, we propose a reoptimizing variant of the general LP-based branch-and-bound algorithm used by modern MIP solvers. It is based on the idea [17] to “continue” the search at the last known search frontier of the branch-and-bound tree. As the performance of state-of-the-art MIP solvers is based to a substantial extend on exploiting dual information, we introduce a mechanism to deal with this. This mechanism is in particular applied to cope with strong branching. It is intuitively clear that continuing the solving process is a poor idea if the objective function has changed a lot or the search frontier is rather huge. To deal with the first situation, we use a similarity measure for objective functions to decide whether to reoptimize or to start from scratch. Moreover, we propose heuristics to start with a reduced search frontier that is still based on the previous one. Our ideas have been carefully implemented using the MIP solver SCIP [2,25]. We test our reoptimization techniques on sequences arising from the generic column generation solver GCG [15] and on instances of the $k$-constrained shortest path problem arising from a ship navigation problem\textsuperscript{1}. More details and computational results can be found in the master thesis of the last author[27].

The paper is outlined as follows. Sec. 2 provides a summary of the relevant ingredients of a state-of-the-art MIP solver (i.e., SCIP) together with an in-depth motivation to reoptimization for MIPs. Sec. 3 presents our technical contributions summarized above. Computational results are discussed in Sec. 4. Finally, Sec. 5 concludes the paper.

2 Mixed Integer Programming and Reoptimization

In this paper we consider mixed integer linear programs (MIPs) of the form

$$z_{MIP} = \min \{ c^T x : Ax \geq b, \ell \leq x \leq u, x_i \in \mathbb{Z} \text{ for all } i \in I \}$$  (1)

with objective function $c \in \mathbb{R}^n$, constraint matrix $A \in \mathbb{R}^{m \times n}$ and constraint right-hand sides $b \in \mathbb{R}^m$, variable lower and upper bounds $\ell, u \in \mathbb{R}^n$ where $\mathbb{R} := \mathbb{R} \cup \{ \pm \infty \}$, and a subset $I \subseteq \mathcal{N} = \{ 1, \ldots, n \}$ of variables which need to be integral in every feasible solution. In the remainder of the paper, we focus on mixed-binary programs, i.e., MIPs with $\ell_i = 0, u_i = 1 \text{ for all } i \in I$.

When omitting the integrality restrictions, we obtain the linear program (LP)

$$z_{LP} = \min \{ c^T x : Ax \geq b, \ell \leq x \leq u \}.$$  (2)

\textsuperscript{1} Thanks to Mirai Tanaka from Tokyo Institute of Technology and Kazuhiro Kobayashi from National Maritime Research Institute Japan for providing the instances.