October 13, 1995. Revised November 1996.

AN INTERIOR POINT METHOD, BASED ON RANK-ONE UPDATES, FOR LINEAR PROGRAMMING

Jos F. Sturm[1] and Shuzhong Zhang[2]

Report 9546/A, Econometric Institute
Erasmus University Rotterdam
P.O. Box 1738, 3000 DR Rotterdam, The Netherlands

ABSTRACT

We propose a polynomial time primal-dual potential reduction algorithm for linear programming. The algorithm generates sequences d^k and v^k rather than a primal-dual interior point (x^k, s^k), where $d_i^k = \sqrt{x_i^k/s_i^k}$ and $v_i^k = \sqrt{x_i^k s_i^k}$ for $i = 1, 2, \ldots, n$. Only one element of d^k is changed in each iteration, so that the work per iteration is bounded by $O(mn)$ using rank-one updating techniques. The usual primal-dual iterates x^k and s^k are not needed explicitly in the algorithm, whereas d^k and v^k are iterated so that the interior primal-dual solutions can always be recovered by afore mentioned relations between (x^k, s^k) and (d^k, v^k) with improving primal-dual potential function values. Moreover, no approximation of d^k is needed in the computation of projection directions.

Key words: linear programming, interior point method, potential function.

[1] Econometric Institute, Erasmus University Rotterdam, The Netherlands, sturm@few.eur.nl.

[2] Econometric Institute, Erasmus University Rotterdam, The Netherlands, zhang@few.eur.nl

1. Introduction

In his seminal paper [8], Karmarkar proposed a projective potential reduction method that solves linear programs in $O(nL)$ main iterations, where n stands for the number of inequality constraints, and L is the input length of the problem. A single iteration of Karmarkar's algorithm requires solving a system of linear equations, which takes $O(n^3)$ operations in the worst case. By using incomplete updates of this equation system, the average work per step can be reduced to $O(n^{2.5})$. Hence, Karmarkar's algorithm achieves an overall complexity of $O(n^{3.5}L)$ for linear programming.

The $O(n^{3.5}L)$ complexity was further reduced to $O(n^3L)$ by Vaidya [17] and Gonzaga [5]. Kojima, Mizuno and Yoshise [9] and Monteiro and Adler [13] showed that this complexity can also be achieved by the primal-dual path-following method, if partial updates are used. However, the algorithms proposed by [8, 17, 5, 9, 13] use a fixed step length per iteration, which makes the algorithms inefficient for practical purposes. Adaptive step algorithms that are based on a partial updating scheme were proposed by Anstreicher [1], Anstreicher and Bosch [2], Den Hertog [6], Den Hertog, Roos and Vial [7], among others. The afore mentioned partial updating methods require $O(n^{2.5})$ operations on average per iteration. Algorithms that require on average only $O(n^2)$ operations per iteration were proposed by Ye [19] and Mizuno [11]. Interestingly, Mizuno [12] also obtained $O(n^3L)$ complexity with a potential reduction algorithm that requires *at most* $O(n^2)$ operations per iteration, as opposed to [19, 11] where each iteration requires $O(n^2)$ operations *on average*. At each iteration of Mizuno's method [12], the amount of required operations is thus comparable to that of one simplex pivot step. A variant of Mizuno's algorithm that allows for up to any fixed number of updates is analysed by Bosch [3].

The algorithm to be proposed in this paper is similar to Mizuno's algorithm [12] in the sense that it requires only $O(n^2)$ operations per main iteration. However, the way in which the new algorithm computes the iterative primal and dual solutions is different. In general, the major computational efforts needed in interior point methods for linear programming are related to the computation of projections. Denote x^k and s^k to be the primal-dual iterates at iteration k. We scale the linear program by a positive vector d^k, and then determine a vector v^k as the unique intersection point of the scaled primal and dual feasible sets. For primal methods, one needs to project onto the null space of AX^k, where $X^k = \mathrm{diag}(x_1^k, x_2^k, ..., x_n^k)$. For dual methods, the projection onto the range space of $(S^k)^{-1}A^{\mathrm{T}}$ is needed with $S^k = \mathrm{diag}(s_1^k, s_2^k, ..., s_n^k)$. Finally, for primal-dual methods, one has to project onto the null space of AD^k with $D^k = \mathrm{diag}(d_1^k, d_2^k, ..., d_n^k)$. Previous partial updating methods work with approximations of either X^k, S^k or D^k, which are updated in such a way that the computational costs are reduced. In our new method, the scaling vector d^k is computed exactly as $(X^k)^{1/2}(S^k)^{-1/2}e$. However, we change only one component of d^k at a time. As a result, only $O(n^2)$ arithmetic operations is needed using a rank-one updating scheme for the projection matrices. In this procedure, no approximation is involved. Another feature of our new algorithm is that the traditional primal and dual iterates x^k and s^k are invisible in the solution procedure. Their functions are completely replaced by the new vectors d^k and v^k. Unfortunately, our algorithm seems to be less effective in reducing the potential function value than Mizuno's algorithm. To be more precise, we arrive at an $O(n^{1.5}L)$ worst case iteration bound. Hence, the new algorithm needs $O(n^{3.5}L)$ operations in total, just like Karmarkar's algorithm.

The paper is organized as follows. In Section 2, we briefly discuss the primal-dual geometry of linear programming. The effect of a rank-1 update on the primal-dual solution pair is derived in Section 3. Subsequently, the new potential reduction algorithm is stated in Section 4, and we proceed in Section 5 by deriving the complexity result. We conclude the paper in Section 6.

2. Problem and discussions

In this paper we consider the standard linear program:

$$(P) \quad \min_x \{c^{\mathrm{T}}x \mid Ax = b, x \geq 0\}$$

and its dual

$$(D) \quad \max_{y,s} \{b^{\mathrm{T}}y \mid A^{\mathrm{T}}y + s = c, s \geq 0\}$$

where A is an $m \times n$ matrix, $n \geq 2$, and b and c are vectors of dimension m and n respectively.

We make the standard assumptions that A has full row rank and that (P) and (D) have interior feasible solutions.

For given positive vector $d \in \Re^n_{++}$, let

$$\mathcal{F}_P(d) := \{x \mid ADx = b\}$$

where

$$D := \mathrm{diag}(d_1, d_2, \ldots, d_n)$$

and let

$$\mathcal{F}_D(d) := \{s \mid DA^{\mathrm{T}}y + s = Dc \text{ for some } y \in \Re^m\}.$$

Denoting the all-one vector by e, it is easily seen that primal feasibility amounts to

$$x \in \mathcal{F}_P(e) \text{ and } x \geq 0$$

and dual feasibility is equivalent to

$$s \in \mathcal{F}_D(e) \text{ and } s \geq 0.$$

As $\mathcal{F}_P(d)$ and $\mathcal{F}_D(d)$ are orthogonal complementary affine spaces, they share a single intersection point, say v_d:

$$\{v_d\} = \mathcal{F}_P(d) \cap \mathcal{F}_D(d).$$

Let

$$x_d := Dv_d$$

and

$$s_d := D^{-1}v_d.$$

Obviously, $x_d \in \mathcal{F}_P(e)$ and $s_d \in \mathcal{F}_D(e)$. Because of the positivity of d, we also have

$$x_d \geq 0 \text{ and } s_d \geq 0$$

if and only if

$$v_d \geq 0.$$

A natural question arises: Is v_d always nonnegative for positive d? Or equivalently: Are x_d and s_d always feasible? As expected, the answer is negative. A simple example is:

Example 2.1. *Let* $A = \begin{bmatrix} 1 & 2 \end{bmatrix}, b = \begin{bmatrix} 4 \end{bmatrix}$ *and* $c^T = \begin{bmatrix} 0 & 7 \end{bmatrix}$. *Then, choosing* $d^T = \begin{bmatrix} 1 & 1 \end{bmatrix}$ *yields*

$$v_d = \begin{bmatrix} -2 \\ 3 \end{bmatrix},$$

which has a negative component.

If, on the other hand, interior feasible solutions $x \in \mathcal{F}_P(e)$, $x > 0$ and $s \in \mathcal{F}_D(e)$, $s > 0$ are known, then it is easy to construct d such that $v_d > 0$. Namely, letting

$$d_i = \sqrt{x_i/s_i} \text{ for } i = 1, 2, \ldots, n$$

yields

$$(v_d)_i = \sqrt{x_i s_i} \text{ for } i = 1, 2, \ldots, n.$$

This triggers the next question: Assuming that we have a pair of interior feasible solutions, is it possible to update the related vector d such that the feasibility is retained and, at the same time, the solutions are improved?

The goal of this paper is to give an affirmative answer to the above question. Namely, we propose an algorithm that generates a sequence $\{d^k \mid k = 0, 1, 2, \cdots\}$ for which the associated sequence of primal-dual feasible pairs (x_{d^k}, s_{d^k}) converges to a pair of optimal solutions for (P) and (D).

First we derive formulas to express v_d, x_d and s_d for a given vector $d \in \Re^n_{++}$. Let Q_d and P_d denote the projection matrices onto the image of DA^{T} and the kernel of AD respectively, i.e.

$$Q_d := DA^{\mathrm{T}}(AD^2A^{\mathrm{T}})^{-1}AD$$

and

$$P_d := I - Q_d.$$

By definition, $v_d \in \mathcal{F}_P(d) \cap \mathcal{F}_D(d)$, i.e.

$$ADv_d = b \tag{1}$$

and

$$DA^{\mathrm{T}}y_d + v_d = Dc, \tag{2}$$

for some $y_d \in \Re^m$. Pre-multiplying (1) with $DA^{\mathrm{T}}(AD^2A^{\mathrm{T}})^{-1}$ yields

$$Q_dv_d = DA^{\mathrm{T}}(AD^2A^{\mathrm{T}})^{-1}b$$

and pre-multiplying (2) with P_d yields

$$P_dv_d = P_dDc.$$

Hence,

$$v_d = Q_dv_d + P_dv_d = DA^{\mathrm{T}}(AD^2A^{\mathrm{T}})^{-1}b + (I - DA^{\mathrm{T}}(AD^2A^{\mathrm{T}})^{-1}AD)Dc,$$

which satisfies (1) and (2) with

$$y_d = (AD^2A^{\mathrm{T}})^{-1}(AD^2c - b). \tag{3}$$

3. Single component update

In this section, we analyze the effect of a change in a single component of d on the associated pair (x_d, s_d). Suppose that we update component d_j, for some $j \in \{1, 2, \cdots, n\}$. For a step size $t > -1$, let the new vector of scalings $d(t)$ be defined as follows:

$$d_j(t) = \sqrt{1+t}\, d_j$$

and

$$d_i(t) = d_i \text{ for } i \in \{1, 2, \cdots, n\} \backslash \{j\},$$

i.e. the components other than j remain unchanged.

We define

$$D(t) := \operatorname{diag}(d_1(t), d_2(t), \ldots, d_n(t)).$$

Let

$$u_d := (AD^2A^{\mathrm{T}})^{-1}ADe_j,$$

where e_j is the j-th unit vector. Since

$$AD(t)^2A^{\mathrm{T}} = AD^2A^{\mathrm{T}} + t(ADe_j)(ADe_j)^{\mathrm{T}}$$

and

$$(ADe_j)^{\mathrm{T}}(AD^2A^{\mathrm{T}})^{-1}(ADe_j) = e_j^{\mathrm{T}}Q_de_j = e_j^{\mathrm{T}}Q_d^{\mathrm{T}}Q_de_j = \|Q_de_j\|^2,$$

using the rank-1 updating formula of Sherman, Morrison and Woodbury, we obtain

$$(AD(t)^2A^{\mathrm{T}})^{-1} = (AD^2A^{\mathrm{T}})^{-1} - \frac{t}{1 + t\|Q_de_j\|^2}u_du_d^{\mathrm{T}}.$$

Furthermore,

$$AD(t)^2c = AD^2c + t(d_jc_j)ADe_j.$$

It follows using (3) that

$$\begin{aligned} y_{d(t)} &= (AD(t)^2A^{\mathrm{T}})^{-1}(AD(t)^2c - b) \\ &= y_d + t(d_jc_j)u_d \\ &\quad -t\frac{t(d_jc_j)\|Q_de_j\|^2 + u_d^{\mathrm{T}}(AD^2c - b)}{1 + t\|Q_de_j\|^2}u_d. \end{aligned}$$

From (3) and (2) we have

$$u_d^{\mathrm{T}}(AD^2c - b) = e_j^{\mathrm{T}}DA^{\mathrm{T}}y_d = d_jc_j - (v_d)_j,$$

and therefore

$$y_{d(t)} = y_d + \frac{t(v_d)_j}{1 + t\|Q_de_j\|^2}u_d.$$

Now it follows that

$$\begin{aligned} Ds_{d(t)} &= Dc - DA^{\mathrm{T}}y_{d(t)} \\ &= v_d - \frac{t}{1 + t\|Q_de_j\|^2}(v_d)_jQ_de_j. \end{aligned} \tag{4}$$

As $x_{d(t)} = D(t)v(t) = D(t)^2 s_{d(t)}$, we have

$$\begin{aligned} D^{-1}x_{d(t)} &= (D^{-2}D(t)^2)Ds_{d(t)} \\ &= (I + te_je_j^{\mathrm{T}})Ds_{d(t)} \\ &= v_d - \frac{t}{1+t\,\|Q_de_j\|^2}(v_d)_jQ_de_j \\ &\quad +t(v_d)_je_j - \frac{t^2\,\|Q_de_j\|^2}{1+t\,\|Q_de_j\|^2}(v_d)_je_j \\ &= v_d + \frac{t}{1+t\,\|Q_de_j\|^2}(v_d)_jP_de_j. \end{aligned} \tag{5}$$

We introduce $\alpha(t)$,

$$\alpha(t) := \frac{t}{1+t\,\|Q_de_j\|^2},$$

which is a strictly increasing function for $t > -1$. This function maps the interval $(-1,\infty)$ into the interval

$$\{\alpha \mid \|P_de_j\|^2\,\alpha > -1 \text{ and } \|Q_de_j\|^2\,\alpha < 1\}.$$

The inverse transformation of $\alpha(t)$ is

$$t(\alpha) = \frac{\alpha}{1-\alpha\,\|Q_de_j\|^2}. \tag{6}$$

From now on, we consider α as the step size parameter, and we compute $t(\alpha)$ from (6). As a matter of notation, we write

$$x(\alpha) := x_{d(t(\alpha))}, s(\alpha) := s_{d(t(\alpha))}$$

and

$$v(\alpha) := v_{d(t(\alpha))}.$$

If no confusion is possible, we will dispense with the argument d and simply write P, Q, v, x, s and y to denote P_d, Q_d, v_d, x_d, s_d and y_d. Now we rewrite (5) and (4) as follows:

$$D^{-1}x(\alpha) = v + \alpha v_jPe_j \tag{7}$$

and

$$Ds(\alpha) = v - \alpha v_jQe_j. \tag{8}$$

These relations imply that

$$\begin{aligned} \|v(\alpha)\|^2 &= x(\alpha)^{\mathrm{T}}s(\alpha) \\ &= (v + \alpha v_jPe_j)^{\mathrm{T}}(v - \alpha v_jQe_j) \\ &= \|v\|^2 + \alpha v_je_j^{\mathrm{T}}(P-Q)v. \end{aligned} \tag{9}$$

Hence, the duality gap between $x(\alpha)$ and $s(\alpha)$ is linear in α.

As is common in the interior point methodology, we restrict ourselves to positive, i.e. interior, solutions $x > 0$ and $s > 0$. Introducing the quantity

$$v_{\min} := \min\{v_1, v_2, \cdots, v_n\},$$

we obtain from (7) and (8) that

$$x(\alpha) > 0 \text{ and } s(\alpha) > 0$$

as long as $\mid \alpha \mid < v_{\min}/v_j$.

4. A potential reduction algorithm

Consider for $x > 0$ and $s > 0$ the Tanabe-Todd-Ye primal-dual potential function [16, 18]

$$\phi(x, s) = (n + l)\log(x^{\mathrm{T}}s) - \sum_{i=1}^{n}\log(x_i s_i) - n\log n,$$

where we let $l := \sqrt{2n}$ in this paper.

From the arithmetic-geometric mean inequality, it is easily seen that

$$\phi(x, s) \geq \sqrt{2n}\log(x^{\mathrm{T}}s),$$

and equality holds only if $v = X^{1/2}S^{1/2}e$ is a positive multiple of the all-one vector e. This observation is the essence of the proof of the following lemma, which is due to Todd and Ye [16] and Ye [18].

Lemma 4.1. *If an algorithm reduces $\phi(x, s)$ by at least a fixed quantity $\delta > 0$ in each iteration, then the algorithm can be used to solve the pair (P), (D) in $O(\sqrt{n}L/\delta)$ iterations.*

Let

$$p_{ij} := e_i^{\mathrm{T}}Pe_j \text{ for } i = 1, 2, \ldots, n$$

and

$$q_{ij} := e_i^{\mathrm{T}}Qe_j \text{ for } i = 1, 2, \ldots, n.$$

For $\mid \alpha \mid < v_{\min}/v_j$, we have from (7), (8) and (9) that

$$\begin{aligned}\phi(x(\alpha), s(\alpha)) \quad = \quad & \phi(x, s) + (n + \sqrt{2n})\log(1 + \alpha\frac{v_j e_j^{\mathrm{T}}(P - Q)v}{\|v\|^2}) \\ & - \sum_{i=1}^{n}\log(1 + \frac{\alpha v_j}{v_i}p_{ij}) - \sum_{i=1}^{n}\log(1 - \frac{\alpha v_j}{v_i}q_{ij}).\end{aligned} \tag{10}$$

Let

$$r := (P - Q)(\frac{(n+\sqrt{2n})}{\|v\|^2}v - V^{-1}e), \tag{11}$$

where

$$V := \mathrm{diag}(v_1, v_2, \ldots, v_n).$$

It is easily seen from (10) that

$$\phi(x(\alpha), s(\alpha)) = \phi(x, s) + \alpha v_j r_j + o(\alpha).$$

We propose to select the index j such that $\mid r_j \mid = \|r\|_\infty$. This choice results in the following algorithm.

Algorithm 1. *(A,b,c,d^0)*
The initial scaling $d^0 > 0$ is chosen such that $v_{d^0} > 0$.

Step 0 Initialization. *Set $k = 0$.*

Step 1 Optimality test. *Let $d = d^k$. Stop if based on (x_d, s_d) an optimal pair (x^*, s^*) can be found.*

Step 2 Choose the updating index. *Compute r according to (11). Set*

$$j = \arg \max_{1 \le i \le n} \mid r_i \mid$$

Step 3 Step size choice. *Compute α such that*

$$\phi(x_d(\alpha), s_d(\alpha)) < \phi(x_d, s_d) - \frac{1}{6(n+2)}$$

Step 4 Take step. *Set $d^{k+1} = d + (\sqrt{1 + t(\alpha)} - 1)d_j e_j$.*

Step 5 *Set $k = k + 1$ and return to Step 1.*

Notice that in order to perform the algorithm it is necessary to iteratively compute v and r. To do this, it is sufficient to update the matrices $(AD^2A^{\mathrm{T}})^{-1}$ and $(AD^2A^{\mathrm{T}})^{-1}AD$. Using the rank-one updating formula, this will take $O(mn)$ operations, whereas a main iteration of other interior point algorithms typically requires $O(m^2n)$ operations in the worst case. Also notice that the index selection rule, i.e. Step 2, is similar to the most negative pivot rule of the simplex method.

5. Convergence analysis

In estimating the reduction $\phi(x,s) - \phi(x(\alpha), s(\alpha))$, we shall make use of the inequality

$$\rho \geq \log(1+\rho) \geq \rho - \frac{\rho^2}{2(1-\mid \rho \mid)} \text{ for any } \rho \in (-1,1), \tag{12}$$

which is proven by Karmarkar [8].

Applying (12) to (10) yields for $\mid \alpha \mid < v_{\min}/v_j$

$$\begin{aligned} \phi(x,s) - \phi(x(\alpha), s(\alpha)) &\geq -\alpha v_j r_j \\ &\quad -\alpha^2 \frac{v_j^2}{v_{\min}^2} \frac{\|Pe_j\|^2 + \|Qe_j\|^2}{2(1-\mid \alpha \mid v_j/v_{\min})} \\ &= -\alpha v_j r_j - \frac{(\alpha v_j/v_{\min})^2}{2(1-\mid \alpha \mid v_j/v_{\min})}. \end{aligned} \tag{13}$$

Due to the index selection rule in Algorithm 1, there holds $\mid r_j \mid = \|r\|_\infty$. The following lemma provides a lower bound on $\|r\|_\infty$.

Lemma 5.1. *It holds that*

$$\|r\|_\infty > \frac{1}{\sqrt{n} v_{\min}}.$$

Proof:

Obviously, we have for any $w \in \Re^n$ that

$$\|(P-Q)w\| = \|(P+Q)w\| = \|w\|.$$

Therefore, we have

$$\|r\|^2 = \left\| \frac{(n+\sqrt{2n})}{\|v\|^2} v - V^{-1}e \right\|^2 = \left\| \frac{\sqrt{2n}}{\|v\|^2} v + \frac{n}{\|v\|^2} v - V^{-1}e \right\|^{2.}$$

Using $v \perp \frac{n}{\|v\|^2} v - V^{-1}e$, it follows that

$$\begin{aligned} \|r\|^2 &= \frac{2n}{\|v\|^2} + \left\| \frac{n}{\|v\|^2} v - V^{-1}e \right\|^2 \\ &\geq \frac{2n}{\|v\|^2} + (\frac{n}{\|v\|^2} v_{\min} - \frac{1}{v_{\min}})^2 \\ &= \frac{1}{v_{\min}^2} + \frac{n^2 v_{\min}^2}{\|v\|^4}. \\ &> \frac{1}{v_{\min}^2}. \end{aligned} \tag{14}$$

Since $\|r\|_\infty \geq \|r\| / \sqrt{n}$, (14) implies the desired result.

□

Combining Lemma 5.1 and (13) we obtain a lower bound on the reduction $\phi(x,s) - \phi(x(\alpha),s(\alpha))$:

Lemma 5.2. *For* $\alpha = -\frac{|r_j|}{r_j}\frac{2v_{\min}/v_j}{3\sqrt{n+2}}$, *we have*

$$\phi(x,s) - \phi(x(\alpha),s(\alpha)) > \frac{1}{3(n+2)}.$$

Proof:

From (13) and the fact that $| r_j |= \|r\|_\infty$ it follows that

$$\phi(x,s) - \phi(x(\alpha),s(\alpha)) \geq -\frac{\alpha r_j}{| r_j |} v_j \|r\|_\infty - \frac{(\alpha v_j/v_{\min})^2}{2(1-\mid \alpha \mid v_j/v_{\min})}.$$

Therefore, by choosing $\alpha = -\frac{|r_j|}{r_j}\frac{2v_{\min}/v_j}{3\sqrt{n+2}}$ and using Lemma 5.1, we obtain

$$\begin{aligned}\phi(x,s) - \phi(x(\alpha),s(\alpha)) &> \frac{2}{3\sqrt{n(n+2)}} - \frac{2}{9(n+2)(1-2/(3\sqrt{n+2}))} \\ &> \frac{2}{3(n+2)}(1-\frac{1}{2}) = \frac{1}{3(n+2)}.\end{aligned}$$

□

Lemma 5.2 provides a possible step size rule for Step 3 in Algorithm 1. A better step size may be obtained by means of some line search procedure. Combining Lemma 4.1 and Lemma 5.2, we obtain the main result of this paper:

Theorem 5.1. *Algorithm 1 solves (P) and (D) in* $O(mn^{2.5}L)$ *operations.*

Proof:

From Lemma 5.2 we know that in every iteration, the potential function is reduced by at least a quantity of $\frac{1}{3(n+2)}$. Using Lemma 4.1, this implies that Algorithm 1 can be used to solve (P) and (D) in $O(n^{1.5}L)$ iterations. A single iteration involves $O(mn)$ operations. Thus, in total the algorithm requires $O(mn^{2.5}L)$ operations in the worst case.

□

6. Concluding Remarks

It is known that interior point algorithms usually require significantly fewer iterations for solving linear programs than simplex algorithms. Yet, the simplex method can outperform the interior point method in running time on a wide range of problems because of the low computational complexity per step. Similar to the algorithm of Mizuno [12], the interior point algorithm that we proposed in this paper has the property that the amount of work per step is comparable to the work in a simplex iteration. Using a new approach for computing the primal and dual iterates, we do not need to distinguish "real" and "approximate" scaling vectors as in [12, 3].

References

[1] K.M. Anstreicher, "A standard form variant, and safeguarded linesearch, for the modified Karmarkar algorithm," *Mathematical Programming* 47 (1990) 337-351.

[2] K.M. Anstreicher and R.A. Bosch, "Long steps in a $O(n^3L)$ algorithm for linear programming," *Mathematical Programming* 54 (1992) 251-265.

[3] R.A. Bosch, "On Mizuno's rank one updating algorithm for linear programming," *SIAM Journal on Optimization* 3 (1993) 861-867.

[4] R.A. Bosch and K.M. Anstreicher, "On partial updating in a potential reduction linear programming algorithm of Kojima, Mizuno, and Yoshise," *Algorithmica* 9 (1993) 184-197.

[5] C.C. Gonzaga, "An algorithm for solving linear programming problems in $O(n^3L)$ operations," in: N. Megiddo, ed. *Progress in Mathematical Programming*, Springer–Verlag (Berlin, 1989) 1-28.

[6] D. Den Hertog, *Interior point approach to linear, quadratic and convex programming, algorithms and complexity.* Kluwer Publishers, Dordrecht, The Netherlands, 1994.

[7] D. Den Hertog, C. Roos and J.-Ph. Vial, "A complexity reduction for the long–step path–following algorithm for linear programming," *SIAM Journal on Optimization* (1992) 71-87.

[8] N.K. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica* 4 (1984) 373-395.

[9] M. Kojima, S. Mizuno and A. Yoshise, "A polynomial–time algorithm for a class of linear complementarity problems," *Mathematical Programming* 44 (1989) 1-26.

[10] S. Mehrotra, "Deferred rank one updates in $O(n^3L)$ interior point algorithms," Technical Report 90-11, Dept. of Industrial Engineering and Management Sciences, Northwestern University (Evanston, IL, 1990).

[11] S. Mizuno, "$O(n^\rho L)$ iteration $O(n^3L)$ potential reduction algorithms for linear programming," *Linear Algebra and its Applications* 152 (1991) 155-168.

[12] S. Mizuno, "A rank one updating algorithm for linear programming," *The Arabian Journal for Science and Engineering* 15 (1990) 671-677.

[13] R.D.C. Monteiro and I. Adler, "Interior path following primal–dual algorithms. Part II: convex quadratic programming," *Mathematical Programming* 44 (1989) 43-66.

[14] J. Renegar, "A polynomial time algorithm, based on Newton's method, for linear programming," *Mathematical Programming* 40 (1988) 59-93.

[15] D.F. Shanno, "Computing Karmarkar projections quickly," *Mathematical Programming* 41 (1988) 61-71.

[16] M.J. Todd and Y. Ye, "A centered projective algorithm for linear programming," *Mathematics of Operations Research* 15 (1990) 508-529.

[17] P.M. Vaidya, "An algorithm for linear programming which requires $O(((m+n)n^2+$

$(m+n)^{1.5}n)L)$ arithmetic operations," *Proceedings of the 19th ACM Symposium on the Theory of Computing* (1987) 29-38.

[18] Y. Ye, "An $O(n^3L)$ potential reduction algorithm for linear programming," *Mathematical Programming* 50 (1991) 239-258.

[19] Y. Ye, "Further developments in potential reduction algorithm," Technical Report, Department of Management Sciences, University of Iowa, 1989.