Algorithmic Operations Research Vol.1 (2006) 3–19

A New Practically Efficient Interior Point Method for LP

Katta G Murty [1]

Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI 48109-2117, USA

Abstract

In this paper we briefly review the importance of LP (linear programming), and Dantzig's main contributions to OR (Operations Research), mathematics, and computer science. In [11, 3] gravitational methods for LP have been introduced. Several versions exist. The three main versions discussed there use a ball of **(a):** *0 radius,* **(b):** *small positive radius, and* **(c):** *the ball of largest possible radius with the given center that will completely fit within the polytope, with the option of changing its radius as the algorithm progresses. In versions* **(a)** *and* **(b)***, after the first move, the center of the ball always remains very close to the boundary (because the ball hugs the boundary), and hence these versions behave like other boundary algorithms such as the simplex algorithm in terms of exponential complexity in the worst case [9].*

Here we discuss a gravitational method of type **(c)** *that behaves like an interior point method [8,20, 21]. To guarantee that the ball has the largest possible radius, it uses a new centering strategy that moves any interior feasible solution x^0 to the center of the intersection of the feasible region with the objective hyperplane through x^0, before beginning the gravitational descent moves. We show that this strategy leads to a strongly polynomial algorithm for LP in terms of the number of centering steps. Also, using this centering strategy, we discuss a method that solves LPs efficiently using no matrix inversions.*

Key words: Linear programming (LP), Dantzig's simplex method, boundary methods, gravitational method, interior point method, avoiding zigzagging, solving LPs without matrix inversions, strongly polynomial algorithm for LP

I dedicate this paper to the memory of George Dantzig who has been instrumental for myself getting into a career of research in Linear Programming and Operations Research.

1. Brief History of Algorithms for Solving Systems of Linear Inequalities

Linear algebra dealing with methods for solving systems of linear equations is the classical subject that initiated the study of mathematics long time ago. The most effective methods for solving systems of linear equations have been discovered over 2500 years ago, these methods are still the leading algorithms in use today. Even though **linear equations** have been conquered thousands of years ago, systems of linear inequalities remained inaccessible until the middle of the 20th century.

For any matrix D, we use the symbols $D_{i.}, D_{.j}$ to denote the i-th row, j-th column of D. If D is of order $m \times n$, and $S \subset \{1, \ldots, m\}$, $D_{S.}$ denotes the submatrix of D consisting of rows $D_{i.}$ for all $i \in S$. We denote the cone consisting of all the nonnegative combinations of row vectors of D by Rpos(D). For any vector y, $||y||$ denotes its Euclidean norm. For any set P, $||P||$ denotes its cardinality.

The following theorem relates systems of linear inequalities to systems of linear equations.

Theorem 1 *Consider the system of linear inequalities*

$$Ax \geq b \tag{1}$$

where $A = (a_{ij})$ is an $m \times n$ matrix and $b = (b_i) \in R^m$. So, the constraints in the system are $A_{i.}x \geq b_i$, $i \in \{1, \ldots, m\}$. If this system has a feasible solution, then there exists a subset **P** *= $\{p_1, \ldots, p_s\} \subset \{1, \ldots, m\}$ such that every solution of the system of equations*

$$A_{i.}x = b_i, \quad i \in \mathbf{P}$$

is also a feasible solution of the original system of linear inequalities (1).

Email: Katta G Murty [murty@umich.edu].

[1] For a brief account of my experience with George B Dantzig, please visit http://www.informs.org/History /dantzig/rem_murty.htm

Proof: Let K denote the set of feasible solutions of (1). For any $x \in K$ the i-th constraint in (1) is said to be *active* at x if $A_{i.}x = b_i$, *inactive* if $A_{i.}x > b_i$.

We will now describe a procedure consisting of repititions of a general step beginning with an initial point $x^0 \in K$.

General Step: Let $x^r \in K$ be the current point, and $\mathbf{P}_r = \{i : i\text{th constraint in (1) is active at } x^r\}$.

Case 1: $\mathbf{P}_r = \emptyset$. In this case x^r is an interior point of K. Let $\bar{x}$ be any solution of one equation $A_{i.}x = b_i$ for some i. If $\bar{x} \in K$, define $x^{r+1} = \bar{x}$.

If $\bar{x} \notin K$, find $\bar{\lambda}$, the maximum value of λ such that $x^r + \lambda(\bar{x} - x^r) \in K$. Then $x^r + \bar{\lambda}(\bar{x} - x^r)$ must satisfy at least one of the constraints in (1) as an equation, define $x^{r+1} = x^r + \bar{\lambda}(\bar{x} - x^r)$.

Go back to another repetition of the general step with x^{r+1} as the current point.

Case 2: $\mathbf{P}_r \neq \emptyset$, and either x^r is the unique solution of the system of equations $\{A_{i.}x = b_i : i \in \mathbf{P}_r\}$, or $\mathbf{P}_r = \{1, \ldots, m\}$. In either of these cases $\mathbf{P} = \mathbf{P}_r$ satisfies the requirement in the theorem, terminate.

Case 3: $\mathbf{P}_r$ is a nonempty proper subset of $\{1, \ldots, m\}$ and the system $\{A_{i.}x = b_i : i \in \mathbf{P}_r\}$ has alternate solutions. Let $H_r = \{x : A_{i.}x = b_i, i \in \mathbf{P}_r\}$. Let t be the dimension of H_r, and let $\{y^1, \ldots, y^t\}$ be a basis for the subspace $\{A_{i.}y = 0 : i \in \mathbf{P}_r\}$.

If each of the points $y \in \{y^1, \ldots, y^t\}$ satisfies $A_{i.}y = 0$ for all $i \in \{1, \ldots, m\}$, then $\mathbf{P} = \mathbf{P}_r$ satisfies the requirement in the theorem, terminate.

Otherwise let $\bar{y} \in \{y^1, \ldots, y^t, -y^1, \ldots, -y^t\}$ satisfy $A_{i.}\bar{y} < 0$ for some $i \in \{1, \ldots, m\} \backslash \mathbf{P}_r$. Find $\bar{\lambda}$, the maximum value of λ such that $x^r + \lambda\bar{y} \in K$, define $x^{r+1} = x^r + \bar{\lambda}\bar{y}$.

Go back to another repetition of the general step with x^{r+1} as the current point.

The subsets of indices generated in this procedure satisfy $\mathbf{P}_r \subset \mathbf{P}_{r+1}$ and $|\mathbf{P}_{r+1}| \geq 1 + |\mathbf{P}_r|$. So after at most m repetitions of the general step the procedure must terminate with a subset $\mathbf{P}$ of $\{1, \ldots, m\}$ satisfying the conditions in the theorem. ■

In systems of linear inequalities like (1) appearing in applications, typically $m \geq n$.

This theorem states that every nonempty polyhedron has a nonempty face that is an affine space. It can be used to generate a finite enumerative algorithm to find a feasible solution to a system of linear constraints containing inequalities. It involves enumeration over subsets of the inequalities in the system. For each subset do the following: eliminate all the inequality constraints in the subset from the system. if there are any inequalities in the remaining system change them into equations. Find any solution of the resulting system of linear equations. If that solution satisfies all the constraints in the original system, done, terminate. Otherwise, repeat the same procedure with the next subset of inequalities. At the end of the enumeration, if no feasible solution of the original system has turned up, it must be infeasible.

However, if the original system has m inequality constraints, in the worst case this enumerative algorithm may have to solve 2^m systems of linear equations before it either finds a feasible solution of the original system, or concludes that it is infeasible. The effort required grows exponentially with the number of inequalities in the system in the worst case.

A Paradox: Many young people develop a fear of mathematics and shy away from it. But since childhood I had a fascination for mathematics because it presents so many paradoxes. Theorem 1 also presents an interesting paradox.

As you know, linear equations can be transformed into linear inequalities by replacing each equation with the opposing pair of inequalities. But there is no way a linear inequality can be transformed into linear equations. This indicates that linear inequalities are more fundamental than linear equations.

But this theorem shows that linear equations are the key to solving linear inequalities, and hence are more fundamental. Again we will show in Section 12 that linear inequalities may play an important role for solving linear equations.

Crude examples of linear programming models have started appearing in published literature from about mid-18th century. In early 19 century Fourier published a geometric version of the principle behind the simplex algorithm for a linear program (vertex to vertex descent along the edges to an optimum) but did not discuss how this descent can be accomplished computationally on systems stated algebraically. Fourier also generalized the classical elimination method for solving linear equations into an elimination method for solving systems of linear inequalities. The method called Fourier elimination, (or Fourier-Motzkin elimination) method is very elegant theoretically. However, the elimination of each variable adds new inequalities to the remaining

system, and the number of these new inequalities grows exponentially as more and more variables are eliminated. So, this method is also not practically viable except for very small problems.

Before the 2nd World War (1930's) Kantarovich [7] developed ideas of LP models, dual multipliers, and the main ideas of the simplex algorithm, but his work was somewhat incomplete for the computational version of the algorithm. In mid-20th century Dantzig developed the complete simplex algorithm, and also showed how systems of linear inequalities can be solved using it on the Phase I linear programming formulation for the system, this was the first complete, practically and computationally viable method for solving systems of linear inequalities. So, **Linear Programming (LP)**, can be considered as the branch of mathematics which is an extension of linear algebra to solve systems of **linear inequalities**. The development of LP is a landmark event in the history of mathematics and its applications that brought our ability to solve general systems of linear constraints (including linear equations, inequalities) to a state of completion.

2. The Importance of LP

LP has now become a dominant subject in the development of efficient computational algorithms, study of convex polyhedra, and in algorithms for decision making. But for a short time in the beginning, its potential was not well recognized. **Dantzig** tells the story of how when he gave his first talk on LP and his simplex method for solving it at a professional conference, Hotelling (a burly person who liked to swim in the sea, the popular story about him was that when he does, the level of the ocean raises perceptibly) dismissed it as unimportant since everything in the world is nonlinear. But Von Neumann came to the defense of Dantzig saying that the subject will become very important. See Page xxvii of [Dantzig, Thapa, 4, 1997]. The preface in this book contains an excellent account of the early history of LP from the inventor of the most successful method in OR and in the mathematical theory of polyhedra.

Von Neumann's early assessment of the importance of LP turned out to be astonishingly correct. Today, the applications of LP in almost all areas of science are so numerous, so well known and recognized that they need no enumeration. Also, LP seems to be the basis for most of the efficient algorithms for many problems in other areas of mathematical programming. Many of the successful approaches in nonlinear programming, discrete optimization, and other branches of optimization are based on LP in their iterations. Also, with the development of duality theory and game theory (Gale [6]), LP has also assumed a central position in economics.

3. Algorithms Used for Solving LPs Today

The simplex method developed by Dantzig in the 1940s is still the dominant algorithm in use for solving LPs. The simplex method exhibits exponential growth in the worst case, but its performance in practice has been outstanding, and is being improved continually by developments in implementation technologies. It is a **one-dimensional boundary method** in the sense that it follows a path along the edges (one-dimensional boundary faces) of the set of feasible solutions of the LP. In each iteration it needs either updating the inverse or its factorization, or computing the inverse of a matrix of order equal to the number of constraints in the LP not counting the bounds on individual variables (typically the smallest dimension of the coefficient matrix for the problem) usually denoted in the literature by the symbol m.

In the late 1900s, Stimulated by Karmarker's work [8] a variety of interior point methods (IPMs) were developed for LP [1, 8, 20, 21]. These IPMs follow a path through the interior of the set of feasible solutions. Among them the ones that give the best performance try to follow the *central path* (path through a mathematically defined center of the set of feasible solutions) approximately. They are based on very elegant theory, and converge to a near optimum in polynomial time. In practice the number of iterations needed by them is much smaller than that for the simplex method, but each of these steps is much harder and more complex as it needs the inversion of a matrix of order m or larger (so each of these steps has complexity m times larger than the complexity of a step in the simplex method). Also taking advantage of sparsity in their implementations is a much more complex task than that in implementations of the simplex method. The IPMs have been observed to give slightly better performance than the simplex method only on large scale sparse problems.

All these methods in use today need either the updating of a matrix inverse or its factorization, or inversion of a matrix in each step, this is the hard part of solving LPs in spite of all the improvements that have occurred so far. It seems that practitioners are quite content with obtaining a solution not necessarily optimal, but close to being so, but they want a method that can obtain such

a solution much faster than existing methods.

4. Dantzig's Contributions to Operations Research (OR), Mathematics, and Computer Science

When he introduced the complete version of the simplex method as a computational algorithm for solving LPs in the late 1940s, Dantzig made many important contributions. We discuss some of these briefly here.

Contributions to OR: The simplex method is the first effective computational algorithm for one of the most versatile mathematical models in OR. Even though LP and also the simplex method for solving it originated much earlier as explained in Section 1, it started becoming prominent only with Dantzig's work, and OR was just beginning to develop around that time. The success of the simplex method is one of the root causes for the phenomenal development and the maturing of LP, mathematical programming in general, and OR, in the 2nd half of the 20th century.

Contributions to Mathematics and Computer Science: Dantzig has made fundamental contributions to the mathematical study of convex polyhedra (a classical subject being investigated by mathematicians for more than 2000 years), and linear algebra (also a classical subject with an even longer history).

We could only see drawings of 2-dimensional polyhedra before Dantzig's work. Polyhedra in higher dimensions could only be visualized through imagination. The primal simplex pivot steps that Dantzig developed are the first computational steps for actually tracing an edge (either bounded or unbounded) of a convex polyhedron. It opened a revolutionary new computational dimension in the mathematical study of convex polyhedra, and made it possible to visualize and explore higher dimensional polyhedra through computation. The simplex method enriched the mathematical study of convex polyhedra manyfold.

Linear algebra is the classical subject with the elimination method (also called the Gaussian (G), Gauss-Jordan (GJ) elimination methods in slightly different versions) that has a history of over 2500 years for solving systems of linear equations. But it did not have effective techniques for handling systems of linear constraints involving inequalities. Even though the Fourier-Motzkin elimination method developed in the 19th century could handle inequalities, it is not practically effective except for very small problems. The simplex method is the first effective computational method for solving general systems of linear constraints. Because of this, LP can be considered as the 20th century extension of linear algebra to solve systems of general linear constraints.

We could also consider important pedagogic improvements that Dantzig contributed to linear algebra. He would state all the algorithmic steps in the GJ elimination method using the fundamental tool of row operations on the detached coefficient tableau for the system with the variable corresponding to each column entered in a top row in every tableau. This makes it easier for young students to see that the essence of this method is to take linear combinations of equations in the original system to get an equivalent but simpler system from which a solution can be read out. In descriptions of the GJ method in most mathematics books on linear algebra, the variables are usually left out. Also, they state the termination condition to be that of reaching the RREF (reduced row echelon form, a tableau is defined to be in RREF if it contains a full set of unit vectors in proper order at the left end). Dantzig realized that it is not important that all unit vectors be at the left end of the tableau (they can be anywhere and can be scattered all over); also it is not important that they be in proper order from left to right. He developed the very simple data structure (this phrase means a strategy for storing information generated during the algorithm, and using it to improve the efficiency of that algorithm, perhaps this is the first instance of such a structure in computational algorithms) of associating the variable corresponding to the rth unit vector in the final tableau as the rth basic variable or basic variable in the rth row; and storing these basic variables in a column on the tableau as the algorithm progresses. This data structure makes it easier to read the solution directly from the final tableau of the GJ method by making all nonbasic variables = 0; and the rth basic variable = the rth updated RHS constant, for all r. Dantzig called this final tableau the **canonical tableau** to distinguish it from the mathematical concept of RREF. It also opened the possibility of pivot column selection strategies instead of always selecting the leftmost eligible column in this method.

Another important contribution of Dantzig, the revised simplex method, has very important consequences to the GJ method. All the linear algebra books state that if the equation "$0 = a$" (where a is a nonzero number) shows up in one of the tableaus during the appli-

cation of the GJ method on the system, then the system is infeasible. Whenever such an equation "$0 = a$" appears, it is also helpful if the method can produce the row vector π of coefficients in a linear combination of constraints in the original system that yields this inconsistent equation. This vector π is called an **evidence (or certificate) of infeasibility** for the original system. But with the usual descriptions of the GJ method to get an RREF or canonical tableau, this evidence is not available when the infeasibility conclusion is reached. Executing the GJ method using the basis inverse in the revised simplex format has the great advantage that at

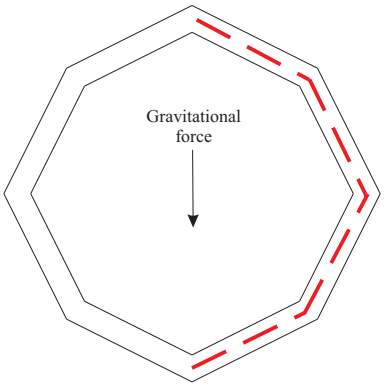

Fig. 1. The dashed lines indicate the path taken by a point ball beginning at the top vertex inside a tubular network for the edges of the feasible region of an LP under the gravitational force pulling it towards the optimum.

termination we either get a feasible solution of the original system, or evidence of its infeasibility; besides being computationally efficient. For details on this version of the GJ method, see the freshman-sophomore level linear algebra web-book with an OR focus and with the pedagogic improvements mentioned above, Murty[14].

5. The Gravitational Method for LP

As pointed out in [2], the path taken by the simplex algorithm to solve an LP can itself be interpreted as the path of a point ball falling under the influence of a gravitational force inside a thin tubular network of the one dimensional skeleton of the feasible region in which each vertex is open to all the edges incident at it. See Figure 1 for a 2-dimensional illustration.

Starting about 20 years ago, Murty[11], Chang and Murty[3] developed newer methods for LP based on the principle of gravitational force. We consider the LP in the form

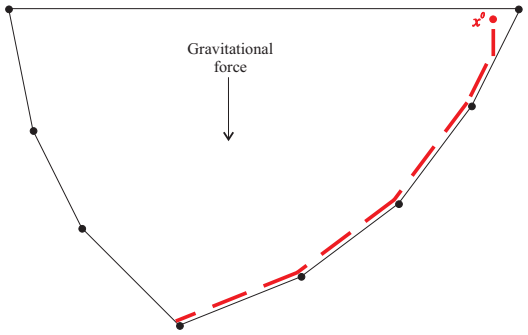

Fig. 2. A 2-dimensional polytope and its faces on which the ball rolls down (dashed path) to the optimum.

$$\begin{aligned} &\text{maximize} \quad \pi b \\ &\text{subject to} \quad \pi A = c, \quad \pi \geq 0 \end{aligned} \tag{2}$$

where A is a matrix of order $m \times n$, $\pi \in R^m$ is the row vector of primal variables.
As explained in Section 1, typically $n \leq m$. Its dual is

$$\begin{aligned} &\text{minimize} \quad z(x) = cx \\ &\qquad \text{subject to} \quad Ax \geq b \end{aligned} \tag{3}$$

where $x \in R^n$ is the column vector of dual variables.

The method is applied on (3). We denote its feasible region $\{x : Ax \geq b\}$ by K, and its interior $\{x : Ax > b\}$ by K^0. The method needs an initial *interior point* $x^0 \in K^0$. It introduces a spherical drop (we will refer to it as the *drop* or the *ball*) of small radius with center x^0 lying completely in the interior of K, and traces the path of its center as the drop falls under a gravitational force pulling it in the direction $-c^T$. The drop cannot cross the boundary of K, so after an initial move in the direction $-c^T$ it will be blocked by the face of K that it touches; after which it will start rolling down along the faces of K of varying dimensions. Hence the center of the drop will follow a piecewise linear descent path completely contained in the interior of K, but since the drop's radius is small, the center remains very close to the boundary of K after the first change in direction in its path. Therefore the method is essentially a boundary method. However, unlike the simplex method which follows a path strictly along the one dimensional boundary of K, this method is a **higher dimensional boundary method** in which the path followed remains very close to faces of K of varying dimensions. See Figures 2, 3, for 2-, 3-dimensional illustrations.

After a finite number of changes in the direction of movement, the drop will reach the lowest possible point in the direction $-c^T$ that it can reach within K and

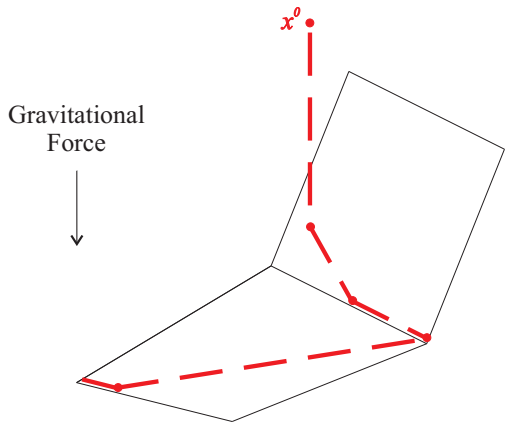

Fig. 3. The ball rolling (dashed path, with dots indicating where its direction changes) inside a 3-dimensional polyhedron. Only the faces along which it rolls to the optimum are shown.

then halt. If the radius of the drop is sufficiently small, the touching constraints (i.e., those whose corresponding facets of K are touching the ball) in (3) at this final halting position will determine an actual optimum solution of the LP (2). If its radius is not small enough, the direction finding step in the method at the final halting position with center x^* yields a feasible solution $\tilde{\pi}$ of (2) and the optimum objective value in (2) lies in the interval $[\tilde{\pi}b, cx^*]$. Then the radius of the drop is reduced and the method continues the same way. In [3] finite termination of the method to find an optimum solution has been proved.

Morin, Prabhu, and Zhang [9] have shown that this version of the gravitational method using point-drops (i.e., drops of radius 0) has exponential growth in the worst case just as the simplex method does.

How to Make the Gravitational Method Efficient?: It is clear that in order to make the gravitational method efficient, it is necessary to keep the center of the drop from hugging the boundary of K all along its path, i.e., make the method a truly interior point method. This can be achieved by making the radius of the drop as large as possible by moving its center to the center of the set of feasible solutions. For this we develop a new centering strategy (discussed briefly in Murty[15]) that is very different from centering strategies used in other IPMs. Its biggest advantage is that it can be solved approximately by a method that needs no matrix inversions, and hence is lot simpler than other centering strategies. We will describe this centering strategy next.

6. The Centering Strategy

We assume that an initial interior feasible solution $x^0 \in K^0$ for (3) is available. If such an initial point is not available, we modify the problem using the usual big-M augmentation with one artificial variable as follows

$$\begin{aligned} &\text{minimize} \quad cx + Mx_{n+1} \\ &\text{subject to} \quad Ax + ex_{n+1} \geq b, \quad x_{n+1} \geq 0 \end{aligned}$$

where $e = (1, ..., 1)^T \in R^m$ and M is a positive number significantly larger than any other number in the problem. Let $x^0_{n+1} > \max\{0, b_1, ..., b_m\}$. Then $(0, ..., 0, x^0_{n+1})^T$ is a strict interior feasible solution of the modified problem which is in the same form as (3).

So, we assume that a strict interior feasible solution x^0 of (3) is available. We also assume that $c \neq 0$, as otherwise 0 is already an optimum solution of (2). We normalize c, so that $||c|| = 1$. We also assume that $||A_{i.}|| = 1$ for all i.

Without any loss of generality, we assume that each constraint in (3) determines a facet of K. This is just for simplicity, the algorithm discussed is not affected by any redundant constraints in (3) which do not correspond to facets of K. For i = 1 to m let $H_i = \{x : A_{i.}x = b_i\}$ be the i-th facetal hyperplane for K. Let $H^0 = \{x : cx = cx^0\}$ be the objective hyperplane through the current point x^0.

Since x^0 is in the interior of K, $A_{i.}x^0 > b_i$ for all i = 1 to m. Then $\delta^0{}_i = A_{i.}x^0 - b_i$ is the distance (Euclidean) of x^0 from H_i. With x^0 as center, the largest sphere we can construct within K has a radius $\min\{\delta^0_i : i = 1 \text{ to } m\}$. This may be too small. To construct even larger drops inside K, we need to move the center of the drop from x^0 to a better interior point. Starting with x^0, the centering strategy tries to find a new position for the center of the drop inside K^0 that maximizes the radius of the drop that can be constructed within K. It does this while keeping the objective value at the new center the same as that at x^0, by including $cx = cx^0$ as a constraint that the new center x has to satisfy.

So the new center x is chosen from $K^0 \cap H^0$ and maximizes $\min\{A_{i.}x - b_i : i = 1 \text{ to } m\}$. The model for this choice is:

$$\begin{aligned} &\text{Maximize} \quad \delta \\ &\text{subject to} \quad \delta \leq A_{i.}x - b_i, \quad i = 1 \text{ to } m \\ &\qquad\qquad cx = cx^0 \end{aligned} \tag{4}$$

This is another LP with variables (δ, x). It may have alternate optimum solutions with different x-vectors, but the optimum value of δ will be unique. If $(\bar{x}, \bar{\delta})$ is

an optimum solution for it, $\bar{x}$ is taken as the new center for the drop, and $\bar{\delta}$ is the maximum radius for the drop within K^0 subject to the constraint that its center lie on $K^0 \cap H^0$.

But this itself is another LP, this type of model may have to be solved several times before we get a solution for our original LP, so solving this model (4) exactly will be counterproductive. But (4) has a very special structure, using it we developed the following procedure to get an approximate solution for it.

6.1. *Procedure for Getting an Approximate Solution for (4)*

In this procedure for finding the new center $x \in K^0 \cap H^0$, we only consider moves in directions perpendicular to the facetal hyperplanes of K, since our goal is to increase the minimum distance of x from a facetal hyperplane. These directions (with positive or negative step lengths) are: $A_{i.}^T$ for i = 1 to m. However since x has to lie on H^0, the actual directions of movement considered are: $P_{.i} = A_{i.}^T - c^T c A_{i.}^T = (I - c^T c)A_{i.}^T$, where $P_{.i}$ is the orthogonal projection of $A_{i.}^T$ on $\{x : cx = 0\}$, for i = 1 to m. Here I is the unit matrix of order n. For an illustration see Figure 4.

So, this procedure consists of a series of moves beginning with x^0, generating a sequence of points $x^0, \ldots, x^r, \ldots$ in $K^0 \cap H^0$. When the current point is x^r we do a line search in one of the directions from $\{P_{.1}, \ldots, P_{.m}\}$ for the position for the next center that will help increase the radius of the ball inside K as much as possible.

Let x^r be the current point. In optimization literature, when considering a line search at x^r in the direction $P_{.i}$, only moves of positive step length α leading to the point $x^r + \alpha P_{.i}$ are considered. Here our step length α can be either positive or negative, so even though we mention $P_{.i}$ as the direction of movement, the actual direction for the move may be either $P_{.i}$ or $-P_{.i}$. With $x^r + \alpha P_{.i}$ as the center, the maximum radius of a ball inside K has radius

$$f_{ir}(\alpha) = \min\{A_{t.}(x^r + \alpha P_{.i}) - b_t : t = 1, \ldots, m\}$$

Since we want the largest ball inside K with its center in $K^0 \cap H^0$, we will call a direction $P_{.i}$ to be a *profitable direction* to move at x^r if $f_{ir}(\alpha)$ increases as α changes from 0 to positive or negative values (i.e., $\max\{f_{ir}(\alpha)$ over $\alpha\}$ is attained at some $\alpha \neq 0$). Likewise, $P_{.i}$ is *unprofitable direction* to move at x^r if $\max\{f_{ir}(\alpha)$ over $\alpha\}$ is attained at $\alpha = 0$).

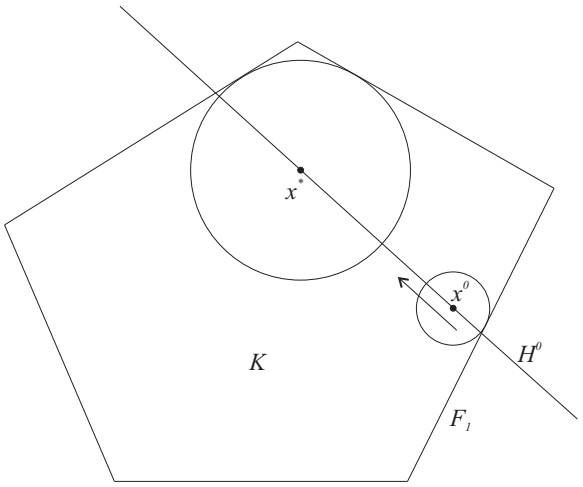

Fig. 4. Moving the center from x^0 along the direction $P_{.1}$ to x^*, leads to a larger ball inside K.

For each direction $P_{.i}$, the value of α that maximizes $f_{ir}(\alpha)$ is known as the optimum step length for it. So, for unprofitable directions at x^r, the optimum step length is 0; for profitable directions it will be nonzero and may be positive or negative.

Let $\bar{\alpha}$ be the optimum step length for direction $P_{.i}$, and let $\bar{\theta} = f_{ir}(\bar{\alpha})$ be the maximum value of $f_{ir}(\alpha)$. Then $(\bar{\theta}, \bar{\alpha})$ is an optimum solution of the following 2-variable LP in which the variables are θ, α.

$$\begin{array}{l} \text{Maximize } \theta \\ \quad \text{subject to} \\ \theta - \alpha A_{t.} P_{.i} \leq A_{t.} x^r - b_t, \ t = 1, \ldots, m \\ \quad \theta \geq 0, \quad \alpha \quad \text{unrestricted in sign.} \end{array} \tag{5}$$

The radius of the largest ball inside K with the current point x^r as center is $\delta_r = \min\{A_{t.}x^r - b_t : t = 1 \text{ to } m\}$, the minimum RHS constant in (5). Let $T = \{t_1, \ldots, t_s\}$ be the set of all t that tie for the minimum in the definition of δ_r. So, the minimum RHS constant in (5) is unique only if $|T| = s = 1$, in this case it is attained for $t = t_1$ only. The following theorem gives a criterion to check very efficiently whether a direction $P_{.i}$ is a profitable or unprofitable direction at x^r.

Theorem 2 *$P_{.i}$ is an unprofitable direction to move at the current point x^r iff: $s = |T|$ defined above is > 1, and the coefficients of α in (5) in rows $t \in T = \{t_1, \ldots, t_s\}$ have both positive and negative values among them.*

Proof: Let Γ denote the set of feasible solutions of (5) in the (θ, α)-space. If the conditions stated in the

theorem are satisfied, $(\delta_r, 0)$ is an extreme point of Γ, and it can be verified that as we move away from this extreme point along either of the two edges of Γ incident at this extreme point, θ decreases. So, when these conditions are satisfied, $(\delta_r, 0)$ is the optimum solution of (5), hence the result in the theorem holds.

Suppose the conditions stated in the theorem are not satisfied, i.e., either $s = 1$, or $s > 1$, but all the coefficients of α in (5) in rows $t \in T$ have the same sign. Then it can be verified that as you move away from $(\delta_r, 0)$ along one of the edges of Γ on which it lies, θ increases. Hence in this case the maximum value of θ in (5) is strictly $> \delta_r = f_{ir}(0)$, so $P_{.i}$ is a profitable direction to move at x^r. ■

If $P_{.i}$ is a profitable direction to move at x^r, the optimum step length for the move can be found by the following procedure. Transform (5) into standard form. Let $u_1, \ldots, u_m$ denote the slack variables corresponding to the constraints in (5) in this order. Then $(u_1, \ldots, u_{t_1-1}, \theta, u_{t_1+1}, \ldots, u_m)$ is a feasible basic vector for this standard form. The BFS corresponding to this basic vector for the standard form corresponds to the extreme point $(\delta_r, 0)$ of Γ in the (θ, α)-space. Starting from this feasible basic vector, the optimum solution of (5) can be found efficiently by the primal simplex algorithm with at most $O(m)$ effort. It may be possible to develop even more efficient ways for finding the optimum value of α in (5), that value is the optimum step length for the move at x^r in the profitable direction $P_{.i}$.

When the current point is x^r, the centering procedure can end in the following way:

(i): If the condition in Theorem 2 determines that there is no profitable direction to move at x^r in $\{P_{.1}, \ldots, P_{.m}\}$, then we terminate the centering procedure and take x^r as the center of the next ball and δ_r as its radius.

(ii): If profitable directions to move at x^r exist in $\{P_{.1}, \ldots, P_{.m}\}$, we select one of them to move, say $P_{.i}$, and find the optimum step length by solving (5). Let it be $\bar{\alpha}$. If $\bar{\alpha} = \infty$, then the objective value in LP (3) is unbounded below, and LP (2) is infeasible, terminate the whole process.

If $\bar{\alpha}$ is finite, $x^{r+1} = x^r + \bar{\alpha} P_{.i}$ is the next point in the sequence. Let $\delta_{r+1} = \min\{A_{i.}x^{r+1} - b_i : i = 1 \text{ to } m\}$. If $\delta_{r+1} - \delta_r$ is larger than some selected tolerance, take x^{r+1} as the current point in the sequence and repeat the centering procedure with it. If $\delta_{r+1} - \delta_r$ is smaller than the selected tolerance, then terminate the centering procedure and take x^{r+1} as the center for the next ball and δ_{r+1} as its radius.

When there are several profitable directions to move at the current point x^r in the set $\{P_{.1}, \ldots, P_{.m}\}$ in this procedure, efficient selection criteria to choose the best among them can be developed. In fact the best may be among the $P_{.i}$ that correspond to i that tie for the minimum in $\delta_r = \min\{A_{i.}x^r - b_i : i = 1 \text{ to } m\}$, or a weighted average of these directions (even though this direction is not included in our list of directions to pursue).

As can be seen, the procedure used in this centering strategy does not need any matrix inversion, and only solves a series of 2-variable LPs which can be solved very efficiently. Hence this centering strategy can be expected to be efficient.

We define the *path of centers* to be the path of the center of the drop in its descent to the optimal face of (3) in this algorithm.

Numerical example: We consider the following LP in 2 variables x_1, x_2 to illustrate the centering strategy, both numerically and with a figure.

$$\begin{aligned} \text{Minimize} \quad & z = -15x_1 - 10x_2 \\ \text{subject to} \quad & 1500 - 2x_1 - x_2 \geq 0 \\ & 1200 - x_1 - x_2 \geq 0 \\ & 500 - x_1 \geq 0 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \end{aligned}$$

Normalizing the coefficient vectors of all the constraints and the objective function to Euclidean norm 1, here it is again:

$$\begin{aligned} \text{Minimize} \quad & z = -0.832x_1 - 0.555x_2 \\ \text{Subject to} \quad & 670.820 - 0.894x_1 - 0.447x_2 \geq 0 \\ & 848.530 - 0.707x_1 - 0.707x_2 \geq 0 \\ & 500 - x_1 \geq 0 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \end{aligned} \tag{6}$$

Let K denote the set of feasible solutions, and let $x^0 = (10, 1)^T$ be the initial interior feasible solution. When we plug in x^0 in the constraints in (6), the left hand side expressions have values 661.433, 840.753, 490, 10, 1 respectively. So, the radius of the largest ball inside K with x^0 as center is $\delta^0 = \min\{661.433, 840.753, 490, 10, 1\} = 1$.

The objective plane through x^0 is the straight line in R^2 defined by $-0.832x_1 - 0.555x_2 = -8.875$. This is

the straight line joining $(10.667, 0)^T$ and $(0, 15.991)^T$ in the x_1, x_2-plane. So, the only direction on it is $P_{.1} = (10.667, -15.991)^T$. Moving from x^0 in the direction of $P_{.1}$ a step length α leads to the new point $(10 + 10.667\alpha, 1 - 15.991\alpha)^T$. Finding the optimum step length α leads to the following 2-variable LP in variables θ, α:

θ	α		
1	2.388	$\leq$	661.433
1	-3.765	$\leq$	840.753
1	10.667	$\leq$	490
1	-10.667	$\leq$	10
1	15.991	$\leq$	1
1	0		Maximize

$\theta \geq 0$, α unrestricted

Since the minimum RHS constant in this problem occurs in only one row, from Theorem 2 we know that the optimum value of α in this problem will be nonzero. Actually the optimum solution of this problem is $(\bar{\theta}, \bar{\alpha})^T = (6.4, -0.338)^T$. See Figure 6. The new position for the center is $x^1 = x^0 - 0.338P_{.1} = (10, 1)^T - 0.338(10.667, -15.991)^T = (6.4, 6.4)^T$, and the maximum radius ball with it as center has radius 6.4. Since $P_{.1}$ is the only direction in $K \cap \{x : cx = cx^0\}$ in this case, this ball is the maximum radius ball inside K with center on the objective plane through x^0.

If we try to get a larger ball by moving from x^1 in the direction $P_{.1}$ a step length of α, it can be verified that in the 2-variable LP to find the optimum step length α, the entries in the RHS vector are: 662.238, 839.48, 493.6, 6.4, 6.4; and the coefficient vector of α remains the same as in the above table. In this problem the minimum RHS constant occurs in both rows 4 and 5; and the coefficients of α in these two rows have opposite signs, indicating by Theorem 2 that the optimum value for step length α will be 0. This illustrates Theorem 2, and indicates that x^1 is the best position for the center of the ball on the objective plane through x^0 in this problem.

7. Stage 1, Repetitions of Iteration 1 in the Gravitational Interior Point Method to Solve (3), (2)

By fixing the initial drop as the largest possible ball with its center in $K^0 \cap H^0$, the centering strategy pushes the center of the ball close to the center of $K^0 \cap H^0$. Stage 1 of the overall method consists of repetitions of a special iteration that exploits this property to get as much reduction in the objective value of (3) as possible using cheap computations consisting of the following two steps repeatedly. These steps in this iteration are described below, some changes in the second step to accelerate convergence will be discussed later. This iteration begins with x^0 as the initial interior feasible solution.

Iteration 1

Select a small positive number ϵ as the tolerance for minimum$\{A_{i.}x - b_i : i = 1 \text{ to } m\}$ for the center x to be in the interior of K.

Step 1.1: Centering: Let $H^0 = \{x : cx = cx^0\}$. Starting with x^0 apply the centering strategy of Section 6 to get the largest ball $B(x^*, \delta)$ with $x^* \in K^0 \cap H^0$ as center and δ as radius. Go to Step 1.2.

Step 1.2: Descent Move Following Centering: This move does not use the ball $B(x^*, \delta)$ constructed in Step 1.1 at all; it only uses its center x^* and its property of being close to the center of $K^0 \cap H^0$. It takes a maximum possible step from x^* in a descent direction for cx.

If this is the first time this step is being carried out, the only descent direction that we have readily available at this time is $-c^T$, and we use that as the direction to move from x^*.

If this is not the first time this step is being carried out, besides $-c^T$ we have another descent direction for cx namely the direction of the path of centers at the current point x^*, which can be approximated by $x^* - \tilde{x}$ where $\tilde{x}$ is the center of the drop when Step 1.2 was last carried out. See Figure 5.

If $d \in \{-c^T, x^* - \tilde{x}\}$ is the direction selected for moving from x^*, we will move in this direction the maximum distance possible while still remaining inside K^0. This gives

$$\bar{x} = x^* + \gamma d$$

$$\gamma = \min\{\frac{-A_{i.}x^* + b_i + \epsilon}{A_{i.}d} : i \text{ such that } A_{i.}d < 0\}$$

If $\gamma = \infty$, the objective function is unbounded below in (3), and (2) is infeasible, terminate the algorithm.

If γ is finite, the decrease in the objective value in this move is $|\gamma cd|$. Select the direction d from $\{-c^T, x^* - \tilde{x}\}$ to be the one which yields the maximum decrease in the objective value in this move. Make $\bar{x}$ obtained after the move the new x^0, and go back to Step 1.1 for another repeat of this Iteration 1.

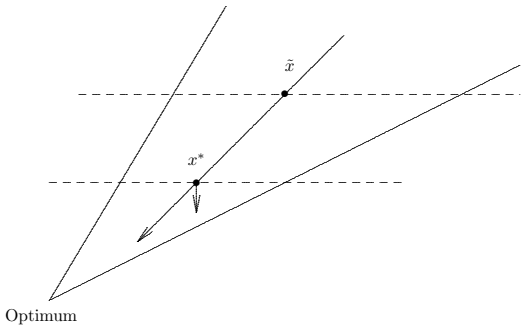

Fig. 5. The two descent directions to move in Step 1.2 when the center is at x^* in an iteration. One is $x^* - \tilde{x}$ where $\tilde{x}$ is the center in the previous iteration, another is $-c^T$ (here shown as pointing down south). The dashed lines are the objective planes in the two iterations.

7.1. *Zigzagging*

Repetitions of this Iteration 1 may encounter *zigzagging* (a phenomenon commonly discussed in nonlinear programming) if the direction to move from x^* is always taken to be $-c^T$. Zigzagging occurs when the center of the ball gets trapped in a narrow cone-like region, with successive balls having the same "touching set" of constraints repeatedly. If this occurs (likely when the center nears the optimal face, if the optimal face is of low dimension) each successive repetition of Iteration 1 makes progressively decreasing improvements. Taking the direction to move in each repetition of Iteration 1 to be the better of $\{-c^T, x^* - \tilde{x}\}$ helps to prevent zigzagging from occurring.

7.2. *Other Descent Directions*

Suppose at this time in Stage 1, Step 1.2 has been carried out r times. Let x^k denote the center of the drop when Step 1.2 is carried out the kth time for k = 1 to r. Then $x^q - x^p$ is a descent direction for the objective function in (3) for all $1 \leq p < q \leq r$. Among all these descent directions, the ones obtained using recent pairs of centers may have useful information about the shape of the feasible region between the objective value at present and at its optimum. So, using a weighted average of these descent directions as the direction to move next (instead of using either $-c^T$ or $x^r - x^{r-1}$ as discussed above) may help in maximizing the improvement in the objective value in this move. The best weighted average to use for maximum practical effectiveness can be determined using computational experiments.

We continue repeating applications of Iteration 1 until the improvement in the objective value in each application becomes small. Let $\bar{x}$ denote the interior feasible solution in this final repetition of Iteration 1 in Stage 1.

If the centering strategy performs well, this final solution $\bar{x}$ in Stage 1 can be expected to be such that $c\bar{x}$ is quite close to the minimum objective value in (3) (i.e., $\bar{x}$ can be expected to be a near optimum to (3)). Several efficient strategies developed in LP theory are available to get an approximate optimum to (2), (3) from $\bar{x}$.

Stage 1 has the aim of getting as close to the optimum as possible without necessitating matrix inversions. The final point obtained in Stage 1 may itself be a reasonable approximation to the optimum in some practical applications.

Numerical example: We will illustrate one iteration of Stage 1 on the 2-variable LP (6) with decision variables x_1, x_2 given in Section 6. There we started with the initial feasible solution $x^0 = (10, 1)^T$ for the problem, applied the centering strategy which moved the center to $x^1 = (6.4, 6.4)^T$. In this initial iteration of Stage 1 on this problem, the only descent direction we have available at x^1 is $-c^T = (0.832, 0.555)^T$. Moving from x^1 a step length γ in the direction $-c^T$ leads to the point $(6.4 + 0.832\gamma, 6.4 + 0.555\gamma)^T$. Taking the tolerance ϵ = 1, we see that the maximum step length is γ = min$\{666.571, 854.72, 592.067\} = 592.067$. Fixing $\gamma = 592.067$, we get the new interior feasible solution $\bar{x} = (499, 335)^T$.

With $\bar{x}$, we need to go to the next iteration in Stage 1 and continue in the same way. Figure 6 illustrates both the centering step carried out in Section 6 beginning with the initial interior feasible solution x^0, and the descent move carried out here.

If a true optimum solution of (2), (3) is needed, starting from the final point $\bar{x}$ obtained at the end of Stage 1 as the current point we go to Stage 2 which carries out applications of the general iteration, Iteration 2, that consists of the following steps besides the centering step:

- Gravitational direction finding step.
- Step length determination and the main move.
- Additional move of the center.
- What to do if the ball halts.

There are several possible options for selecting the gravitational direction along which the ball will move, we will first discuss these in detail next.

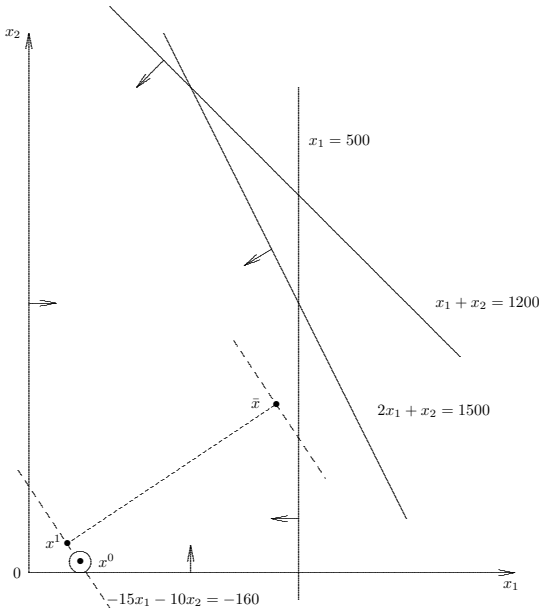

Fig. 6. Figure (not drawn to scale) shows feasible region K with 5 facets, each has an arrow pointing its feasible side. Only a small sphere of radius 1 can be drawn inside K with initial point x^0 as center. Dashed line through x^0 is the objective plane, centering strategy moves point to $x^1 = (6.4, 6.4)^T$ on this plane. With x^1 as center, a sphere of radius 6.4 can be inscribed inside K. The descent move from x^1 in Step 1.2 in direction $-c^T$ (dotted line) leads to $\bar{x} = (499, 335)^T$ with objective value -10835. The dashed line through $\bar{x}$ is the objective plane $\{x : -15x_1 - 10x_2 = -10835\}$. Another iteration of Stage 1 begins with $\bar{x}$.

8. The Gravitational Direction Finding Step

In the versions of the gravitational methods discussed in [3, 11], the initial ball is always selected to have a very small radius so that it is completely inside K^0 without any of the boundary faces of K touching it, so the initial move in the method always takes place in the direction $-c^T$ and could be of very short length depending on the location of the center of the initial ball. But here, the centering step in Section 6 used for selecting the initial ball makes sure that it is already touching some facets of K; these are called the *touching facets*, and the constraints that define them are called the *touching constraints*. Let:

$B(x^*, \delta) =$ the current ball inside K with center x^* and radius δ

$J(x^*, \delta) = \{i : A_{i.}x^* = b_i + \delta\}$, the index set of touching constraints for $B(x^*, \delta)$

$Q =$ the matrix consisting of rows $A_{i.}$ for $i \in J(x^*, \delta)$

$G(x^*, \delta) = \{y : cy < 0, A_{i.}y \geq 0$ for all $i \in J(x^*, \delta)\}$, the set of descent feasible directions for $B(x^*, \delta)$.

The gravitational direction at x^* is a direction selected from $G(x^*, \delta)$ along which the entire current ball $B(x^*, \delta)$ will be moved. Various options for selecting this direction are given below.

8.1. *The Steepest Descent Gravitational Direction (SDGD)*

Defined in [3] and used in the SDGM (Steepest Descent Gravitational Method) discussed there, this is the steepest descent direction among all those in $G(x^*, \delta)$. So, the SDGD is the optimum solution of

$$\begin{aligned} &\text{Minimize} && cy \\ &\text{subject to} && Qy \geq 0 \\ &&& 1 - y^T y \geq 0 \end{aligned} \tag{7}$$

In [3] it has been proved that this problem is equivalent to the problem

$$\begin{aligned} &\text{Minimize} && (c - \eta Q)(c - \eta Q)^T \\ &\text{subject to} && \eta \geq 0 \end{aligned} \tag{8}$$

which is a nearest point problem (finding nearest point to c in Rpos(Q) = the nonnegative hull of row vectors in the matrix Q defined above).

If $\bar{\eta}$ is an optimum solution of (8), then $\bar{y}$ = 0 if $\bar{\xi} = (c - \bar{\eta}Q) = 0$, or $-\bar{\xi}^T/||\bar{\xi}||$ otherwise, is an optimum solution of (7).

Also, if $\bar{\xi} = 0$, $G(x^*, \delta) = \emptyset$, i.e., the ball $B(x^*, \delta)$ cannot move from its present position in gravitational descent, hence it halts in its present position. If this happens, let $\bar{\pi} = (\bar{\pi}_i)$ where $\bar{\pi}_i = \bar{\eta}_i$ if $i \in J(x^*, \delta)$, 0 otherwise. Then $\bar{\pi}$ is feasible to (2), and the optimum objective value in (2) lies in the interval $[\bar{\pi}b, cx^*]$. In this case the method goes to the step to carry out when the ball halts, discussed in Section 9.

If $\bar{\xi} \neq 0$, go to the main move step with $\bar{y}$ as the gravitational direction for the move. Now the method

goes to the main move discussed in Step 2.3 of Section 9.

8.2. *Modified Gravitational Directions MGD1, MGD2*

Computing the SDGD becomes simplified if the cone Rpos(Q) is simplicial, i.e., if Q is of full row rank, which may not be the case always. So, in [3], simplified versions of gravitational directions MGD1, MGD2 are discussed. In these versions, the nearest point problem (8) for finding the gravitational direction is modified by replacing the matrix Q by a submatrix D of it consisting of a maximal linearly independent subset of row vectors of Q. So, computing MGD1 requires the solution of the nearest point problem

$$\begin{array}{ll} \text{Minimize} & (c-\eta D)(c-\eta D)^T \\ \text{subject to} & \eta \geq 0 \end{array} \tag{9}$$

which is the problem of finding the nearest point in Rpos(D) to c.

So, MGD1 is the direction obtained as in the SDGD, with $\bar{\eta}$ being the optimum solution of (9) instead of (8). (9) can be solved very efficiently by geometric methods discussed in [12, 13, 17, 18, 19] using the concept of projection faces of the simplicial cone Rpos(D).

MGD2 simplifies the effort needed to find the gravitational direction even further by taking the vector $\bar{\eta}$ to be not the optimum solution of (9), but the one corresponding to a projection face of Rpos(D) that is closer than the initial one.

8.3. *The Gradient Projection Direction (GPD)*

Define $y^0 = -c^T$, and $T = J(x^*, \delta)$. Computing this direction defined in [11] involves the following steps:

Define $J(y^0) = \{i : i \in T, A_{i.}y^0 < 0\}$. Each of the constraints $A_{i.}x \geq b_i$ for $i \in J(y^0)$ is currently blocking the movement of the ball in the direction y^0, so $J(y^0)$ is called the *index set of blocking constraints*.

If $J(y^0) = \emptyset$, take the GPD to be y^0.

If $J(y^0) \neq \emptyset$, . Let E be a submatrix of A consisting of rows $A_{i.}$ which form a maximal linearly independent subset of $\{A_{i.} : i \in J(y^0)\}$, and let $P \subset J(y^0)$ be $\{i : A_{i.}$ is a row of $E\}$. Let ξ be the orthogonal projection of y^0 in the subspace $\{x : Ex = 0\}$, so $\xi = ((I - E^T(EE^T)^{-1}E)y^0$.

If $\xi \neq 0$, the GPD is ξ.

If $\xi = 0$, let $\mu = -(EE^T)^{-1}Ey^0 = (\mu_i : i \in P)$. Then $\mu^T E = c$. So if $\mu \geq 0$, then $\bar{\pi} = (\bar{\pi}_i)$ defined by $\bar{\pi}_i = 0$ if $i \notin P$, $= \mu_i$ if $i \in P$, is a BFS of (2). In this case the ball $B(x^*, \delta)$ halts in its present position and cannot move any further under the gravitational force. In this case the method goes to the step to carry out when the drop halts (Step 2.5), discussed in Section 9.

If $\xi = 0$ and $\mu \not\geq 0$, then delete the i corresponding to the most negative μ_i from the set T, and repeat all this work with the new T.

9. The Gravitational Interior Point Method for LP

Here we state the whole method.

Stage 1: This stage consists of repeated applications of Iteration 1.

Iteration 1: Initial Iteration: Starting with the initial interior feasible solution x^0, apply Iteration 1 (Steps 1.1, 1.2) described in Section 7 repeatedly until the improvement in the objective value cx in each application becomes small.

Let $\bar{x}$ denote the interior feasible solution obtained at the end of Stage 1. With $\bar{x}$ go to Stage 2.

Stage 2: This stage consists of repeated applications of Iteration 2 starting with the point obtained at the end of Stage 1.

Iteration 2: General Iteration: The first application of this iteration begins with $\bar{x}$, the interior feasible solution obtained at the end of Stage 1.

Step 2.1: Centering: Let $\bar{H} = \{x : cx = c\bar{x}\}$. Starting with $\bar{x}$, apply the centering strategy discussed in Section 6 to get the largest ball $B(x^*, \delta)$ with $x^* \in K^0 \cap \bar{H}$ as center and δ as radius. Go to Step 2.2.

Step 2.2: Gravitational Direction Finding: Find the gravitational direction at x^* along which the current ball will be moved, using one of the options described in Section 8 (the options discussed there are: SDGD, MGD 1, MGD 2, and GPD). If the ball halts, go to Step 2.5. Otherwise denote the gravitational direction selected by $\bar{y}$, go to Step 2.3 with it.

Step 2.3: Step Length Determination and the Main Move: Let $\bar{y}$ denote the gravitational direction selected for the move of the ball $B(x^*, \delta)$. The step length is the maximum distance the ball can move in this direction while still remaining completely within K. So, this step length in the direction $\bar{y}$ is:

$$\theta = \min\{\frac{A_{i.}x^* - b_i - \delta}{-A_{i.}\bar{y}} : i \in J(\bar{y})\}$$

where $J(\bar{y}) = \{i : A_{i.}\bar{y} < 0\}$, the blocking set of constraints corresponding to the direction $\bar{y}$.

If $\theta = \infty$, the objective function is unbounded below in (3), and (2) is infeasible, terminate the algorithm.

If θ is finite, move the present ball $B(x^*, \delta)$ to $B(x^* + \theta\bar{y}, \delta)$ and go to Step 2.4.

Step 2.4: The Additional Move: Suppose the main move in this iteration has moved the ball to the new position $B(x^* + \theta\bar{y}, \delta)$. The center of this ball $x^* + \theta\bar{y}$, is strictly in the interior of K. Now get rid of this ball, and take a maximum possible step from its center $x^* + \theta\bar{y}$ in the direction $-c^T$ while still remaining inside K^0. The maximum step length in this move is:

$$\gamma = \min\{\frac{A_{i.}(x^* + \theta\bar{y}) - b_i - \epsilon}{A_{i.}c^T} : \quad i \quad \text{such that} \quad A_{i.}c^T > 0\}$$

So, now we move from $x^* + \theta\bar{y}$ to the point $\bar{x} = x^* + \theta\bar{y} - \gamma c^T$.

With this $\bar{x}$ as the new interior point solution, go back to Step 2.1 to repeat this Iteration 2.

Step 2.5: When the Ball Halts: Since the ball $B(x^*, \delta)$ has halted, it cannot move from its present position under the gravitational force, because its movement is blocked by the blocking constraints among the touching constraints. In this case we obtain a feasible solution $\bar{\pi}$ for (2) as described in Section 8. Let $F = \{i : \bar{\pi}_i > 0\}$, and $E \subset F$ such that $\{A_{i.} : i \in E\}$ is a maximal linearly independent subset of $\{A_{i.} : i \in F\}$, and $d = (b_i : i \in E)$. Let $\hat{x} = \bar{x} + E^T(EE^T)^{-1}(d - E\bar{x})$, the orthogonal projection of $\bar{x}$ on the flat $\{x : A_{i.}x = b_i, i \in E\}$. If $\hat{x}$ is feasible to (3), then it is optimal to (3), and $\bar{\pi}$ is optimal to (2), terminate the algorithm.

Suppose $\hat{x}$ is not feasible to (3). Here the center x^* of $B(x^*, \delta)$ is at the center of K on the present objective plane. Make the additional move at x^* as in Step 2.4, and continue.

10. Convergence Proofs

In this section we give convergence proofs for the algorithm consisting of Stages 1, 2; and the algorithm consisting of Stage 1 alone under the assumption that the centering strategy is carried to optimality.

Theorem 3 *Consider the following variant of problem (4) with the value of cx as a parameter t.*

$$\begin{aligned} \delta(t) \quad &= \textit{ maximum value of } \delta \\ \textit{subject to} \\ &\delta - A_{i.}x \le -b_i, \quad i = 1 \textit{ to } m \\ &cx = t \end{aligned} \tag{10}$$

Here $\delta(t)$, the optimum objective value in this problem as a function of the parameter t, is a piecewise linear concave function of t in the interval of t for which the problem has feasible solutions.

Proof: $\delta(t)$ is the optimum objective value in a parametric right hand side linear program in which the objective function is to be maximized. It is well known that $\delta(t)$ is a piecewise linear concave function [10]. ■

$\delta(t)$ is the maximum radius of a sphere that can be inscribed in K with center in $\{x : cx = t\}$. Let

$$\begin{aligned} \bar{t} &= \max\{cx : x \in K\} \\ \underline{t} &= \min\{cx : x \in K\} \\ t^* &= \text{ the value of } t \text{ where } \delta(t) \\ &\quad \text{attains its maximum value.} \end{aligned}$$

From Theorem 3 we know that $\delta(t)$ satisfies the following properties.

(i): $\delta(t)$ is monotonic increasing as t increases from $\underline{t}$ to t^*, and from t^* it is monotonic decreasing as t increases on to $\bar{t}$.

(ii): If the centering strategy is carried to accuracy, the radii of the spheres encountered in Steps 1.2, 2.2 in the algorithms of Sections 7, 9 may increase and peak in the beginning, after crossing the peak they will be monotonic decreasing.

Theorem 4 *Starting from an interior feasible solution for (3), if the centering strategy is carried to accuracy, the algorithm of Section 9 consisting of Stages 1, 2 converges to optimum solutions of the LPs (2), (3) if they exist, or conclude that the objective value is unbounded in (3). Also, the Stage 1 algorithm of Section 7 alone converges to an optimum solution of (3) if it exists, or conclude that the objective value is unbounded below in it.*

Proof: From (ii) above we know that the radii of the balls encountered increase in the beginning until reaching a peak. Therefore, during this phase, moving

in descent directions leads to increasing reductions in the objective value.

Once the radii of the balls begins to decrease, they are monotonically decreasing. Using this, the theorem follows from the convergence proof of the gravitational method discussed in [3]. ■

These results show the convergence of both the algorithms. But we will now show that the Stage 1 algorithm by itself has much stronger convergence properties.

Definitions: The *set of touching constraints at* t is the set of all constraints in (10) satisfied as equations by any of the optimum solutions of (10).

The *essential touching constraint index set at* t is the set $J(t) = \{i : A_{i.}x = b_i + \delta(t)\}$ for every optimum solution (δ, x) of (10)$\}$. The i-th constraint in (3), (10) is said to be in the set of essential touching constraints at t if $i \in J(t)$.

For the rest of this section we will assume that the center selected in the centering strategy is an $x(t)$ satisfying the property that the facets of K touching the ball $B(x(t), \delta(t))$ (the ball with $x(t)$ as center and $\delta(t)$ as radius) are those corresponding to the essential touching constraint set at t. Then for any t, $\delta(t)$ = Euclidean distance of $x(t)$ to any of the constraints of (3) corresponding to the index set $J(t)$. There may be different points satisfying this assumption, but all of them lead to the same value for $\delta(t)$ and hence a ball of the same size inside K.

Theorem 5 *If $J(t)$ remains the same for all $t_1 \leq t \leq t_2$, then $\delta(t)$ is linear in this interval.*

Proof: For $t = t_1 + \alpha(t_2 - t_1)$, $0 \leq \alpha \leq 1$, the point $x(t_1) + \alpha(x(t_2) - x(t_1))$ along the line segment joining $x(t_1)$ and $x(t_2)$ satisfies the assumption made for the center $x(t)$. By the hypothesis $J(t) = J(t_2)$ for all t in this interval. So for any t in this interval, $\delta(t)$ is the Euclidean distance from this point to the i-th facetal hyperplane of K for any $i \in J(t_2)$, and hence is linear; i.e., $\delta(t_1 + \alpha(t_2 - t_1))$ = $\delta(t_1)$ + $\alpha(\delta(t_2) - \delta(t_1))$ for all $0 \leq \alpha \leq 1$. ■

Theorem 5 shows that if t_3 is a point where the slope of $\delta(t)$ changes, then as t is decreases through t_3 the set $J(t)$ changes, some constraints in it leave, and others outside it enter into it.

Theorem 6 *For t in the interval t^* to $\underline{t}$, $x(t)$, an optimum solution of (10), is also an optimum solution of*

$$\begin{aligned} &\textit{minimize} \quad cx \\ &\textit{subject to} \quad Ax \geq b + e\delta(t) \end{aligned}$$

where e is the column vector of all 1s of appropriate dimension. And for t in the interval $\bar{t}$ to t^*, $x(t)$, is also an optimum solution of

$$\begin{aligned} &\textit{maximize} \quad cx \\ &\textit{subject to} \quad Ax \geq b + e\delta(t) \end{aligned}$$

Proof: For any t, $(x(t), \delta(t))$ is an optimum solution for (10). So, there must exist an optimum dual solution $\pi(t) = (\pi_1(t), \ldots, \pi_m(t))$, $\mu(t)$, optimal to (11), the dual of (10).

$$\begin{aligned} \text{Minimize} \quad & -\pi b + \mu t \\ \text{subject to} \quad & \sum_{i=1}^{m} \pi_i = 1 \\ & -\pi A + \mu c = 0 \\ \mu \ \text{unrestricted}, \quad & \pi \geq 0 \end{aligned} \tag{11}$$

such that $((x(t), \delta(t)), \pi(t), \mu(t))$ satisfy the CS (complementary slackness) conditions (12)

$$\pi_i(A_{i.}x - b_i, -\delta) = 0, \quad i = 1, \ldots, m \tag{12}$$

We have already seen above that for $\underline{t} \leq t \leq t^*$, $\delta(t)$ is increasing with t, so the dual variable $\mu(t)$ will be strictly positive in this interval. And for $t^* \leq t \leq \bar{t}$, $\delta(t)$ is decreasing as t increases, so $\mu(t)$ will be strictly negative. So, we can define $y(t) = (y_i(t) = \pi_i(t)/\mu(t))$, then it satisfies the following properties:

(a): For t between t^* and $\underline{t}$, from (11) and (12) we see that $(x(t), y(t))$ are an optimum primal, dual pair for the LP

$$\begin{aligned} &\text{Minimize} \quad cx \\ &\text{subject to} \quad Ax \geq b + e\delta(t). \end{aligned}$$

(b): For t between $\bar{t}$ and t^* , from (11) and (12) we see that $(x(t), y(t))$ are an optimum primal, dual pair for the LP

$$\begin{aligned} &\text{Maximize} \quad cx \\ &\text{subject to} \quad Ax \geq b + e\delta(t). \end{aligned}$$

■

Theorem 7 *Consider the parameter t lying in the interval $[\underline{t}, t^*]$, decreasing from t^* to $\underline{t}$. Suppose a constraint, say the 1st, enters the set of essential touching constraint set as t keeps decreasing through t_1 for some t_1 in this interval; i.e., $1 \in J(t_1)$ and $1 \notin J(t)$ for values of t slightly greater than t_1. Then $1 \notin J(t)$ for any value of $t > t_1$ in this interval.*

Proof: Here t_1 is between t^* and $\underline{t}$. For any t if $(x(t), \delta(t))$ is the optimum solution of (10) and $1 \in J(t)$, then they satisfy

$$\delta(t) - A_{1.}x(t) = -b_1$$
$$\delta(t) - A_{i.}x(t) \leq -b_i \quad i = 2 \text{ to } m$$

So, a necessary condition for 1 belonging to $J(t)$ is that the system

$$cx = t$$
$$\delta - A_{1.}x = -b_1$$
$$\delta - A_{i.}x \leq -b_i \quad i = 2 \text{ to } m$$

must have a feasible solution. From Theorem 3 we know that both cx (= t) and $\delta(t)$ are monotonic decreasing as t decreases from t^* to $\underline{t}$. So, the largest value of t for which $J(t)$ contains 1 is $\leq$ the optimal t in the LP (13).

$$\begin{array}{ll} \text{Maximize} & \alpha\delta + cx \\ \text{subject to} & \delta - A_{1.}x = -b_1 \\ & \delta - A_{i.}x \leq -b_i, \quad i = 2 \text{ to } m \end{array} \tag{13}$$

for any positive coefficient α. The dual of (13) is

$$\begin{array}{rl} \text{Minimize} & -\pi b \\ \text{subject to} & \sum \pi_i = \alpha \\ & -\pi A = c \end{array}$$
$$\pi_1 \text{ unrestricted, } \pi_2 \text{ to } \pi_m \geq 0$$

As t is decreasing through t_1, 1 is entering the set $J(t)$, so (13) is infeasible for values of t slightly greater than t_1. At t_1 from the proof of Theorem 6 we have $y(t_1) = (y_i(t_1))$ satisfying

$$\begin{array}{rl} y(t_1)A = c & \\ \sum y_i(t_1) = (1/\mu(t_1)) & > 0 \\ y_i(t_1)(A_{i.}x(t_1) - b_i - \delta(t_1)) = 0 & i = 2 \ldots m \\ y(t_1) \geq 0 & \end{array} \tag{14}$$

It can be verified that if we take $\alpha = (1/\mu(t_1))$, then $(x(t_1), \delta(t_1)), \pi = y(t_1)$ satisfy the conditions for being optimal to (13) and its dual. This shows that t_1 is the largest value of $cx = t$ for which 1 can appear in the set $J(t)$ as t decreases in this interval, i.e., as t is decreasing from t^* to $\underline{t}$, t_1 is its first value at which the 1st constraint can appear in the set of essential touching constraints. ■

Theorem 8 *Consider the parameter t lying in the interval $[t^*, \bar{t}\,]$, decreasing from $\bar{t}$ to t^*. Suppose a constraint, say the 2nd, drops out of the set of essential touching constraints as t keeps decreasing through t_2 for some t_2 in this interval, i.e., $2 \in J(t_2)$ and $2 \notin J(t_2)$ for values of t slightly less than t_2. Then $2 \notin J(t)$ for any $t < t_2$ in this interval.*

Proof: As in the proof of Theorem 7 it can be argued that the smallest value of t for which $J(t)$ can contain 2 is $\geq$ optimal t in the LP (15).

$$\begin{array}{ll} \text{Minimize} & \beta\delta + cx \\ \text{subject to} & \delta - A_{2.}x = -b_2 \\ & \delta - A_{i.}x \leq -b_i, \quad i = 1, 3, \ldots, m \end{array} \tag{15}$$

where β is an arbitrary negative coefficient. The dual of (15) is

$$\begin{array}{rl} \text{Maximize} & -\pi b \\ \text{subject to} & \sum \pi_i = \beta \\ & -\pi A = c \end{array}$$
$$\pi_2 \text{ unrestricted, } \pi_i \leq 0, \quad i \neq 2.$$

From the proof of Theorem 6 we know that there is a $y(t_2)$ satisfying

$$y(t_2)A = c$$
$$\sum y_i(t_2) = (1/\mu(t_2)) < 0$$
$$y(t_2) \leq 0$$
$$y_i(t_2)(A_{i.}x(t_2) - b_i - \delta(t_2)) = 0, \quad i = 1, \ldots, m$$

It can be verified that if we take $\beta = 1/\mu(t_2)$, then $(x(t_2), \delta(t_2)), \pi = y(t_2))$ satisfies the conditions for being optimal to (15) and its dual, because for values of t slightly less than t_2, (15) is infeasible. This shows that t_2 is the smallest value of t for which 2 appears in the set $J(t)$ as t decreases in this interval; i.e., after t decreases below t_2 the 2nd constraint will not appear again in the set of essential touching constraints while t is in this interval. ■

Theorem 9 *As t, the value of cx, decreases to $\underline{t}$, the set of essential touching constraints can change at most $2m$ times.*

Proof: The argument in Theorem 8 shows that each constraint can leave the set of essential touching constraints at most once in the interval $\bar{t}$ to t^*. Each time the essential set of touching constraints changes, a constraint outside it must enter it. Thus between $\bar{t}$ to t^* the essential set of touching constraints can change at most m times.

The argument in Theorem 7 shows that between t^* to $\underline{t}$, a constraint can enter the set of essential touching constraints at most once. But each time the set of essential touching constraints changes at least one constraint drops out of it. So, between t^* and $\underline{t}$ also, there can be at most m changes in the essential touching constraint

set. So, as t is decreasing to $\underline{t}$, there can be at most $2m$ changes in the set of essential touching constraints. ■

Theorem 10 *Suppose for $t_1 \geq t \geq t_2$ the index set of essential touching constraints $J(t)$ does not change. Then the Stage 1 algorithm will descend from objective value t_1 to t_2 in no more than three repetitions of Step 1.2.*

Proof: Starting from a point with objective value t_1, after two applications of Step 1.2 the direction of the path of centers will be a direction along which the objective value can decrease all the way to t_2 in the next application (this will definitely be the case if the set of essential touching constraints identifies the center uniquely). ■

Theorems 9, 10 together show that the Stage 1 algorithm will be a strongly polynomial algorithm in terms of the number of centering steps, if centering is carried out exactly. The centering problem is a maxmin problem (an LP with a very special structure). Using its special structure we developed (Section 6) an iterative method for it based on line searches, each line search step involves solving a two variable LP. Its complexity has not been established, it is being investigated. So, with the present centering procedure, the most interesting question in LP complexity theory (whether general LP can be solved in strongly polynomial time) remains open.

If the centering strategy is carried to good accuracy, these results indicate that Stage 1 will have superior computational performance. We are planning to carry out a computational experiment comparing it with other methods on test problems available in the literature.

11. Intelligent Modeling

If an LP model is appropriate in an application, practitioners may be able to use the many flexible options usually available in applications to model the problem in the form (3) directly with a feasible set of full dimension. In this case it is possible to solve it using just Stage 1 described in Section 7.

Even if there are some equality constraints, by using each equation to eliminate a variable from the problem, it may be possible to transform it easily into form (3), and then use the approach in Section 7 to solve it.

If the centering strategy performs well, the method discussed in Section 7 combined with intelligent modeling offers many potential benefits for practical problem solving.

12. Application to Solve a Large Scale Nonsingular System of Linear Equations

Research in many branches of science often leads to large square nonsingular systems of linear equations

$$Ax = b \tag{16}$$

which need to be solved. When m, the order of A is very large, applying Gaussian elimination to solve (16) becomes very difficult. For such large systems, simpler iterative methods like Gauss-Seidel method, SOR methods, etc. are normally used to get approximate solutions. Here we show that Stage 1 of the method discussed in Section 7 can be used as an alternate method to get an approximate solution of (16).

Let x^* denote the unknown solution of (16). Now consider the LP

$$\begin{aligned} \text{Minimize} \quad & z = \sum_{i=1}^{m} A_{i.}x \\ \text{subject to} \quad & Ax \geq b \end{aligned} \tag{17}$$

x^* is the unique optimum solution of (17), its set of feasible solutions is the translate of a simplical cone to x^*. Stage 1 of Section 7 can be applied on (17) to get x^*. If the centering strategy of Section 6 works well, this can be expected to be a practically viable approach to solve (16) fast.

13. Some Advantages, and a Geometric Interpretation

Redundant constraints in (3) can effect the efficiency for solving it by the simplex method, or the earlier central path-following interior point methods. In fact in [5] Deza, Nematollahi, Peyghami, Terlaky show that when redundant constraints are added to the Klee-Minty problem over the n-dimensional cube, the central path in these methods takes $2^n - 2$ turns as it passes through the neighborhood of all the vertices of the cube before converging to the optimum solution.

Since gravitational methods operate only with the touching constraints, their performance is not affected by redundant constraints. Also, redundant constraints in (3) do not correspond to facets of K. So, in Stage 1 of Section 7, having redundant constraints in (3) just adds some additional directions $P_{.i}$ in the set of directions used in the centering procedure (Section 6). Programming tricks can be developed for efficiently selecting

promising directions in this set to search for improving the value of $f_{ir}(\alpha)$ in this procedure, and keep this centering procedure and Stage 1 efficient.

Also, since Stage 1 needs no matrix inversion, it can be used even when A is dense.

We will now show that Stage 1 (Section 7) can be viewed as computationally duplicating the geometric algorithm for solving 2-variable LPs discussed in undergraduate OR texts. In that method, the graph of the feasible region K is drawn on paper, a point $x^0 \in K$ is selected visually, and the straight line $z(x) = cx = cx^0$ (objective plane through x^0) is drawn. Looking at the picture of the feasible region, the objective line is moved parallel to itself in the desirable direction as far as possible until any further move will make the line loose contact with the feasible region K. The intersection of K with the final position of the line is the set of optimum solutions of the LP.

Due to lack of visibility in higher dimensional spaces to check if the objective plane can be moved further in the desirable direction while still keeping its contact with the feasible region, this simple geometric method could not be generalized to dimensions ≥ 3. In Iteration 1 (Section 7), the centering step in Step 1.1 guarantees that in Step 1.2, the objective plane through the center x^* of the current ball $B(x^*, \delta)$ can move a distance of δ in the descent direction and still keep its contact with the feasible region. Thus Stage 1 of Section 7 can be viewed as a generalization of the objective plane moving step in the geometric method for two dimensional LPs.

Acknowledgements: Discussions with many friends have been very helpful in preparing this paper. Also I acknowledge the latex help I received from Gong Yun Zhao.

References

[1] D. A. Bayer and J. C. Lagarias, "The Nonlinear Geometry of Linear Programming, I. Affine and Projective Scaling Trajectories, II. Legendre Transform Coordinates and Central Trajectories, III. Projective Legendre Transform Coordinates and Hilbert Geometry", *Transactions of the American Mathematical Society*, 314(1989)499-581.

[2] S. Y. Chang, "The Steepest Descent Gravitational Method for Linear Programming", Ph. D. dissertation, University of Michigan, Ann Arbor, MI, 1988.

[3] S. Y. Chang and K. G. Murty, "The Steepest Descent Gravitational method for Linear Programming", *Discrete Applied Mathematics* 25(1989)211-239.

[4] G. B. Dantzig and M. N. Thapa *Linear Programming, 1. Introduction*, Springer-Verlag New York, 1997.

[5] A. Deza, E. Nematollahi, R. Peyghami, T. Terlaky, "The Central Path Visits All the Vertices of the Klee-Minty Cube", AdvOL-Report No. 2004/11, McMaster University, Hamilton, Ontario, Canada.

[6] D. Gale, *The Theory of Linear Economic Models*, McGraw-Hill, NY, 1960.

[7] L. V. Kantorovich, *The Mathematical Method of Production Planning and Organization*, 1939, in Russian. Translated and published in *Management Science*, 6, no. 4(July 1960)363-422.

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

[9] T. L. Morin, N. Prabhu, and Z. Zhang, "Complexity of the Gravitational Method for Linear Programming", *Journal of Optimization Theory and Applications* 108(2001)633-658.

[10] K. G. Murty, *Linear Programming*, Wiley, 1983.

[11] K. G. Murty, "The Gravitational method for Linear Programming", *Opsearch* 23(1986)206-214.

[12] K. G. Murty and Y. Fathi, "A Critical Index Algorithm for Nearest Point Problems on Simplicial Cones", *Mathematical Programming*, 23(1982)206-215.

[13] K. G. Murty, *Linear Complementarity, Linear and Nonlinear Programming*, Helderman Verlag, Berlin, 1988, can be accessed on the web from Murty's webpage at: http://www-personal.engin.umich.edu/~ murty/

[14] K. G. Murty, *Computational and Algorithmic Linear Algebra and n-dimensional Geometry*, Freshman-Sophomore level linear algebra book available as free download at: http://ioe.engin.umich.edu/people/fac/books/murty/algorithmic_linear_algebra/

[15] K. G. Murty, "A Gravitational Interior Point Method for LP", *Opsearch*, 42, No. 1 (2005)28-36.

[16] R. Saigal, *Linear Programming A Modern Integrated Analysis*, Kluwer Academic Publishers, Boston, 1995.

[17] D. R. Wilhelmsen, "A Nearest Point Algorithm for Convex Polyhedral Cones and Applications to Positive Linear Approximation", *Mathematics of Computation*, 30(1976)48-57.

[18] P. Wolfe, "Algorithm for a Least Distance Programming Problem", *Mathematical Programming Study 1*, 1974, 190-205.

[19] P. Wolfe, "Finding the Nearest Point in a Polytope", *Mathematical Programming*, 11(1976)128-149.

[20] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1960.

[21] Y. Ye, *Interior Point Algorithms, Theory and Analysis*, Wiley-Interscience, 1997.

Received 7 January 2005; revised 25 May 2005; accepted 19 Jun 2005