TUTORIALS IN
OPERATIONS RESEARCH
INFORMS 2009

 | ISBN 978-1-877640-24-7
DOI 10.1287/educ.1090.0060

New Sphere Methods for Linear Programs

Katta G. Murty
Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan 48109, murty@umich.edu, http://www-personal.umich.edu/∼murty

Abstract A new method being developed for solving linear programs is discussed in this tutorial. This method uses matrix inversion operations sparingly, and it thus seems well suited to solve large-scale problems and those that may not have the property of being very sparse.

Keywords linear programming (LP); interior point methods; ball centers of a polytope; solving LPs using matrix inversions sparingly; touching set of constraints; sphere methods 1 and 2 for LPs

1. LP and Evolution of Algorithms for It

Examples of applications involving linear inequality models have started appearing in published literature from about mid-18th century. The mathematician Joseph Fourier was one of the first to recognize the importance of inequalities as opposed to equations in applications of mathematics. Moreover, he is the pioneer who seems to have recognized the equivalence of systems of linear inequalities and linear programs (LPs), which involve optimization of a linear objective function subject to linear inequality constraints, as early as the beginning of the 19th century.

The famous elimination method (nowadays called the Gaussian elimination method) for solving systems of linear equations was discovered in China and India over 2,000 years ago. In contrast, there was no method known for solving systems of linear inequalities until modern times. The first method for them, a generalization of the elimination method for solving systems of linear equations, is now known as the Fourier, or Fourier-Motzkin, elimination method. Dantzig [4] and Dantzig and Thapa [5] show how this method can be adopted directly to solve linear programs. However, starting with a system involving m inequalities, the number of inequalities can jump to $O(m^2)$ after eliminating one variable, so this method is not practical except for very small problems.

The second method developed for solving linear inequalities and LPs is the simplex method. The geometric version of the principle behind the simplex method was published by Fourier in 1827. After development by several people, this work culminated in mid-20th century with the publication of the primal simplex method by Dantzig (see Dantzig [4] for a detailed historical account). This method is practical, and it stimulated the widespread use of the LP model as a decision-making tool. In the second half of the 20th century, the simplex method underwent extensive development, and nurtured by demands from practical applications, many variants of it have been developed and implemented. These continue to be popular today.

In the last quarter of the 20th century, Karmarker [6] pioneered a new method for LP, an interior point method (IPM). His work attracted worldwide attention, not only from operations researchers but also from scientists in other areas.

Let me relate a personal experience. When news of his work broke out in world press, I was returning from Asia. The person sitting next to me on the flight was a petroleum geologist. When he learned that I was on the OR faculty at Michigan, he asked me excitedly,

"I understand that an OR scientist from India at Bell Labs made a discovery that is going to revolutionize petroleum exploration. Do you know him?!"

In talks on his algorithm that he gave at that time, Karmarker repeatedly emphasized the following points:

(I) The boundary of a convex polyhedron with its faces of varying dimensions has a highly complex combinatorial structure. Any method that operates on the boundary or close to the boundary will get caught up in this combinatorial complexity, and there is a limit on improvements we can make to its efficiency.

(II) Methods that operate in the central portion of the feasible region in the direction of descent of the objective function have the ability to take longer steps toward the optimum before being stopped by the boundary, and hence have the potential of being more efficient than boundary methods for larger problems.

(III) From an interior point, one can move in any direction locally without violating feasibility; hence, powerful methods of unconstrained optimization can be brought to bear on the problem.

Researchers saw the validity of these arguments; hence, his talks stimulated a lot of work on these methods that stay "away" from the boundary. In the tidal wave of research that ensued, many different classes of IPMs have been developed for LP and extended to wider classes of problems including convex quadratic programming, monotone linear complementarity problems, and semidefinite programming problems. Alizadeh [1] was the first to propose a polynomial algorithm based on IPM for semidefinite programming, and Alizadeh and Goldfarb [2] introduced an IPM for second-order cone programming (for other references, see Kojima et al. [7], Megiddo [8], Mehrotra [9], Monteiro and Adler [10], Saigal [17], Wright [18], and Ye [19]).

1.1. Newer Methods

Many practical applications lead to large-scale LP models. The methods discussed so far are based on matrix inversion operations. Also, in every step of these methods, the matrix inversion operations involve every constraint in the problem, including any redundant constraints in the model. The process of identifying and eliminating the redundant constraints, if any, from the model using the methods available in theory is prohibitively expensive (in fact the effort needed by these methods is many times more than that needed to solve the original model with all the redundant constraints in it) and hence is not practical. Presolvers based on various heuristics are commonly employed in LP solvers to identify and eliminate redundant constraints in the model; they are reported to be effective, albeit not optimal (see Cartis and Gould [3]). Methods that have the ability to avoid redundant constraints in their computational work by themselves instead of depending on these presolvers have a definite advantage over existing methods.

The *density* of an LP (or its coefficient matrix) refers to the percentage of nonzero entries in the data. Problems in which this percentage is small (typically $\leq 1\%$) are said to be *sparse*; as this percentage increases, the problem becomes more and more dense. Many LP models in real-world applications tend to be very sparse. In matrix inversion operations of the simplex method and IPMs, programmers have developed techniques to exploit this sparsity. Using these they produced implementations of these methods with reasonable memory requirements that can solve large-scale models fast. However, the effectiveness of these techniques fades as the density of the coefficient matrix increases, which is why solving large-scale dense LPs is hard by these methods.

Dense LP models do arise in several important application areas. For example, a typical LP model for a data envelopment analysis problem has a coefficient matrix that is essentially 100% dense. Also, many LP models arising in location problems, distribution problems, and

supply chain problems are typically dense. That is why there is a lot of research being carried out targeting dense LP models. In solving LPs that are not very sparse, it is definitely advantageous to have to deal with smaller number of constraints.

It seems that practitioners are quite content with obtaining solutions not necessarily optimal but close to being so, but they want a method that can obtain this approximate solution faster than existing methods. In many applications, this requires algorithms that can give good performance on models that may not be very sparse. For this, we need to investigate fast methods that satisfy the following properties:

1. The method should be a descent method (i.e., starting with a feasible solution, the method should maintain feasibility throughout, and the objective value should improve monotonically in every step),
2. It should be implementable with no matrix inversions or use matrix inversion operations only sparingly, and
3. If matrix inversion operations are used, they should never involve any redundant constraints in the model and should only involve a small subset of the other constraints selected intelligently.

The future of algorithmic research in LP is in this area.

1.2. The Sphere Methods for LP

The *sphere methods* that are discussed in this tutorial are IPMs that are descent methods. In contrast to all other IPMs, these methods can be implemented with no matrix inversions, or using them only sparingly (see Murty [11, 12], Murty and Oskoorouchi [15, 16]). Another advantage of these methods is that if matrix inversion operations are used in any iteration of the algorithm, only a small subset of constraints (called the *touching set of constraints* in that iteration) will be involved in those matrix inversion operations. Also, amazingly, redundant constraints, if any in the model, never enter into the touching set. So, they offer the prospect of extending the superior performance of existing software systems for LP to models that do not have the property of being very sparse. So, they seem to be well suited to the goals mentioned above. The sphere methods belong to the class of predictor/corrector-type IPMs. They consider LPs in the form

$$\begin{aligned} \text{minimize} \quad & z(x) = cx \\ \text{subject to} \quad & Ax \geq b, \end{aligned} \tag{1}$$

where A is an $m \times n$ data matrix, with a known initial interior feasible solution x^0 (i.e., satisfying $Ax^0 > b$). The constraints in (1) include all bound constraints on individual variables. We assume that the rows of A, denoted by $A_{i.}$ for $i = 1$ to m, have been normalized, so that $\|A_{i.}\| = 1$ ($\|.\|$ denotes the Euclidean norm) for all $i = 1$ to m.

If $c = 0$, every feasible solution is optimal, so we can change the objective function in the problem to that of minimizing $A_{1.}x$ instead. So without any loss of generality we can assume that $c \neq 0$. We will assume that c is also normalized so that $\|c\| = 1$.

If the set of feasible solutions does not have an interior, or if it does but an interior feasible solution is not known, we modify the problem with the usual big-M augmentation involving one artificial variable as follows:

$$\begin{aligned} \text{minimize} \quad & z(x) = cx + Mx_0 \\ \text{subject to} \quad & Ax + ex_0 \geq b, \\ & x_0 \geq 0, \end{aligned}$$

where e is the column vector of all 1s in R^m, and M is a large positive penalty parameter. $(x^0 = 0, x_0^0)$, where x_0^0 is a sufficiently large positive number, gives an initial interior feasible

solution to this Phase I problem. This Phase I problem is in the same form as (1), so we solve it instead. For other such Phase I strategies, see Cartis and Gould [3].

1.3. How to Implement these Algorithms for Solving a General LP

At present, the most popular interior point method for software implementations is the primal-dual interior point method, because (i) it gives optimum solutions to both the primal and the dual when both have feasible solutions, and (ii) it provides a lower bound that serves as an indicator to check how far left to go to reach the optimum. Also, in many algorithms, specifying a good termination condition to be used in practice is not easy. The lower bound in the primal-dual format provides an automatic practical termination condition.

We will show how to convert the sphere methods into primal-dual algorithms for LPs in general form. Consider an LP in general form in which there may be equality constraints on the variables, inequality constraints, and bounds on individual variables. By combining the bounds on individual variables with the inequality constraints, the problem is in the form

$$\begin{aligned} \text{Minimize} \quad & f\xi \\ \text{subject to} \quad & F\xi = h, \\ & G\xi \geq g, \end{aligned} \tag{2}$$

where F is a matrix of order, say, $p \times q$. Let π, μ be dual vectors corresponding to the constraints in the two lines in (2). Solving (2) and its dual involves finding a feasible solution to the following system

$$\begin{aligned} F\xi &= h, \\ \pi F + \mu G &= f, \\ (G\xi, \mu, -f\xi + \pi h + \mu g) &\geq (g, 0, 0). \end{aligned} \tag{3}$$

Solving (3) is the same as solving the LP

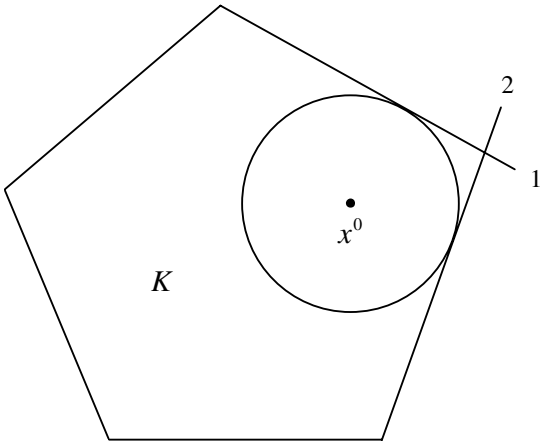
$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^{p} (F_{i\bullet}\xi - h_i) + \sum_{j=1}^{q} (\pi F_{\bullet j} + \mu G_{\bullet j} - f_j) \\ \text{subject to} \quad & (F\xi, G\xi) \geq (h, g), \\ & (\pi F + \mu G, \mu, -f\xi + \pi h + \mu g) \geq (f, 0, 0). \end{aligned} \tag{4}$$

LP (4) is in same form as (1). Also, because we are applying the algorithm using matrix inversion operations sparingly, having all these additional constraints over those in the original LP (2) in the model may not make it numerically difficult to handle. The objective value in (4) will be 0 at the optimum if both (2) and its dual have feasible solutions, so this provides a convenient lower bound to judge how far is left to go. So, in the sequel we will discuss the methods for solving the LP in the form of (1).

The sphere method needs an initial interior feasible solution. Each iteration of the method begins with the best interior feasible solution obtained at the end of the previous iteration and consists of two steps: a *centering step* and a *descent step*.

In concept, the aim of the centering step is to find a *ball center*, which is an interior feasible solution with objective value equal to or better than that of the current interior feasible solution, and is the center of a largest radius ball inside the feasible region of the original LP subject to the constraint on its center. This centering step takes up most of the computational effort in the iteration. Once the ball center is obtained, the descent step, which is computationally cheap, carries out several descent steps from it, and the iteration stops with the best point obtained in all these descent steps.

FIGURE 1. $x^0 \in K^0$, and the ball shown is $B(x^0, \delta(x^0))$, the largest ball inside K with x^0 as center.



Note. Facetal hyperplanes of K corresponding to indices 1, 2 are tangent planes to this ball, so $T(x^0) = \{1, 2\}$.

2. Notation

The orthogonal distance from a point $y \in R^n$ to the facetal hyperplane $\{x\colon A_{i.}x = b_i\}$ of the set of feasible solutions of (1) is minimum$\{\|x-y\|\colon A_{i.}x = b_i\} = A_{i.}y - b_i$, because $\|A_{i.}\| = 1$. We will use the following notation:

K = Set of feasible solutions of (1).
$K^0 = \{x\colon Ax > b\}$ = interior of K.
$\delta(x)$ = Min$\{A_{i.}x - b_i\colon i = 1$ to $m\}$, defined for $x \in K^0$, it is the radius of the largest ball inside K with x as its center, since $\|A_{i.}\| = 1$ for all i.
$B(x, \delta(x))$ = Defined for $x \in K^0$, it is the largest ball inside K with x as its center.
$T(x)$ = Defined for $x \in K^0$, it is the set of all indices i satisfying $A_{i.}x - b_i =$ Minimum$\{A_{p.}x - b_p\colon p = 1$ to $m\} = \delta(x)$. The hyperplane $\{x\colon A_{i.}x = b_i\}$ is a tangent plane to $B(x, \delta(x))$ for each $i \in T(x)$, therefore $T(x)$ is called the *index set of touching constraints in* (1) at $x \in K^0$. See Figure 1.
$\Gamma_1 = \{A_{i.}^T, -A_{i.}^T\colon i = 1$ to $m\}$; set of directions normal to facetal hyperplanes of K.
$\Gamma_2 = \{P_{.1}, \ldots, P_{.m}, -P_{.1}, \ldots, -P_{.m}\}$, where $P_{.i} = (I - c^T c)A_{i.}^T$, orthogonal projections of the directions normal to the facets of K on the hyperplane $\{x\colon cx = 0\}$.
$c^i = (I - A_{i.}(A_{i.})^T)c^T$, the orthogonal projection of c^T on $\{x\colon A_{i.}x = 0\}$ for $i = 1$ to m.

3. Ball Centers of Polytopes

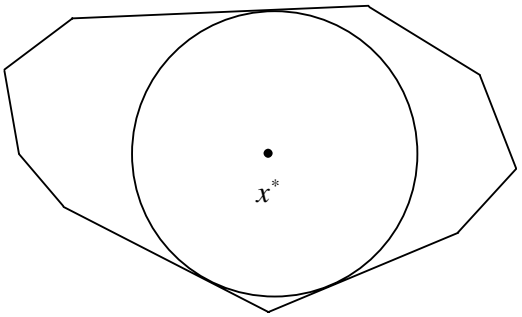
A *ball center* of a polytope (set of feasible solutions of a system of linear constraints, when it is bounded) is the center of a largest ball inside the polytope. The concept of ball centers of unbounded convex polyhedra is discussed at the end of this section.

For a detailed discussion on ball centers of polytopes, see Murty and Oskoorouchi [15, 16] and Murty [13]. Assuming that K is a polytope (i.e., bounded), a ball center of K, or for the system (1), is an optimum solution x for (5), and the δ in that optimum solution is the radius of the largest ball inside K with x as center.

$$\begin{aligned} &\text{Maximize } \delta \\ &\text{subject to } \delta \le A_{i.}x - b_i, \quad i = 1 \text{ to } m. \end{aligned} \tag{5}$$

An optimum solution of (5) may not be unique, so a ball center for a polytope may not be unique. Figure 2 illustrates a polytope which has a unique ball center, and Figure 3 illustrates one in which the ball center is not unique.

FIGURE 2. The polytope K and the largest ball inside it are shown.



Note. When the largest inscribed ball in K is unique as here, its center x^* is the ball center of K.

When the polytope K has alternate ball centers, Murty and Oskoorouchi [15] give the geometric concept of defining one of them as *the ball center of* K. However, for computational efficiency, sphere methods use any point that is optimal to (5) as a ball center of K.

Procedures for computing ball centers approximately use a series of line search steps. The search direction for each line search step is selected such that it is a *profitable direction*, i.e., the move in that direction at the current point is guaranteed to lead to a point that increases the radius of the inscribed ball with that point as center. Theorem 1 given below provides a simple criterion to check whether a give direction is a profitable direction at the current interior feasible solution or not. After the theorem, we will show that the optimum step length in this line search step can be found by solving a two-variable LP.

Let $\bar{x}$ be the current interior feasible solution. So, a direction $y \neq 0$ is defined to be a *profitable direction* at $\bar{x}$ if $\delta(\bar{x} + \alpha y)$ strictly increases as α increases from 0, and an *unprofitable direction* to move at $\bar{x}$ if the maximum value of $\delta(\bar{x} + \alpha y)$ over $\alpha \geq 0$ is attained at $\alpha = 0$. We have the following result.

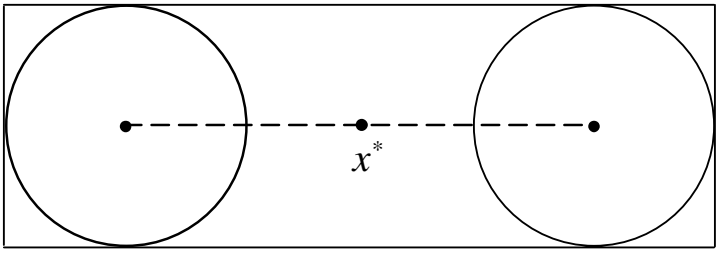
Theorem 1. *A given direction y is a profitable direction to move at the current interior feasible solution $\bar{x}$ iff $A_{i.}y > 0$ for all $i \in T(\bar{x})$. Also, $\bar{x}$ is a ball center of K iff the system $A_{i.}y > 0$ for all $i \in T(\bar{x})$ has no solution in y.*

Proof. For any α we have $\delta(\bar{x} + \alpha y) = \text{minimum}\{A_{i.}(\bar{x} + \alpha y) - b_i\colon\ i = 1 \text{ to } m\}$, and $\delta(\bar{x} + \alpha y) = \delta(\bar{x})$ when $\alpha = 0$. From the definition of $T(\bar{x})$, we have $\delta(\bar{x}) = A_{i.}x - b_i$ for all $i \in T(\bar{x})$. So, we have the following facts.

Fact 1. For all $i \notin T(\bar{x})$, we have $A_{i.}\bar{x} - b_i > \delta(\bar{x})$; hence, when α is positive but sufficiently small, $A_{i.}(\bar{x} + \alpha y) - b_i > \delta(\bar{x})$ whatever the sign of $A_{i.}y$ may be.

Fact 2. If $A_{i.}y > 0$ for all $i \in T(\bar{x})$, then each of these $A_{i.}(\bar{x} + \alpha y)$ is strictly monotonically increasing as α increases from 0.

FIGURE 3. A two-dimensional polytope K for which the largest inscribed ball is not unique.



Note. S, the set of centers of all such balls, the optimum face of (5) in the x-space, is the dashed-line segment in this polytope, and x^* is a point in it.

Facts 1 and 2 together imply that when α is positive but sufficiently small, and y satisfies $A_{i\cdot}y > 0$ for all $i \in T(\bar{x})$, then $\delta(\bar{x}+\alpha y) = \text{minimum}\{A_{i\cdot}(\bar{x}+\alpha y) - b_i\colon i = 1 \text{ to } m\}$ is $> \delta(\bar{x})$; i.e., $\delta(\bar{x}+\alpha y)$ strictly increases as α increases from 0. This shows that y is a profitable direction to move at $\bar{x}$ under the conditions stated in the theorem.

If there is at least one index, say, $i_1 \in T(\bar{x})$ satisfying $A_{i_1\cdot}y \le 0$, then for positive but sufficiently small values of α, we have $A_{i_1\cdot}(\bar{x}+\alpha y) - b_{i_1} \le A_{i_1\cdot}\bar{x} - b_{i_1} = \delta(\bar{x})$ because $i_1 \in T(\bar{x})$. From the definition of $\delta(\bar{x}+\alpha y)$, this implies that in this case $\delta(\bar{x}+\alpha y)$ either stays the same or strictly decreases as α increases from 0, implying that y is an unprofitable direction to move at $\bar{x}$.

For $x \in K$, $\delta(x) = \text{minimum}\{A_{i\cdot}x - b_i\colon i = 1 \text{ to } m\}$ is a piecewise linear function that is the pointwise infimum of a finite set of affine functions; hence, $\delta(x)$ is a concave function. If there is no profitable direction at $\bar{x}$, then $\bar{x}$ is a local maximum for the concave function $\delta(x)$ over K, and so it is the global maximum for $\delta(x)$ over K; i.e., $\bar{x}$ is a ball center of K. □

Because $\delta(\bar{x}+\alpha y) = \text{minimum}\{A_{i\cdot}(\bar{x}+\alpha y)\colon i = 1 \text{ to } m\}$, we have $\delta(\bar{x}+\alpha y) \le A_{i\cdot}\bar{x} + \alpha A_{i\cdot}y$ for all $i = 1$ to m.

So, for a given direction y, suppose $\bar{\delta} = \text{maximum value of } \delta(\bar{x}+\alpha y)$ over $\alpha \ge 0$, and $\bar{\alpha} \ge 0$ is the value of α where this maximum is attained in $\delta(\bar{x}+\alpha y)$. Then, $(\bar{\delta}, \bar{\alpha})$ is the optimum solution of the following two-variable LP in which the variables are δ and α.

$$\begin{aligned} &\text{Maximize} && \delta \\ &\text{subject to} && \delta - \alpha A_{i\cdot}y \le A_{i\cdot}\bar{x} - b_i, \quad i = 1 \text{ to } m, \\ &&& \delta, \alpha \ge 0. \end{aligned} \tag{6}$$

Therefore, the optimum step length for the line search problem of maximizing $\delta(\bar{x}+\alpha y)$ over $\alpha \ge 0$ in the given direction y can be found by solving the two-variable LP (6).

One technique for computing the ball center of a polytope approximately, discussed in Murty [11, 12] and Murty and Oskoorouchi [15], uses two procedures called line search steps using facetal normal directions (LSFN) and line search steps using computed profitable directions (LSCPD) both using series of line search steps. However, we do not discuss that technique here because sphere methods use ball centers subject to constraints on the objective that the center is required to satisfy, which we discuss next.

3.1. Ball Centers Used in Sphere Methods

Consider the stage in solving the LP (1) by sphere methods when the objective value at the current interior feasible solution is t.

At this stage, a *ball center of K on the current objective plane* $\{x\colon cx = t\}$ is defined as a center of a largest ball inside K subject to the constraint that the center lie on the current objective plane. So, if $(\bar{x}, \bar{\delta})$ is an optimum solution of the LP (5) with the additional constraint $cx = t$, then $\bar{x}$ is the ball center of K on the current objective plane at this stage, and $\bar{\delta}$ is the radius of that ball.

The ball center used in the sphere methods at the current stage is defined as the center of a largest ball inside K subject to the constraint that the objective value at this center is $\le t$; it is an optimum solution of (5) with the additional constraint $cx \le t$. Murty [11, 12] and Murty and Oskoorouchi [15, 16] discuss a technique to compute these ball centers approximately using the two procedures called LSFN and LSCPD (adapted to this problem), each based on a sequence of line search steps; these are discussed below.

3.2. Approximate Computation of Ball Centers in Sphere Methods

Both LSFN and LSCPD involve a series of line search steps, with search directions that are profitable directions y, satisfying $cy \le 0$; i.e., they are also descent directions for the objective function to be minimized in (1).

3.3. LSFN: Using Line Search Steps in Facetal Normal Directions

One advantage of this technique is that it needs no matrix inversions. Beginning with the initial $x^0 = \bar{x}$, it generates a sequence of points x^k, $k = 1, 2, \ldots$, along which the radius of the inscribed ball δ is strictly increasing.

Whether a given direction y is profitable or not at the current point x^k can be checked very efficiently using the conditions proved in Theorem 1 (these are $A_{i.}y > 0$ for all $i \in T(x^k)$). Because the goal in this centering step is to increase the minimum distance of x from each facetal hyperplane of K, the LSFN procedure uses the directions normal to the facetal hyperplanes of K for the line searches, i.e., directions y in $\Gamma_1 = \{A_{i.}^T, -A_{i.}^T\colon\ i = 1 \text{ to } m\}$, satisfying $cy = 0$ (if the goal is to find a ball center for the polytope K on the current objective plane) or $cy \leq 0$ (if the center is required to satisfy $cx \leq$ current objective value). Once a profitable direction y at the current point x^k has been found, the optimum step length α in this direction that maximizes $\delta(x^k + \alpha y)$ over $\alpha \geq 0$ is found by solving the two-variable LP (6) with $\bar{x}$ replaced by x^k.

This procedure continues as long as profitable directions for line search are found in Γ_1 satisfying the above constraints, and terminates with the final point as an approximation to a required ball center of K, obtained by this procedure.

3.4. LSCPD: Sequence of Line Search Steps Using Computed Profitable Directions

This technique, discussed in (Murty [12]), consists of a sequence of at most n line search steps in profitable directions computed by solving a system of linear equations. After each step in the sequence, the index set of touching constraints at the current solution x keeps growing by at least one more constraint, and we will stop the sequence when the set of coefficient vectors of the touching set of constraints becomes linearly dependent. That is why the number of steps in the sequence is at most n. The entire sequence needs a single matrix inversion, carried out in stages adding one row and column to the matrix at a time, so it uses matrix inversion operations sparingly.

The sequence begins with the initial point x^k, say, obtained at the end of LSFN. When x^g is the current solution, from Theorem 1 we know that any solution y of the system

$$A_{i.}y = d_i \quad \text{for all } i \in T(x^g), \tag{7}$$

where $d = (d_i\colon\ i \in T(x^g)) > 0$ is any positive vector, is a profitable direction to move at x^g.

We will use (7) with $d = e$, the column vector of all 1s of appropriate dimension, to generate a profitable direction to move that is a basic solution of (7). Once a profitable direction y is determined, the step length to move in this direction is determined by solving a two-variable LP of the form of (6).

So, this sequence begins with the initial point x^g and solves (7) with $d = e$ for a basic solution. If this system has no solution, then this technique is terminated with x^g as an approximate ball center required.

If it does yield a solution y^0, suppose it is the basic solution with respect to a basic vector y_{B_0} and basis B_0 for the system (7). There are two cases to consider.

Case 1. If $cy^0 \leq 0$, then y^0 is not only profitable at x^g (i.e., $\delta(x^g + \alpha y^0)$ increases as α increases from 0), but it is also a descent direction for cx. So, we carry out a line search step at x^g in the direction y^0 exactly as described earlier and continue. This moves increases $\delta(x)$ and also may decrease cx.

Case 2. If $cy^0 > 0$, there are two subcases to consider here. Let $(B_0 \vdots D_0)$ be the partition of the coefficient matrix of (7) into basic, nonbasic parts with respect to the basic vector y_{B_0} for it. Let $(c_{B_0} \vdots c_{D_0})$ be the corresponding partition of the row vector c.

Subcase 2.1. $c_{D_0} - c_{B_0}(B_0)^{-1}D_0 = 0$. In this subcase, cy = a constant = $cy^0 > 0$ in every solution of (7). So, in this subcase using any solution of (7) as the direction for the move helps increase $\delta(x)$ but also increases cx. So we terminate the sequence at this stage, and take the current point x^g as the approximation to the ball center on the objective plane through x^g.

Subcase 2.2: There is a nonzero entry in $c_{D_0} - c_{B_0}(B_0)^{-1}D_0$. Suppose it occurs in the column of the nonbasic variable y_j, and that nonzero entry is denoted by $\bar{c}_j$. Let the column vector of this nonbasic variable y_j in D_0 be denoted by $(D_0)_{\cdot j}$. Then the solution y^1 of (7) given by the following: all the variables except y_j in the nonbasic vector y_{D_0} are = 0, and

$$y_{B_0} = y^0_{B_0} - \theta(B_0)^{-1}(D_0)_{\cdot j}$$
$$\text{Nonbasic } y_j = \theta,$$

where $\theta = (-1 - cy^0)/\bar{c}_j$ satisfies $cy^1 = -1$. Now carry out a line search step at the current point x^g in the direction y^1 exactly as described earlier and continue. As under Case 1, this move not only increases $\delta(x)$ but also decreases cx.

If α^g is the optimum step length maximizing $\delta(x^g + \alpha y)$, notice that as α increases from 0, $A_{i\cdot}(x^g + \alpha y) = A_{i\cdot}x^g + \alpha$ increases at the same rate as α for all $i \in T(x^g)$. This implies that at the solution $x^{g+1} = x^g + \alpha^g y$ obtained after this line search step, we will have $T(x^{g+1}) \supset T(x^g)$.

The sequence will now continue the same way with x^{g+1} as the initial solution for the next step. Suppose $T(x^g) = \{1, \ldots, s\}$ and $T(x^{g+1}) = \{1, \ldots, s, s+1\}$. Then, the profitable direction y to be used at x^{g+1} is computed by solving the system

$$A_{i\cdot}y = 1 \quad \text{for all } i \in \{1, \ldots, s+1\}. \tag{8}$$

Let $\mathcal{A}^0 = (B_0 \vdots D_0)$ denote the coefficient matrix of (7) with rows $\{A_{i\cdot}: i \in T(x^g)\}$, with columns partitioned into the basic and nonbasic parts with respect to the basic vector y_{B_0} for (7), and let $(A^{B_0}_{s+1\cdot}, A^{D_0}_{s+1\cdot})$ denote the corresponding partition of $A_{s+1\cdot}$.

The set of touching constraint coefficient vectors $\{A_{i\cdot}: i \in T(x^{g+1})\}$ is linearly independent iff $A^{D_0}_{s+1\cdot} - A^{B_0}_{s+1\cdot}(B_0)^{-1}D_0 \neq 0$.

Therefore, if $A^{D_0}_{s+1\cdot} - A^{B_0}_{s+1\cdot}(B_0)^{-1}D_0 = 0$, terminate the sequence with x^{g+1} as the final approximation of a ball center required. On the other hand, if this vector is $\neq 0$, select a nonzero entry in it, suppose it is in the column of the variable y_j for some j, and then let $\mathcal{A}^0_{\cdot j}$ be the column corresponding to y_j in $\mathcal{A}^0$, the coefficient matrix of (7). Then, $y_{B^1} = (y_{B_0}, y_j)$ is a basic vector for (8). The corresponding basis for (8) is

$$B_1 = \begin{pmatrix} B_0 & \vdots & \mathcal{A}^g_{\cdot j} \\ \ldots & & \ldots \\ A^{B_0}_{s+1\cdot} & \vdots & a_{s+1,\,j} \end{pmatrix},$$

where $a_{s+1,\,j}$ is the coefficient corresponding to y_j in $A_{s+1\cdot}$. Hence,

$$(B_1)^{-1} = \begin{pmatrix} P & \vdots & Q \\ \ldots & & \ldots \\ R & \vdots & S \end{pmatrix},$$

where

$$S = 1/\left(a_{s+1,j} - A^{B_0}_{s+1.}(B_0)^{-1}\mathcal{A}^g_{.j}\right),$$
$$R = \left(-A^{B_0}_{s+1.}(B_0)^{-1}\right)/\left(a_{s+1,j} - A^{B_0}_{s+1.}(B_0)^{-1}\mathcal{A}^g_{.j}\right),$$
$$Q = -(B_0)^{-1}\mathcal{A}^g_{.j}S,$$
$$P = (B_0)^{-1} + QR/S.$$

So, $(B_1)^{-1}$ can be obtained by updating $(B_0)^{-1}$ using the above formulas.

The sequence repeats the same way with x^{g+1} until it terminates at some stage. Thus, in this sequence, whenever system (8) is augmented by a new constraint in a step, the basic vector and basis inverse in this step can be updated quite efficiently for the next step as discussed above.

In the centering steps of sphere methods, we first apply the LSFN sequence of steps as described above, selecting profitable directions from the set Γ_2, or $\Gamma_1 \cap \{y\colon cy \leq 0\}$ as appropriate, until the improvement in the radius of the ball δ per step decreases below some selected tolerance. At that time, we switch and initiate an LSCPD sequence starting with the final point obtained at the end of the LSFN sequence of steps. When the LSCPD sequence terminates, the final point obtained in that sequence is an approximate ball center required.

3.5. The Case of Unbounded Convex Polyhedra

Suppose the set of feasible solutions, K, of (1) is an unbounded polyhedron. Typically, the radius of a maximum inscribed ball in K will be ∞ (i.e., δ is unbounded above in the LP (5)). Even if the radius of the maximum inscribed ball is finite, the optimum face of (5) in the x-space may be an unbounded polyhedron itself, so the ball center of K is not defined.

Even when K is an unbounded convex polyhedron, the hyperplane H may be such that $K \cap H$ is bounded, i.e., is a polytope. In this case the ball center of K on H is well defined. If $K \cap H$ is also unbounded, then the ball center of K on H is also not defined.

What happens to the LP (1) if we do not know whether its set of feasible solutions is bounded or not, or if we know that it is unbounded? In the unbounded case, even though the concepts of ball centers may not be well defined, the implementation of the algorithm based on the approximate computation of ball centers can be carried out as usual. In this case, in one of the descent steps, the step length for the move may turn out to be $+\infty$. That is an indication that the objective value in (1) is unbounded below on its set of feasible solutions, and the algorithm terminates if this happens.

4. Sphere Method 1

This method is discussed in Murty and Oskoorouchi [15, 16]. In concept, the centering step in this method has the aim of finding a ball center of K on the objective plane through the current point. So, the LSFN sequence of steps in it use profitable directions from the set $\Gamma_2 = \{\pm P_{.i}\colon i = 1 \text{ to } m\}$, where $P_{.i}$ is the orthogonal projection of $A^T_{i.}$ on $\{y\colon cy = 0\}$. However, the LSCPD steps in it generate and use profitable directions y, which satisfy $cy \leq 0$, so some of them may also decrease the objective value cy. Here is the description of the general iteration $r+1$ in this method.

4.1. The Centering Step in Iteration $r+1$ in Sphere Method 1

Let x^r be the initial interior feasible solution for this iteration. This step consists of a series of line searches in profitable directions with the aim of finding an x that maximizes $\delta(x)$ subject to the constraint $cx \leq cx^r$. First it carries out the LSFN sequence of line searches;

then with the point obtained at the end of the LSFN sequence, it carries out a LSCPD sequence of line searches.

Beginning with the initial point x^r, this generates a sequence of points $x^{r,k}$, $k = 1, 2, \ldots$, along which the radius of the ball δ is strictly increasing, as described in §3, by selecting profitable directions from the set Γ_2. The procedure continues as long as profitable directions for line search are found in Γ_2, and this sequence terminates with the final point, which we denote by $\tilde{x}^r$.

Beginning with the initial point $\tilde{x}^r$ obtained at the end of the LSFN sequence, this generates a sequence of points $\tilde{x}^{r,k}$, $k = 1, 2, \ldots$, along which the radius of the ball δ is strictly increasing.

When $\tilde{x}^{r,k}$ is the current solution, it selects the profitable direction to move as described in §3. Let $\bar{x}^r$ denote the final point obtained at the end of the LSCPD sequence; it is the approximate ball center obtained in this centering step, and the iteration moves to the descent step with it.

4.2. The Descent Step in Iteration $r+1$ in Sphere Method 1

The centering step in each iteration is the computationally expensive step. Compared to it, each descent step requires only a minimum ratio computation, which is cheap in comparison. So, once the approximate ball center is computed in the centering step, sphere methods carry out several descent steps from it and take the best point obtained from all of them as the output of this step. All the descent directions used need very little computation, if any, to obtain them. Here is a summary.

Let $\bar{x}^r$ denote the approximate ball center obtained in the centering step of this iteration.

Each descent step carried out in this iteration requires one minimum ratio computation. For example, consider a descent step from the current center $\bar{x}^r$ in the descent direction y (i.e., satisfying $cy < 0$). If the step length is λ, the move leads to the point $\bar{x}^r + \lambda y$. Select a small positive number ϵ_1 as the tolerance for minimum $\{A_{i.}x - b_i\colon i = 1 \text{ to } m\}$ for the point x to be in the interior of K. Then, we will take the step length from $\bar{x}^r$ in the direction y to be $(-\epsilon_1)$ + (the maximum step length possible while remaining inside K), which is

$$\gamma = \text{minimum}\left\{\frac{-A_{i.}\bar{x}^r + b_i + \epsilon_1}{A_{i.}y}\colon i \text{ such that } A_{i.}y < 0\right\},$$

and then the point obtained at the end of this descent step will be $\bar{x}^r + \gamma y$ if γ is finite.

If $\gamma = \infty$, the objective function $z(x)$ is unbounded below in (1), and $\{\bar{x}^r + \lambda y\colon \lambda \geq 0\}$ is a feasible half-line along which $z(x)$ diverges to $-\infty$ on K. Terminate the method if this occurs.

We now list the various descent steps carried out in this iteration. After each descent step, include the point obtained at the end of it, along with its objective value, in a *list*.

Descent Step 1 (D1). From the ball center $\bar{x}^r$ take a descent step in the direction $d^1 = -c^T$.

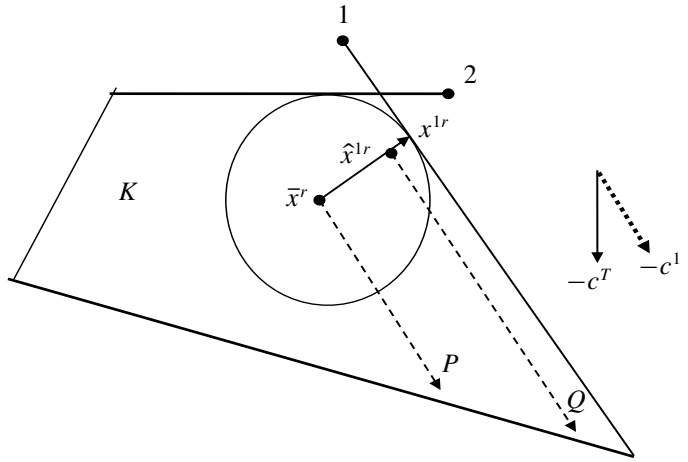
Descent Step 2 (D2). From the ball center $\bar{x}^r$ take a descent step in the direction $d^2 = \bar{x}^r - \bar{x}^{r-1}$, where $\bar{x}^{r-1}$ denotes the ball center computed in the previous iteration r. So, this direction is the direction of the path of ball centers generated in the algorithm.

Descent Step 3 (D3). The directions $-c^i$ for each $i \in T(\bar{x}^r)$ are called the GPTC (gradient projection on touching constraints) at $\bar{x}^r$. Carry out descent steps from the ball center $\bar{x}^r$ in each of the GPTC directions at $\bar{x}^r$.

Descent Step 4 (D4). From the ball center $\bar{x}^r$, take a descent step in the direction $d^4 = (\sum(-c^i\colon \text{for } i \in T(\bar{x}^r))/|T(\bar{x}^r)|)$, the average direction of all the GPTC directions at $\bar{x}^r$.

Descent Step 5.1 (D5.1). For $i \in T(\bar{x}^r)$, let x^{ir} denote the orthogonal projection of the center $\bar{x}^r$ on the touching facetal hyperplane $\{x\colon A_{i.}x = b_i\}$; it is the point where this facetal hyperplane touches the ball $B(\bar{x}_r, \delta(\bar{x}^r))$. The points x^{ir} for $i \in T(\bar{x}^r)$ are called the *touching points* (*TPs*) of the ball $B(\bar{x}_r, \delta(\bar{x}^r))$ with its touching facetal hyperplanes of K. See Figure 4.

FIGURE 4. Illustrating D5.1.



Notes. Here, $T(\bar{x}^r) = \{1, 2\}$ $-c^T$ pointing down south and $-c^1$ equal to the orthogonal projection of $-c^T$ on the facetal hyperplane of Constraint 1 are shown. x^{1r}, $\hat{x}^{1r}$ = TP and NTP, respectively, corresponding to Constraint 1. Descent steps from $\bar{x}^r$ [$\hat{x}^{1r}$] in direction $-c^1$ are shown, leading to P, [Q], and Q is a much better point than P.

Let $0 < \epsilon < 1$ be a small positive tolerance ($\epsilon = 0.1$ works well). Then for $i \in T(\bar{x}^r)$, the point on the line segment joining $\bar{x}^r$ and x^{ir} close to the TP x^{ir},

$$\hat{x}^{ir} = \epsilon \bar{x}^r + (1 - \epsilon) x^{ir},$$

is called the *near touching point* (*NTP*), corresponding to the tangent plane $\{x\colon A_{i.}x = b_i\}$ of the ball $B(\bar{x}_r, \delta(\bar{x}^r))$.

D5.1 consists of $|T(\bar{x}^r)|$ descent steps: for each $i \in T(\bar{x}^r)$, it carries out a descent step in the GPTC direction $-c^i$ from the NTP $\hat{x}^{ir}$. The output of D5.1, denoted by $\tilde{x}^{r1}$, is the best point obtained in it.

When all these descent steps are carried out, the best point among the output points of all the descent steps is the output of this iteration, and with that point the method goes to the next iteration.

Just as other IPMs, this method also terminates when the change in the final points obtained in successive iterations is smaller than some tolerance (i.e., it terminates at the end of the iteration $r+1$ if $\|x^{r+1} - x^r\|/\|x^r\| < \epsilon$, concluding that x^{r+1} is an optimum solution of (1)).

In the next section, we will discuss the improved version, called Sphere Method 2.

5. Sphere Method 2

In Sphere Method 1 the set of feasible solutions considered remains unchanged (i.e., remains the original K) throughout the algorithm, but the current objective plane $\{x\colon cx = t\}$ keeps on sliding parallely toward decreasing values of t from one iteration to the next. The centering step in this method in each iteration has the aim of finding a ball center on the current objective plane, at least in principle. Even though line search directions y used in LSCPD in the centering step in Sphere Method 1 may satisfy $cy < 0$, all the search directions used in LSFN satisfy $cy = 0$, and hence leave the objective value unchanged.

In Sphere Method 2, discussed in Murty and Oskoorouchi [16], in contrast, the set of feasible solutions K considered is updated by the current objective value after each iteration, and hence gets smaller. So, to distinguish, we will denote by K^r the set of feasible solutions considered in step r, and we will have $K^r \subset K^{r-1} \subset K$ for all r. In addition, in the centering step of Sphere Method 2, all line search directions used (in both the LSFN and LSCPD

sequences) will be both profitable and strict descent directions for the original objective function $z = cx$.

Also, in Murty and Kabadi [14], two additional descent steps, D5.2 and D6, to be used in every iteration of the sphere method have been proposed. Of these, D6 performed poorly compared with other descent steps in preliminary steps, hence we will ignore it. However, we will include D5.2 as an additional descent step in every iteration of Sphere Method 2.

The first iteration of Sphere Method 2 begins with the initial interior feasible solution x^0. We will now describe the general iteration, $r+1$, in this method.

5.1. General Iteration $r+1$

The initial point for this iteration is x^r, the interior feasible solution obtained at the end of the previous iteration. Define the set of feasible solutions to be considered for this iteration to be K^{r+1}, where

$$K^{r+1} = \{x\colon Ax \geq b, \text{ and } cx \leq cx^r + \epsilon\}$$

where ϵ is a small positive tolerance parameter. Go to the centering step in this iteration. See Figure 5.

Centering step. The aim of this step is to find a ball center of K^{r+1} approximately, as described earlier.

LSFN. The set of facetal normal directions of K^{r+1} is $\Gamma_1^{r+1} = \{\pm c^T, \pm A_{i\bullet}^T\colon i = 1 \text{ to } m\}$. Apply the LSFN sequence to find a ball center for K^{r+1} as described above using profitable directions for K^{r+1} from Γ_1^{r+1}.

LSCPD. This sequence begins with the interior feasible solution obtained at the end of LSFN.

Let $\hat{x}$ denote the interior feasible solution in a step of this sequence. The touching constraint set at $\hat{x}$ for K^{r+1} will typically include the objective constraint in the definition of K^{r+1}. If it does not, then apply this sequence as discussed above.

On the other hand, if the touching constraint set includes the objective constraint, let $T^{r+1}(\hat{x})$ denote the touching constraint index set for K^{r+1}. Solve the system

$$\begin{aligned} A_{i\bullet}y &= 1 \quad \text{for all } i \in T^{r+1}(\hat{x}) \\ -cy &= \beta, \end{aligned} \tag{9}$$

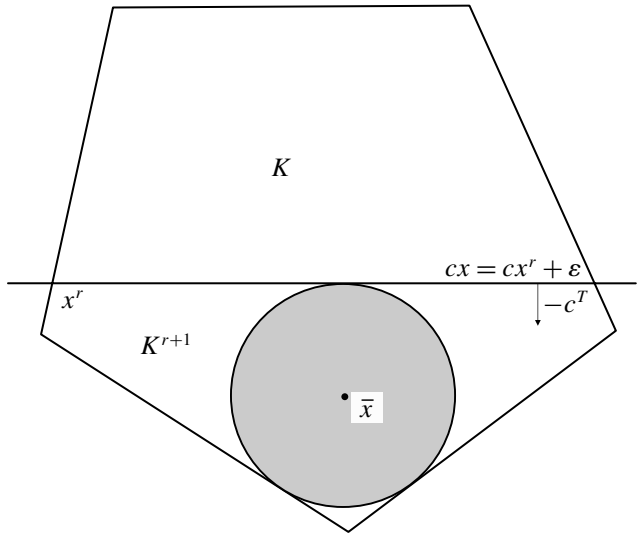
where β is a positive parameter. Earlier we used only $\beta = 1$. However, here we will leave it as a parameter that is restricted to take positive values only and obtain a solution of (9) as a function of this parameter β. Let this solution be denoted by $p + \beta q$.

As in §2.1 of Murty and Oskoorouchi [16], if B is a basis associated with the basic vector y_B obtained for (9), let y_D denote the vector of remaining nonbasic variables in (9) associated with the basic vector y_B. Let $p = (p_B, p_D), q = (q_B, q_D)$ be the partition of the vectors p, q corresponding to the partition of y into basic and nonbasic parts (y_B, y_D). Then, $q_D = p_D = 0$, q_B is the last column of B^{-1}, and p_B is the sum of the remaining columns of B^{-1}.

So, for all $\beta > 0$, $p + \beta q$ is a profitable direction at $\hat{x}$ for K^{r+1}. With $p + \beta q$ as line search direction, the optimum step length α (maximizing $\delta(\hat{x} + \alpha(p + \beta q))$, the radius of the maximum radius ball inscribed in K^{r+1} with $\hat{x} + \alpha(p + \beta q)$ as center) is determined by solving the 3 variable LP in variables δ, α, and γ:

$$\begin{aligned} \text{Maximize } & \delta \\ \text{subject to } & \delta - \alpha A_{i\bullet}p - \gamma A_{i\bullet}q \leq A_{i\bullet}\hat{x} - b_i, \quad i = 1, \ldots, m, \\ & \delta - \alpha(-c)p - \gamma(-c)q \leq (-c)\hat{x} - ((-c)x^r - \epsilon), \\ & \delta, \alpha, \gamma \geq 0. \end{aligned}$$

FIGURE 5. Illustrating K, K^{r+1}.



Notes. K is the original set of feasible solutions of the LP being solved. The current set of feasible solutions in an iteration when x^r is the initial interior feasible solution is K^{r+1}. The ball shown is the largest ball inside K^{r+1}, and its center $\bar{x}$ is a ball center obtained in the centering step in this iteration.

Here, α, γ will both be > 0 at optimum. Actually, this γ is $(\alpha)(\beta)$.

If $(\bar{\delta}, \bar{\alpha}, \bar{\gamma})$ is an optimum solution of this three-variable LP, then the point obtained at the end of this step is $\hat{x} + \bar{\alpha}p + \bar{\gamma}q$. With that, the next LSCPD step is applied again as here, and so on until the LSCPD sequence is completed.

Let $\bar{x}$ denote the point obtained at the end of LSCPD; it is the approximate ball center of K^{r+1} obtained in this iteration. See Figure 5.

The descent steps. With the point $\bar{x}$ obtained at the end of the centering step, the iteration moves to the descent steps in this iteration for the current set of feasible solutions K^{r+1}.

It first applies descent steps D1 to D5.1 as described in Sphere Method 1 in the current set of feasible solutions K^{r+1}. Let $\tilde{x}^{r1}$ denote the best point (by objective value) obtained in Descent Steps D1 to D5.1. This $\tilde{x}^{r1}$ is the initial interior feasible solution for Descent Step 5.2.

Descent Step 5.2 (D5.2). By the way the descent steps are carried out, it is clear that $\tilde{x}^{r1}$ is close to the boundary of K^{r+1}, and $\delta(\tilde{x}^{r1}) \leq \epsilon_1$. Find the touching set $T(\tilde{x}^{r1}) =$ set of all constraint indices for the current set of feasible solutions that tie for the minimum in $\{A_{i.}\tilde{x}^{r1} - b_i\colon i = 1 \text{ to } m; -c\tilde{x}^{r1} + cx^r + \epsilon\}$.

For each $i \in T(\tilde{x}^{r1})$, from $\tilde{x}^{r1}$ take a descent step in the GPTC direction $-c^i$ and include the resulting point along with its objective value in a new List 5.2.

At the end, let $\tilde{x}^{r2}$ denote the best point in List 5.2 by objective value. If $c\tilde{x}^{r1} - c\tilde{x}^{r2}$ is

$\leq$ some selected tolerance for objective value reduction, take $\tilde{x}^{r2}$ as the output of this Descent Step 5.2, and put $\tilde{x}^{r2}$ along with its objective value in the list.

$>$ the selected tolerance for objective value reduction, with $\tilde{x}^{r2}$ as the initial interior feasible solution repeat the Descent Step 5.2, and continue the same way.

After all these descent steps are carried out, the best point among the outputs of all the descent steps carried out in this iteration is taken as the output of this iteration. With that point the method goes to the next iteration. Termination criteria are the same as in Sphere Method 1.

Instead of giving β the specific value 1 as in earlier methods, leaving it as a positive parameter in (9) improves the performance of the centering step in Sphere Method 2.

5.2. Computational Performance of Sphere Methods

Murty and Oskoorouchi [15] present results on preliminary computational tests with Sphere Method 1 on randomly generated problems and compare its performance with that of the simplex method. The results indicate that Sphere Method 1 is faster than the simplex method, particularly as problem size (m, n) increases. They show that on randomly generated LPs with $m = 300$, $n = 100$, and densities ranging from 10% to 100%, the percentage move toward the optimum objective value per iteration varied from 10 to 20 in Sphere Method 1, whereas in the simplex method it varied from 0.22 to 0.50.

Murty and Oskoorouchi [16] presents results comparing the performance of Sphere Methods 1 and 2, and show that Sphere Method 2 is about 20% to 40% faster than Sphere Method 1.

6. How to Improve the Performance of Sphere Methods Even Further

The most computationally expensive step in the sphere methods is the centering step in each iteration, so reducing the number of times this step has to be carried out will improve the performance of the methods.

The computation of ball centers of polytopes poses the following problems:

Problem 1. Given an interior point x in the polytope P, what is the radius $\delta(x)$ of the largest ball with x as center that is contained inside P?

Problem 2. We want to find a largest ball inside the polytope P. Is it unique, and if so, what is its center and radius? If not, what is the set of all points, each of which is the center of a largest ball inside P? Also, given an interior point of P, how can we check whether it is the center of a largest ball inside P?

Problem 3. Given an interior point $\bar{x}$ of P of full dimension in R^n, a $y \in R^n, y \neq 0$, is said to be a *profitable direction* at $\bar{x}$ if $\delta(\bar{x} + \alpha y)$ increases strictly as α increases from 0. How can we check efficiently whether a given $y \neq 0$ is a profitable direction at $\bar{x}$? How can we check whether there exists a profitable direction at $\bar{x}$, and if so, how can we compute one such direction efficiently?

There are two ways of representing a convex polytope. One way represents it as the set of feasible solutions of a given system of linear constraints. This is the representation we used for the polytope K in earlier sections, and we have seen that Problems 1, 2, and 3 can be solved efficiently under this representation of the convex polytope.

Another way is to give the set of all the extreme points of the polytope and represent the polytope as the convex hull of this set. Let $\{x^1, \ldots, x^L\}$ be the set of all extreme points of a convex polytope P. In this way, P is represented as the convex hull of this set of its extreme points, denoted by $< \{x^1, \ldots, x^L\} >$.

However, there are no efficient methods known to solve any of Problems 1, 2, and 3 listed above on the polytope P represented as the convex hull of its extreme points. All these problems on P are open problems at the moment.

However, when P is a special polytope, a simplex, or the convex hull of a set of vectors that is linearly independent, we show in the next two sections that its ball center is unique and can be computed efficiently and directly. These results are taken from Murty [13]. Then in §9 we discuss how these results can be used to improve the performance of sphere methods.

7. Ball Center of an n-Dimensional Simplex in R^n

7.1. Ball Center of a Simplex Represented by Constraints

Let S be an n-dimensional simplex in R^n; i.e., its representation using linear constraints is of the form

$$S = \{x\colon D_{i\cdot} \geq d_i \text{ for } i = 1 \text{ to } n+1\},$$

where $d = (d_i) \in R^{n+1}$, and the coefficient matrix D of order $(n+1) \times n$ with rows $D_{i\cdot}$, $i = 1$ to $n+1$, satisfies the properties that all the $(n+1)$ submatrices of it of order $n \times n$ are nonsingular, and that each row vector of D is a linear combination of the other rows with strictly negative coefficients. Without any loss of generality we will assume that all the rows of D have been normalized so that $\|D_{i\cdot}\| = 1$ for all i.

We will now show that S has a unique ball center that is the unique solution of the system of linear equations

$$Dx - \delta e = d,$$

where x will be the ball center of S in the solutions, and δ is the radius the largest ball inside S with center at that x.

It can be verified that the coefficient matrix of this system is nonsingular; hence, this system has a unique solution, $(\bar{x}, \bar{\delta})$. The ball $B(\bar{x}, \bar{\delta})$ with $\bar{x}$ as the center and $\bar{\delta}$ as radius is inside S and touches all the facets of S, so it is the largest ball with $\bar{x}$ as center inside S.

Also, the system $Dy \geq 0$ has 0 as its unique solution, because $D_{n+1\cdot} = \alpha_1 D_{1\cdot} + \cdots + \alpha_n D_{n\cdot}$ with all $\alpha_1, \ldots, \alpha_n < 0$.

Applying the conditions mentioned in Theorem 1, we conclude that $\bar{x}$ is the ball center of K, and it is unique. So, when the simplex S is represented using linear constraints, its ball center can be computed efficiently as described above.

7.2. Ball Center of a Simplex Represented as the Convex Hull of Its Extreme Points

Now suppose that S is represented as the convex hull of its set of extreme points $\Gamma = \{x^1, \ldots, x^{n+1}\}$. So, an $x \in S$ is represented as

$$x = \beta_1(x^1 - x^{n+1}) + \cdots + \beta_n(x^n - x^{n+1}) + x^{n+1}, \tag{10}$$

where $\beta_1, \ldots, \beta_n \geq 0$, and $\beta_1 + \cdots + \beta_n \leq 1$.

Let B denote the $n \times n$ matrix with its jth column $B_{\cdot j} = x^j - x^{n+1}$ for $j = 1$ to n, and $\beta = (\beta_1, \ldots, \beta_n)^T$. Because S is a simplex we know that B is nonsingular. So, from (10), we have

$$B^{-1}x = \beta + B^{-1}x^{n+1}.$$

Using this, from the bounds on β we have

$$\begin{aligned} B^{-1}x &\geq B^{-1}x^{n+1}, \\ -eB^{-1}x &\geq -1 - eB^{-1}x^{n+1}. \end{aligned} \tag{11}$$

So, (11) is the representation of S = convex hull of Γ here, through linear constraints. To derive the ball center of S using this representation, (11), we need to normalize each constraint in (11) so that the Euclidean norm of its coefficient vector is 1. For this we need $\gamma_i = \|(B^{-1})_{i\cdot}\|$ for $i = 1$ to n, and $\gamma_{n+1} = \|eB^{-1}\|$. Then, from the results in §7.1, we know that the ball center x of S and the radius δ of the largest ball inside S are the solution of the system

$$\begin{pmatrix} B^{-1} & -\gamma \\ -eB^{-1} & -\gamma_{n+1} \end{pmatrix} \begin{pmatrix} x \\ \delta \end{pmatrix} = \begin{pmatrix} B^{-1}x^{n+1} \\ -1 - eB^{-1}x^{n+1} \end{pmatrix},$$

where $\gamma = (\gamma_1, \ldots, \gamma_n)^T$. By adding the sum of the first n equations in this system to the last, we find that in the solution of this system,

$$\delta = 1/(\gamma_1 + \cdots + \gamma_{n+1}),$$

and from the first n equations we see that the ball center of the simplex S here is

$$x = x^{n+1} + (1/(\gamma_1 + \cdots + \gamma_{n+1}))B\gamma.$$

8. Ball Center of the Convex Hull of a Linearly Independent Set of Vectors in R^n

Let $P = \{x^1, \ldots, x^r\}$ be a linearly independent set of column vectors in R^n, and $S_2 =$ convex hull of P, where $r \leq n$. Then, S_2 is an $(r-1)$-dimensional simplex in its affine hull. In this section we discuss how to compute the ball center of S_2 directly.

In this case, the $n \times (r-1)$ matrix

$$B_2 = (x^1 - x^r \vdots \cdots \vdots x^{r-1} - x^r)$$

is of full column rank. Find a row partition of it into $(B_{21}, B_{22})^T$ such that B_{21} and B_{22} are of orders $(r-1) \times (r-1)$ and $(n-r+1) \times (n-r+1)$, respectively, and B_{21} is nonsingular.

Let $(\underline{x}_1^j, \underline{x}_2^j)^T$, $(\underline{x}_1, \underline{x}_2)^T$ be the corresponding row partitions of the column vectors x^j for each $j = 1$ to r and each $x \in S_2$, respectively.

Then, for each $x = (\underline{x}_1, \underline{x}_2)^T \in S_2$, it can be verified that

$$\underline{x}_2 = B_{22}\left[B_{21}^{-1}(\underline{x}_1 - \underline{x}_1^r)\right] + \underline{x}_2^r. \tag{12}$$

Equation (12) is the system of linear equations that defines the affine hull of S_2.

Now, the convex hull of $\{\underline{x}_1^1, \ldots, \underline{x}_1^r\} \subset R^{r-1}$ is a full-dimensional simplex in R^{r-1}, and its ball center $\underline{x}_1$ can be found by applying the formula derived in §7.2 to it. Then in the original space R^n, the ball center of S_2 is $(\underline{x}_1, \underline{x}_2)^T$, where $\underline{x}_2$ is obtained from $\underline{x}_1$ using (12).

9. Application to Improve the Performance of Sphere Methods

Let $\{\hat{x}^1, \ldots, \hat{x}^s\}$ denote the set of all points obtained at the end of the various descent steps in D5.1 in an iteration. $s \leq m$, and typically $s \leq n+1$, so s is not a large number in comparison to m, n.

Let K_1 denote the convex hull of $\{\hat{x}^1, \ldots, \hat{x}^s\}$. Typically, $\hat{x}^1, \ldots, \hat{x}^s$ are spread out in different directions all around K, each one in the interior of K but close to the boundary of K. So, intuitively, it seems that a ball center for K_1 may be close to a ball center for K on the objective plane through it.

We checked whether the average of $\hat{x}^1, \ldots, \hat{x}^s$ is a reasonable approximation to a ball center of K_1, but it is not.

Let $\hat{x}^s$ be the best point in $\{\hat{x}^1, \ldots, \hat{x}^s\}$ by objective value. So it is the output of D5.1 in this iteration.

Case 1. First consider the case where $s = n+1$. Typically, we find in this case that $\langle\{\hat{x}^1, \ldots, \hat{x}^s\}\rangle$ is an n-dimensional simplex. Let us assume that this is true. Then, from §7.2 we have the following: let

$$B = (x^1 - x^s \vdots \cdots \vdots x^{s-1} - x^s),$$

and let $\gamma_j = \|(B^{-1})_{j\bullet}\|$ for $j = 1$ to $s-1$. We can of course compute B^{-1} and all γ_j exactly, but we can take $\bar{\gamma}_j = (1/(\|B_{\bullet j}\|)) = 1/(\|x_j - x_s\|)$ as a crude approximation of γ_j. Let $\bar{y} = \sum_{j=1}^{s-1} \bar{\gamma}_j(\hat{x}^j - \hat{x}^s)$. Then, from the results in §7.2 we know that in this case $x^s + (\bar{y}) \times$ (a constant) is an approximation to the ball center of $\langle\{\hat{x}^1, \ldots, \hat{x}^s\}\rangle$.

However, in Sphere Method 1, what we need is a point which is a ball center of K on the objective plane through it. So, the above result suggests that we can use the two-variable LP (6) with $\bar{x}$ replaced by x^s and y replaced by $\bar{y}$ here, and take the optimal $\tilde{x}$ for this two-variable LP as an approximation to the ball center required. Because $c\bar{y} > 0$, $\tilde{x}$ will not be a better point than x^s by objective value, but carrying out all the descent steps in Sphere Method 1 with $\tilde{x}$ as center can be expected to yield a point better than x^s.

Even when $s \neq n+1$, we can carry out the same work using the same formulas to get $\tilde{x}$ as an approximate center.

This leads to a new step, which we call D5.3, to be carried out immediately after completing the step D5.1 in an iteration, which we describe below.

Descent Step 5.3 (D5.3). This is carried out immediately after applying D5.1. Let $\{\hat{x}^1, \ldots, \hat{x}^s\}$ be the set of all points obtained at the end of the various descent steps in D5.1, with x^s as the best point among them.

For each $j = 1$ to $s-1$, let $\bar{\gamma}_j = 1/(\|x_j - x_s\|)$, and let $\bar{y} = \sum_{j=1}^{s-1} \bar{\gamma}_j(\hat{x}^j - \hat{x}^s)$.

Solve the two-variable LP (6) with $\bar{x}$ replaced by x^s, and y replaced by $\bar{y}$ here, and let $\tilde{x}$ be the optimum solution to it.

With $\tilde{x}$ as the approximate ball center, apply all the descent steps D1, D2, D3, D4, and D5.1 in Sphere Method 1, and let $\bar{\tilde{x}}$ denote the best point obtained in all these descent steps.

Repeat this work with the set of points obtained at the end of the current D5.1, and if the new point $\bar{\tilde{x}}$ shows good improvement over the last one, continue the same.

This step stops with the point $\bar{\tilde{x}}$ obtained in the final repetition; with that go to the next iteration in Sphere Method 1 beginning with centering again.

Sphere Method 1 with this additional descent step has not yet been tested computationally; we will be testing it shortly.

References

[1] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization* 5:13–51, 1995.

[2] F. Alizadeh and D. Goldfarb. Second-order cone programming. *Mathematical Programming* 95(1):3-51, 2003.

[3] C. Cartis and N. I. M. Gould. Finding a point in the relative interior of a polyhedron. RAL Technical Report 2006-016. *Optimization Online*, http://www.optimization-online.org, 2006.

[4] G. B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ, 1963.

[5] G. B. Dantzig and M. N. Thapa. *Linear Programming*, Vol. 1. *Introduction*, Vol. 2. *Theory and Extensions*. Springer-Verlag, New York, 1997.

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

[7] M. Kojima, S. Mizuno, and A. Yoshise. A primal-dual interior point algorithm for linear programming. N. Megiddo, ed. *Progress in Mathematical Programming: Interior Point and Related Methods*. Springer-Verlag, New York, 29–47, 1989.

[8] N. Megiddo. Pathways to the optimal set in linear programming. N. Megiddo, ed. *Progress in Mathematical Programming: Interior Point and Related Methods*. Springer-Verlag, New York, 131–158, 1989.

[9] S. Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal on Optimization* 2:575–601, 1992.

[10] R. D. C. Monteiro and I. Adler. Interior path-following primal-dual algorithms, Part I: Linear programming. *Mathematical Programming* 44:27–41, 1989.

[11] K. G. Murty. A new practically efficient interior point method for LP. *Algorithmic Operations Research* 1:3–19, 2006. http://journals.hil.unb.ca/index.php/AOR/index.

[12] K. G. Murty. Linear equations, inequalities, linear programs (LP), and a new efficient algorithm. *INFORMS Tutorials in Operations Research* 1–36, 2006.

[13] K. G. Murty. Ball centers of special polytopes. Working paper, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, 2009.

[14] K. G. Murty and S. N. Kabadi. Additional descent steps in the sphere method. Working paper, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, 2008.

[15] K. G. Murty and M. R. Oskoorouchi. Note on implementing the new sphere method for LP using matrix inversions sparingly. *Optimization Letters* 137–160, 2008.

[16] K. G. Murty and M. R. Oskoorouchi. Sphere methods for LP. Working paper, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, 2008.

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

[18] S. J. Wright. *Primal-Dual Interior-Point Methods*. SIAM, Philadelphia, 1997.

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