Journal of Applied Mathematics, Statistics and Informatics (JAMSI), **4** (2008), No. 2

A Nearest Point Approach Algorithm for a Class of Linear Programming Problems

TOMÁŠ JURÍK

Abstract

We propose a new algorithm for solving linear programming problems. It is based on the observation that any optimal solution of a linear programming problem is the closest feasible point to a hyperplane whose normal is given by the vector c consisting of the cost coefficients provided that the hyperplane does not intersect the feasible region. To minimize the distance between the feasible region and the hyperplane the algorithm employs the multiple solving of the quadratic programming problem – to find the nearest feasible solution to a line perpendicular to the vector c. Beside the geometric idea we present some theoretical results which guarantee correctness and local convergence of the algorithm. The efficiency of the algorithm is supported by the numerical experiments on the randomly generated problems.

Mathematics Subject Classification 2000: 90C05, 90C20, 90C55
Additional Key Words and Phrases: linear programming, nearest point problem, quadratic programming, line search, active set method

1. INTRODUCTION

Since the linear programming (LP) was discovered by Dantzig, it has experienced an enormous growth. In the first decades of LP the simplex method was the only tool for solving those problems. After the surprising complexity result [Klee and Minty 1972] and the Karmarkar's breakthrough [Karmarkar 1984], the interior point methods appeared. Instead of these two methods and their more or less sophisticated variants [Terlaky and Zhang 1993], [Cardoso and Clímaco 1992], [Wright 1996], [Roos et al. 2006], the different approaches has been studied [Barahona and Anbil 2000], [Barnes et al. 2002], [Paparrizos et al. 2003], [Palomo 2004], [Golikov and Evtusenko 2004], [Murty 2006], [Jurík 2007]. An exterior pivot algorithm [Paparrizos et al. 2003] constructs two paths: a feasible path and an infeasible (exterior) path. The exterior path, which is created due to the dual basis, speeds up the progress of the feasible path to the optimal solution, the feasible path is a piecewise linear path that crosses an interior of the feasible region and its break points lie on the boundary of feasible region. The key point of the projective method [Golikov and Evtusenko 2004] is solving the nonlinear optimization problem and to project the set of feasible solutions to the set of the optimal solutions. In this paper, the auxiliary Lagrangian is constructed and the unconstrained problem is solved in the nonlinear manner. Our approach was inspired by these ideas.

We propose a new algorithm for solving linear problems. We consider an LP problem in the form

$$\max_{x\in\mathbb{R}^n} \left\{c^T x : Ax \le b\right\}, \tag{1}$$

where $0 \ne c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and A is an $m \times n$ matrix. We assume that the given problem (1) has an optimal solution. This assumption is not constraining as one can see from the following. To detect the infeasibility of the problem (1) we have to solve, similarly to the two-phase simplex method, an auxiliary LP problem $\min_{\mu\in\mathbb{R}}\{\mu : Ax-\mu e \leq b,\ \mu \geq 0\}$ whose optimal value is μ^*, where $e \in \mathbb{R}^n$ is the all one vector. Then the problem (1) is feasible only iff $\mu^* = 0$. The problem (1) is unbounded iff there exists a vector $v \in \mathbb{R}^n$ such that $c^T v > 0$ and $A^T v \leq 0$. This is the same as to decide whether an LP problem $\max\{0^T v : c^T v \geq 1, Av \leq 0\}$ is feasible and this is the previous case. For the sake of simplicity, we assume moreover that the given problem (1) is nondegenerate. This assumption is also not too constraining since we are focused on the real-value optimization, which we modeled by the randomly generated problems. In those problems the degeneracy occurs very rarely.

Throughout this paper we denote the j-th row of a matrix A by a_j and the i-th component of a vector v by v_i. The vectors are considered as columns. We say that a point $x \in \mathbb{R}^n$ is a *feasible solution* of a problem (1) if $Ax \leq b$. The set of all feasible solutions (feasible region) is denoted by $\mathcal{P}$. The feasible solution x^* is *optimal* if for every $x \in \mathcal{P}$ we have $c^T x \leq c^T x^*$. The rows of the system $Ax \leq b$, i .e. $a_i^T x \leq b_i$ for $i = 1, \ldots, m$, are called constraints. A constraint is called *active at* x, if $a_i^T x = b_i$. A set of indices of all active constraints at a feasible solution x defines an *active-set* $I(x)$. In addition, $\|v\|$ stands for the 2-norm of the vector v.

The proposed algorithm is based on the observation that the minimum distance between a feasible region $\mathcal{P}$ and a hyperplane which is perpendicular to the cost coefficient vector c is attained at the optimal solution (cf. Theorem 2.2). For minimizing this distance, the algorithm employs a multiple solving of a quadratic programming (QP) problem – to find the nearest feasible solution to a line which is perpendicular to the vector c. In k-th main iteration we compute a direction s^k of a line $\mathcal{L}^k = \{w^{k,0} + ts^k,\ t \in \mathbb{R}\}$ and solve a problem to find the nearest feasible solution of the problem (1) to the line $\mathcal{L}^k$. This problem is solved in the manner of moving the point $x^{k,l}$, for $l = 0, 1, \ldots$ along the boundary of $\mathcal{P}$ in the direction that is a projection of the vector s^k onto an affine space determined by the point $x^{k,l}$. At the beginning of each main iterations, the direction s^k is changed.

After this introduction we state and prove some basic statements. Section 3 describes the new method in full details including the correctness and finiteness proofs. The last section gives the numerical results that certify the efficiency of our approach for solving randomly generated LP problems.

2. BASIC THEORY

Our reference monograph for nonlinear programming is [Boyd and Vanderberghe 2004]. We remark that the dual problem to the problem (1) is

$$\min_{u\in\mathbb{R}^m} \left\{b^T u : A^T u = c, u \geq 0\right\}. \tag{2}$$

The following well-known complementary slackness conditions (cf. [Roos et al. 2006]) claim that a primal and a dual feasible solution are optimal if and only if the products of the corresponding residuals are zero.

THEOREM 2.1. *Suppose that $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ are the feasible solutions for the problems (1) and (2) respectively. Then x and u are optimal for their respective problems if and only if*

$$u_i \left(Ax - b\right)_i = 0, \ \textit{for all} \ \ i = 1, 2, \ldots, m. \tag{3}$$

Let x^* be any optimal solution of the problem (1) and let $z^* = c^T x^*$. If the minimum distance between the feasible region $\mathcal{P}$ and a hyperplane $\mathcal{H}(z) = \{w \in \mathbb{R}^n : c^T w = z\}$ where $z > z^*$ is achieved in a pair (x^*, w^*), then x^* is an optimal solution of the problem (1).

THEOREM 2.2. *Let z^* be the optimal value of the problem (1) and z be a real number such that $z > z^*$. If*

$$(x^*, w^*) = \arg\min_{x,w} \left\{ \frac{1}{2}\|w - x\|^2 : Ax \le b,\ c^T w = z \right\} \tag{4}$$

then x^* is the optimal solution of the problem (1).

PROOF. A function $\frac{1}{2}\|w-x\|^2$ is strictly convex and quadratic; therefore due to Karush-Kuhn-Tucker (KKT) conditions the point (x^*, w^*) is an optimal solution for the problem (4) if and only if

$$\begin{aligned}
x^* - w^* + A^T\lambda^* &= 0, && \text{(5a)}\\
w^* - x^* - \nu^* c &= 0, && \text{(5b)}\\
Ax^* - b &\le 0, && \text{(5c)}\\
\lambda_i^*(Ax^* - b)_i &= 0, \quad i = 1, 2, \ldots, m, && \text{(5d)}\\
\lambda^* &\ge 0, && \text{(5e)}\\
c^T w^* - z &= 0 && \text{(5f)}
\end{aligned}$$

for some $\lambda^* \in \mathbb{R}^m$ and $\nu^* \in \mathbb{R}$. Since $z > z^*$ then $w^* \ne x^*$. From (5a) and (5b) we have $A^T\lambda^* = w^* - x^* = \nu^* c$ for some $\nu^* \in \mathbb{R}$. Moreover, the assumption $z > z^*$ and the equality (5b) multiplied by c^T give that $\nu^* > 0$. Now the system (5) implies the complementary slackness conditions for the pair $(x^*, \lambda^*/\nu^*)$ in Theorem 2.1. □

3. A NEAREST POINT APPROACH ALGORITHM

In this paper we propose an algorithm that solves the problem (4) in several main iterations. In each k-th main iteration we find the nearest feasible solution to the line $\mathcal{L}^k = \{w^{k,0} + ts^k,\ t \in \mathbb{R}\}$, i.e.

$$\min_{x \in \mathbb{R}^n, t \in \mathbb{R}} \left\{ \frac{1}{2} \left\| w^{k,0} + ts^k - x \right\|^2 : Ax \le b \right\} \tag{6}$$

for a fixed point $w^{k,0}$ and a direction s^k. The k-th iteration starts with a pair of points $(w^{k,0}, x^{k,0})$ such that the point $x^{k,0}$ is the nearest feasible solution of the problem (1) to the point $w^{k,0}$. In the l-th inner iteration, the point $w^{k,l}$ is moved in the direction s^k and the point $x^{k,l}$ is moved in the direction $\Delta x^{k,l}$ which is the projection of the vector s^k onto the set $\mathcal{P}$. The inner iterations repeat until a pair $(x^{k,l}, \sum_l t^{k,l})$ solves the problem (6). In that case the inner iterations terminate, the new line l^{k+1} is constructed and the main iteration is repeated.

In this section we give an outline of steps involved in the nearest point approach algorithm (NPAA), and then we prove the correctness of the algorithm. For a simplicity we denote the active set $I(x^{k,l})$ by $I_{k,l}$, the matrix whose rows are $\{a_j^T : j \in I_{k,l}\}$ by $A_{k,l}$ and the matrix $(A_{k,l}A_{k,l}^T)^{-1}A_{k,l}$ by $R_{k,l}$. The identity matrix of order n is denoted by $\mathbb{I}_n$.

Inputs: An $m \times n$ matrix A, vectors $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, a feasible solution x^0 and an upper bound z of an optimal value of the problem (1).

Step 0 (initialization)

Set $x^{0,0} = x^0 + \overline{\alpha}c$, where $\overline{\alpha} = \arg\max_{\alpha \in \mathbb{R}} \{\alpha :\ A(x^0 + \alpha c) \le b\}$. Construct a point $w^{0,0} \in \mathcal{H}(z) = \{w \in \mathbb{R}^n : c^T w = z\}$ such that the nearest feasible solution to the point $w^{0,0}$ is the point $x^{0,0}$ (see formula (10)). Set $k = 0$ and $l = 0$.

Step 1 (optimality condition)
If the system $A_{k,l}^T u = c$, $u \geq 0$ has any solution, then the point $x^{k,l}$ is the optimal solution to the problem (1). Stop.

Step 2 (line $\mathcal{L}^k$)
Set $w^{k+1,0} = w^{k,l}$, $x^{k+1,0} = x^{k,l}$, $k = k+1$. Calculate a direction s^k of a line $\mathcal{L}^k$ as a projection of a vector $x^{k,0} - w^{k,0}$ onto the hyperplane $\mathcal{H}(z)$. The problem is to find the nearest feasible solution x of the problem (1) to the line $\mathcal{L}^k = \{w^{k,0} + ts^k, t \in \mathbb{R}\}$. Set $l = 0$.

Step 3 (solving a nearest point problem)
Construct a subset $J_{k,l}$ of all active constraints $I_{k,l}$ of the present solution $x^{k,l}$

$$J_{k,l} = \left\{ i \in I_{k,l} : \left[R_{k,l}(w^{k,l} - x^{k,l})\right]_i > 0 \text{ or } \left[R_{k,l} s^k\right]_i \geq 0 \right\} \tag{7}$$

and an update direction $\Delta x^{k,l}$ that is a projection of the vector s^k onto a set $\{y \in \mathbb{R}^n : B_{k,l} y = 0\}$, where $B_{k,l}$ is a matrix whose rows are $\{a_j^T : j \in J_{k,l}\}$, i.e.

$$\Delta x^{k,l} = \left(\mathbb{I}_n - B_{k,l}^T \left(B_{k,l} B_{k,l}^T\right)^{-1} B_{k,l} \right) s^k. \tag{8}$$

Find the maximum value of $t_1^{k,l}$ such that a vector $\lambda^{k,l}(t)$ is nonnegative, where

$$\lambda^{k,l}(t) = R_{k,l} \left[w^{k,l} - x^{k,l} + t(s^k - \Delta x^{k,l}) \right].$$

Find the maximum value of $t_2^{k,l}$ such that a point $x^{k,l} + t_2 \Delta x^{k,l}$ is a feasible solution and

$$t_3^{k,l} = -(w^{k,l} - x^{k,l})^T s^k / \left(s^{kT} B_{k,l}^T \left(B_{k,l} B_{k,l}^T\right)^{-1} B_{k,l} s^k \right) \tag{9}$$

Set $t^{k,l} = \min\{t_1^{k,l}, t_2^{k,l}, t_3^{k,l}\}$, $x^{k,l+1} = x^{k,l} + t^{k,l} \Delta x^{k,l}$, $w^{k,l+1} = w^{k,l} + t^{k,l} \Delta s^k$.
If $t^{k,l} = t_3^{k,l}$ then the solution $x^{k,l}$ is the nearest feasible solution to the line $\mathcal{L}^k$, go to Step 1.
Else set $l = l+1$ and repeat Step 3.

3.1 Correctness of algorithm

The algorithm starts with any feasible solution x^0. An initialization step assures that $I_{0,0} \neq \emptyset$, moreover, it assures that a set $M_0 = I_{0,0} \bigcap \left\{ j : c^T a_j > 0 \right\}$ is nonempty. Since if the set M_0 is empty, then the given problem is unbounded. Note that the algorithm keeps the set $I_{k,l}$ nonempty since the nearest feasible solution $x^{k,l} \in \mathcal{P}$ to the line $\mathcal{L}^k \subset \mathcal{H}(z)$ lies at the boundary of $\mathcal{P}$ and the algorithm moves along the boundary of the set $\mathcal{P}$. We construct an initial point $w^{0,0}$ in a way that a vector $w^{0,0} - x^{0,0}$ is a multiple of the average of the vectors $\{a_i : i \in M_0\}$, i. e.

$$w^{0,0} = x^{0,0} + t \sum_{i \in M_0} a_i, \text{ where } t > 0 \text{ fulfills } c^T \left(x^{0,0} + t \sum_{i \in M_0} a_i \right) = z, \tag{10}$$

where z is an upper bound for the optimal value z^*. The value of t is well defined, since $M_0 \neq \emptyset$, $c^T x^{0,0} < z$ and for all $j \in M_0$ is $c^T a_j > 0$. The point $w^{0,0}$ is defined such that a vector $w^{0,0} - x^{0,0}$ is a nonnegative combination of the vectors $\{a_i : i \in I_{0,0}\}$. It means that a pair $(x^{0,0}, w^{0,0})$ fulfills the system (5) except the equality (5b) for some vector $\lambda \in \mathbb{R}^m$. In other words, the point $x^{0,0}$ is the nearest feasible solution to the point $w^{0,0}$. This property is demanded to be fulfilled whenever entering Step 2.

The optimality condition of the present feasible solution x (Step 1) is a direct corollary of Farkas's lemma.

The second step of the algorithm involves computing a vector s^k as a projection of a vector $x^{k,0} - w^{k,0}$ onto the hyperplane $\mathcal{H}(z)$, i.e.

$$s^k = \left(\mathbb{I}_n - \frac{c \cdot c^T}{\|c\|^2} \right) (x^{k,0} - w^{k,0}) \tag{11}$$

where $\mathbb{I}_n$ is the identity matrix of order n. To prove the correctness of Step 2 we have to show that the projection s^k is nonzero if the present feasible solution $x^{k,0}$ is not optimal. The vector s^k is zero if and only if we have $w^{k,0} - x^{k,0} = \nu c$ for some $\nu \in \mathbb{R}$. Since $c^T w^{k,0} = z > z^*$ then we have that $\nu > 0$ and consequently the pair $(x^{k,0}, w^{k,0})$ is the solution of the KKT system (5), i.e. $x^{k,0}$ is an optimal solution.

The aim of the main iteration loop (Step 3) is to find the nearest feasible solution to the line $\mathcal{L}^k = \{w^{k,0} + ts^k, t \in \mathbb{R}\} \subset \mathcal{H}(z)$. The algorithm presents a successive line search method which efforts to fulfill the corresponding KKT conditions (15). In each main iteration we minimize the function $f^k(t) : \mathbb{R} \to \mathbb{R}$

$$f^k(t) = \min_{x \in \mathbb{R}^n} \left\{ \frac{1}{2} \|w^{k,0} + ts^k - x\|^2 : \ Ax \le b, \right\} \tag{12}$$

for a fixed point $w^{k,0}$ and a fixed vector s^k. For this purpose we study the trajectory of the points $x^k(t) : \mathbb{R} \to \mathbb{R}^n$

$$x^k(t) = \arg\min_{x \in \mathbb{R}^n} \left\{ \frac{1}{2} \|w^{k,0} + ts^k - x\|^2 : \ Ax \le b \right\}. \tag{13}$$

First we prove that the function $f^k(t)$ is convex.

Theorem 3.1. *Let A be a given $m \times n$ matrix, $b \in \mathbb{R}^m$ and $w^{k,0}, s^k \in \mathbb{R}^n$ are given vectors such that the set $\mathcal{P} = \{x \in \mathbb{R}^n : \ Ax \le b\}$ is nonempty. Then the function $f^k(t) : \mathbb{R} \to \mathbb{R}$ given by (12) is convex.*

Proof. We verify that for any numbers t_1, t_2 and $0 \le \lambda \le 1$ is

$$\lambda f^k(t_1) + (1-\lambda) f^k(t_2) \ge f^k(\lambda t_1 + (1-\lambda) t_2).$$

We denoted by $y^i \in \mathcal{P}$ the points in which the value $f^k(t_i)$ for $i = 1, 2$ is attained, i.e.

$$f^k(t_i) = \frac{1}{2} \left\| w^{k,0} + t_i s^k - y^i \right\|^2, \ Ay^i \le b, \ i = 1, 2.$$

Because the set $\mathcal{P}$ is convex, we have that $\lambda y^1 + (1-\lambda) y^2 \in \mathcal{P}$. The convexity of the Euclidean norm completes the proof

$$\begin{aligned} \lambda f^k(t_1) + (1-\lambda) f^k(t_2) &= \lambda \ \frac{1}{2} \left\| w^{k,0} + t_1 s^k - y^1 \right\|^2 + (1-\lambda) \ \frac{1}{2} \left\| w^{k,0} + t_2 s - y^2 \right\|^2 \\ &\ge \frac{1}{2} \left\| w^{k,0} + (\lambda t_1 + (1-\lambda) t_2) s^k - \left(\lambda y^1 + (1-\lambda) y^2 \right) \right\|^2 \\ &\ge \min_{x \in \mathbb{R}^n} \left\{ \frac{1}{2} \left\| w^{k,0} + (\lambda t_1 + (1-\lambda) t_2) s^k - y \right\|^2 : \ Ay \le b \right\}. \end{aligned}$$

□

Since the function $f^k(t)$ is convex then every solution $x^{k,l}$ of the KKT conditions for the problem (6) is the nearest feasible solution to the line $\mathcal{L}^k$. Here we describe one inner iteration loop: how to step from the pair $(x^{k,l}, w^{k,l})$ to the pair $(x^{k,l} + t^{k,l} \Delta x^{k,l}, w^{k,l} + t^{k,l} s^k)$. This step is illustrated in the figure 1. We suppose that $x^{k,l}$ is the nearest feasible solution to

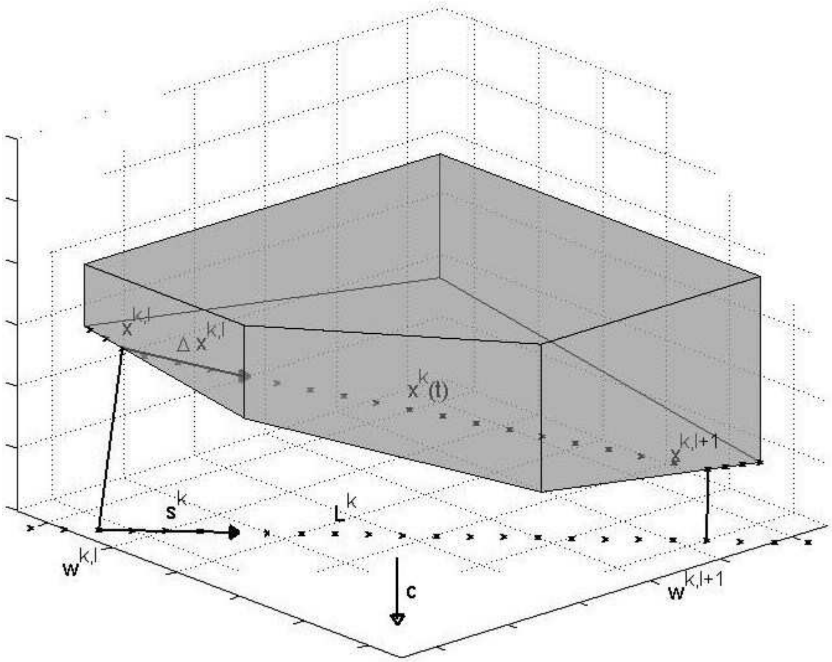

Fig. 1. Inner iteration step

the point $w^{k,l} \in \mathbb{R}^n$. We are interested in a direction $\Delta x^{k,l}$ such that the point $x^{k,l}+t^{k,l}\Delta x^{k,l}$ is the nearest feasible solution to the point $w^{k,l}+t^{k,l}s^k$, where s^k is a fixed vector and $t^{k,l} \in \mathbb{R}$ is a sufficiently small positive number. The point $x^{k,l}+t^{k,l}\Delta x^{k,l}$ remains feasible if and only if $A_{k,l}\Delta x^{k,l} \le 0$. Our QP problem (6) can be written as

$$\min_{\Delta x, t} \left\{ \frac{1}{2} \left\| w^{k,l} + ts^k - (x^{k,l} + t\Delta x) \right\|^2 : \; A_{k,l}\Delta x \le 0 \right\}. \tag{14}$$

We reduced the number of constraints in the problem (6), i.e. ignore some constraints and focus on the local geometry around the feasible solution $x^{k,l}$. The corresponding KKT conditions for $\Delta x = \Delta x^{k,l}$ and $t = t^{k,l}$ are

$$x^{k,l} + t\Delta x - w^{k,l} - ts^k + A_{k,l}^T \lambda \;=\; 0 \tag{15a}$$

$$\left(w^{k,l} + ts^k - x^{k,l} - t\Delta x\right)^T s^k \;=\; 0 \tag{15b}$$

$$A_{k,l}\Delta x \;\le\; 0 \tag{15c}$$

$$\lambda_i \left(A_{k,l}\Delta x\right)_i \;=\; 0, \quad i = 1, 2, \ldots, m \tag{15d}$$

$$\lambda \;\ge\; 0. \tag{15e}$$

Since $x^{k,l}$ is the nearest feasible solution to the point $w^{k,l}$ then it is true that the vectors $(x^{k,l}, w^{k,l})$, $\Delta x = 0$ and a real numbers $t = 0$ and some $\lambda \ge 0$ fulfill the system (15) except the equality (15b). We remark that since $(w^{k,0} - x^{k,0})^T s^k < 0$ (cf. (11)) then every solution of the system (15) fulfills the condition $t > 0$. Our aim is to find the direction Δx for a small positive value of t such that the pair $(x^{k,l} + t\Delta x, w^{k,l} + ts^k)$ fulfills the system (15) except (15b) for some nonnegative vector $\lambda(t)$. Every main iteration loop starts from a pair $(x^{k,l}, w^{k,l})$ which fulfills the system (15) except the equality (15b) and in a finite number of steps find the solution of the system (15).

Under the assumption that the problem (1) is nondegenerate we derive from the equality (15a) that

$$\lambda(t) = R_{k,l} \left[w^{k,l} + ts^k - x^{k,l} - \Delta x \right]. \tag{16}$$

The nonnegativity of vector $\lambda(t)$ is assured by the choice of the value of $t^{k,l}$, i.e. $t_1^{k,l} \geq t^{k,l}$. It is known that the direction Δx is a projection of the vector s^k onto some affine space (cf. [Bernstein 2005]). This space is the null space of $B_{k,l}$, where $J_{k,l} \subseteq I_{k,l}$ (cf. (8)). From the equality (15b) we have

$$(w^{k,l} - x^{k,l})^T s^k + t \left(s^{kT} B_{k,l}^T \left(B_{k,l} B_{k,l}^T \right)^{-1} B_{k,l} s^k \right) = 0, \tag{17}$$

therefore the value of t that fulfills the equality (15b) is given by the formula (9). Under the nondegeneracy assumption is the symmetric matrix $B_{k,l}B_{k,l}^T$ positive definite. The only case when the value of t cannot be derived from the equality (17) is if $B_{k,l}s^k = 0$. Then we set $t_3^{k,l} = \infty$ which does not affect the value of $t^{k,l}$. Note that all points $\{w^{k,l}\}_{l \in \mathbb{N}}$ lie on the line $\mathcal{L}^k$ and a sequence $\{x^{k,l}\}_{l \in \mathbb{N}}$ contains the break points of the (piecewise linear) path $x^k(t)$.

It remains to give a rule how to construct the set $J_{k,l}$. We have to distinguish two possible cases. If $i \in J_{k,l} \subseteq I_{k,l}$ then $a_i^T \Delta x = 0$; therefore $\lambda_i(t) = \left[R_{k,l}(w^{k,l} + ts^k - x^{k,l}) \right]_i$. Otherwise, if $i \notin J_{k,l}$, then we have $a_i^T \Delta x < 0$; moreover, from (15d) yields that $\lambda_i(t) = 0$ for all feasible values t. Since $x^{k,l}$ is the nearest feasible solution to the point $w^{k,l}$ then we have that $\overline{\lambda} \geq 0$ where

$$\overline{\lambda} = R_{k,l}(w^{k,l} - x^{k,l}). \tag{18}$$

Now it is not hard to see that the rule is based on the positivity of the value $\overline{\lambda}_i$. If $\overline{\lambda}_i > 0$ then from the equation (15d) we have $i \in J_{k,l}$. If $\overline{\lambda}_i = 0$ and $(R_{k,l}s^k)_i < 0$ then $i \notin J_{k,l}$. Reversely, if $\overline{\lambda}_i = 0$ and $i \notin J_{k,l}$; then from $a_i^T \Delta x < 0$ we have $(R_{k,l}s^k)_i < 0$. Therefore the construction of the set $J_{k,l} \subseteq I_{k,l}$ is given by the formula (7).

Finally we set the value of $t^{k,l}$ in a way that the pair $(x^{k,l} + t^{k,l}\Delta x^{k,l}, w^{k,l} + t^{k,l}s^k)$ and $\lambda(t^{k,l})$ fulfills the system (5) except the equality (5b) and the point $x^{k,l} + t^{k,l}\Delta x^{k,l}$ is a feasible solution to the problem (1). If the equality (5b) is fulfilled for some $t_3^{k,l} \leq t^{k,l}$ then the point $x^{k,l} + t_3^{k,l}\Delta x^{k,l}$ is the nearest feasible solution to the line $\mathcal{L}^k$.

3.2 Finiteness of algorithm

To our best knowledge, we can prove only that a number of iterations that are needed to solve the nearest point problem (6) using the described algorithm is finite. It is an direct corollary of Theorem 3.2 which states that in one main iteration the active set of the points $\{x^{k,l}\}_{l \in \mathbb{N}}$ generated from Step 3 cannot repeat, i.e. for $r \neq s$ is $I_{k,r} \neq I_{k,s}$.

Theorem 3.2. *Let A be a given $m \times n$ matrix, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ are the given vectors, $k \in \mathbb{N}$, $x^{k,0}$ is a given feasible solution and z is an upper bound of the problem (1). Let $\mathcal{L}^k \subset \mathcal{H}(z)$ be a line constructed according to the Step 2 of the proposed algorithm. If $\{x^{k,l}\}_{l \in \mathbb{N}}$ is a sequence of solutions generated by Step 3 for a fixed line $\mathcal{L}^k$, then the sequence $\{I_{k,l}\}_{l \in \mathbb{N}}$ cannot contain the same set twice.*

Proof. We want to prove that the iteration points $x^{k,l}$ for $l \in \mathbb{N}$ cannot have the same active-set.

During the proof we will reuse the definition of the set $J_{k,l}$. The rule (7) for constructing the set $J_{k,l} \subset I_{k,l}$ is applied if and only if some element of the corresponding vector $\lambda(t)$

given by (16) attained zero. Therefore exists an element of the set $I_{k,l}$ such that it does not belong to the set $J_{k,l}$.

We proceed by contradiction. Suppose that there exist two indices $r < s$ such that $I_{k,r} = I_{k,s}$. Let the corresponding points on the line $\mathcal{L}^k$ be $w^{k,0} + \tau^r s^k$ and $w^{k,l} + \tau^s s^k$. We remark that $\tau^r = \sum_{l=0}^{r} t^{k,l}$ and $\tau^s = \sum_{l=0}^{s} t^{k,l}$. An element which leaves the set $I_{k,r}$ is denoted by i. Since $0 < \tau^r$ and $r < s$ we have $0 < \tau^r < \tau^s$. Using the equality (18) and the rule (7) for a feasible solution $x^{k,r}$ we have

$$\left[R_{k,r}(w^{k,0} + \tau^r s^k - x^{k,r})\right]_i \;=\; 0.$$

Since $i \notin J_{k,r}$ we have that

$$Q := \left[R_{k,r} s^k\right]_i \;<\; 0.$$

The crucial fact in this proof is an assumption that $I_{k,r} = I_{k,s}$. As a consequence we have $R_{k,r}x^{k,r} = R_{k,s}x^{k,s}$. Finally for a feasible solution $x^{k,s}$ we have

$$\begin{aligned} \left[R_{k,s}(w^{k,0} + \tau^s s^k - x^{k,s})\right]_i \;&=\; \left[R_{k,s}(w^{k,0} - x^{k,s})\right]_i + \tau^s Q \\ &=\; \left[R_{k,r}(w^{k,0} - x^{k,r})\right]_i + \tau^s Q = (\tau^s - \tau^r)Q < 0, \end{aligned}$$

which contradicts the inequality (18) for the feasible solution $x^{k,s}$. □

4. NUMERICAL RESULTS

In this section we describe our numerical experiments and present computational results that demonstrate an efficiency of the new approach on sparse linear programs. In order to facilitate the computations, we confine our experiments to the random problems that were small and medium in size. The results were reached in the environment Matlab 6 performed on a PC with 2.6 GHz Athlon processor, RAM 2 GB and the Windows Vista operating system. The tested problems were in the form

$$\max\left\{c^T x : Ax \le b\right\},$$

where, $A \in \mathbb{R}^{m\times n}$, $c, x \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$. The matrix A involved the nonnegativity constraints $x \ge 0$. The matrix $A = (a_{ij})_{m\times n}$ and the vector $c \in \mathbb{R}^n$ have the nonzero elements with the density $d = 5\%$, where $a_{ij} \in [10, 15]$, $c \in [-1, 1]$. The vector $b \in [50, 100]$ was dense with randomly distributed elements. The vector $b \in \mathbb{R}^m$ was positive in order to generate an initial point $x \ge 0$ without a need to solve an auxiliary LP problem. Ten problems were generated for each problem size and the average values are presented. The feasibility and precision tolerance were set to 10^{-10}.

The new algorithm was compared to the implemented procedure *linprog* in the environment Matlab and to the LIPSOL package (c.f. [Zhang 1999]). We used the procedure linprog with 'LargeScale' option turned off (simplex method - SM) and turned on (interior point method - IPM).

The first two columns represent the size of the problems, the next columns represent the average CPU time (in seconds) and the number of iterations. The last column indicates the average number of repeating of Step 3.

From these results can be seen that the calculation time is superior to the both implemented Matlab functions and is only slightly higher comparing to the package LIPSOL which exploits sparse-matrix data manipulations in Matlab environment. Since the number of repetitions of Step 3 is very small we propose a conjecture that the algorithm terminates in a finite number of iterations for all nondegenerated problems.

$d = 0.05$		linprog - SM		linprog - IPM		LIPSOL		NPAA		
m	n	time	iter	time	iter	time	iter	time	iter	k
50	100	0.17	100.6	0.04	9.6	0.09	8.4	0.03	24.7	1.0
100	200	0.95	200.9	0.21	11.3	0.18	9.1	0.10	32.9	1.0
150	300	4.01	302.2	0.53	9.8	0.36	10.6	0.44	70.6	1.0
200	400	9.79	402.7	1.38	11.4	0.44	11.5	0.87	80.4	1.0
250	500	18.74	502.1	3.22	11.3	0.58	13.8	1.75	93.8	1.2
300	600	32.86	602.7	5.82	11.9	1.34	15.2	2.71	100.8	1.4
350	700	53.59	703.6	9.25	12.4	1.82	16.1	3.97	118.6	1.0
400	800	70.83	803.6	13.10	13.9	3.25	17.6	4.82	124.6	1.0
450	900	120.25	903.5	21.86	14.0	5.38	19.7	7.39	149.1	1.0
500	1000	158.49	1037.1	25.56	12.6	7.12	22.3	9.88	156.6	1.0

Table I. Computational results for solving sparse LP problems

5. REMARKS

This paper does not present a complete theory for the new algorithm, its global convergence is still an open question. The finiteness of the inner iteration has been proved the finiteness of the outer iterations is unknown but the numerical experiments present that only a small number of lines $\mathcal{L}^k$ is needed to find an optimal solution.

At least two facts could stipulate the global convergence. The presented algorithm, which minimizes the value $\|w^{k,l} - x^{k,l}\|$, is strongly dependent on the vector s^k. Firstly, in the beginning of every main loop the direction s^k is constructed as a *gradient* of convex function $f^k(t)$ onto the hyperplane $\mathcal{H}(z)$. The second fact is a consequence of Theorem 2.2. The set $\mathcal{W}$ of all points $w \in \mathbb{R}^n$ such that $\arg\min_{x\in\mathbb{R}^n}\{\|w - x\| : \ Ax \leq b\} = x^*$, i.e. the nearest feasible solution of the problem (1) to the point w is an optimal solution x^*, is an unbounded polyhedra given by the formula

$$\mathcal{W} = \left\{w^* + A^T u : \ u \in \mathbb{R}^n, -\lambda^* \leq u\right\},$$

where (w^*, λ^*) are the solution of the system (5a)–(5e). Therefore if the algorithm finds any point $w \in \mathcal{W}$ then it terminates with an optimal solution x^*. Algorithm produce only the optimal solution x^* not the value of w^*. There is no need to find the point w^* exactly. In other words, the algorithm successfully terminates whenever the present point $w^{k,l}$ belongs in the polyhedra $\mathcal{W} \cap \mathcal{H}(z)$. The greater is a value of z the larger is a set $\mathcal{W} \cap \mathcal{H}(z)$.

One vague point of the new algorithm is an estimator (upper bound) of the optimal value z^*. We can handle this using different approaches. One of them is to construct an LP problem, which the optimal value is known for, e.g. the optimal value of the problem

$$\max\left\{c^T x - b^T u : \ Ax \leq b, A^T u = c, u \geq 0\right\}$$

is zero (provided that the problem (1) is optimal) and the part x^* of its optimal solution (x^*, u^*) is the optimal solution for the problem (1). A different approach is based on the successive updating the value z. If the minimum of the function given by (12) is zero and the current solution x is not optimal, the value of z is increased. Or, if we are able to estimate some bounds for the values of an optimal solution, we can calculate an estimate of the appropriate value of z due to [Kallio 2006].

ACKNOWLEDGMENTS

This work was supported by VEGA grant No. 1/3022/06. The author is grateful to Margaréta Halická and Milan Hamala for helpful comments, corrections and remarks on an earlier draft of the paper.

REFERENCES

Barahona, F. and Anbil, R. 2000. The volume algorithm: Producing primal solutions with a subgradient method. *Mathematical Programming 87*, 385–399.

Barnes, E., Chen, V., Gopalakrishnan, B., and Johnson, E. 2002. A least-square primal-dual algorithm for solving linear programming problems. *Operations Research Letters 30*, 289–294.

Bernstein, D. S. 2005. *Matrix mathematics: Theory, facts, and formulas with application to linear systems theory.* Princeton University Press, New Jersey.

Boyd, S. and Vanderberghe, L. 2004. *Convex optimization.* Cambridge University Press, United Kingdom.

Cardoso, D. M. and Clímaco, J. C. N. 1992. The generalized simplex method. *Operations Research Letters 12*, 337–348.

Golikov, A. I. and Evtusenko, Y. G. 2004. Solution method for large-scale linear programming problems. *Doklady Mathematics 70*, 615–619.

Jurík, T. 2007. Computational experience with a new active-set method for linear programming. *Journal of Electrical Engineering 58*, 24–27.

Kallio, M. 2006. Computing bounds for the optimal value in linear programming. *Naval Research Logistics Quarterly 24*, 301–308.

Karmarkar, N. 1984. A new polynomial-time algorithm for linear programming. *Combinatorica 4*, 373–395.

Klee, V. and Minty, G. J. 1972. How good is the simplex algorithm. In *Proc.* 3^{rd} *Symposium Inequalities.* Academic Press, New York, 159–175.

Murty, K. G. 2006. A new practically efficient interior point method for lp. *Algorithmic Operations Research 1*, 3–19.

Palomo, A. S. 2004. The sagitta method for solving linear programs. *European Journal of Operational Research 157*, 527–539.

Paparrizos, K., Samaras, N., and Stephanides, G. 2003. An efficient simplex type algorithm for sparse and dense linear programs. *European Journal of Operational Research 148*, 323–334.

Roos, C., Terlaky, T., and Vial, J.-P. 2006. *Interior Point Methods for Linear Optimization.* Springer Science + Business Media, New York.

Terlaky, T. and Zhang, S. 1993. Pivot rules for linear programming: A survey on recent theoretical developments. *Annals of Operations Research 46*, 203–233.

Wright, S. J. 1996. *Primal-dual Interior-Point Methods.* SIAM, Philadelphia.

Zhang, Y. 1999. User's guide to lipsol. *Optimization Methods and Software 11-12*, 385–396.

Tomáš Jurík,
Department of Mathematical Analysis and Numerical Mathematics,
Comenius University,
Mlynská dolina, 842 48 Bratislava, Slovak Republic
e-mail: jurik@fmph.uniba.sk

Received April 2008;