

A VIEW OF INTERIOR POINT METHODS FOR LINEAR PROGRAMMING

Vera V. KOVAČEVIĆ-VUJČIĆ *)

*Laboratory for Operations Research,
Faculty of Organizational Sciences, University of Belgrade,
Jove Ilića 154, 11000 Belgrade, Yugoslavia*

Abstract: The paper gives a view of interior point methods for linear programming inspired by the work of Karmarkar. Two possible classifications of the existing methods are presented through the detailed discussion of typical representatives. The paper also discusses some implementation details and provides a selected bibliography.

Keywords: Linear programming, Interior methods, Computational complexity

1. INTRODUCTION

The introduction of polynomial-time interior methods is one of the most important developments in mathematical programming in the last decade. The first step in that direction was made in 1984 when Karmarkar in his seminal paper [35] proposed a new polynomial algorithm for linear programming. The method had not only a better complexity bound than the earlier method of Khachian [37] but it also enjoyed computational efficiency on practical problems. Karmarkar's paper stimulated an enormous research interest which has resulted in the development of numerous interior point methods both for linear and nonlinear programming problems and for certain discrete optimization problems. The ten years of research resulted in about 2000 papers dealing with interior point methods (see Kranich's bibliography [41]). Recently, Nesterov and Nemirovskii [55] have developed a general framework for polynomial-time interior point methods, based on self-concordant functions. Interior point methodology has also been used to develop a new approach to the theory of linear programming, including duality and sensitivity analysis (see [34]).

This paper reviews the most important aspects of interior point methods for linear programming inspired by work of Karmarkar. Section 2 describes the main idea

*) This research was supported by Science Fund of Serbia, grant number 0401F, through Mathematical Institute

of interior point methods and gives two possible classifications. Section 3 is devoted to projective methods with the emphasis on the original method of Karmarkar. Affine methods, including the original and modified version of Dikin's method [18, 19] and Gonzaga's polynomial modification [27] are analyzed in Section 4. Section 5 describes the main idea of path-following methods on the example of Gonzaga's method [25]. Section 6 gives a survey of search directions in interior point methods with the emphasis on numerical stability problems. Finally, Section 7 discusses some details related to practical implementation of interior point methods.

2. THE BASIC PRINCIPLES OF INTERIOR POINT METHODS

Consider the linear programming problem in standard form

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax = b \quad x \geq 0 \end{aligned} \quad (1)$$

where $A = [a_{ij}]_{m \times n}$, $c \in R^n$, $x \in R^n$, $b \in R^m$, $m < n$, $\text{rank} A = m$ and assume that A , b and c have integer entries (which is equivalent to the assumption that the entries are rational). Let X be the feasible set of (1) and, for the sake of simplicity, assume that X is bounded and $\text{ri } X = \{x \in R^n \mid Ax = b, x > 0\} \neq \emptyset$. It is well known that the problem dual to (1) can be written in either of the two forms:

$$\begin{aligned} \max \quad & b^T y \\ \text{subject to} \quad & A^T y \leq c \end{aligned} \quad \begin{aligned} \max \quad & b^T y \\ \text{subject to} \quad & A^T y + z = c, \quad z \geq 0 \end{aligned} \quad (2)$$

The idea of interior point methods for solving (1) is quite opposite to that of the simplex method: instead of visiting different extreme points of X , a typical interior point method generates a sequence $(x^k) \subset \text{ri } X$, with the property that for k large enough the approximate solution x^k can be transformed into an optimal solution x_{opt} . The most important issue in this approach is to estimate polynomially the number of elementary operations (additions, multiplications, comparisons, etc.) needed to get an optimal solution. In addition, it has to be shown that generated points have polynomial descriptions. This is usually done by showing that it is sufficient to do all the calculations in certain fixed precision which depends on the data of problem (1) (see e.g. [54]). We shall here shortly discuss a theoretical background for the transformation of x^k into x_{opt} .

We shall denote by $\langle h \rangle$ the number of bits needed for binary representation of an integer h . Then $\langle h \rangle = 1 + \lceil \log_2 (|h| + 1) \rceil$. Let r be the bit length of the input data in problem (1), i.e.

$$r = \langle a_{11} \rangle + \dots + \langle a_{mn} \rangle + \langle c_1 \rangle + \dots + \langle c_n \rangle + \langle b_1 \rangle + \dots + \langle b_m \rangle$$

and let $L = r + \langle n \rangle$. The next Lemma, basically due to Khachian [37], shows that the quantity 2^{-L} can be used for the identification of the optimal solution.

Lemma 1. Suppose that the feasible set X of problem (1) is nonempty and bounded and let $f_{opt} = c^T x_{opt}$ be the optimal objective function value. If \bar{x} is an arbitrary extreme point of X then either $c^T \bar{x} = f_{opt}$ or $c^T \bar{x} > f_{opt} + 2^{-L}$. ■

Interior point methods for linear programming use the results of Lemma 1 in the following way: Suppose that the optimal value f_{opt} is known and that we seek a point where it is achieved. Let (x^k) be a sequence generated by an interior method and suppose that for some k we have

$$c^T x^k - f_{opt} \leq 2^{-L} \quad (3)$$

Furthermore, suppose that the point x^k is transformed into an extreme point \bar{x} of X with the property $c^T \bar{x} \leq c^T x^k$. Then according to Lemma 1 $c^T \bar{x} = f_{opt}$, i.e. \bar{x} is an optimal solution (equal to x_{opt} in the case of the unique solution). An obvious difficulty with this approach lies in the fact that in general case f_{opt} is not known and there are several ways to get around it. Karmakar in [35] proposes replacing problem (1) with a new problem in which the duality gap, known to have 0 as the optimal value, is minimized. Several authors propose generating a sequence (y^k, z^k) of dual feasible solutions. Since $c^T x_{opt} - b^T y_{opt} = 0$ and $b^T y^k \leq b^T y_{opt}$ we have $(z^k)^T x^k = c^T x^k - b^T y^k = c^T x^k - c^T x_{opt} + b^T y_{opt} - b^T y^k \geq c^T x^k - c^T x_{opt}$ and the test (3) can be replaced by $c^T x^k - b^T y^k \leq 2^{-L}$ or $(z^k)^T x^k \leq 2^{-L}$. There are also more efficient approaches which avoid explicit evaluation of dual variables (see e.g. [63]).

It remains to be seen how a point x^k satisfying (3) can be transformed into an extreme point \bar{x} with the property $c^T \bar{x} \leq c^T x^k$. Such transformations are often called rounding procedures. The idea of the rounding procedure below is to start from x^k and at each step generate a feasible point which has at least one more zero coordinate than the previous point and at the same time does not increase the objective function value. The Algorithm is taken from [20], see also [40].

Let $x^k > 0$ be a feasible point and let the vectors z^1, \dots, z^{n-m} constitute a basis of the vector space $N(A) = \{x \in R^n \mid Ax = 0\}$. Let us note that z^1, \dots, z^{n-m} can be obtained with $O(m^3)$ operations by pivoting on the matrix A . Namely, if matrix A is transformed into the matrix $[I_m \ M]$ then the rows of the matrix $[M^T - I_{n-m}]$ are a basis of $N(A)$, where I_m and I_{n-m} are the unit matrices of order m and $n-m$, respectively.

Algorithm 1.

Step 0: Set $s = n - m$, $y = x^k$

Step 1: Set $z = z^s$. If $c^T z^s < 0$ set $z = -z^s$. Compute
 $\lambda = \min \{y_j / z_j \mid z_j > 0\} = y_r / z_r$.

Step 2: Compute $\bar{y} = y - \lambda z$. If $s = 1$ set $\bar{x} = \bar{y}$ STOP. Otherwise compute

$$\bar{z}^j = z^j - (z_r^j / z_r) z, \quad j = 1, \dots, s-1.$$

Step 3: Set $y = \bar{y}$, $z^j = \bar{z}^j$, $j = 1, \dots, s-1$, replace s by $s-1$ and go to Step 1. ■

It can be shown that the point \bar{x} generated by Algorithm 1 is an extreme point of X with the property $c^T \bar{x} \leq c^T x^k$. Since each cycle of the algorithm uses $O(n^2)$ elementary operations and the number of cycles is $O(n)$, the total complexity is $O(n^3)$.

All known interior point methods use the outlined ideas for the transformation into an exact solution. The difference occurs in the rule for generating the sequence (x^k) and the number of steps needed to get a solution satisfying (3) or a related stopping criterion. A possible classification of interior methods, based on the geometric interpretation, is the following:

1. Projective methods (Karmarkar [35], Anstreicher [3], De Ghellinck and Vial [14], Todd and Ye [64], Ye [73], Yamashita [72], Gonzaga [26], etc.).

2. Affine methods (Dikin [18, 19], Barnes [9], Gonzaga [27], Vanderbei et al. [68], Monma and Morton [51], Monteiro et al. [53], Ye [74], etc.).

3. Path-following methods (Renegar [57], Gonzaga [25, 30, 31], Roos and Vial [59, 60], Den Hertog et al. [16], Monteiro and Adler [52], Kojima et al. [38], etc.).

In Sections 2, 3, and 4 we will shortly outline typical representatives of the three classes. Interior methods can also be classified according to the type of problem being solved as:

1. Primal methods, working on (1), (Karmarkar [35], Anstreicher [3], De Ghellinck and Vial [14], Ye [73, 74], Gonzaga [30, 31], Roos and Vial [59, 60], Den Hertog et al. [16], Dikin [18, 19], Barnes [9], Vanderbei et al. [68], etc.).

2. Dual methods, working on (2), (Yamashita [72], Gonzaga [25], Monma and Morton [51], Renegar [57], etc.).

3. Primal-dual methods (Todd and Ye [64], Monteiro et al. [53], Monteiro and Adler [52], Kojima et al. [38], etc.).

3. PROJECTIVE METHODS

The original version of Karmarkar's method [35] is the most important representative of the class of projective methods. Karmarkar's method is formulated for the problems in the following special form:

$$\begin{aligned} \min c^T x \\ Ax = 0, \quad e^T x = 1, \quad x \geq 0 \end{aligned} \tag{4}$$

where $A=[a_{ij}]_{m \times n}$, $\text{rank} A=m$, $x, c \in R^n$, $e=(1, \dots, 1) \in R^n$ and all the coefficients in A and c are integer. Let X be the feasible set of problem (4), i.e. $X = N(A) \cap S$, where S is the $(n-1)$ -dimensional simplex, $S = \{x \in R^n \mid e^T x = 1, x \geq 0\}$. It is assumed that $Ae=0$, i.e. the center of S defined by $a=(1/n)e$ is feasible, and that the optimal objective function value $f_{\text{opt}}=0$.

Karmarkar's algorithm can be described as the steepest descent method with the projective transformation at each iteration. At the iteration k the transformation is defined by

$$y = \frac{D^{-1}x}{e^T D^{-1}x}$$

where $D = \text{diag}(x_1^k, \dots, x_n^k)$ and $x^k > 0$ is a point generated at the iteration $k-1$. It is easy to see that the transformation maps S into itself, that the extreme points remain fixed and that x^k is mapped into a . Let \bar{X} be the image of X . Then $\bar{X} = N(\bar{A}) \cap S$, $\bar{A} = AD$. The inverse transformation is given by

$$x = \frac{Dy}{e^T Dy}$$

The next so-called potential function plays an important role in the analysis of Karmarkar's algorithm:

$$f: R_+^n \rightarrow R, \quad f(x) = n \ln(c^T x) - \sum_{i=1}^n \ln x_i$$

Let \bar{f} be the transformed potential function defined by $\bar{f}(y) = f(x)$. Then

$$\bar{f}(y) = n \ln\left(\frac{c^T Dy}{e^T Dy}\right) - \sum_{i=1}^n \ln \frac{x_i^k y_i}{e^T Dy} = n \ln \bar{c}^T y - \sum_{i=1}^n \ln y_i - \sum_{i=1}^n \ln x_i^k$$

where $\bar{c} = Dc$, i.e. functions \bar{f} and f are of the same form.

The k -th iteration of Karmarkar's algorithm can now be shortly outlined as follows:

- With the projective transformation $y = D^{-1}x / e^T D^{-1}x$ map the feasible set X onto \bar{X} ; the image of x^k is $a = (1/n)e$.
- Take one step of the projected gradient method for decreasing \bar{f} on \bar{X} , i.e. compute $\bar{y} = a - \lambda P \nabla \bar{f}(a) / \|P \nabla \bar{f}(a)\|$.
- Get x^{k+1} as the inverse image of \bar{y} .

The precise formulation of the method is:

Algorithm 2 (Karmarkar):

Step 0: Let $L > 0$ and $0 < \alpha < 1$ be given. Set $\lambda = \alpha / \sqrt{n(n-1)}$, $x^0 = a$, $k = 0$.

Step 1: Set $D = \text{diag}(x_1^k, \dots, x_n^k)$. Compute $\bar{A} = AD$, $\bar{c} = Dc$. Let

$$B = \begin{bmatrix} \bar{A} \\ e^T \end{bmatrix}_{(m+1) \times n}$$

be the matrix defining \bar{X} .

Step 2: Let $P = I - B^T(BB^T)^{-1}B$ be the projection matrix onto $N(B)$. Compute

$$h = P \nabla \bar{f}(a) / \|P \nabla \bar{f}(a)\| = P \bar{c} / \|P \bar{c}\|$$

Step 3: Set $\bar{y} = a - \lambda h$.

Step 4: Compute $x^{k+1} = D\bar{y} / e^T D\bar{y}$.

Step 5: Replace k by $k+1$. If $e^T x^k < 2^{-L}$ STOP, else go to Step 1. ■

Let us note that

$$BB^T = \begin{bmatrix} \bar{A} \\ e^T \end{bmatrix} \begin{bmatrix} \bar{A}^T e \end{bmatrix} = \begin{bmatrix} \bar{A} \bar{A}^T & 0 \\ 0 & n \end{bmatrix} \quad (BB^T)^{-1} = \begin{bmatrix} (\bar{A} \bar{A}^T)^{-1} & 0 \\ 0 & 1/n \end{bmatrix}$$

so that the main computational effort in one iteration of Algorithm 2 is the inversion of $\bar{A} \bar{A}^T = AD^2 A^T$. We shall see that this is the case with all interior point methods.

The next theorem shows that Algorithm 2 is polynomial:

Theorem 1 ([35]). Let r be the length of the input data in (4), let $L = r + \langle n \rangle$ and $\alpha \leq 1/3$. Then Algorithm 2 in $O(nL)$ iterations finds a point x^k satisfying $e^T x^k < 2^{-L}$. ■

Since the number of elementary arithmetic operations per iteration of Algorithm 2 is $O(n^3)$, it follows that the overall complexity of Algorithm 2 is $O(n^4 L)$. Karmarkar also shows that Algorithm 2 can be modified using partial updating so that the complexity bound is reduced to $O(n^{3.5} L)$. The rate of convergence of Karmarkar's algorithm is analysed by Ašić, Kovačević-Vujčić and Radosavljević-Nikolić in [7]. Under the assumption that $\bar{x} = (\bar{x}_1, \dots, \bar{x}_m, 0, \dots, 0)$, $\bar{x}_1 > 0, \dots, \bar{x}_m > 0$ is the unique solution of (4) it is shown that

$$\lim_{k \rightarrow \infty} \frac{e^T x^{k+1}}{e^T x^k} = \frac{\sqrt{(n-1)(n-m-1)(m+1)} - \alpha(m+1)}{\sqrt{(n-1)(n-m-1)(m+1)} + \alpha(n-m-1)}$$

Furthermore, Ašić et al. show in [8] that the normalized sequence of search directions in Karmarkar's method converges, i.e.

$$\lim_{k \rightarrow \infty} \frac{x^{k+1} - x^k}{\|x^{k+1} - x^k\|} = s,$$

where the components of s are uniquely determined by the reduced cost vector at the optimal vertex. Intuitively speaking, this means that the successive iterates tend to line up along a direction which points towards the solution. In [42] Kovačević-Vujčić uses this property in order to improve the rate of convergence of Karmarkar's method by taking long steps in promising directions. Namely, given a sequence (x^k) , $x^k \rightarrow \bar{x}$, generated by Karmarkar's method, let \bar{x}^k , $k \in N$, be defined as a point at which the ray $x^{k-1} + t(x^k - x^{k-1})$ intersects the boundary of X . Then

$$\lim_{k \rightarrow \infty} \frac{\|\bar{x}^k - \bar{x}\|}{\|x^k - \bar{x}\|} = 0,$$

i.e. the auxiliary sequence (\bar{x}^k) , generated at a low cost ($O(n)$ per step), converges faster than (x^k) .

In [24] Gill, Murray, Saunders, Tomlin and Wright show that Karmarkar's algorithm can be viewed as the Newton's method applied to the logarithmic barrier function. Todd and Burrell [63] were the first to show that the assumption of known optimal value in Karmarkar's method can be removed using a lower bound technique. A variant of Karmarkar's method which works with linear programming problems in standard form was developed by Gonzaga [26] and several other authors. Further analysis of Karmarkar's method and descriptions of some other related methods belonging to the same class can be found in [3, 12, 14, 20, 23, 28, 64, 72, 73, 76], etc.

4. AFFINE METHODS

Soon after Karmarkar's method was published several authors have come up with the idea to simplify it using affine rather than projective transformations. (Barnes [9], Vanderbei et al. [68], etc.). It has come out later that the idea with the affine transformation was used as early as 1967 in a paper by Dikin [18]. We shall outline here Dikin's method and its polynomial modification.

Consider the following linear programming problem:

$$\min c^T y \tag{5}$$

$$Ax = b, \quad x \geq 0$$

where $A = [a_{ij}]_{m \times n}$, $x, c \in R^n$, $b \in R^m$ and $\text{rank} A = m$. We shall assume that an x^0 such that $Ax^0 = b$, $x^0 > 0$ is known. At the k -th iteration we have a feasible point x^k , $x^k > 0$

which is used to define a linear transformation $y = D^{-1}x$, where, as in Karmarkar's method, $D = \text{diag}(x_1^k, \dots, x_n^k)$. The image of x^k is $e = (1, \dots, 1)$, and the inverse transformation is $x = Dy$. In the transformed space the problem is the following:

$$\min \bar{c}^T y \quad (6)$$

$$\bar{A}y = b \quad y \geq 0$$

where $\bar{c} = Dc$, $\bar{A} = AD$. The algorithm takes one step of the projected gradient method for decreasing the objective function of problem (6) starting from the point e , i.e. computes $\bar{y} = e - \lambda P\bar{c} / \|P\bar{c}\|$, and then maps it by the inverse transformation to get x^{k+1} . The details are given in:

Algorithm 3 (Dikin)

Step 0: Let λ and x^0 satisfying $0 < \lambda < 1$ and $x^0 > 0$, $Ax^0 = b$ be input parameters.
Set $k = 0$.

Step 1: Set $D = \text{diag}(x_1^k, \dots, x_n^k)$. Compute $\bar{A} = AD$, $\bar{c} = Dc$.

Step 2: Let $P = (I - \bar{A}^T(\bar{A}\bar{A}^T)^{-1}\bar{A})$ be the projection matrix onto $N(\bar{A})$. Compute $h = P\bar{c} / \|P\bar{c}\|$.

Step 3: Compute $\bar{y} = e - \lambda h$.

Step 4: Set $x^{k+1} = D\bar{y}$.

Step 5: Replace k by $k+1$ and go to Step 1. ■

It can be shown that for a suitable value of λ (e.g. $\lambda = 1/8$) the sequence generated by Algorithm 3 converges to an optimal solution \bar{x} (see e.g. [9, 18, 65]). As in the case of Karmarkar's algorithm it can be shown that the rate of convergence is linear and that the sequence of normalized search directions converges [8, 56]. The main result in [56] is that under the appropriate nondegeneracy assumptions we have

$$\lim_{k \rightarrow \infty} \frac{c^T x^{k+1} - c^T \bar{x}}{c^T x^k - c^T \bar{x}} = 1 - \frac{\lambda}{\sqrt{n-m}}$$

The procedure proposed in [42] can again be used to speed up the convergence.

Recently, Dikin [19] has proposed an interesting modification of the affine method. It is easy to see that Algorithm 3 can be reformulated as follows:

$$x^0 = e, \quad x^{k+1} = x^k - \lambda_k D_k^2 z^k, \quad \lambda_k = \lambda / ((x_1^k z_1^k)^2 + \dots + (x_n^k z_n^k)^2)^{1/2}$$

where now $D_k = \text{diag}(x_1^k, \dots, x_n^k)$, $z^k = c - A^T y^k$, $y^k = (AD_k^2 A^T)^{-1} AD_k^2 c$. The modified method differs from the original only in the choice of the step-size:

$$x^0 = e, \quad x^{k+1} = x^k - \lambda_k D_k^2 z^k, \quad \lambda_k = \lambda / \min(x_j^k, s_j^k)$$

and it has a nice property that without any nondegeneracy assumptions it can be shown that both sequences (x^k) and (y^k) converge to optimal solutions in the interior of the optimal faces of primal and dual problem, respectively.

Up to now it has remained unknown whether Algorithm 3 is polynomial or not, although the results obtained by Megiddo and Shub [48] indicate that a negative answer is to be expected. However, it is not hard to obtain different modifications that are polynomial. One of such modifications is proposed by Gonzaga [27]. Gonzaga assumes that the optimal objective function value is 0 and uses the following potential function:

$$f: R_+^n \rightarrow R, \quad f(x) = q \ln(c^T x) - \sum_{i=1}^n \ln x_i,$$

where $q > 0$. The linear transformation $y = D^{-1}x$, $D = \text{diag}(x_1^k, \dots, x_n^k)$, transforms the potential function into

$$\bar{f}(y) = q \ln(c^T D y) - \sum_{i=1}^n \ln x_i^k y_i = q \ln \bar{c}^T y - \sum_{i=1}^n \ln y_i - \sum_{i=1}^n \ln x_i^k$$

and it is easy to see that

$$\nabla \bar{f}(e) = \frac{q}{\bar{c}^T e} \bar{c} - e.$$

Algorithm 4 (Gonzaga)

Step 0: Let L and x^0 satisfying $L > 0$, $x^0 > 0$, $Ax^0 = b$ be input parameters.
Set $k=0$.

Step 1: Set $D = \text{diag}(x_1^k, \dots, x_n^k)$. Compute $\bar{A} = AD$, $\bar{c} = Dc$.

Step 2: Let $P = (I - \bar{A}^T (\bar{A} \bar{A}^T)^{-1} \bar{A})$ be the projection matrix onto $N(\bar{A})$.
Compute $h = P \nabla \bar{f}(\bar{e}) / \|P \nabla \bar{f}(\bar{e})\|$.

Step 3: Compute $\bar{y} = e - 0.3h$.

Step 4: Set $x^{k+1} = D\bar{y}$.

Step 5: Replace k by $k+1$. If $c^T x^k < 2^{-L}$ STOP. Otherwise go to Step 1. ■

The next theorem shows that Algorithm 4 is polynomial:

Theorem 2 ([27]). Suppose that problem (5) has integer coefficients and let r be the length of the input data and $L = r + \langle n \rangle$.

- (i) If $q = n$ then Algorithm 4 stops after $O(n^2L)$ iterations.
- (ii) If $q \geq n + \sqrt{n}$ then Algorithm 4 stops after $O(nL)$ iterations. ■

The number of elementary arithmetic operations per step of Algorithm 4 is $O(n^3)$, so that the overall complexity of finding \bar{x} by Algorithm 4 is $O(n^5L)$ for $q = n$, i.e. $O(n^4L)$ for $q \geq n + \sqrt{n}$. As in the case Karmakar's method, using partial updating, the later bound can be reduced to $O(n^{3.5}L)$. It should be noted that in Karmarkar's method we have $P\bar{c} / \|P\bar{c}\| = P\nabla\bar{f}(a) / \|P\nabla\bar{f}(a)\|$, which does not hold in the case of affine methods. Namely, Dikin's method uses the direction $P\bar{c}$, while Gonzaga's polynomial modification uses $P\nabla\bar{f}(e)$. Using a different potential function Ye [74] proposes an affine algorithm which finds a satisfactory approximate solution in $O(n^{0.5}L)$ iterations. The overall complexity of Ye's algorithm is $O(n^{3.5}L)$, i.e. $O(n^3L)$ if the partial updating is included. The bound $O(n^3L)$ is the best complexity bound obtained by now and it also holds for most of the path-following methods, which are the subject of the next section.

Affine methods have been studied by many other authors, e.g. Anstreicher [4], Freund [21], Anstreicher and Bosch [5], Monteiro et al. [53], Kojima et al. [39], etc.

5. PATH-FOLLOWING METHODS

The idea of the path following methods will be illustrated on the example of a simple dual method proposed by Gonzaga [25]. Let us consider the dual problem:

$$\begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & A^T y \leq c \end{aligned} \tag{7}$$

and denote by Y its feasible set. We shall assume that

$$\text{int } Y = \{y \in R^m \mid A^T y < c\} \neq \emptyset$$

and for the sake of simplicity that Y (instead of X) is compact. Let

$$f(y, t) = b^T y + t \sum_{i=1}^n \ln(c_i - A_i y)$$

be a barrier function associated with (7), where $t > 0$ is the penalty parameter and A_i , $i = 1, \dots, n$ are the rows of A^T . For a fixed t the function f is strictly concave and the problem

$$\begin{aligned} \max \quad & f(y, t) \\ \text{s.t.} \quad & A^T y < c \end{aligned} \tag{8}$$

has the unique solution $y(t)$. The trajectory $y(t)$, $t > 0$ is called the central path associated with (7). The trajectory has the important property that it converges to an optimal solution of (7) when $t \rightarrow 0$.

The classical barrier method generates a monotone decreasing sequence (t_k) , $t_k \rightarrow 0$ and the corresponding sequence $(y(t_k))$. In that way the constrained optimization problem is replaced by an infinite sequence of unconstrained problems. Although the computation of $y(t_k)$ is usually replaced by the computation of a suitable approximation y^k , since the number of steps needed to compute y^k is not known in advance we can roughly say that classical methods generate infinitely many infinite sequences. Path-following methods for linear programming are characterized not only by the fact that for each k a fixed (independent of k) number of steps is taken in order to obtain y^k , but it is also possible to estimate polynomially the overall complexity needed to produce a solution of (7). The simplest algorithm of this type is Gonzaga's method [25], where $p=1$, i.e. the penalty parameter is changed at each iteration.

Algorithm 5 (Gonzaga)

Step 0: Let $y^0 \in Y$ and $t_0 > 0$ be input parameters. Set $k=0$.

Step 2: Compute t_{k+1} :

$$t_{k+1} = \left(1 - \frac{1}{41\sqrt{n}}\right) t_k$$

Step 3: Compute y^{k+1} :

$$y^{k+1} = y^k + \nabla_{yy}^2 f(y^k, t_{k+1})^{-1} \nabla_y f(y^k, t_{k+1})$$

Step 3: Replace k by $k+1$ and go to Step 1. ■

Let us note that Gonzaga's method uses Newton steps for approximate maximization, which is the case with most of the path-following methods. It is easy to see that

$$\nabla_y f(y, t) = b - tAD^{-1}e \quad \nabla_{yy}^2 f(y, t) = -tAD^{-2}A^T$$

where $D = \text{diag}(c_1 - A_1 y, \dots, c_n - A_n y)$, so that the main computational effort per iteration of Algorithm 5 is the inversion of the positively definite matrix $AD^{-2}A^T$. The proof of polynomiality of Algorithm 5 can be obtained directly [25] or as a consequence of Kantorovich's theorem [58]. Namely, if f_{opt} is the optimal value of the objective function in (7) the following result holds:

Theorem 3. Let all the data in (7) be integer with the bit length r and let $L=r+\langle n \rangle$. If y^0 is such that

$$\|DA^T(y^0 - y(t_0))\| \leq 1/20 \quad (9)$$

where $D = \text{diag}(c_1 - A_1 y^0, \dots, c_n - A_n y^0)$, then Algorithm 5 in $O(n^{0.5}L)$ iterations generates a point y^k such that

$$f_{\text{opt}} - b^T y^k \leq 2^{-L}. \quad \blacksquare \quad (10)$$

It can be shown that a starting point y^0 satisfying (9) can be obtained either directly after a suitable reformulation of problem (7) (e.g. [47, 25]) or using so called centralization procedures. It also should be noted that y^k satisfying (10) can be transformed into an exact solution \bar{y} with $O(n^3)$ operations using algorithms related to Algorithm 1 (see e.g. [57]). If Algorithm 5 is modified to include a stopping rule and if y^0 satisfies the conditions of Theorem 3, it is easy to see that the overall complexity is $O(n^{3.5}L)$. The algorithm can be modified using the idea of partial updating so that the complexity bound is reduced to $O(n^3L)$ (see [25, 58]).

The central path can also be defined for the primal problem:

$$\begin{aligned} \min \quad & c^T x \\ \text{subject to} \quad & Ax = b \quad x \geq 0 \end{aligned} \quad (11)$$

Consider the logarithmic barrier function associated with (11):

$$g(x, t) = c^T x - t \sum_{i=1}^n \ln x_i, \quad x \in \text{ri } X$$

The function $g(x, t)$ is strictly convex and it is easy to see that

$$\nabla_x g(x, t) = c - tD_x^{-1}e \quad \nabla_{xx}^2 g(x, t) = tD_x^{-2}$$

where $D_x = \text{diag}(x_1, \dots, x_n)$. Under the assumption $\text{int } Y \neq \emptyset$ it follows that the problem

$$\begin{aligned} \min \quad & g(x, t) \\ \text{subject to} \quad & Ax = b, \quad x > 0 \end{aligned} \quad (12)$$

has the unique solution $x(t)$. The central path associated with (11) is defined as the trajectory $x(t)$, $t > 0$. There is an interesting relationship between central paths associated with primal and dual problems which can be obtained from the optimality conditions for (12). The necessary and sufficient optimality conditions have the form

$$c - tD_x^{-1}e = A^T y \quad Ax = b, \quad x > 0. \quad (13)$$

If $z = c - A^T y$ is the slack variable and $D_z = \text{diag}(z_1, \dots, z_n)$ it follows that $D_z e = z$ and (13) reduces to

$$\begin{aligned} D_x D_z e &= te \\ Ax &= b, \quad x > 0 \\ A^T y + z &= c, \quad z > 0 \end{aligned} \quad (14)$$

A point $x(t)$ is an optimal solution to (12) if and only if there exist $y(t)$ and $z(t)$ such that $x(t), y(t), z(t)$ is a solution to (14). It is easy to show that $y(t)$ is precisely the optimal solution to (8) (see [47]). Hence, we can say that the system (14) characterizes central paths of both primal and dual problems and for that reason the central path is often defined as the triple $x(t), y(t), z(t)$ satisfying (14) (see e.g. [71]). Many path-following methods operate directly on the system (14) or on some of its equivalent formulations. We shall illustrate it on the example of a primal-dual method which uses Newton's method for finding approximate solutions of (14).

Let (x^k, y^k, z^k) be an approximate solution of the system (14) for $t=t_k$ and let $(x^{k+1}, y^{k+1}, z^{k+1})$ be an approximate solution of the system:

$$\Phi(x, y, z) = \begin{bmatrix} D_x D_z e - t_{k+1} e \\ Ax - b \\ A^T y + z - c \end{bmatrix} = 0$$

obtained from (x^k, y^k, z^k) using one step of Newton's method. Let $p = (p_x, p_y, p_z)$, $p_x = x^{k+1} - x^k$, $p_y = y^{k+1} - y^k$, $p_z = z^{k+1} - z^k$ be the Newton's direction. As it is well known, p is the solution of the system

$$J(x^k, y^k, z^k)p = -\Phi(x^k, y^k, z^k)$$

where J is the Jacobian of Φ , i.e. p_x, p_y and p_z satisfy the system

$$J(x^k, y^k, z^k) \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} D_{z^k} & 0 & D_{x^k} \\ A & 0 & 0 \\ 0 & A^T & I \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} t_{k+1} e - D_{x^k} D_{z^k} e \\ b - Ax^k \\ c - A^T y^k - z^k \end{bmatrix}$$

where $D_{x^k} = \text{diag}(x_1^k, \dots, x_n^k)$, $D_{z^k} = \text{diag}(z_1^k, \dots, z_n^k)$.

There are obviously three sets of equations induced by the blocks of J . From the third set we obtain

$$p_z = -A^T p_y + c - A^T y^k - z^k,$$

and the first two sets of equations can be equivalently transformed into

$$\begin{bmatrix} D_{x^k}^{-1} D_{z^k} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} p_x \\ -p_y \end{bmatrix} = \begin{bmatrix} t_{k+1} D_{x^k}^{-1} e - c + A^T y^k \\ b - Ax^k \end{bmatrix}$$

From the first subsystem we have

$$p_x = D_{z^k}^{-1} D_{x^k} A^T p_y + D_{z^k}^{-1} D_{x^k} (t_{k+1} D_{x^k}^{-1} e - c + A^T y^k).$$

and p_x can be eliminated from the second subsystem, yielding

$$AD_z^{-1}D_xA^T p_y = AD_z^{-1}D_x(c - t_{k+1}D_x^{-1}c - A^T y^k) + b - Ax^k.$$

Computational effort for computing p_y is $O(n^2)$, while p_x and p_z can be obtained by a direct substitution.

Path-following methods are subject of investigation of many authors, e.g. Renegar [57], Kojima et al. [38], Monteiro and Adler [52], Roos and Vial [59, 60], Den Hertog et al. [16]. A unified approach to path-following methods is given e.g. in [58, 71], while the recent book by Den Hertog [17] gives a systematic analysis of short-, medium- and long-step path-following methods for linear, quadratic and convex programming. Analytical properties of the central path as well as the definition and properties of trajectories associated with projective and affine methods were investigated by Bayer and Lagarias [10,11], Witzgall et al. [70], Megiddo and Shub [48] and others.

6. SEARCH DIRECTIONS IN INTERIOR METHODS

Consider the primal-dual pair of problems (1) and (2) and suppose that $riX \neq \emptyset$, $intY \neq \emptyset$. As we have indicated, interior methods can be classified into primal, dual and primal-dual methods. Typically, a primal method generates $(x^k) \subset riX$, with the property $x^k \rightarrow x_{opt}$, a dual method generates $(y^k) \subset intY$ with the property $y^k \rightarrow y_{opt}$ while primal-dual methods generate simultaneously (x^k) , (y^k) and $(z^k) = (c - A^T y^k)$. All these methods can be viewed as iterative procedures which at each iteration make positive steps in certain search directions. Yamashita [72] was the first who noticed that search directions in different methods are linear combinations of the same two vectors, the so called affine scaling direction and centering direction. The investigation in that direction was continued by other authors (Gonzaga [28], Den Hertog and Roos [15], etc.). We shall shortly outline the results from [15].

Let p_{aff} and p_{cent} respectively denote the affine and the centering direction associated with the problem (1). For a given x and $D_x = \text{diag}(x_1^k, \dots, x_n^k)$ we define

$$p_{aff} = -D_x(I - D_x A^T (AD_x^2 A^T)^{-1} AD_x) D_x c.$$

The primal centering direction is defined as

$$p_{cent} = D_x(I - D_x A^T (AD_x^2 A^T)^{-1} AD_x) e.$$

It can be proved that p_{cent} is in fact Newton's direction for the problem of finding the analytical center of X . Roos and Den Hertog show that the search directions for a variety of primal methods and primal versions of dual methods (Gonzaga [25, 27, 28, 30, 31], De Ghellinck and Vial [14], Gay [23], Ye and Kojima [76], Ye [73, 74], Barnes [9], Dikin [18], Vanderbei et al. [68], Gill et al. [24], Roos and Vial [59, 60], Renegar [57], Den Hertog et al. [16], Freund [21], Anstreicher and Bosch [5], Iri and Imai [33] are given by

$$p_x = p_{\text{aff}} + \mu p_{\text{cent}},$$

where μ varies with the method.

The affine and the centering direction associated with the dual problem (2) at (y, z) are defined by

$$d_{\text{aff}} = (AD_z^{-2}A^T)^{-1}b, \quad d_{\text{cent}} = -(AD_z^{-2}A^T)^{-1}AD_z^{-1}e$$

where $D_z = \text{diag}(c_1 - A_1y, \dots, c_n - A_ny) = \text{diag}(z_1, \dots, z_n)$. It is shown in [15] that search directions in many dual methods as well as in dual versions of primal methods are of the form

$$p_y = d_{\text{aff}} + \mu d_{\text{cent}},$$

The values for μ are obtained in the following cases: Gonzaga [25, 27, 29, 30, 31], Yamashita [72], Barnes [9], Dikin [18], Vanderbei et al. [68], Gill et al. [24], Roos and Vial [59, 60], Renegar [57], Den Hertog et al. [16], Freund [21], Ye [74], Anstreicher and Bosch [5], Iri and Imai [33].

In primal-dual case let $D_{xz} = (D_z^{-1}D_x)^{1/2} = \text{diag}(\sqrt{x_1/z_1}, \dots, \sqrt{x_n/z_n})$ and let the directions at x and (y, z) be defined by

$$\begin{aligned} p_{\text{aff}}^* &= -D_{xz}(I - D_{xz}A^T(AD_{xz}^2A^T)^{-1}AD_{xz})D_{xz}c, & d_{\text{aff}}^* &= (AD_{xz}^2A^T)^{-1}b \\ p_{\text{cent}}^* &= D_{xz}(I - D_{xz}A^T(AD_{xz}^2A^T)^{-1}AD_{xz})D_{xz}D_x^{-1}e, & d_{\text{cent}}^* &= -(AD_{xz}^2A^T)^{-1}AD_z^{-1}e. \end{aligned}$$

It is shown that primal-dual methods move from x in the direction

$$p_x = p_{\text{aff}}^* + \mu p_{\text{cent}}^*$$

and from y in the direction

$$p_y = d_{\text{aff}}^* + \mu d_{\text{cent}}^*,$$

(which uniquely defines the direction at z). This result is obtained for the methods: Monteiro et al. [53], Kojima et al. [38, 39], Monteiro and Adler [52], Todd and Ye [64].

From the results in [15] it follows that interior point methods at the k -th iteration use a direction defined by the matrix $(AD_k^2A^T)^{-1}$, where D_k is equal to D_{x^k} , $D_{z^k}^{-1}$ or $(D_{z^k}^{-1}D_{x^k})^{1/2}$ and D_{x^k} , D_{z^k} are the diagonal matrices with x^k and z^k at the diagonal. Another interesting property of interior point methods is that the majority of them generate a solution sequence in which every limit point satisfies the strict complementarity condition, i.e. belongs to the relative interior of the optimal face (see [32]). It is easy to see now that if the optimal face $X^* = \{x \mid A_Bx_B = b, x_B \geq 0, x_N = 0\}$

is degenerate ($\text{rank } A_B < m$) then $\text{cond}(AD_k^2 A^T) \rightarrow \infty$, $k \rightarrow \infty$ and all three classes of interior point methods are faced with numerical stability problems, no matter which computational technique is used to find the direction. Stabilization procedures are analyzed by Ašić et al. [7] and Kovačević-Vujčić [43].

7. IMPLEMENTATION OF INTERIOR POINT METHODS

Practical implementation of interior point methods imposes many problems ranging from ill-conditioning to efficient data structures for handling large problems and it is a subject of many papers, e.g. [1, 2, 13, 36, 44].

One of the most important issues in the implementation of interior point methods is the strategy for computing the direction. Instead of using the projection formula indicated e.g. in Algorithms 2 and 3, a suitable linear system is solved. As it is well known, the projection of a vector h onto the subspace defined by $ADx = 0$ is the solution to the problem

$$\begin{aligned} \min \|h - x\|^2 \\ ADx = 0 \end{aligned}$$

In this case the Lagrange optimality conditions are given with

$$\begin{aligned} x + DA^T r &= h \\ ADx &= 0 \end{aligned} \tag{15}$$

or, equivalently, with

$$\begin{aligned} AD^2 A^T r &= ADh \\ x &= h - DA^T r \end{aligned}$$

The first set of equations is usually solved by a stable Cholesky factorization, although some authors recommend preconditioned conjugate gradients method, and x is then obtained by a direct substitution. On the other hand, some authors (see e.g. [6]) propose using the matrix structure of (15):

$$\begin{bmatrix} I & DA^T \\ AD & 0 \end{bmatrix} \begin{bmatrix} x \\ r \end{bmatrix} = \begin{bmatrix} h \\ 0 \end{bmatrix}$$

and choose the pivots so that the sparsity of the coefficient matrix is preserved, which is an important issue for large problems. Minimum fill in strategies were analyzed by Vanderbei [66], Vanderbei and Carpenter [67] and others.

Another important issue is the so-called early rounding. The idea is to apply Algorithm 1 or a similar procedure at early stages of an interior point algorithm, when a condition of the type (3) is not yet satisfied. Very often the obtained point turns out to be optimal, which can save a lot of computational effort (see e.g. [49, 61]).

An interesting point is also the comparison with the simplex method. We shall quote here a part of the results reported by Lustig, Marsten and Shanno [45] who compared the OSL program for the simplex method developed by IBM and the OB1 program for the primal-dual method developed by the authors. The comparison was performed on the IBM RISC System 1600, model 530. A series of standard problems (Table 1) was tested and the results are presented in Table 2. It is evident that with growing dimension the interior methods become more efficient than the simplex method, but that the simplex method stays concurrent. The general conclusion is that both approaches to linear programming will remain in use, with the recommendation to use interior methods at early stages of the iteration process and then switch to the simplex method.

Table 1: Test problems

NAME	#ROWS	#COLUMNS	#NONZEROS
MARKAL	5,831	7,689	59,435
PILOT87	2,030	4,883	73,152
MFMPTE16	6,507	13,610	100,523
NCM	3,352	16,298	128,057
TRIP	8,335	21,200	161,160
FORD	43,387	106,908	189,864
OILCO	18,800	38,540	219,880

Table 2: OSL(simplex) vs. OB1(interior)

NAME	OSL		OB1		OSL/OB1
	#ITER	TIME(sec)	#ITER	TIME(sec)	
MARKAL	16,041	664.22	41	132.03	5.03
PILOT87	11,578	1,459.42	42	621.15	2.35
MFMPTE16	27,257	2,017.97	30	133.76	15.09
NCM	8,938	309.06	44	69.14	4.47
TRIP	104,687	12,417.22	41	418.97	29.64
FORD	19,935	7,430.95	56	632.20	11.75
OILCO	95,857	19,889.51	64	3,688.76	5.39

A great scientific activity indicates that further progress in the area of interior point methods is to be expected. One of the open questions comes from the fact that interior methods, similarly as the simplex method, have much better practical performance than the theoretical bounds indicate (20-60 iterations, almost independent of the problem dimensions). Some attempts to explain this phenomenon are given in [62, 75], etc. The next important issue is how to use the information on the solution of a given program in order to solve a closely related program more effi-

ently. Namely, large programs are often being resolved after slight modification of the input data, which can be done efficiently with a few iterations of the simplex method. It is still an open question how to make interior point methods more efficient if a possibly infeasible "warm start" is given. Attempts in that direction were made e.g. by Freund [22]. An interesting issue is also the application of interior point methods to linear programming problems of the special structure and discrete optimization problems (see e.g. [46, 50, 69]). Finally we make a brief reference to infeasible interior-point algorithms for linear programming and linear complementarity problems which work with strictly positive but infeasible points (e.g. [77, 78]).

REFERENCES

- [1] Adler I., Karmarkar N., Resende M.G.C. and Veiga G., "An implementation of Karmarkar's algorithm for linear programming", *Mathematical Programming* 44 (1989) 297-335.
- [2] Adler, I., Karmarkar, N., Resende M.G.C. and Veiga, G., "Data structures and programming techniques for the implementation of Karmarkar's algorithm", *ORSA Journal on Computing* 1 (1989) 84-106.
- [3] Anstreicher, K.M., "A monotonic projective algorithm for fractional linear programming", *Algorithmica* 1 (1985) 483-498.
- [4] Anstreicher, K.M., "A combined phase I - phase II scaled potential algorithm for linear programming", *Mathematical Programming* 52 (1991) 429-440.
- [5] Anstreicher, K.M., and Bosch, R.A., "Long steps in an $O(n^3)$ algorithm for linear programming", *Mathematical Programming* 54 (1992) 251-267.
- [6] Arioli, M., Duff, I.S., and De Rijk, P.P.M., "On the augmented system approach to sparse least-squares problems", *Numerische Mathematik* 55 (1989) 667-684.
- [7] Ašić, M.D., Kovačević-Vujčić, V.V., and Radosavljević-Nikolić, M.D., "Asymptotic behaviour of Karmarkar's method for linear programming", *Mathematical Programming* 46 (1990) 173-190.
- [8] Ašić, M.D., Kovačević-Vujčić, V.V., and Radosavljević-Nikolić, M.D., "A note on limiting behaviour of the projective and the affine rescaling algorithms", *Contemporary Mathematics* 114 (1990) 151-157.
- [9] Barnes, E.R., "A variation on Karmarkar's algorithm for solving linear programming problems", *Mathematical Programming* 36 (1986) 174-182.
- [10] Bayer, D.A., and Lagarias, J.C., "The nonlinear geometry of linear programming: I, affine and projective scaling trajectories", *Trans. Amer. Math. Soc.* 314 (1989) 499-526.
- [11] Bayer, D.A., and Lagarias, J.C., "The nonlinear geometry of linear programming: II, Legendre transform coordinates and central trajectories", *Trans. Amer. Math. Soc.* 314 (1989) 527-581.
- [12] Bloom, L., "Towards an asymptotic analysis of Karmarkar's algorithm", *Information Processing Letters* 23 (1986) 189-194.
- [13] Choi, I.C., Monma, C.L., and Shanno, D.F., "Further development of a primal-dual interior point method", *ORSA Journal on Computing* 2 (1990) 304-311.
- [14] De Ghellinck, D., and Vial, J.Ph., "A polynomial Newton method for linear programming", *Algorithmica* 1 (1986) 425-453.
- [15] Den Hertog, D., Ross, C., De Ghellinck, G. and Vial, J.Ph., "A survey of search directions in interior point methods for linear programming", *Mathematical Programming* 52, (1991) 481-510.
- [16] Den Hertog, D., Ross, C. and Terlaky, T., "A potential reduction variant of Renegar's short-step path-following method for linear programming", *Linear Algebra and its Applications* 152 (1991) 43-68.

- [17] Den Hertog, D., *Interior Point Approach to Linear, Quadratic and Convex Programming*, Kluwer Academic Publishers, Dordrecht, 1994.
- [18] Dikin, I.I., "Iterative solution of problems of linear and quadratic programming", *Doklady Akademii Nauk SSSR* 174 (1967) 747-748.
- [19] Dikin, I.I., "The convergence of dual variables", *Russian Journal of Operations Research* 1 (1994).
- [20] Franklin, J., "Convergence in Karmarkar's algorithm for linear programming", *SIAM J. Numer. Anal.* 24, 1987, 928-944.
- [21] Freund, R.M., "Polynomial-time algorithms for linear programming based only on primal scaling and projected gradients of a potential function", *Mathematical Programming* 51 (1991) 203-222.
- [22] Freund, R.M., "A potential-function reduction algorithm for solving linear program directly from an infeasible 'warm start'", *Mathematical Programming* 52 (1991) 441-466.
- [23] Gay, D.M., "A variant of Karmarkar's linear programming algorithm for problems in standard form", *Mathematical Programming* 37 (1987) 81-90.
- [24] Gill, P.E., Murray, W., Saunders, M.A., Tomlin, J.A. and Wright, M.H., "On projected Newton barrier methods for linear programming and an equivalence to Karmarkar projective method", *Mathematical Programming* 36 (1986) 183-209.
- [25] Gonzaga, C.C., "An algorithm for solving linear programming problems in $O(n^3L)$ operations", in: N. Megiddo (ed.), *Progress in Mathematical Programming, Interior Point and Related Methods*, Springer, New York, 1988, 1-28.
- [26] Gonzaga, C.C., "Conical projection algorithms for linear programming", *Mathematical Programming* 43 (1989) 151-173.
- [27] Gonzaga, C.C., "Polynomial affine algorithms for linear programming", *Mathematical Programming* 49 (1990) 7-21.
- [28] Gonzaga, C.C., "Search directions for interior linear programming methods", *Algorithmica* 6 (1991) 153-181.
- [29] Gonzaga, C.C., "Interior point algorithms for linear programming with inequality constraints", *Mathematical Programming* 52 (1991) 209-225.
- [30] Gonzaga, C.C., "Large-step path-following methods for linear programming, Part I: barrier function method", *SIAM Journal on Optimization* 1 (1991) 268-279.
- [31] Gonzaga, C.C., "Large-step path-following methods for linear programming, Part II: potential reduction method", *SIAM Journal on Optimization* 1 (1991) 280-292.
- [32] Guler, O. and Ye, Y., "Convergence behavior of interior-point algorithms", *Mathematical Programming* 60 (1993) 215-228.
- [33] Iri, M. and Imai, H., "A multiplicative method for linear programming", *Algorithmica* 1 (1986) 455-482.
- [34] Jansen, B., Roos, C. and Terlaky, T., "An interior point approach to postoptimal and parametric analysis in linear programming", Report No. 92-21, Faculty of Mathematics and Informatics, Delft University of Technology, Delft, 1992.
- [35] Karmarkar, N., "A new polynomial-time algorithm for linear programming", *Combinatorica* 4 (1984) 373-395.
- [36] Karmarkar, N. and Ramakrishnan, K., "Computational results of an interior point algorithm for large scale linear programming", *Mathematical Programming* 52 (1991) 555-586.
- [37] Khachian, L.G., "A polynomial algorithm in linear programming", *Doklady Akademii Nauk SSSR* 244 (1979) 1093-1096.
- [38] Kojima, M., Mizuno, S. and Yoshise, A., "A primal-dual interior point algorithm for linear programming", in: N. Megiddo, (ed.), *Progress in Mathematical Programming, Interior Point and Related Methods*, Springer, New York, 1988, 29-47.
- [39] Kojima, M., Mizuno, S. and Yoshise, A., "An $O(n^{0.5}L)$ iteration potential reduction algorithm for linear complementarity problems", *Mathematical Programming* 50 (1991) 331-342.
- [40] Kortanek, K. and Jishan, Z., "New purification algorithms for linear programming", *Naval Research Logistic* 35 (1988) 571-583.

- [41] Kranich, E., "Interior point methods for mathematical programming: A bibliography", Diskussionsbeitrag Nr. 171, Fern Universität Hagen, Hagen, 1991, (An updated version available on Netlib).
- [42] Kovačević-Vujčić, V.V., "Improving the rate of convergence of interior point methods for linear programming", *Mathematical Programming* 52, (1991) 467-480.
- [43] Kovačević-Vujčić, V.V., "Stabilization of path-following interior point methods for linear programming", in: D. Herceg and I.J. Cvetković (eds.), *Proceedings of IX Yugoslav Conference on Applied Mathematics PRIM'94*, University of Novi Sad, Novi Sad, 1995, 47-53.
- [44] Lustig, I.J., Marsten, R.E. and Shanno, D.F., "Computational experience with a primal-dual interior point method for linear programming", *Linear Algebra and its Applications* 152 (1991) 191-222.
- [45] Lustig, I.J., Marsten, R.E. and Shanno, D.F., "Interior method vs simplex method: Beyond NETLIB", *COAL Newsletter* 19 (1991) 41-44.
- [46] Masuzawa, K., Mizuno, S. and Mori, M., "A polynomial time interior point algorithm for minimum cost flow problems", *Journal of Operations Research Society of Japan* 33 (1990) 157-167.
- [47] Megiddo, N., "Pathways to the optimal set in linear programming", in: N. Megiddo, (ed.), *Progress in Mathematical Programming*, Springer, New York, 1987, 131-158.
- [48] Megiddo, N. and Shub, M., "Boundary behaviour of interior point algorithms in linear programming", *Mathematics of Operations Research* 14 (1989) 97-146.
- [49] Mehrotra, S. and Ye, Y., "Finding an interior point in the optimal face of linear programs", *Mathematical Programming* 62 (1993) 497-516.
- [50] Mitchell, J.E. and Todd, M.J., "Solving combinatorial optimization problems using Karmarkar's algorithm", *Mathematical Programming* 56 (1992) 245-284.
- [51] Monma, C.L. and Morton, J., "Computational experience with a dual affine variant of Karmarkar's method for linear programming", *Operations Research Letters* 6 (1987) 261-267.
- [52] Monteiro, R.C. and Adler, I., "Interior path-following primal-dual algorithm. Part I: Linear Programming", *Mathematical Programming* 44 (1989) 27-41.
- [53] Monteiro, R.D.C., Adler, I. and Resende, M.G.C., "A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension", *Mathematics of Operations Research* 15 (1990) 191-214.
- [54] Nemhauser, G.L. and Wolsey, L.A., *Integer and Combinatorial Optimization*, J.Wiley & Sons, London, 1988.
- [55] Nesterov, Y. and Nemirovskii, A., *Interior Point Polynomial Methods in Convex Programming*, SIAM, Philadelphia, 1993.
- [56] Radosavljević-Nikolić, M.D., "Asymptotic behaviour of affine methods for linear programming" (in Serbian), in: J. Petrić, S. Krčevinac, V. Kovačević-Vujčić and N. Mladenović (eds.), *Proceedings of XVII Yugoslav Symposium on Operations Research SYM-OP-IS'90*, Naučna Knjiga, Beograd, 1990, 713-720.
- [57] Renegar, J., "A polynomial-time algorithm, based on Newton's method, for linear programming", *Mathematical Programming* 40 (1988) 59-93.
- [58] Renegar, J. and Shub, M., "Unified complexity analysis for Newton LP methods", *Mathematical Programming* 53 (1992) 1-16.
- [59] Roos, C. and Vial, J.Ph., "Long steps with the logarithmic penalty barrier function in linear programming", in: J. Gabszewicz, J.F. Richard and L. Wolsey, (eds.), *Economic Decision-making: Games, Economics and Optimisation. Contributions in Honour of Jacques H. Dreze*, Elsevier Science Publisher, Amsterdam, 1990, 433-441.
- [60] Roos, C. and Vial, J.Ph., "A polynomial method of approximate centers for linear programming", *Mathematical Programming* 54 (1992) 295-306.
- [61] Tapia, R. and Zhang, Y., "An optimal basis identification technique for interior linear programming algorithms", *Linear Algebra and Its Applications* 152 (1991).

- [62] Todd, M.J., "Anticipated behaviour of Karmarkar's algorithm", Technical Report 879, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY, 1989.
- [63] Todd, M.J. and Burrell, B.P., "An extension of Karmarkar's algorithm for linear programming using dual variables", *Algorithmica* 1 (1986) 409-424.
- [64] Todd, M.J. and Ye, Y., "A centered projective algorithm for linear programming", *Mathematics of Operations Research* 15 (1990) 508-529.
- [65] Tsuchiya, T., "Global convergence of the affine scaling methods for degenerate linear programming problems", *Mathematical Programming* 52 (1991) 377-404.
- [66] Vanderbei, R.J., "Splitting dense columns in sparse linear systems", *Linear algebra and its applications* 152 (1991) 107-117.
- [67] Vanderbei, R.J. and Carpenter, T., "Symmetric indefinite systems for interior point methods", *Mathematical Programming* 58 (1993) 1-32.
- [68] Vanderbei, R.J., Meketon, M.S. and Freedman, B.A., "A modification of Karmarkar's linear programming algorithm", *Algorithmica* 1 (1986) 395-407.
- [69] Wallacher, C. and Zimmermann, U., "A combinatorial interior point method for network flow problems", *Mathematical Programming* 56 (1992) 321-336.
- [70] Witzgall, C., Boggs, P. and Domich, P., "On the convergence behaviour of trajectories for linear programming", *Contemporary Mathematics* 114 (1990) 161-187.
- [71] Wright, M., "Interior methods for constrained optimization", *Acta Numerica* 1 (1992), 341-407.
- [72] Yamashita, H., "A polynomially and quadratically convergent method for linear programming", Mathematical Systems Institute, Tokyo, Japan, 1986.
- [73] Ye, Y., "A class of projective transformations for linear programming", *SIAM Journal on Computing* 19, (1991), 457-466.
- [74] Ye, Y., "An $O(n^3L)$ potential reduction algorithm for linear programming", *Mathematical Programming* 50 (1991) 239-258.
- [75] Ye, Y., "Comparative analysis of affine scaling algorithms based on simplifying assumptions", *Mathematical Programming* 52 (1991) 405-414.
- [76] Ye, Y. and Kojima, M., "Recovering optimal basis in Karmarkar's polynomial algorithm for linear programming", *Mathematical Programming* 39 (1987) 305-317.
- [77] Zhang, Y., "On the convergence of a class of infeasible interior-point algorithms for the horizontal linear complementarity problem", *SIAM Journal on Optimization* 4, (1994), 208-227.
- [78] Zhang, Y. and Zhang, D., "Superlinear convergence of infeasible-interior-point methods for linear programming", *Mathematical Programming* 66, (1994), 361-337.