6

Interior point algorithms for integer programming

John E. Mitchell

Department of Mathematical Sciences,
Rensselaer Polytechnic Institute,
Troy, NY 12180-3590, USA

1 Introduction

An algorithm for solving linear programming problems can be used as a subroutine in methods for more complicated problems. Such methods usually involve solving a sequence of related linear programming problems, where the solution to one linear program is close to the solution to the next, that is, it provides a *warm start* for the next linear program. The *branch and cut* method for solving integer programming problems is of this form: linear programming relaxations are solved until eventually the integer programming problem has been solved. Within the last ten years, interior point methods have become accepted as powerful tools for solving linear programming problems. It appears that interior point methods may well solve large linear programs substantially faster than the simplex method. A natural question, therefore, is whether interior point methods can be successfully used to solve integer programming problems. This requires the ability to exploit a warm start.

As discussed in Chapter 5 of this book, branch and cut methods for integer programming problems are a combination of cutting plane methods and branch and bound methods. In a cutting plane method for solving an integer programming problem, the linear programming relaxation of the integer program is solved. If the optimal solution to the linear program is feasible in the integer program, it also solves the integer program; otherwise, a constraint is added to the linear program which *separates* the optimal solution from the set of feasible solutions to the integer program, the new linear program is solved, and the process is repeated. In a branch and bound method for solving an integer programming problem, the first step is also to solve the linear programming relaxation. If the optimal solution is feasible in the integer program, it solves the integer program. Otherwise, the relaxation is split into two subproblems, usually by fixing a particular variable at zero or one. This is the start of a tree of subproblems. A subproblem is selected and the linear programming relaxation of that subproblem is solved. Four outcomes are possible: the linear programming relaxation is infeasible, in which case the integer subproblem is also infeasible, and the tree can be pruned at this node; the optimal solution to the linear programming relaxation is feasible in the integer subproblem, in which case it also solves the subproblem, and the tree can be pruned at this node; the optimal solution has a worse objective function value than a known integer solution to the original problem, in which case any solution to the integer subproblem is also worse than the known solution, and the tree can be pruned at this node; finally, none of these situations occur, in which case it is necessary to split the node into two further subproblems.

The difficulty with using an interior point method in a cutting plane algorithm is that the solution to one relaxation is usually not a good starting point for an interior point method, because it is close to the boundary of the feasible region. Thus, it is usually necessary to attempt to stop working on the current relaxation before it is completely solved to optimality: the earlier we are able to find good cutting planes, the better the initial solution to the next relaxation. Early termination obviously reduces the number of iterations spent solving the current relaxation; in addition, it reduces the number of iterations spent solving the next relaxation, because the initial point to the next relaxation is more centered. There are two potential disadvantages from trying to find cutting planes early: if the search for cutting planes is unsuccessful, we have wasted time; secondly, it may well be that superfluous constraints are added, with the result that the algorithm requires extra iterations and extra stages of adding cutting planes.

It is also necessary to be able to use a warm start when using an interior point method in a branch and bound algorithm; again, it is possible to use early termination to improve the algorithm. Usually, the only time it is necessary to solve the subproblem to optimality in the branch and bound tree is when the optimal solution to the linear programming relaxation is feasible in the integer subproblem.

We will concentrate on branch and cut algorithms for integer programming in the rest of this chapter, with the principal ideas being presented in Section 2. A cutting plane method can be regarded as a *column generation method* applied to the dual of the linear programming relaxation, because the addition of a cutting plane, or constraint, to the relaxation adds a column to the dual. There has been research on using interior point methods in column generation algorithms for various problems, and we will discuss some of this work later in the chapter, because of its relevance to branch and cut algorithms. For now, we give two examples. First, consider a linear programming problem with many more variables than constraints. If all the variables were included in the working set of columns, the matrix algebra would become impractical. Therefore, a pricing criterion is used to select a subset of the variables, and this subset is updated periodically – see Bixby *et al.* [3], Kaliski and Ye [20], and Mitchell [26], for example. Secondly, column generation methods are also useful for solving nonsmooth optimization problems. The constraints and the objective function are approximated by piecewise linear functions, and the approximation is improved as the algorithm proceeds. The problem solved at each stage is a linear programming problem, and refining the approximation corresponds to adding columns to the linear program. For more details, see, for example, the papers by Goffin and Vial and their coauthors [10, 12].

Another approach for solving integer programming problems using interior point methods has been presented by Karmarkar *et al.* [21]. We discuss this method in Section 3.2.

2 Interior point branch and cut algorithms

In this section, we discuss the use of interior point methods in cutting plane and branch and bound approaches to solving integer programming problems. The two approaches can be combined into a branch and cut algorithm; however, we intend to deal with them separately in order to make the basic ideas clear. Various observations about interior point algorithms will affect the design of efficient algorithms in both the cutting plane and branch and bound approaches. In particular, it is vital to use early termination in order to obtain good performance from these algorithms.

The standard form integer programming problem we consider has the form:

$$\begin{array}{lll} \text{minimize} & c^T x, & \\ \text{subject to} & Ax \geq b, & (IP) \\ & x \in \{0,1\}, & \end{array}$$

where x and c are n-vectors, b is an m-vector and A is an $m \times n$ matrix. Note that we restrict attention to problems where the integer variables are constrained to be binary. We define the polyhedron Q to be the convex hull of feasible solutions to the problem (IP). The problem (IP) can be solved by solving the linear programming problem $\min\{c^T x : x \in Q\}$. (If this linear program has multiple optimal solutions, an interior point method may well return a point which is non-integral, but this can be rounded to give an integer optimal extreme point of Q using the methods of, for example, Megiddo [24].) The difficulty with this approach, obviously, is that in general a closed form expression for Q is not known. In a cutting plane method, we consider *linear programming relaxations* of (IP), where the feasible region gradually becomes a better approximation to Q, at least in the neighbourhood of the optimal solution. The linear programming relaxation of (IP) is

$$\begin{array}{lll} \text{minimize} & c^T x, & \\ \text{subject to} & Ax \geq b, & (LPP) \\ & 0 \leq x \leq e, & \end{array}$$

where e denotes a vector of ones of the appropriate dimension. We denote the feasible region of this problem by $Q^{LPP} := \{x \in \mathbb{R}^n : Ax \geq b,\ 0 \leq x \leq e\}$. The dual of this problem is

$$\text{maximize} \qquad b^T y - e^T w,$$