SCHOOL OF OPERATIONS RESEARCH
AND INDUSTRIAL ENGINEERING
COLLEGE OF ENGINEERING
CORNELL UNIVERSITY
ITHACA, NEW YORK 14853

TECHNICAL REPORT NO. 903

April 1990
Revised December 1990

A LOW COMPLEXITY INTERIOR-POINT ALGORITHM FOR LINEAR PROGRAMMING

By

Michael J. Todd

Research was supported in part by NSF grant DMS-8904406 and by NSF, AFORS and ONR through NSF grant DMS-8920550.

Abstract

We describe an interior-point algorithm for linear programming that is almost as simple as the affine-scaling method and yet achieves the currently best complexity of $O(\sqrt{n}t)$ iterations to attain precision t. The basic algorithm needs neither dual estimates nor lower bounds, although its analysis is based on Ye's results for the primal-dual potential function. We also present some computationally preferable variants.

1. Introduction

The polynomial-time interior-point algorithms that have been developed in the last few years can be roughly classified as follows:

i) projective-scaling algorithms, stemming from Karmarkar's original method [16], such as those of Anstreicher [1], de Ghellinck and Vial [9], and Todd and Burrell [28];

ii) path-following methods, which attempt to follow closely the central path studied by Megiddo [20] and Bayer and Lagarias [6], such as the dual algorithm of Renegar [26], the primal algorithm of Gonzaga [11], and the primal-dual algorithms of Kojima-Mizuno-Yoshise [17,18] and Monteiro-Adler [24, 25]; and

iii) potential-reduction algorithms, such as those of Gonzaga [12], Ye [31], Freund[8], Kojima-Mizuno-Yoshise [19], Gonzaga [13], and Anstreicher [2, 3].

While the derivations of these methods follow very different lines, the search directions employed are invariably linear combinations of two directions: the affine-scaling direction and the centering direction, which try respectively to improve the objective function and to drive the current iterate towards the analytic center (Sonnevend [27]) of the feasible region. This property of the search direction was noted in several papers: Yamashita [30], Gonzaga [10], Mitchell-Todd [22], Zimmerman [32], and the recent survey of den Hertog and Roos [15]. (In case an algorithm does not require a feasible starting point and generates infeasible iterates, a third "feasibility" direction is also included in the search direction; see, e.g., de Ghellinck and Vial [9] and Anstreicher [2].)

The affine-scaling direction mentioned above is the basis of the affine-scaling algorithm first proposed by Dikin [7] and rediscovered by Barnes [4] and Vanderbei-Meketon-Freedman [29]. This method is believed not to be polynomial on the basis of results of Megiddo and Shub [21], although a variant that includes centering steps does possess a polynomial time bound (Barnes-Chopra-Jensen [5]). The convergence results assume a step a fixed proportion of the way either to the boundary of the feasible region [29] or to the boundary of the inscribed ellipsoid [7,4], which corresponds to a step of fixed Euclidean length in the transformed space.

In this paper we propose a new algorithm whose search direction is a very simple combination of the affine-scaling and constant-cost centering directions. The step length is a constant in the transformed space. This simple algorithm attains the best-known complexity for the number of iterations without requiring the generation of lower bounds on the objective value or of dual feasible iterates. The proof however does make use of results of Ye [31] concerning the primal-dual potential function used in his potential reduction algorithm.

Complexity results given in the literature typically address the case of a linear programming problem with integer data, and bound the computational work in terms of the number of inequalities n (the number of variables in a standard form problem) and the length L of the input (the total number of bits necessary to describe the problem). Hence, after suitable initialization, the projective-scaling algorithms require $O(nL)$ iterations (and $O(n^{3.5}L)$ or $O(n^4L)$ arithmetic operations in total) and the path-following and potential-reduction methods $O(\sqrt{n}L)$ iterations (and $O(n^3L)$ or $O(n^{3.5}L)$ arithmetic operations). This number of iterations guarantees a feasible solution that is close enough to optimal that an exact solution can be obtained with modest additional computational effort. However, we feel it is more appropriate for linear programming (where the data are usually regarded as real) to state the complexity results in terms of n and a parameter t, which represents the precision required as well as the quality (initial objective value and "closeness" to the central path) of the initial solution. Our algorithms require $O(\sqrt{n}t)$ or $O(nt)$ iterations in this sense, and easily translate to $O(\sqrt{n}L)$ or $O(nL)$ iteration methods in the integer data case.

Section 2 describes the basic algorithm, and the $O(\sqrt{n}t)$ complexity result is derived in section 3. Section 4 describes two variants, one of which maintains this complexity while the other requires $O(nt)$ iterations. These variants sacrifice some of the simplicity of the basic algorithm for improved practical behavior, and, in particular, recur lower bound estimates of the optimal value; the algorithms then bear a strong resemblance to those of Gonzaga [12], Ye [31] and Freund [8]. Section 5 contains the results of preliminary computational experience showing the superiority of the second variant. This illustrates a phenomenon which has also been observed elsewhere: the better practical versions of interior-point algorithms frequently do not have the best theoretical complexity bounds. Our results also suggest the most important modifications in making the basic algorithm efficient; line searches are essential, and then the tightest lower bounds significantly improve performance.

2. The basic algorithm

We consider the linear programming problem in standard form:

$$(P) \qquad \begin{aligned} \min\ & c^T x \\ & Ax = b \\ & x \geq 0, \end{aligned}$$

where A is $m \times n$. Let $F(P) := \{x \in \mathbb{R}^n\colon Ax = b, x \geq 0\}$ denote its feasible region, and $F_+(P) := \{x \in F(P)\colon x > 0\}$ the relative interior of the feasible region. We assume that $F_+(P)$ is nonempty, and that (P) has a nonempty bounded set of optimal solutions. Let v(P) denote the optimal value of (P).

We suppose that an initial point $x^0 \in F_+(P)$ is available. At iteration k we will have the current iterate $x^k \in F_+(P)$, and we define a scaled problem $(\bar{P})$ as follows. Let $X_k := \mathrm{diag}(x^k)$ be the diagonal matrix with the components of x^k down its diagonal, and consider the affine transformation $x \to \bar{x} := X_k^{-1}x$. The image of x^k under this transformation is e, the vector of ones in $\mathbb{R}^n$. In terms of $\bar{x}$, (P) becomes

$$(\bar{P}) \qquad \begin{aligned} \min\ & \bar{c}^T\bar{x} \\ \bar{A}\bar{x} &= b \\ \bar{x} &\geq 0, \end{aligned}$$

where $\bar{A} := AX_k$ and $\bar{c} := X_k c$.

In the scaled problem $(\bar{P})$, there are two very important directions. Let $P_{\bar{A}}$ denote projection into the null space of $\bar{A}$. The first direction is the affine-scaling direction

$$-\bar{c}_P := -P_{\bar{A}}\,\bar{c}; \tag{1}$$

the second is the projection of the negative gradient of the barrier function

$$p(\bar{x}) := -\sum_j \ell n\,\bar{x}_j \tag{2}$$

evaluated at the point $\bar{x} = e$, which is

$$e_P := P_{\bar{A}}e. \tag{3}$$

If $\bar{c}_P = 0$, then it is easy to see that all feasible points of $(\bar{P})$ have the same objective function value, so are optimal, and hence x^k is optimal in (P). Henceforth we assume that $\bar{c}_P \neq 0$.

Most of the directions we are concerned with are combinations of the form

$$\bar{d}_\beta := -\beta\bar{c}_P + e_P \tag{4}$$

of our two basic directions, for some scalar β. In particular, the direction $\bar{d}_\alpha$, where

$$\alpha := \bar{c}_P^T e \,/\, \bar{c}_P^T\bar{c}_P, \tag{5}$$

will be very important to us. We note that it has three properties:

$$\bar{d}_\alpha = \text{argmin } \{||\bar{d}||: \bar{d} = \bar{d}_\beta \text{ for some } \beta\}; \tag{6}$$

$$\bar{d}_\alpha^T \bar{c}_P = 0; \text{ and} \tag{7}$$

$$\bar{d}_\beta^T \bar{d}_\alpha = \bar{d}_\alpha^T \bar{d}_\alpha \text{ for all } \beta. \tag{8}$$

It is easy to see that $\bar{d}_\alpha$ is the steepest descent direction for the barrier function in the set $\{\bar{x} : \bar{A}\bar{x} = b, \bar{c}^T\bar{x} = \bar{c}^T e\}$, so we call $\bar{d}_\alpha$ the <u>constant-cost</u> <u>centering</u> <u>direction</u>. This direction appears in the centered version of the affine-scaling algorithm due to Barnes, Chopra and Jensen [5] and in the monotonic versions of the standard-form projective variant (Anstreicher [1]) and of the scaled potential algorithm (Anstreicher [3]).

The direction of our algorithm is then chosen as follows:

<u>Case 1:</u> $||\bar{d}_\alpha|| \geq \cdot 3$. Then set

$$\bar{d} = \bar{d}_\alpha \,/\, ||\bar{d}_\alpha||. \tag{9}$$

<u>Case 2:</u> $||\bar{d}_\alpha|| < \cdot 3$. Then set

$$\bar{d} = - \bar{c}_P \,/\, ||\bar{c}_P||. \tag{10}$$

Thus our direction is proportional either to the constant-cost centering direction or the affine-scaling direction, and in either case is normalized to have length 1.

Having defined the direction $\bar{d}$, we take a step of length $\cdot 2$, so that

$$\bar{x}_+ = e + \cdot 2\bar{d} \tag{11}$$

in the transformed space, and then

$$x^{k+1} = X_k(e + \cdot 2\bar{d}) = x^k + \cdot 2X_k\bar{d}, \tag{12}$$

Since $||\bar{d}|| = 1$, $\bar{x}_+ > 0$ and it is easy to check that $\bar{A}\bar{x}_+ = b$; hence $x^{k+1} \in F_+(P)$. Moreover, $\bar{c}^T\bar{x}_+ \leq \bar{c}^T e$ (strictly if case 2 obtains) so that $c^T x^{k+1} \leq c^T x^k$.

3. Analysis

In this section we show that, if x^0 is suitably chosen, in $O(\sqrt{n}t)$ iterations we will have an iterate $x^k \in F_+(P)$ with

$$c^T x^k - v(P) \leq 2^{-t}. \tag{13}$$

The argument uses the results proved by Ye [31] in his scaled potential reduction algorithm.

The dual of (P) is

$$(D) \qquad \begin{aligned} \max\ & b^T y \\ & A^T y + s = c \\ & s \geq 0, \end{aligned}$$

and, for any $x \in F(P)$ and (y, s) feasible in (D), the duality gap is $b^T y - c^T x = x^T s \geq 0$. Let $F(D) = \{s \in \mathbb{R}^n : A^T y + s = c \text{ for some } y \text{ and } s \geq 0\}$ and $F_+(D) = \{s \in F(D) : s > 0\}$. The fact that (P) has a nonempty bounded optimal solution set implies that $F_+(D) \neq \emptyset$. For an $x \in F_+(P)$ and $s \in F_+(D)$, and for any $q \geq 0$, we define the primal-dual potential function (with parameter q) to be

$$\begin{aligned} \phi_q(x, s) &:= q\,\ell n(x^T s) - \sum_j \ell n\, x_j - \sum_j \ell n\, s_j - \ell n\, n \\ &= (q - n)\,\ell n(x^T s) - \sum_j \ell n \frac{x_j s_j}{x^T s/n} \\ &\geq (q - n)\,\ell n(x^T s) \end{aligned} \tag{14}$$

since the $x_j s_j/(x^T s/n)$ terms are positive with arithmetic mean one. We also use $\phi(x, s)$ to denote $\phi_{\bar{q}}(x, s)$ where $\bar{q} := n + \sqrt{n}$.

The condition we require on $x^0 \in F_+(P)$ is that, for some $s^0 \in F_+(D)$ (which need not be known), $\phi(x^0, s^0) = O(\sqrt{n}t)$. (When the data of (P) are integer, Monteiro and Adler [24, 25] show how to construct a related linear programming problem for which such an initial (x^0, s^0) can easily be obtained, with $t = L$, the size of the input.)

Suppose that at each iteration we can reduce ϕ by a constant. Then in $O(\sqrt{n}t)$ iterations we will have (x^k, s^k) with $\phi(x^k, s^k) \leq -\sqrt{n}t$, so that by (14)

$$c^T x^k - v(P) \leq (x^k)^T s^k \leq 2^{-t}$$

as required.

We aim to show that this constant reduction is achieved even though the iterates s^k are not explicitly computed. For each $x^k \in F_+(P)$, we use an associated $s^k = s(x^k) \in F_+(D)$ that minimizes $\phi(x^k, \cdot)$. First we show the existence and uniqueness of such an s^k. (In fact, it is not hard to see that s^k is on the central path [6,20] for the dual.)

Proposition 1. If $\hat{x} \in F_+(P)$ then $\inf\{\phi(\hat{x}, s) : s \in F_+(D)\}$ is attained by a unique $\hat{s} \in F_+(D)$. Write $\hat{s} = s(\hat{x})$. If $\tilde{x} \in F_+(P)$ with $c^T\tilde{x} \leq c^T\hat{x}$, then $c^T\hat{x} - \hat{x}^T\hat{s} \leq c^T\tilde{x} - \tilde{x}^T\tilde{s}$, where $\hat{s} = s(\hat{x})$, $\tilde{s} = s(\tilde{x})$.

Proof. Choose any $\bar{s} \in F_+(D)$; then we can confine the minimization to those $s \in F_+(D)$ with $\phi(\hat{x}, s) \leq \phi(\hat{x}, \bar{s})$. By (14) this shows that we can add the constraint $\hat{x}^T s \leq \mu$ for some μ, and since $\hat{x} > 0$, that s can be confined to a bounded set. Next, since $\phi(\hat{x}, s) \geq \sqrt{n}\, \ell n(\hat{x}^T s) - \ell n\, (\hat{x}_j s_j) + \ell n(\hat{x}^T s) - \ell n\, n$, and $\hat{x}^T s \geq c^T\hat{x} - v(P) > 0$, we can further restrict s_j to be at least some positive $\underline{s}_j$, and this argument applies to each j. But $\phi(\hat{x}, \cdot)$ is continuous on the compact set $\{s \in F(D): \hat{x}^T s \leq \mu, s_j \geq \underline{s}_j \text{ all } j\}$, so it attains its minimum there, and existence follows. Uniqueness is then implied by the argument of [28, Lemma 2.3]. So we can write $\hat{s} = s(\hat{x})$.

Now suppose $\hat{s} = s(\hat{x}), \tilde{s} = s(\tilde{x})$, where $c^T\tilde{x} \leq c^T\hat{x}$. If $c^T\tilde{x} = c^T\hat{x}$, then $\phi(\hat{x}, \cdot)$ and $\phi(\tilde{x}, \cdot)$ differ by a constant, so $\hat{s} = \tilde{s}$ and the second part follows easily; both sides of the inequality equal $b^T\hat{y}$, where $(\hat{y}, \hat{s})$ is feasible in (D). So assume that $c^T\tilde{x} < c^T\hat{x}$. Then from $\phi(\hat{x}, \hat{s}) \leq \phi(\hat{x}, \tilde{s})$ and $\phi(\tilde{x}, \tilde{s}) \leq \phi(\tilde{x}, \hat{s})$ we deduce that $\ell n\, \hat{x}^T\hat{s} + \ell n\, \tilde{x}^T\tilde{s} \leq \ell n\, \hat{x}^T\tilde{s} + \ell n\, \tilde{x}^T\hat{s}$. Suppose $(\hat{y}, \hat{s})$ and $(\tilde{y}, \tilde{s})$ are feasible in (D). By exponentiating the last inequality and simplifying, we find

$$(c^T\hat{x} - c^T\tilde{x})(b^T\hat{y} - b^T\tilde{y}) \leq 0.$$

Then $b^T\hat{y} \leq b^T\tilde{y}$, which yields the second part.

We will show that $\phi(x^{k+1}, s(x^k)) \leq \phi(x^k, s(x^k)) - \cdot 02$ for each k, so that

$$\phi(x^{k+1}, s(x^{k+1})) \leq \phi(x^k, s(x^k)) - \cdot 02 \tag{15}$$

a fortiori. This will prove the desired complexity result. (In effect, we are working with the primal-only potential function

$$\psi(x) := \min\{\phi(x, s) : s \in F_+(D)\} = \phi(x, s(x)), \tag{16}$$

which is the only such function we know that can ensure an $O(\sqrt{n}t)$ iteration bound.)

Note that $\phi(\Lambda^{-1}x, \Lambda s) = \phi(x,s)$ for any positive definite diagonal matrix Λ. We can therefore always scale so that our current iterate x^k is e, and it is straightforward to check that the algorithm of section 2 is invariant under such scaling. We will therefore assume until the statement of Theorem 1 that such a scaling has already been performed, so that $x^k = e$, and we omit the overbars in our notation, so that $c_P = P_A c$, $e_P = P_A e$, $d_\beta = -\beta c_P + e_P$, $\alpha = c_P^T e / c_P^T c_P$, and d is our search direction. We wish to show that

$$\phi(e + \cdot 2d, s(e)) \leq \phi(e, s(e)) - \cdot 02; \tag{17}$$

this will then imply (15) as desired.

A key point is that, for any $q \geq 0$,

$$- P_A(\nabla_x \phi_q(e, s(e))) = - P_A\left(\frac{q}{e^T s(e)} s(e) - e\right)$$

$$= - \frac{q}{e^T s(e)} c_P + e_P$$

(since $A^T y + s(e) = c$ for some y, $P_A s(e) = P_A c = c_P$), and this is of the form d_β for some $\beta \geq 0$. We now have

<u>Lemma 1</u>. For any $q \geq n + \sqrt{n}$,

$$\left\| - \frac{q}{e^T s(e)} c_P + e_P \right\| \geq \cdot 4. \tag{18}$$

<u>Proof</u>. Assume the contrary, so that $\|h\| < \cdot 4$ where $h = (q/e^T s(e))\, c_P - e_P$ for some $q \geq n + \sqrt{n}$. Then, for some y,

$$A^T y + \frac{e^T s(e)}{q}(e + h) = c,$$

so that

$$\bar{s} := \frac{e^T s(e)}{q} (e + h) \in F_+(D). \tag{19}$$

Moreover, the associated duality gap is

$$e^T \bar{s} = \frac{e^T s(e)}{q} (e^T(e + h)) \leq \frac{e^T s(e)}{n + \sqrt{n}} (n + \cdot 4 \sqrt{n})$$

$$\leq (1 - \frac{\cdot 6\sqrt{n}}{n + \sqrt{n}}) e^T s(e). \tag{20}$$

Finally, lemma 2 of Ye [30] shows that

$$|| \bar{s} - \frac{e^T\bar{s}}{n} e || \leq \cdot 5 \frac{e^T\bar{s}}{n}; \tag{21}$$

note that Ye's argument does not require that q (his ρ) equals $n + \sqrt{n}$. We can then argue as in theorem 1 of Ye [30] that

$$\phi(e, \bar{s}) \leq \phi(e, s(e)) - \frac{\cdot 6}{2} + \frac{(\cdot 5)^2}{2(1 - \cdot 5)}$$

$$< \phi(e, s(e)),$$

which contradicts our choice of s(e). Hence (18) must hold.

We also require the following standard result; see for instance Ye [31].

Lemma 2. If $Ad = 0$, $||d|| = 1$, and $0 < \gamma < 1$, then

$$\phi_q(e + \gamma d, s) \leq \phi_q(e, s) + \gamma \nabla_x \phi_q(e, s)^T d + \frac{\gamma^2}{2(1 - \gamma)}. \tag{22}$$

Now we consider the two cases of our algorithm, and prove that in each case (17) holds. Define δ by $-P_A \nabla_x \phi(e, s(e)) = d_\delta$.

First assume that $||d_\alpha|| \geq \cdot 3$ so that $d = d_\alpha / ||d_\alpha||$. Then

$$\nabla_x \phi(e, s(e))^T d = (P_A \nabla_x \phi(e, s(e)))^T d_\alpha / ||d_\alpha||$$

$$= -d_\delta^T d_\alpha / ||d_\alpha||$$

$$= -d_\alpha^T d_\alpha / ||d_\alpha|| = -||d_\alpha|| \leq -\cdot 3,$$

where the first equality holds since d is in the null space of A, the second by definition of δ, and the third by (8). Hence with $\gamma = \cdot 2$, lemma 2 yields

$$\phi(e + \cdot 2d, s(e)) \leq \phi(e, s(e)) - \cdot 2 \times \cdot 3 + \frac{(\cdot 2)^2}{2(1 - \cdot 2)}$$

$$\leq \phi(e, s(e)) - \cdot 02. \tag{23}$$

Now suppose that $||d_\alpha|| < \cdot 3$. Then $\delta > \alpha$ (otherwise $\alpha = q/e^T s(e)$ for some $q \geq n + \sqrt{n}$, contradicting lemma 1) and in fact, since $||d_\delta|| \geq \cdot 4$, d_δ makes an angle of at least arc $\cos(3/4)$ with d_α, and hence an angle of at most arc $\cos(5/8)$ with $-c_p$ (see Figure 1). Hence, with $d = -c_p/||c_p||$ we have $d_\delta^T d \leq -(5/8) \times \cdot 4 = -\cdot 25$. Using lemma 2 we can conclude

$$\phi(e + \cdot 2d, s(e)) \leq \phi(e, s(e)) - \cdot 2 \times \cdot 25 + \frac{(\cdot 2)^2}{2(1 - \cdot 2)}$$

$$\leq \phi(e, s(e)) - \cdot 02.$$

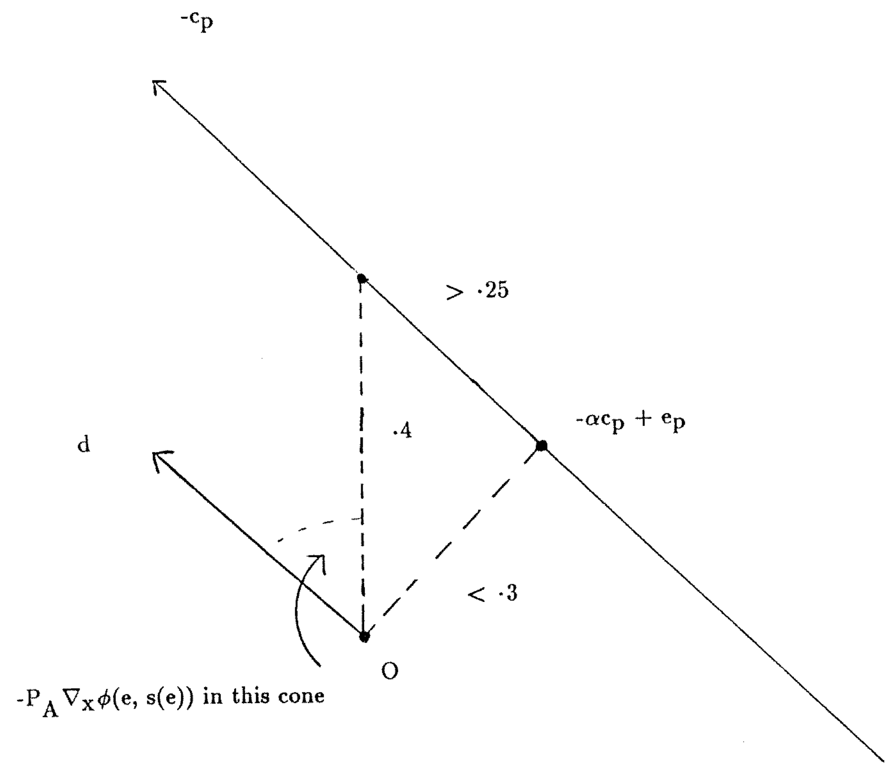
To summarize, we have shown

Theorem 1. At each iteration of the algorithm of section 2,

$$\phi(x^{k+1}, s(x^{k+1})) \leq \phi(x^k, s(x^k)) - \cdot 02. \tag{24}$$

If $\phi(x^0, s^0) = O(\sqrt{n}t)$ for some $s^0 \in F_+(D)$, then after $O(\sqrt{n}t)$ iterations we have x^k with

$$c^T x^k - v(P) \leq 2^{-t}.$$



4. Refinements

The simple algorithm of section 2 does not perform well in practice. Indeed, (12) shows that each component of x^k can decrease by at most 20% in each iteration, so that the best we can hope for is linear convergence with ratio ·8. In this section we describe refinements that attempt to improve the convergence while maintaining the simple structure of the algorithm. Usually, an interior-point method can be improved significantly by incorporating a line search. Here, however, the potential function on which we would like to perform a line search cannot easily be computed, and hence complications arise. We use the insights obtained from the analysis of section 3 and update lower bounds on the optimal value. The resulting methods bear a strong resemblance to those of Gonzaga [12], Ye [31] and Freund [8].

Our first variant maintains inequality (24), and hence provides an $O(\sqrt{n}t)$ iteration method ($O(\sqrt{n}L)$ for a problem with integer data), while our second variant is preferable in practice but has only an $O(nt)$ iteration bound.

At each iteration, we try to update the current lower bound z_k, then choose a search direction and make a line search in this direction to minimize approximately some potential function. In the second variant we only need $z_k \leq v(P)$. In the first, we require further that $c^Tx^k - z_k \geq (x^k)^Ts(x^k)$, and when we update z_k to z_{k+1}, we insist that z_{k+1} also satisfies this inequality. Since our algorithm is monotonic, Proposition 1 then implies that $c^T x^{k+1} - z_{k+1} \geq (x^{k+1})^T s(x^{k+1})$. In both variants we can initialize with $z_0 = -\infty$.

To describe the iteration it is again convenient to assume that the problem has been scaled if necessary so that the current iterate is e. Suppose c_p, e_p and α have been calculated. In the first variant, z_k is updated as follows.

$$\begin{aligned}
&\text{If } \|d_\alpha\| \geq .4,\ z_{k+1} := z_k.\\
&\text{Otherwise, let } \epsilon > \alpha \text{ be such that } \|d_\epsilon\| = .4.\\
&\text{If } \epsilon \leq 0,\ z_{k+1} := z_k.\\
&\text{If } \epsilon > 0,\ \text{ let } z_+ := c^T e - (n + \sqrt{n})/\epsilon \text{ and } z_{k+1} := \max\{z_k, z_+\}.
\end{aligned} \tag{25}$$

This method of updating z_k is very similar to that of Gonzaga [12, section 5] and closely related to those of Freund [8] and Ye [31]. We then have

Lemma 3. Assume that $c^T e - z_k \geq e^T s(e)$, so that z_k is a valid lower bound on $v(P)$. Then if z_{k+1} is updated by (25), $c^T e - z_{k+1} \geq e^T s(e)$ also, so that z_{k+1} is a valid lower bound too.

Proof. There is nothing to show if $z_{k+1} = z_k$. Otherwise, $z_{k+1} = z_+$. Because of lemma 1, $(n + \sqrt{n})/e^T s(e) \geq \epsilon = (n + \sqrt{n})/(c^T e - z_+)$, which gives the desired inequality. Then z_{k+1} is a valid lower bound because it is at most $c^T e - e^T s(e)$, the value of the dual solution corresponding to $s(e)$.

Next we compute the search direction d as follows. First, let

$$\zeta := \begin{cases} q/(c^T e - z_{k+1}) & \text{if } z_{k+1} > -\infty, \\ 0 & \text{if } z_{k+1} = -\infty, \end{cases} \tag{26}$$

for $q = n + \sqrt{n}$. Note that, if $\|d_\alpha\| < .4$, then $\zeta \geq \epsilon > \alpha$ so that $\|d_\zeta\| \geq .4$ by definition of ϵ, while if $\|d_\alpha\| \geq .4$ then $\|d_\zeta\| \geq .4$ by (6).

Case A $\zeta < \alpha$. Then $\|d_\alpha\| \geq .4$. In this case, set

$$d = \frac{d_\alpha}{\|d_\alpha\|}. \tag{27}$$

Case B $\zeta \geq \alpha$. Let

$$\tilde{d} = \frac{d_\zeta}{||d_\zeta||} + \frac{-c_P}{||c_P||}, \tag{28}$$

$$d = \tilde{d}/||\tilde{d}||.$$

(Note that Gonzaga [12], Ye [31], and Freund [8], given a lower bound z_{k+1} and resulting ζ with $||d_\zeta||$ sufficiently large, would choose $d = d_\zeta$.)

Lemma 4. If d is defined as above, then

$$\nabla_x \phi_q(e, s(e))^T d \leq -\cdot 28$$

for any $q \geq 0$ if $z_{k+1} = -\infty$ and for any $q \geq n + \sqrt{n}$ if $z_{k+1} > -\infty$.

Proof. Define δ by $-P_A \nabla_x \phi_q(e, s(e)) = d_\delta$. Then $\nabla_x \phi_q(e, s(e))^T d = -d_\delta^T d$. In case A, $d = d_\alpha/||d_\alpha||$ and $-d_\delta^T d = -d_\delta^T d_\alpha/||d_\alpha|| = -d_\alpha^T d_\alpha/||d_\alpha|| = -||d_\alpha|| \leq -\cdot 4$ using (8).

In case B, lemma 3 and (26) show that $\delta \geq \zeta \geq \alpha$ as long as $z_{k+1} = -\infty$ ($\zeta = 0$, $\delta \geq 0$ for any $q \geq 0$) or $q \geq n + \sqrt{n}$ (using lemma 1). Hence $||d_\delta|| \geq ||d_\zeta|| \geq \cdot 4$. Now d is a unit vector that bisects the angle between the two extreme directions for d_δ, namely d_ζ and $-c_P$. Since $\zeta \geq \alpha$, these two directions form a non-obtuse angle, so the angle between d_δ and d is at most $\pi/4$. Then

$$-d_\delta^T d \leq -||d_\delta|| \cos(\pi/4) \leq -\cdot 4 \cos(\pi/4) \leq -\cdot 28.$$

Now we search on the half-line $\{e + \lambda d: \lambda \geq 0\}$. If $z_{k+1} = -\infty$, we seek to minimize

$$\phi_0^P(x, -\infty) := -\sum_j \ell n\, x_j, \tag{29}$$

the barrier function. It is easy to see that, in either case A or case B, ϕ_0^P can be decreased by at least $\cdot 03$. Moreover, since $c^T d \leq 0$, we obtain at least as great a decrease in the primal potential function

$$\phi_q^P(x, z) := q\, \ell n(c^T x - z) - \sum_j \ell n\, x_j \tag{30}$$

for any $q \geq n$ and $z \leq v(P)$, and hence in $\phi(x, s(e))$, since this differs by a constant from $\phi_q^P(x, z)$ with $q = n + \sqrt{n}$ and $z = c^T e - e^T s(e)$.

If $z_{k+1} > -\infty$, we seek to minimize

$$\phi_q^P(x, z_{k+1}) \tag{31}$$

for $q = n + \sqrt{n}$. Again, our updating scheme for z_k ensures that this can be decreased by at least $\cdot 03$, and that $\phi(x, s(e))$ can be reduced by at least as much.

Hence our first variant of the basic algorithm preserves (24), while employing directions probably closer to the negative of the projected scaled gradient of the appropriate potential function and allowing line searches to achieve greater reductions in such potential functions.

The following argument, which improves the author's original and is due to Y. Ye (private communication), shows that a finite lower bound must be generated in $O(\sqrt{n}t)$ iterations. Indeed, $\phi(x^0, s^0) = O(\sqrt{n}t)$ implies $\phi_n(x^0, s^0) = O(\sqrt{n}t)$. Until a lower bound is generated, ϕ_n decreases by a constant at each iteration, while it is bounded below by zero. This yields the desired complexity.

Once again we have a monotonic algorithm, so all iterates lie in the compact set $\{x \in F(P): c^T x \le c^T x^0\}$. Let $\bar{\xi} := \max\{e^T x/n: x \in F(P),\ c^T x \le c^T x^0\}$ and let $\underline{\xi}^0 := (\prod_{j=1}^n x_j^0)^{1/n}$, the geometric mean of the components of x^0. Our remarks above on decreasing ϕ_q^p ensure that,

$$\phi_q^P(x^k, z_k) \le \phi_q^P(x^0, z_k) - \cdot 03k \tag{32}$$

for $q = n + \sqrt{n}$, and using the estimates above and the fact that $n \le n + \sqrt{n} \le 2n$, we find

$$(c^T x^k - z_k) \le (\bar{\xi}/\underline{\xi}^0) \cdot \exp(-\cdot 01k/n) \cdot (c^T x^0 - z_k). \tag{33}$$

Assuming that $c^T x^0 - z_0$ and $\bar{\xi}/\underline{\xi}^0$ are bounded by 2^t, we can obtain from (33) an $O(nt)$ bound on the number of iterations to obtain $c^T x^k - z_k \le 2^{-t}$. This contrasts with the $O(\sqrt{n}t)$ bound in Theorem 1, which is still valid; one reason for the difference is that $v(P)$ is not known in Theorem 1, whereas here z_k is a known (and possibly not very tight) lower bound on $v(P)$. By contrast, the algorithms of Ye [31] and Freund [8, section 4] require only $O(\sqrt{n}t)$ iterations to obtain this inequality, where now z_k is the lower bound associated with an explicitly computed dual solution.

This first variant of the basic algorithm generates better directions than the original, but it still converges rather slowly in practice. The reason appears to be that the lower bounds generated are rather poor, so that in the line search to minimize $\phi_{n+\sqrt{n}}^P(x, z_{k+1})$, the barrier term $-\sum_j \ell n\, x_j$ forces a small step size, typically of the order of a tenth the maximum feasible step size.

In order to obtain better lower bounds, we reconsider the argument used in lemma 1. For any $\beta > 0$, if $\|-\beta c_p + e_p\| \leq 1$, we find that $s := (1/\beta)(e + \beta c_p - e_p) = c_p + (e - e_p)/\beta$ belongs to F(D), and the associated duality gap is $e^T c_p + e^T(e - e_p)/\beta = e^T c_p + \|e - e_p\|^2/\beta$, which is decreasing as a function of β. In fact, it is not even necessary that $\|-\beta c_p + e_p\| \leq 1$; as long as s is nonnegative, it provides such a bound. Hence we have

Lemma 5. If there is some $\beta > 0$ with $c_p + (e - e_p)/\beta \geq 0$, let $\bar{\beta}$ be the maximum such. Then $z_+ := c^T e - c_p{}^T e - \|e - e_p\|^2/\bar{\beta}$ is a lower bound on v(P).

This lower bound was also obtained independently by Gonzaga [14]. In fact, it is also implicit in the statement on performing a line search for Δ in Freund [8, section 3].

Thus our second variant sets z_+ as above if the criterion of the lemma is met, and then sets

$$z_{k+1} := \max\{z_k, z_+\}.$$

In this second variant, z_{k+1} is always a lower bound, but we may not have $c^T e - z_{k+1} \geq e^T s(e)$ as we did before. Given z_{k+1}, we define the search direction d as above (see (27)-(28)) and perform a line search on (29) or (31) as before. However, since we no longer have $z_{k+1} \leq c^T x^k - (x^k)^T s(x^k)$, we may not obtain a corresponding decrease in $\phi(x, s(e))$. Nevertheless, our line search can guarantee a decrease of at least ·03 in $\phi_0^P(x, -\infty)$ (and in $\phi_q^P(x, z)$ for any z and any q) if $z_{k+1} = -\infty$ and the same decrease in $\phi_q^P(x, z_{k+1})$ for $q = n + \sqrt{n}$ if $z_{k+1} > -\infty$. Thus (32) - (33) remain valid, and we deduce that the second variant provides an O(nt) iteration algorithm. This remains true if q in (26) and (31) is O(n), rather than $n + \sqrt{n}$.

5. Computational Results

We conclude the paper by giving the results of some very preliminary computational testing. These results suggest that the key refinement is the incorporation of a line search, followed by the use of impoved lower bounds.

The test problems were obtained as follows. For a given m and n, we generated each entry of A, y and s as an independent standard normal random variable, then set $b = Ae$ and $c = A^T y + |s|$, where $|s| = (|s_j|)$; the initial solution was $x^0 = e$. We describe the results of several variants. In all of them, we generated a sequence of lower bounds z_k, even if they were not used in the algorithm, in order to terminate when $(c^T x - z_k)/\max\{1, |c^T x|\}$ was less than 10^{-4}. Whenever a search direction d^k was obtained, we checked this termination criterion at the point $x = x^k + \lambda_{max} d_k$, where $\lambda_{max} = \max\{\lambda : x^k + \lambda d^k \geq 0\}$, so that the algorithm could terminate in a reasonable number of iterations even if it chose rather small step sizes.

The first variant differs from the basic algorithm of section 2 in three respects. It uses a sequence of lower bounds, generates improved directions, and performs a line search on a suitable potential function. The second variant adds to this an improved lower bound update. For one 50×100 problem, we tested all combinations of these features, as well as trying $q = n + \sqrt{n}$ and $q = 2n$ for the second variant. Thus we tried nine algorithms: the basic algorithm of section 2 (with termination based on lower bounds as in the first variant), then this basic method with a line search, the basic method with improved search directions, and the basic method with improved lower bounds (which only affect the termination criterion in this case); next, the basic algorithm with two of these features, namely, with improved directions and improved lower bounds, with a line search and improved lower bounds and with a line search and improved directions (the first variant); and finally, with all three features and $q = n + \sqrt{n}$ or $q = 2n$ (the second variant). The results are given in table 1, which also presents a typical value of λ_k/λ_{max}, where λ_k is the step size chosen and λ_{max} the maximum feasible step size. All runs used PRO-MATLAB [23] Version 3.5e on a Sun SPARCstation 1.

It is clear that the most significant enhancement is the incorporation of a line search; when this is present, improved lower bounds are more important than better directions because they allow longer step sizes to be used. Finally, all three features together allow a considerable decrease in the number of iterations required, especially for $q = 2n$. (If we also recur the improved lower bounds in the first variant, but only use them in the termination criterion (thus maintaining the theoretical $O(\sqrt{n}t)$ bound) then the number of iterations required decreases only slightly, to 77. Thus the difficulty is slow primal convergence, not a poor termination criterion.) Similar results hold for other test problems solved.

We also solved ten random 50×100 problems using the second variant. With $q = n + \sqrt{n}$, the average number of iterations was $12{\cdot}2$, with λ_k/λ_{max} typically $\cdot 87$; with $q = 2n$, the figures become $11{\cdot}0$ and $\cdot 97$. Five random 100×200 problems needed an average of $14{\cdot}0$ iterations with $q = n + \sqrt{n}$ and $12{\cdot}2$ with $q = 2n$. For 150×300 problems the figures were $14{\cdot}4$ and $13{\cdot}0$ respectively, and for 200x400 problems $15{\cdot}4$ and $13{\cdot}6$. The typical step size ratio was similar to those reported above.

Finally, a single 100×200 problem, which required 16 or 14 iterations for the second variant, needed 161 for the first variant (144 if termination was based on the improved lower bounds) and 626 for the basic algorithm and a 200×400 problem also needing 16 or 14 iterations for the second variant required 240 (or 202) for the first variant and 739 for the basic algorithm.

Acknowledgment. I am very grateful to the referees for their helpful comments, which considerably improved the presentation.

Method	Number of Iterations	Typical λ_k/λ_{max}
Basic	347	·04
Basic with line search	98	·13
Basic with improved directions	347	·04
Basic with improved bounds	282	·04
Basic with improved directions and improved bounds	262	·04
Basic with line search and improved bounds	27	·46
Basic with line search and improved directions (first variant)	95	·14
Second variant, $q=n+\sqrt{n}$	12	·87
Second variant, $q=2n$	11	·98

Table 1. Computational Results on a 50×100 problem

References

1. Anstreicher, K.M., (1986), "A monotonic projective algorithm for fractional linear programming," Algorithmica 1, 483-498.

2. Anstreicher, K.M., (1989), "A combined phase I - phase II scaled potential algorithm for linear programming," CORE Discussion Paper 8939, Catholic University of Louvain, Belgium.

3. Anstreicher, K.M., (1990), "On monotonicity in the scaled potential algorithm for linear programming," report, CORE, Catholic University of Louvain, Belguim.

4. Barnes, E.R., (1986), "A variation on Karmarkar's algorithm for solving linear programming problems," Mathematical Programming 36, 174-182.

5. Barnes, E.R, Chopra, S., and Jensen, D.L., (1988), "A polynomial time version of the affine scaling algorithm," IBM T.J. Watson Research Center, Yorktown Heights, NY.

6. Bayer, D. and Lagarias, J.C., (1989), "The nonlinear geometry of linear programming: I. Affine and projective scaling trajectories" and "II. Legendre transform coordinates and central trajectories," Transactions of the American Mathematical Society 314, 499-526 and 527-581.

7. Dikin, I.I. 1967), "Iterative solution of problems of linear and quadratic programming," Doklady Akademiia Nauk SSSR 174, 747-748 [English translation: Soviet Mathematics Doklady 8, 674, 675].

8. Freund, R. , (1988), "Polynomial-time algorithms for linear programming based only on primal scaling and projected gradients of a potential function," manuscript, Sloan School of Management, M.I.T., Cambridge, Massacusetts.

9. de Ghellinck, G. and Vial, J.-Ph., (1986), "A polynomial Newton method for linear programming," Algorithmica 1, 425-453.

10. Gonzaga, C., (1987), "Search directions for interior linear programming methods," Memorandum UCB/ERL M87/44, Electronics Laboratory, College of Engineering, University of California, Berkeley, California.

11. Gonzaga, C., (1988), "An algorithm for solving linear programming in $O(n^3L)$ operations," in: Progress in Mathematical Programming (N. Megiddo, ed.), Springer-Verlag, Berlin, 1-28.

12. Gonzaga, C., (1990), "Polynomial affine algorithms for linear programming," Mathematical Programming 49, 7-21.

13. Gonzaga, C., (1989), "Large-steps path following methods for linear programming: potential reduction method," Report ES-211/89, COPPE, Federal University of Rio de Janeiro, Brazil.

14. Gonzaga, C., (1990), "On lower bound updates in primal potential reduction methods for linear programming," Report ES-227/90, COPPE, Federal University of Rio de Janeiro, Brazil.

15. den Hertog, D. and Roos, C., (1989), "A survey of search directions in interior-point methods for linear programming," Report 89-65, Faculty of Technical Mathematics and Informatics, Delft University of Technology, The Netherlands.

16. Karmarkar, N., (1984), "A new polynomial time algorithm for linear programming," Combinatorica 4, 373-395.

17. Kojima, M., Mizuno, S. and Yoshise, A., (1988), "A primal-dual interior point method for linear programming," in : Progress in Mathematical Programming (N. Megiddo, ed.), Springer Verlag, Berlin, 29-47.

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

19. Kojima, M., Mizuno, S. and Yoshise, A., (1988), "An $O(\sqrt{n}\ L)$ iteration potential reduction algorithm for linear complementarity problems," Research Report 8-217, Department of Information Sciences, Tokyo Institute of Technology, Tokyo, Japan.

20. Megiddo, N., (1988), "Pathways to the optimal set in linear programming," in: Progress in Mathematical Programming (N. Megiddo, ed.), Springer Verlag, Berlin, 131-158.

21. Megiddo, N. and Shub, M., (1989), "Boundary behavior of interior point algorithms in linear programming," Mathematics of Operations Research 14, 97-146.

22. Mitchell, J.E., and Todd, M.J., (1989), "On the relationship between the search directions in the affine and projective variants of Karmarkar's linear programming algorithm," in: Contributions to Operations Research and Economics (B. Cornet and H. Tulkens, eds.), M.I.T. Press, 237-250.

23. Moler, C.B., Little, J., Bangert, S., and Kleiman, S., (1987), "Pro-Matlab User's Guide," MathWorks, Sherborn, MA.

24. Monteiro, R.C., and Adler, I., (1989), "Interior path following primal-dual algorithms, Part I: Linear programming," Mathematical Programming 44, 27-41.

25. Monterio, R.C. and Adler, I., (1989), "Interior path following primal-dual algorithms, Part II: Convex quadratic programming," Mathematical Programming 44, 43-66.

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

27. Sonnevend, G., (1986), "An analytical centre for polyhedrons and new classes of golbal algorithms for linear (smooth, convex) programming," Lecture Notes in Control and Information Sciences 84, Springer Verlag, New York, 866-876.

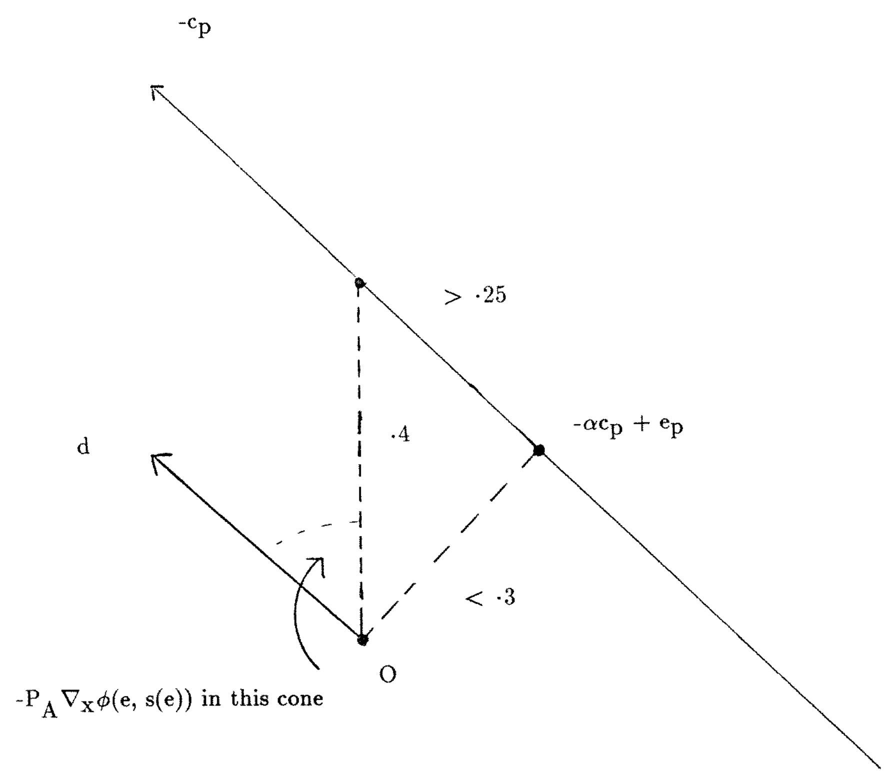
28. Todd, M.J. and Burrell, B.P., (1986), "An extension of Karmarkar's algorithm for linear programming using dual variables," Algorithmica 1, 409-424.

29. Vanderbei, R.J., Meketon, M.S., and Freedman, B.A., (1986), "A modification of Karmarkar's linear programming algorithm," Algorithmica 1, 395-407.

30. Yamashita, H., (1986), "A polynomially and quadratically convergent method for linear programming," report, Mathematical Systems Institute, Tokyo, Japan.

31. Ye, Y., (1989), "An $O(n^3\ L)$ potential reduction algorithm for linear programming," Technical Report, Department of Management Sciences, The University of Iowa (Iowa City, IA), to appear in Mathematical Programming.

32. Zimmerman, U., (1988), "Search directions for projective methods," Institut fuer Angewandte Mathematik, Technische Universitaet Carolo-Wilhelmina, Braunschweig, W. Germany.

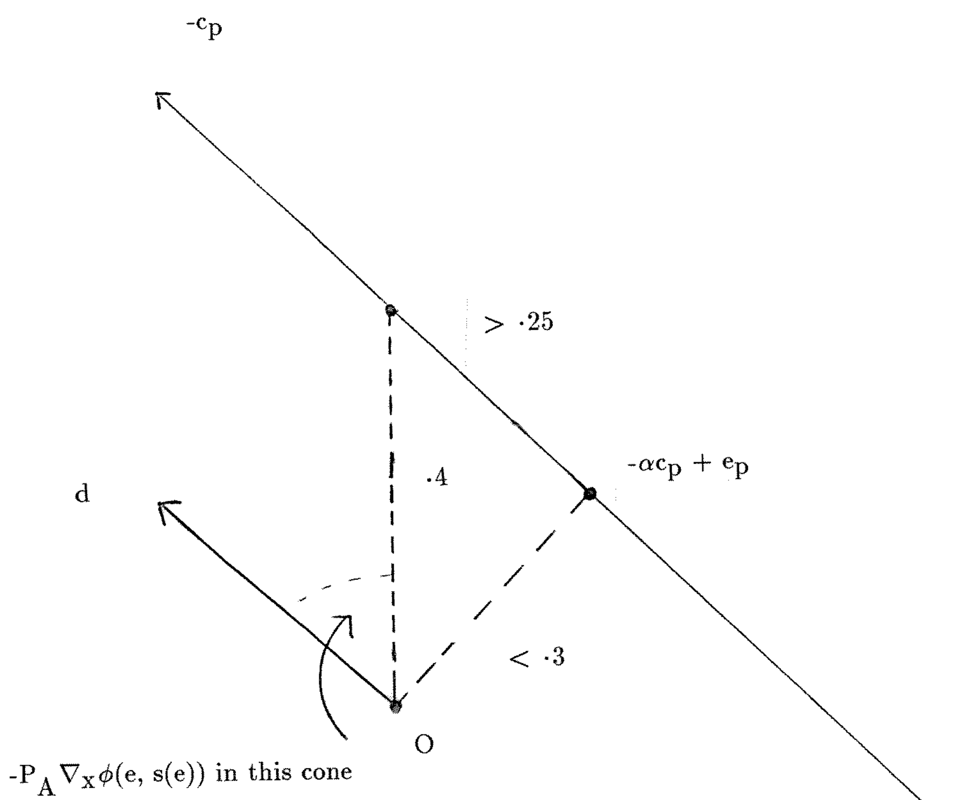


4. Refinements

The simple algorithm of section 2 does not perform well in practice. Indeed, (12) shows that each component of x^k can decrease by at most 20% in each iteration, so that the best we can hope for is linear convergence with ratio $\cdot 8$. In this section we describe refinements that attempt to improve the convergence while maintaining the simple structure of the algorithm. Usually, an interior-point method can be improved significantly by incorporating a line search. Here, however, the potential function on which we would like to perform a line search cannot easily be computed, and hence complications arise. We use the insights obtained from the analysis of section 3 and update lower bounds on the optimal value. The resulting methods bear a strong resemblance to those of Gonzaga [12], Ye [31] and Freund [8].

Our first variant maintains inequality (24), and hence provides an $O(\sqrt{n}t)$ iteration method ($O(\sqrt{n}L)$ for a problem with integer data), while our second variant is preferable in practice but has only an $O(nt)$ iteration bound.

At each iteration, we try to update the current lower bound z_k, then choose a search direction and make a line search in this direction to minimize approximately some potential function. In the second variant we only need $z_k \leq v(P)$. In the first, we require further that $c^T x^k - z_k \geq (x^k)^T s(x^k)$,



-cp
> ·25
·4
-αcp + ep
d
< ·3
O
-PA ∇x φ(e, s(e)) in this cone