April 1, 1990 LIDS-P-1963

A Path-Following Algorithm for Linear Programming using Quadratic and Logarithmic Penalty Functions[1]

by

Paul Tseng[2]

Abstract

We propose a path-following algorithm for linear programming using both logarithmic and quadratic penalty functions. In the algorithm, we place a logarithmic and a quadratic penalty on, respectively, the non-negativity constraints and an arbitrary subset of the equality constraints; we apply Newton's method to solve the penalized problem, and after each Newton step we decrease the penalty parameters. This algorithm maintains neither primal nor dual feasibility and does not require a Phase I. We show that if the initial iterate is chosen appropriately and the penalty parameters are decreased to zero in a particular way, then the algorithm is linearly convergent. We also present numerical results showing that the algorithm can be competitive with interior point algorithms in practice, requiring between 30 to 45 iterations to accurately solve each Netlib problem tested.

Key Words: linear program, path-following, Newton step, penalty function.

---

[1] This research is partially supported by the U.S. Army Research Office, contract DAAL03-86-K-0171 (Center for Intelligent Control Systems).

[2] Laboratory for Information and Decision Systems and Center for Intelligent Control Systems, Room 35-205, Massachusetts Institute of Technology, Cambridge, MA 02139.

Acknowledgement: We thank Dr. R. Setiono for providing us with a copy of the YSMP codes and for helpful discussions regarding implementation.

1. Introduction

Since the pioneering work of Karmarkar [Kar84], much interest has focussed on solving linear programs using interior point algorithms. These interior point algorithms may be classified roughly as either (i) projective-scaling (or potential reduction), or (ii) affine-scaling, or (iii) path-following. We will not attempt to review the literature on this subject, which is vast (see for example [Meg89], [Tod88] for surveys). Our interest is in algorithms of the path-following type, of the sort discussed in [GaZ81]. These interior point algorithms typically penalize the non-negativity constraints by a logarithmic function and use Newton's method to solve the penalized problem, with the penalty parameters decreased after each Newton step (see, e.g., [Gon89], [KMY89], [MoA87], [Ren88], [Tse89a]).

One disadvantage of interior point algorithms is the need of an initial interior feasible solution. A common technique for handling this is to add an artifical column (see [AKRV89], [BDDW89], [GMSTW86], [MMS88], [MSSPB88], [MoM87]), but this itself has disadvantages. For example, the cost of the artificial column must be estimated, and some type of rank-1 updating is needed to solve each least square problem which can significantly increase the solution time and degrade the numerical accuracy of the solutions.

Recently, Setiono [Set89] proposed an interesting algorithm that combines features of a path-following algorithm with those of the method of multipliers [HaB70], [Hes69], [Pow69] (also see [Roc76], [Ber82]). This algorithm does not require a feasible solution to start, and is comparable to interior point algorithms both in terms of work per iteration and, according to the numerical results reported in [Set89], in terms of the total number of iterations. To describe the basic idea in Setiono's algorithm, consider a linear program in the standard dual form

$$\begin{array}{ll} \text{minimize} & -b^T p \\ \text{subject to} & t + A^T p = c, \ t \geq 0, \end{array} \tag{1.1}$$

where A is some matrix and b, c are vectors of appropriate dimension. Let us attach a Lagrange multiplier vector x to the constraints $t + A^T p = c$ and apply the method of multipliers to the above linear program. This produces the following iterations

$$x^{k+1} = x^k + \frac{1}{\varepsilon^k}(t^k + A^T p^k - c), \qquad k = 1, 2, ..., \tag{1.2}$$

where $\{\varepsilon^k\}$ is a sequence of monotonically decreasing positive scalars and (t^k,p^k) is some (inexact) solution of the augmented Lagrangian subproblem

$$\begin{array}{ll} \text{minimize} & -b^Tp + (x^k)(t + A^Tp - c) + \frac{1}{2\varepsilon^k}\|t + A^Tp - c\|^2 \\ \text{subject to} & t \geq 0. \end{array} \tag{1.3}$$

[An advantage of the above multiplier iterations is that they do not need a feasible solution to start.] A key issue associated with the above multiplier iterations concerns the efficient generation of an inexact solution (t^k,p^k) of the convex quadratic program (1.3) for each k. [Notice that as ε^k decreases, the objective function of (1.3) becomes progressively more ill-conditioned.] Setiono's algorithm may be viewed as the method of multipliers in which (t^k,p^k) is generated according to the following scheme, reminiscent of the path-following idea: Add a logarithmic penalty function $-\gamma^k\sum_{j=1}^{m} \ln(t_j)$ to the objective of (1.3), where γ^k is some positive scalar monotonically decreasing with k, and apply a single Newton step, starting from (t^{k-1},p^{k-1}), to the resulting problem. [If the t^k thus obtained lies outside the positive orthant, it is moved back towards t^{k-1} until it becomes positive.[1]]

In this paper, inspired by the work of Setiono, we study an algorithm that also adds to the objective a quadratic penalty on the equality constraints and a logarithmic penalty on the nonnegativity constraints; and then solves the penalized problem using Newton's method, with the penalty parameters decreased after each Newton step. Unlike Setiono's algorithm, our algorithm does not use the multiplier vector x^k (so it may be viewed as a pure penalty method) and allows <u>any</u> subset of the equality constraints to be penalized. We show that if the problem is primal non-degenerate and the iterates start near the optimal solution of the initial penalized problem, then the penalty parameters can be decreased at the rate of a geometric progression and the iterates converge linearly. To the best of our knowledge, this is the first linear convergence result for an non–interior point path-following algorithm. We also present numerical results indicating that the algorithm can be competitive with interior point algorithms in practice. We remark that penalty methods which use

---

[1]More precisely, t^k is given by the formula

$$t^k = t^{k-1} + .98\lambda^k\Delta t^k,$$

where Δt^k is the Newton direction (projected onto the space of t) and λ^k is the largest $\lambda \in (0,1]$ for which $t^{k-1} + \lambda\Delta t^k$ is nonnegative. [The choice of .98 is arbitrary – any number between 0 and 1 would do.]

either the quadratic or the logarithmic penalty function have been well studied (see, e.g., [Ber82], [FiM68], [Fri57], [JiO78], [Man84], [WBD88]), but very little is known about penalty methods which use both types of penalty functions (called mixed interior point–exterior point algorithms in [FiM68]).

This paper proceeds as follows: In §2 we describe the basic algorithm; in §3 we analyze its convergence; and in §4 we recount our numerical experience with it. In §5 we discuss possible extensions of this work.

In our notation, every vector is a column vector in some k-dimensional real space $\Re^k$, and superscript T denotes transpose. For any vector x, we denote by x_j the j-th coordinate of x, by Diag(x) the diagonal matrix whose j-th diagonal entry is x_j, and by $\|x\|_1$, $\|x\|$, $\|x\|_\infty$ the L_1-norm, the L_2-norm and the L_∞-norm of x, respectively. For any matrix A, we denote by A_j the j-th column of A. We also denote by e the vector of 1's (whose dimension will be clear from the context) and denote by $\ln(\cdot)$ the natural logarithm function.

2. Algorithm Description

Let A be an n×m matrix, B be an l×m matrix, b be an n-vector, c be an m-vector, and d be an l-vector. Consider the following linear program associated with A, B, b, c and d:

$$\begin{array}{ll} \text{minimize} & -b^T p \\ \text{subject to} & t + A^T p = c, \;\; Bt = d, \;\; t \geq 0, \end{array} \qquad \text{(D)}$$

which we call the <u>dual</u> problem. The dual problem may be viewed as a standard linear program in t, in which we arbitrarily partition the equality constraints into two subsets and write one of which in the generator form $t + A^T p = c$. [There is no loss of generality in assuming that t is absent from the objective since t can always be replaced in the objective by $c - A^T p$.] The constraints $t + A^T p = c$ can be thought of as the <u>complicating</u> constraints which, if removed, would make (D) much easier to solve. [The form in which we write the equality constraints is not very important, except that one form may be easier to work with than another.]

By attaching Lagrange multiplier vectors x and y to the constraints $c - A^T p = t$ and $Bt = d$ respectively, we obtain the following dual of (D):

$$\begin{array}{ll} \text{minimize} & c^T x + d^T y \\ \text{subject to} & Ax = b, \;\; x + B^T y \geq 0, \end{array} \qquad \text{(P)}$$

which we call the <u>primal</u> problem.

We make the following blanket assumptions, which are standard for interior point algorithms, regarding (P) and (D):

Assumption A:
(a) $\{ x \mid Ax = b, x + B^T y > 0 \text{ for some } y \}$ is nonempty and bounded.
(b) $\{ t \mid Bt = d, t > 0, t + A^T p = c \text{ for some } p \}$ is nonempty and bounded.
(c) A has full row rank.

Consider the dual problem (D). Suppose that we place a quadratic penalty on the constraints $t + A^T p = c$ with a penalty parameter $1/\varepsilon$ ($\varepsilon > 0$) and we place a logarithmic penalty on the constraints $t \geq 0$ with a penalty parameter $\gamma > 0$. This gives the following approximation to (D):

$$\begin{array}{lll} \text{minimize} & f_{\varepsilon,\gamma}(t,p), & (D_{\varepsilon,\gamma}) \\ \text{subject to} & Bt = d,\ t > 0, & \end{array}$$

where $f_{\varepsilon,\gamma}:(0,\infty)^m\times\Re^n$ is the penalized objective function given by

$$f_{\varepsilon,\gamma}(t,p) = \|c - t - A^Tp\|^2/2 - \varepsilon\gamma\sum_{j=1}^{m} \ln(t_j) - \varepsilon b^Tp, \quad \forall\, t > 0, \forall\, p. \tag{2.1}$$

The penalized problem $(D_{\varepsilon,\gamma})$ has the advantage that its objective function $f_{\varepsilon,\gamma}$ is twice differentiable and the Hessian $\nabla^2 f_{\varepsilon,\gamma}$ is positive definite (cf. the full row rank assumption on A). Since there can not exist u with $A^Tu \geq 0$, $A^Tu \neq 0$, $BA^Tu = 0$ [cf. Assumption A (b)], it can be seen that the intersection of any level set of $f_{\varepsilon,\gamma}$ with the constraint set of $(D_{\varepsilon,\gamma})$ is bounded. This, together with the fact that $f_{\varepsilon,\gamma}$ tends to ∞ at the boundary of its domain, implies that such an intersection is compact, so $(D_{\varepsilon,\gamma})$ has an optimal solution. By the strict convexity of $f_{\varepsilon,\gamma}$, this optimal solution is unique.

Note 1: We can use penalty functions other than the quadratic and the logarithmic. For example, we can use a cubic in place of the quadratic and $-t_j\ln(t_j)$ in place of $\ln(t_j)$. The quadratic and the logarithmic function, however, have nice properties (such as the second derivative of the logarithmic function is minus the square of its first derivative) which make global convergence analysis possible.

It is well-known that (t,p) is the optimal solution of $(D_{\varepsilon,\gamma})$ if and only if it satisfies, together with some $u\in\Re^l$, the Kuhn-Tucker conditions (see [Roc70])

$$t > 0, \qquad Bt = d, \qquad \nabla f_{\varepsilon,\gamma}(t,p) + \begin{bmatrix} B^Tu \\ 0 \end{bmatrix} = 0. \tag{2.2}$$

Straightforward calculation using (2.1) finds that

$$\nabla f_{\varepsilon,\gamma}(t,p) = \begin{bmatrix} t + A^Tp - c - \varepsilon\gamma(T)^{-1}e \\ \\ A(t + A^Tp - c) - \varepsilon b \end{bmatrix}, \tag{2.3}$$

and

$$\nabla^2 f_{\varepsilon,\gamma}(t,p) \quad = \quad \begin{bmatrix} I + \varepsilon\gamma(T)^{-2} & A^T \\ & \\ A & AA^T \end{bmatrix}, \qquad (2.4)$$

where T = Diag(t). The above formulas will be used extensively in the subsequent analysis. Notice that $\nabla^2 f_{\varepsilon,\gamma}$ is ill-conditioned at the boundary of its domain.

It is not difficult to show that, as ε and γ tend to zero, the optimal solution of $(D_{\varepsilon,\gamma})$ approaches the optimal solution set of (D) (see Lemma 1). This suggests the following algorithm for solving $(D_{\varepsilon,\gamma})$. At each iteration, we are given ε, γ and a (t,p) which is an approximate solution of $(D_{\varepsilon,\gamma})$; we apply a Newton step to $(D_{\varepsilon,\gamma})$ at (t,p) to generate a new (t,p) and then we decrease γ, ε. In other words, we consider a sequence of penalized problems $\{(D_{\varepsilon^k,\gamma^k})\}_{k=1}^{\infty}$ with $\varepsilon^k \downarrow 0$ and $\gamma^k \downarrow 0$, and use a Newton step to follow the optimal solution of one penalized problem to that of the next. We now formally state this algorithm, which we call the QLPPF (short for Quadratic–Logarithmic Penalty Path–Following) algorithm:

QLPPF Algorithm

Iter. 0 Choose $\varepsilon^1 > 0$ and $\gamma^1 > 0$. Choose $(t^1,p^1) \in (0,\infty)^m \times \Re^n$ with $Bt^1 = d$.

Iter. k Given $(t^k,p^k) \in (0,\infty)^m \times \Re^n$ with $Bt^k = d$, compute $(\Delta t^k, \Delta p^k, u^k)$ to be a solution of

$$\nabla^2 f_{\varepsilon^k,\gamma^k}(t^k,p^k) \begin{bmatrix} \Delta t^k \\ \Delta p^k \end{bmatrix} + \nabla f_{\varepsilon^k,\gamma^k}(t^k,p^k) + \begin{bmatrix} B^T u^k \\ 0 \end{bmatrix} = 0, \quad B\Delta t^k = 0, \qquad (2.5)$$

and set

$$t^{k+1} = t^k + \Delta t^k, \qquad p^{k+1} = p^k + \Delta p^k, \qquad (2.6)$$

$$\gamma^{k+1} = \alpha^k \gamma^k, \qquad \varepsilon^{k+1} = \alpha^k \varepsilon^k, \qquad (2.7)$$

where α^k is some scalar in (0,1).

Note 2: It can be seen from (2.3)-(2.4) that, in the special case where B is the zero matrix, the direction finding problem (2.5) differs from that in Setiono's algorithm [Set89, Eq. (6)] by only an order ε^k term in the right hand side (which tends to zero as ε^k tends to zero).

3. Global Convergence

In this section, we show that if (D) is in some sense primal non-degenerate and if (t^1,p^1) is "close" to the optimal solution of $(D_{\varepsilon^1,\gamma^1})$ in the QLPPF algorithm, then, by decreasing the α^k's at an appropriate rate, the iterates $\{(t^k,p^k)\}$ generated by the QLPPF algorithm approach the optimal solution set of (D) (see Theorem 1). Because the Hessian $\nabla^2 f_{\varepsilon,\gamma}$ is ill-conditioned at the boundary of its domain, the proof of this result is quite involved and relies critically on finding a suitable Lyapunov function to monitor the progress of the algorithm.

For any $\varepsilon > 0$ and $\lambda > 0$, let $\rho_{\varepsilon,\gamma}:(0,\infty)^m\times\Re^n\times\Re^l$ be the function given by

$$\rho_{\varepsilon,\gamma}(t,p,u) = \max\{ \|T(c-t-A^Tp-B^Tu)+\varepsilon\gamma e\|/(\varepsilon\gamma), \|A(c-t-A^Tp)+\varepsilon b\|/\sqrt{\varepsilon\gamma} \}, \qquad \forall\, t>0, \forall\, p, \forall\, u, \tag{3.1}$$

where $T = \text{Diag}(t)$. From (2.2)-(2.3) we see that (t,p) is a solution of $(D_{\varepsilon,\gamma})$ if and only if $t > 0$, $Bt = d$ and, for some u, $\rho_{\varepsilon,\gamma}(t,p,u) = 0$. Hence $\rho_{\varepsilon,\gamma}$ acts as a Lyapunov function which measures how far (t,p) is from solving $(D_{\varepsilon,\gamma})$. This notion is made precise in the following lemma.

For any $\varepsilon > 0$, let

$$\mathcal{Y}_\varepsilon = \{ (t,p) \mid A(c-t-A^Tp) = -\varepsilon b,\ Bt = d,\ t > 0 \}. \tag{3.2}$$

Lemma 1. Fix $\varepsilon > 0$, $\gamma > 0$ and $\beta\in(0,1]$. For any $(t,p)\in\mathcal{Y}_\varepsilon$ and any u with $\rho_{\varepsilon,\gamma}(t,p,u) \le \beta$, the following hold:

(a) (x,y), where $x = (t+A^Tp-c)/\varepsilon$ and $y = u/\varepsilon$, is feasible for (P).

(b) (x,y) and (t,p) are optimal primal and dual solution pairs of a linear program which is obtained from (D) by perturbing each right hand coefficient by at most $\max\{\varepsilon M_0, (1+\beta)\sqrt{\gamma}\}$ and each cost coefficient by at most $\sqrt{\gamma}$, where $M_0 = \max\{ \|x\|_\infty \mid (x,y) \text{ is feasible for (P) for some } y \}$.

Proof: (a) Since (t,p) is in $\mathcal{Y}_\varepsilon$, it follows from the definition of $\mathcal{Y}_\varepsilon$ [cf. (3.2)] that $Ax = b$. Since $\rho_{\varepsilon,\gamma}(t,p,u) \le \beta$, we have from the definition of $\rho_{\varepsilon,\gamma}$ [cf. (3.1)] and x that $\|T(-\varepsilon x - B^Tu) + \varepsilon\gamma e\| \le \varepsilon\gamma\beta$, where $T = \text{Diag}(t)$, implying

$$\gamma(1-\beta)e \le T(x+B^Tu/\varepsilon) \le \gamma(1+\beta)e. \tag{3.3}$$

Since $t > 0$ and $\beta \le 1$, the first inequality in (3.3) yields $x + B^T u/\varepsilon \ge \gamma(1-\beta)T^{-1}e \ge 0$. Hence, $(x,u/\varepsilon)$ is feasible for (P).

(b) From $y = u/\varepsilon$ and the second inequality in (3.3) we have $T(x + B^T y) \le (1 + \beta)\gamma e$, so

$$x_j + B_j^T y > \sqrt{\gamma} \qquad \Rightarrow \qquad t_j \le (1+\beta)\sqrt{\gamma}. \tag{3.4}$$

Since (x,y) is feasible for (P), then $\|x\|_\infty \le M_0$ and thus $\|t + A^T p - c\|_\infty \le \varepsilon M_0$. This together with (3.4) and the facts $t > 0$, $Bt = d$, $x + B^T y \ge 0$ shows that (x,y) and (t,p) satisfy the optimality conditions for a perturbed linear program (which is obtained from (D) by perturbing each component of c by at most εM_0, each right hand side coefficient of the constraints $t \ge 0$ by at most $(1+\beta)\sqrt{\gamma}$, and each cost coefficient on t by at most $\sqrt{\gamma}$). Q.E.D.

Since we are dealing with linear programs, part (b) of Lemma 1 implies that, as $\varepsilon \to 0$ and $\gamma \to 0$, the (x,y) of part (a) approaches the optimal solution set of (P) and (t,p) approaches the optimal solution set of (D). In fact, it suffices to decrease ε and γ as far as $2^{-\kappa L}$, where κ is some scalar constant and L is the size of the problem encoding in binary (defined as, say, in [Kar84]), at which time an optimal solution of (P) and of (D) can be recovered by using the techniques described in, for example, [Kar84], [PaS82].

For each $\lambda > 0$, let $\theta_\lambda : (0,\infty)^m \to [0,\infty)$ be the function given by

$$\theta_\lambda(t) = (\|ED^{1/2}A^T FAD^{1/2}E + E\| + \|ED^{1/2}A^T F\|)^2, \qquad \forall\, t > 0, \tag{3.5a}$$

where

$$D = (I + \lambda T^{-2})^{-1}, \tag{3.5b}$$
$$E = I - D^{1/2}B^T[BDB^T]^{-1}BD^{1/2}, \tag{3.5c}$$
$$F = [A(I - D^{1/2}ED^{1/2})A^T]^{-1}, \tag{3.5d}$$

and $T = \mathrm{Diag}(t)$. [F is well-defined because $\|E\| \le 1$ (E is a projection matrix) and $\|D\| < 1$, so that $I - D^{1/2}ED^{1/2}$ is positive definite. We also use the assumption that A has full row rank.] The quantity $\theta_\lambda(t)$ estimates the norm squared of certain projection-like operator depending on λ and t, and it will be used extensively in our analysis. In general, $\theta_\lambda(t)$ is rather cumbersome to evaluate, but, as we shall see, it suffices for our analysis to upper bound $\theta_\lambda(t)$ [see Lemma 3 (b)].

3.1. Analyzing a Newton Step

In this subsection we prove a key lemma which states that if $(\bar{t},\bar{p})$ is "close" to the optimal solution of $(D_{\bar{\varepsilon},\bar{\gamma}})$, then (t,p) generated by applying one Newton step to (D) at $(\bar{t},\bar{p})$ is close to the optimal solution of $(D_{\varepsilon,\gamma})$ for some $\varepsilon < \bar{\varepsilon}$ and some $\gamma < \bar{\gamma}$. The notion of "closeness" is measured by the Lyapunov function $\rho_{\varepsilon,\gamma}$ and the proof of the lemma is based on the ideas used in [Tse89a, §2].

Lemma 2. For any $\bar{\varepsilon} > 0$, any $\bar{\gamma} > 0$ and any $(\bar{t},\bar{p},\bar{u}) \in (0,\infty)^m \times \Re^n \times \Re^1$ with $B\bar{t} = d$, let (t,p,u) be given by

$$t = \bar{t} + \Delta t, \tag{3.6a}$$

$$p = \bar{p} + \Delta p, \tag{3.6b}$$

where u and $(\Delta t, \Delta p)$ together solve the following system of linear equations

$$\nabla^2 f_{\bar{\varepsilon},\bar{\gamma}}(\bar{t},\bar{p}) \begin{bmatrix} \Delta t \\ \Delta p \end{bmatrix} + \nabla f_{\bar{\varepsilon},\bar{\gamma}}(\bar{t},\bar{p}) + \begin{bmatrix} B^T u \\ 0 \end{bmatrix} = 0, \qquad B\Delta t = 0. \tag{3.6c}$$

Suppose that $\rho_{\bar{\varepsilon},\bar{\gamma}}(\bar{t},\bar{p},\bar{u}) \le \beta$ for some $\beta < \min\{\ 1,\ 1/\theta_{\bar{\varepsilon}\bar{\gamma}}(\bar{t})\}$. Then the following hold:

(a) $(t,p) \in \mathcal{Y}_{\bar{\varepsilon}}$.

(b) For any α satisfying

$$\max\{\ \sqrt{(\theta_{\bar{\varepsilon}\bar{\gamma}}(\bar{t})\beta^2 + \sqrt{m})/(\beta + \sqrt{m})}\ ,\ 1/(1 + \beta\sqrt{\bar{\gamma}/\bar{\varepsilon}/\|b\|})\}\ \le\ \alpha\ \le 1, \tag{3.7}$$

there holds $\rho_{\alpha\bar{\varepsilon},\alpha\bar{\gamma}}(t,p,u) \le \beta$.

Proof: Let

$$\bar{r} = \bar{T}(c - \bar{t} - A^T\bar{p} - B^T\bar{u}) + \bar{\varepsilon}\bar{\gamma}e, \tag{3.8}$$

and

$$\bar{s} = A(c - \bar{t} - A^T\bar{p}) + \bar{\varepsilon}b, \tag{3.9}$$

where $\bar{T} = \mathrm{Diag}(\bar{t})$. Then [cf. (3.1)]

$$\max\{\|\bar{r}\|/(\bar{\varepsilon}\bar{\gamma}),\ \|\bar{s}\|/\sqrt{\bar{\varepsilon}\bar{\gamma}}\} \quad = \rho_{\bar{\varepsilon},\bar{\gamma}}(\bar{t},\bar{p},\bar{u}) \le \beta. \tag{3.10}$$

By using (2.3)-(2.4) and (3.8)-(3.9), we write (3.6c) equivalently as

$$\begin{aligned} D^{-1}\Delta t + A^T\Delta p + B^T(u-\bar{u}) &= \bar{T}^{-1}\bar{r}, \\ A\Delta t + AA^T\Delta p &= \bar{s}, \\ B\Delta t &= 0, \end{aligned}$$

where for convenience we let $D = (I + \bar{\varepsilon}\bar{\gamma}\bar{T}^{-2})^{-1}$. Solving for Δt gives

$$\Delta t \quad = D^{1/2}(ED^{1/2}A^TFAD^{1/2}E + E)D^{1/2}\bar{T}^{-1}\bar{r} - D^{1/2}ED^{1/2}A^TF\bar{s},$$

where we let $E = I - D^{1/2}B^T[BDB^T]^{-1}BD^{1/2}$ and $F = [A(I - D^{1/2}ED^{1/2})A^T]^{-1}$. [F is well-defined by the same reasoning that the matrix given by (3.5d) is well-defined.] Then, we can bound $\bar{T}^{-1}\Delta t$ as follows:

$$\|\bar{T}^{-1}\Delta t\| \quad \le \|\bar{T}^{-1}D^{1/2}\|^2\|ED^{1/2}A^TFAD^{1/2}E + E\|\ \|\bar{r}\| + \|\bar{T}^{-1}D^{1/2}\|\ \|ED^{1/2}A^TF\|\ \|\bar{s}\|.$$

Since $\bar{T}^{-2}D$ is diagonal and each of its diagonal entry is less than $1/(\bar{\varepsilon}\bar{\gamma})$, we obtain that $\|\bar{T}^{-2}D\| < 1/(\bar{\varepsilon}\bar{\gamma})$ and hence

$$\begin{aligned} \|\bar{T}^{-1}\Delta t\| &\le \|ED^{1/2}A^TFAD^{1/2}E + E\|\ \|\bar{r}\|/(\bar{\varepsilon}\bar{\gamma}) + \|ED^{1/2}A^TF\|\ \|\bar{s}\|/\sqrt{\bar{\varepsilon}\bar{\gamma}} \\ &\le \sqrt{\theta_{\bar{\varepsilon}\bar{\gamma}}(\bar{t})}\beta, \end{aligned} \tag{3.11}$$

where the last inequality follows from (3.10) and the definition of $\bar{T}$, D, E, F and $\theta_{\bar{\varepsilon}\bar{\gamma}}(\bar{t})$ [cf. (3.5a)-(3.5d)].

Now, by using (2.3)-(2.4), we can write (3.6c) equivalently as $B\Delta t = 0$ and

$$\bar{\varepsilon}\bar{\gamma}\bar{T}^{-1}\Delta t \quad = \bar{T}(c - \bar{t} - A^T\bar{p} - \Delta t - A^T\Delta p - B^Tu) + \bar{\varepsilon}\bar{\gamma}e, \tag{3.12}$$

$$0 \quad = A(c - \bar{t} - A^T\bar{p} - \Delta t - A^T\Delta p) + \bar{\varepsilon}b, \tag{3.13}$$

so from (3.6a)-(3.6b) we obtain

$$\begin{aligned} T(c - t - A^Tp - B^Tu) + \bar{\varepsilon}\bar{\gamma}e &= (\bar{T} + \Delta T)(c - \bar{t} - \Delta t - A^T\bar{p} - A^T\Delta p - B^Tu) + \bar{\varepsilon}\bar{\gamma}e \\ &= \bar{T}(c - \bar{t} - \Delta t - A^T\bar{p} - A^T\Delta p - B^Tu) + \bar{\varepsilon}\bar{\gamma}e \\ &+ \Delta T(c - \bar{t} - \Delta t - A^T\bar{p} - A^T\Delta p - B^Tu) \\ &= \bar{\varepsilon}\bar{\gamma}\bar{T}^{-1}\Delta t + \Delta T(c - \bar{t} - \Delta t - A^T\bar{p} - A^T\Delta p - B^Tu) \\ &= \bar{T}^{-1}\Delta T(\bar{\varepsilon}\bar{\gamma}e + \bar{T}(c - \bar{t} - A^T\bar{p} - \Delta t - A^T\Delta p - B^Tu)) \\ &= \bar{\varepsilon}\bar{\gamma}\bar{T}^{-2}\Delta T\Delta t, \end{aligned}$$

where $T = \text{Diag}(t)$, $\Delta T = \text{Diag}(\Delta t)$, and the third and the last equality follow from (3.12). This implies

$$\begin{aligned} \|T(c - t - A^Tp - B^Tu) + \bar{\varepsilon}\bar{\gamma}e\| &\le \bar{\varepsilon}\bar{\gamma}\|\bar{T}^{-2}\Delta T\Delta t\| \\ &\le \bar{\varepsilon}\bar{\gamma}\|\bar{T}^{-2}\Delta T\Delta t\|_1 \\ &= \bar{\varepsilon}\bar{\gamma}\|\bar{T}^{-1}\Delta t\|^2 \\ &\le \bar{\varepsilon}\bar{\gamma}\theta_{\bar{\varepsilon}\bar{\gamma}}(\bar{t})\beta^2, \end{aligned} \tag{3.14}$$

where the third inequality follows from (3.11).

(a) We have from (3.11) and the hypothesis $\beta < 1/\theta_{\bar{\varepsilon}\bar{\gamma}}(\bar{t})$ that $\|\bar{T}^{-1}\Delta t\|^2 \le \theta_{\bar{\varepsilon}\bar{\gamma}}(\bar{t})\beta^2 < \beta < 1$, so (3.6a) and $\bar{t} > 0$ yields $t = \bar{t} + \Delta t > 0$. Also, $B\bar{t} = d$ together with $B\Delta t = 0$ [cf. (3.6c)] and (3.6a) yields $Bt = d$ and (3.13) together with (3.6a)-(3.6b) yields $0 = A(c - t - A^Tp) + \bar{\varepsilon}b$. Hence $(t,p) \in \mathcal{Y}_{\bar{\varepsilon}}$.

(b) Fix any α satisfying (3.7) and let $\gamma = \alpha\bar{\gamma}$, $\varepsilon = \alpha\bar{\varepsilon}$. [Notice that because $\theta_{\bar{\varepsilon}\bar{\gamma}}(\bar{t})\beta < 1$, the left hand quantity in (3.7) is strictly less than 1, so such an α exists.] Let

$$r \quad = T(c - t - A^Tp - B^Tu) + \varepsilon\gamma e. \tag{3.15}$$

Then the triangle inequality and (3.14) imply

$$\|r\|/(\varepsilon\gamma) \quad \le \|T(c - t - A^Tp - B^Tu) + \bar{\varepsilon}\bar{\gamma}e\|/(\varepsilon\gamma) + (1 - \alpha^2)\sqrt{m}/\alpha^2$$

$$\leq\ \theta_{\overline{\varepsilon\gamma}}(\bar{t})\beta^2/\alpha^2 + (1/\alpha^2 - 1)\sqrt{m},$$

which together with the fact [cf. (3.7)]

$$\left(\theta_{\overline{\varepsilon\gamma}}(\bar{t})\beta^2 + \sqrt{m}\right)/(\beta + \sqrt{m})\ \leq\ \alpha^2,$$

yields

$$\|r\|/(\varepsilon\gamma)\quad\leq\quad\beta. \tag{3.16}$$

Let

$$s\quad=\ A(c - t - A^T p) + \varepsilon b. \tag{3.17}$$

By using (3.6a), (3.6b) and (3.13), we have

$$\begin{aligned} s\quad &= A(c - \bar{t} - \Delta t - A^T\bar{p} - A^T\Delta p) + \alpha\bar{\varepsilon}b \\ &= (\alpha - 1)\bar{\varepsilon}b, \end{aligned}$$

which together with the fact [cf. (3.7)]

$$1/(1 + \beta\sqrt{\bar{\gamma}/\bar{\varepsilon}}\,/\|b\|)\ \leq\ \alpha\ \leq\ 1,$$

yields

$$\|s\|/\sqrt{\varepsilon\gamma}\ =\ (1/\alpha - 1)\|b\|\sqrt{\bar{\varepsilon}/\bar{\gamma}}\ \leq\ \beta.$$

This together with (3.16) and the definition of $\rho_{\varepsilon,\gamma}(t,p,u)$ [cf. (3.1)] proves our claim. Q.E.D.

3.2. Bounds

Lemma 2 shows that if the rate α at which the penalty parameters ε and γ are decreased is not too small [cf. (3.7)], then a single Newton step suffices to keep the current iterate close to the optimal solution of the penalized problem $(D_{\varepsilon,\gamma})$. Thus, in order to establish the (linear) convergence of the QLPPF algorithm, it suffices to bound α away from 1 which, according to (3.7), amounts to bounding $\theta_{\varepsilon\gamma}(t)$ by some quantity independent of t and ε, γ. It is not difficult to see that such a bound does not exist for arbitrary t. Fortunately, we need to consider only those t which, together with some p, are close to the optimal solution of $(D_{\varepsilon,\gamma})$, in which case, as we show below, such a bound does exist (provided that a certain primal non-degeneracy assumption also holds). The proof of this is somewhat intricate: For ε and γ large, we argue by showing that t can not be too large, i.e., of the order $\varepsilon + \sqrt{\varepsilon\gamma}$ (see Lemma 3 (a)) and, for ε and γ small, we argue by showing that, under the primal non-degeneracy assumption, the columns of A corresponding to those components of t which are small (i.e., of the order γ) are of rank n.

Lemma 3.

(a) There exist constants $M_1 > 0$, $M_2 > 0$ depending on A only such that $\|t\| \le M_1(\varepsilon + \sqrt{\varepsilon\gamma}) + M_2$ for all $(t,p)\in \mathcal{Y}_\varepsilon$ satisfying $\rho_{\varepsilon,\gamma}(t,p,u) \le 1$ with some u, for all $\varepsilon > 0$ and all $\gamma > 0$.

(b) Suppose that (P) is primal non-degenerate in the sense that, for every optimal solution (x^*,y^*) of (P), those columns of A corresponding to the positive components of $x^* + B^Ty^*$ have rank n. Then, for all $(t,p)\in \mathcal{Y}_\varepsilon$ satisfying $\rho_{\varepsilon,\gamma}(t,p,u) \le 1$ with some u, all $\varepsilon > 0$, all $\gamma > 0$ and all $\lambda \ge \varepsilon\gamma\sqrt{m}/(1+\sqrt{m})$, there holds $\theta_\lambda(t) \le \psi(\varepsilon/\gamma)$, where

$$\psi(\omega) = (M_3 + M_4/\omega + M_5\omega)^2, \qquad \forall\, \omega > 0, \tag{3.18}$$

and $M_3 \ge 1$, $M_4 > 0$, $M_5 > 0$ are scalars depending on A, B, b, c and d only.

Proof: (a) The proof is by contradiction. Suppose the contrary, so that there exists a sequence $\{(t^k,p^k,u^k,\varepsilon^k,\gamma^k)\}$ such that

$$(t^k,p^k)\in \mathcal{Y}_{\varepsilon^k}, \quad \rho_{\varepsilon^k,\gamma^k}(t^k,p^k,u^k) \le 1, \quad \forall\, k, \tag{3.19}$$

and

$$\|t^k\|/(\varepsilon^k + \sqrt{\varepsilon^k\gamma^k}) \to \infty, \quad \|t^k\| \to \infty. \tag{3.20}$$

By passing into a subsequence if necessary we will assume that $(t^k,p^k)/\|(t^k,p^k)\|$ converges to some limit point, say (t^∞,p^∞) (so $(t^\infty,p^\infty) \neq 0$).

Since we have from (3.19) [also using (3.1)-(3.2)] that

$$
\begin{aligned}
&A(c - t^k - A^Tp^k) = -\varepsilon^k b, \quad Bt^k = d, \quad t^k > 0, && \forall\, k,\\
&\|T^k(c - t^k - A^Tp^k - B^Tu^k) + \varepsilon^k\gamma^k e\| \le \varepsilon^k\gamma^k, && \forall\, k,
\end{aligned}
$$

where $T^k = \mathrm{Diag}(t^k)$, then, upon dividing both sides of first three relations (respectively, fourth relation) by $\|(t^k,p^k)\|$ (respectively, $\|(t^k,p^k)\|^2$) and letting $k \to \infty$, we obtain from (3.20) and $(t^k,p^k)/\|(t^k,p^k)\| \to (t^\infty,p^\infty)$ that $t^\infty \ge 0$ and

$$A(t^\infty + A^Tp^\infty) = 0, \tag{3.21}$$

$$Bt^\infty = 0, \tag{3.22}$$

$$\|T^\infty(t^\infty + A^Tp^\infty) + v^\infty\| \le 0, \tag{3.23}$$

where $T^\infty = \mathrm{Diag}(t^\infty)$ and $v^\infty = \lim_{k\to\infty} T^kB^Tu^k/\|(t^k,p^k)\|^2$. From (3.21) (and using the full row rank property of A) we have $p^\infty = -(AA^T)^{-1}At^\infty$, so $t^\infty \neq 0$ (otherwise $p^\infty = 0$ also). Let us, by re-indexing the coordinates if necessary, partition t^∞ into $t^\infty = \begin{bmatrix} t^{\infty\prime} \\ t^{\infty\prime\prime} \end{bmatrix}$ with $t^{\infty\prime} = 0$ and $t^{\infty\prime\prime} > 0$. We correspondingly partition $A = [A'\ A'']$, $B = [B'\ B'']$, $v^\infty = \begin{bmatrix} v^{\infty\prime} \\ v^{\infty\prime\prime} \end{bmatrix}$, $t^k = \begin{bmatrix} t^{k\prime} \\ t^{k\prime\prime} \end{bmatrix}$. Then, from (3.21), (3.22) and (3.23) we have, respectively,

$$A'(A')^Tp^\infty + A''(t^{\infty\prime\prime} + (A'')^Tp^\infty) = 0, \tag{3.24}$$

$$B''t^{\infty\prime\prime} = 0, \tag{3.25}$$

$$t^{\infty\prime\prime} + (A'')^Tp^\infty + (T^{\infty\prime\prime})^{-1}v^{\infty\prime\prime} = 0. \tag{3.26}$$

Also, from $t^{k\prime\prime}/\|(t^k,p^k)\| \to t^{\infty\prime\prime} > 0$ and $T^{k\prime\prime}(B'')^Tu^k/\|(t^k,p^k)\|^2 \to v^{\infty\prime\prime}$, where $T^{k\prime\prime} = \mathrm{Diag}(t^{k\prime\prime})$, we see that $(B'')^Tu^k/\|(t^k,p^k)\|$ converges to $(T^{\infty\prime\prime})^{-1}v^{\infty\prime\prime}$, where $T^{\infty\prime\prime} = \mathrm{Diag}(t^{\infty\prime\prime})$, so that

$$(T^{\infty\prime\prime})^{-1}v^{\infty\prime\prime} = (B'')^Tu^\infty, \tag{3.27}$$

for some u^∞. Multiplying both sides of (3.26) by A'' and using (3.24) and (3.27) gives

$$0 \quad = -A'(A')^T p^\infty + A''(B'')^T u^\infty. \tag{3.28}$$

Multiplying both sides of (3.26) by B'' and using (3.25), (3.27) gives

$$0 \quad = B''(A'')^T p^\infty + B''(B'')^T u^\infty.$$

Multiplying both sides of the above relation by $(u^\infty)^T$ and using (3.28) gives

$$0 \quad = (p^\infty)^T A'(A')^T p^\infty + (u^\infty)^T B''(B'')^T u^\infty.$$

Hence, $(A')^T p^\infty = 0$ and $(B'')^T u^\infty = 0$, which together with (3.26), (3.27) and the fact $t^{\infty\prime} = 0$ shows $t^{\infty\prime} = -(A')^T p^\infty$ and $t^{\infty\prime\prime} = -(A'')^T p^\infty$, i.e., $t^\infty = -A^T p^\infty$. Since $t^\infty \geq 0$, $t^\infty \neq 0$ and [cf. (3.22)] $Bt^\infty = 0$, this contradicts the boundedness of the dual feasible set [cf. Assumption A (b)].

(b) Fix any $\varepsilon > 0$, $\gamma > 0$, $\lambda \geq \varepsilon\gamma\sqrt{m}/(1+\sqrt{m})$, and any $(t,p) \in \mathcal{Y}_\varepsilon$ satisfying $\rho_{\varepsilon,\gamma}(t,p,u) \leq 1$ with some u. Let $T = \mathrm{Diag}(t)$ and let D, E, F be given by, respectively, (3.5b), (3.5c), and (3.5d). Then, $F^{-1} = A(I - D^{1/2}ED^{1/2})A^T$, $\|E\| \leq 1$ and $D = (I + \lambda T^{-2})^{-1}$. From the definition of $\theta_\lambda(t)$ [cf. (3.5a)] we then obtain

$$\begin{aligned}
\theta_\lambda(t) &= (\|E + ED^{1/2}A^TFAD^{1/2}E\| + \|ED^{1/2}A^TF\|)^2 \\
&\leq (\|E\| + \|E\|^2\|D^{1/2}\|^2\|A\|^2\|F\| + \|E\|\,\|D^{1/2}\|\,\|A\|\,\|F\|)^2 \\
&< (1 + \|A\|^2\|F\| + \|A\|\,\|F\|)^2,
\end{aligned} \tag{3.29}$$

where the strict inequality follows from the facts $\|D\| < 1$, $\|E\| \leq 1$. Now we bound $\|F\|$. We have

$$\begin{aligned}
z^T(F^{-1})z &= z^TA(I - D^{1/2}ED^{1/2})A^Tz \\
&\geq z^TA(I - D)A^Tz \\
&= \textstyle\sum_j (A_j^Tz)^2/((t_j)^2/\lambda + 1) \\
&\geq \lambda\textstyle\sum_j (A_j^Tz)^2/(t_j)^2 \\
&\geq \lambda\textstyle\sum_j (A_j^Tz)^2/\|t\|^2 \\
&\geq \lambda\sigma\|z\|^2/\|t\|^2, \qquad \forall\, z,
\end{aligned} \tag{3.30}$$

where the first inequality follows from $\|E\| \leq 1$ and $\sigma > 0$ denotes the smallest eigenvalue of AA^T. [$\sigma > 0$ because A has full row rank.] Hence, part (a) and $\lambda \geq \sqrt{m}/(1 + \sqrt{m})\varepsilon\gamma$ yield

$$\|F\| \quad \le \|t\|^2/(\lambda\sigma) \quad \le (1 + 1/\sqrt{m})(M_1(\varepsilon + \sqrt{\varepsilon\gamma}) + M_2)^2/(\varepsilon\gamma\sigma). \tag{3.31}$$

For ε and γ near zero, we give a different bound on $\|F\|$. By the primal non-degeneracy assumption, there exists a constant $\delta > 0$ depending on A, B, b, c and d only such that if (x,y) is any optimal primal solution of a perturbed linear program, which is obtained from (D) by perturbing the cost coefficients and the right hand side coefficients by at most δ, then the columns $\{ A_j \mid x_j + B_j^T y \ge \delta \}$ have rank n. Suppose that $\varepsilon \in (0,\delta/M_0]$ and $\gamma \in (0,(\delta)^2/4]$. Since $(t,p) \in \mathcal{Y}_\varepsilon$ and $\rho_{\varepsilon,\gamma}(t,p,u) \le 1$, it follows from Lemma 1 that the columns $\{ A_j \mid (t_j + A_j^T p + B_j^T u - c_j)/\varepsilon \ge \delta \}$ have rank n. Since $(t,p) \in \mathcal{Y}_\varepsilon$ and $\rho_{\varepsilon,\gamma}(t,p,u) \le 1$, we have $T(c - t - A^T p - B^T u) \ge -2\varepsilon\gamma e$ [cf. (3.1), (3.2)] so that $(t_j + A_j^T p + B_j^T u - c_j)/\varepsilon \ge \delta$ implies $t_j \le 2\gamma/\delta$. Hence, we obtain from (3.30) that

$$\begin{aligned} z^T(F^{-1})z \quad & \ge \lambda \textstyle\sum_j (A_j^T z)^2/(t_j)^2 \\ & \ge \lambda(\delta)^2 \sum_{t_j \le 2\gamma/\delta} (A_j^T z)^2/(2\gamma)^2 \\ & \ge \lambda(\delta)^2 \sigma' \|z\|^2/(2\gamma)^2, \qquad \forall\ z, \end{aligned}$$

where the last inequality follows from the fact that those A_j for which $t_j \le 2\gamma/\delta$ have rank n, and σ' is some positive scalar depending on A only. Since $\lambda \ge \sqrt{m}/(1 + \sqrt{m})\varepsilon\gamma$, we then have

$$\|F\| \quad \le 4(\gamma)^2/(\lambda(\delta)^2\sigma') \quad \le 4(1 + 1/\sqrt{m})\gamma/(\varepsilon(\delta)^2\sigma'). \tag{3.32}$$

For convenience, let $\omega = \varepsilon/\gamma$. Then, $\gamma \le \min\{(\delta)^2/4, \delta/(M_0\omega)\}$ implies $\gamma \le (\delta)^2/4$, $\varepsilon \le \delta/M_0$ so (3.32) yields $\|F\| \le 4(1 + 1/\sqrt{m})/(\omega(\delta)^2\sigma')$; otherwise $\gamma > \min\{(\delta)^2/4, \delta/(M_0\omega)\}$ so (3.31) yields $\|F\| \le (1 + 1/\sqrt{m})(M_1(\sqrt{\omega} + 1) + M_2/(\sqrt{\omega}\min\{(\delta)^2/4, \delta/(M_0\omega)\}))^2/\sigma$. Therefore, there exist positive scalars K_1, K_2, K_3 depending on A, B, b, c and d only such that

$$\|F\| \quad \le \quad \begin{cases} K_1/\omega & \text{if } \gamma \le \min\{(\delta)^2/4, \delta/(M_0\omega)\}; \\ \left(K_2\sqrt{\omega} + K_2 + \dfrac{K_3\sqrt{\omega}}{\min\{\delta\omega, 4/M_0\}}\right)^2 & \text{otherwise.} \end{cases}$$

Combining the above bound with (3.29) and we conclude that there exist scalars $M_3 \ge 1$, $M_4 > 0$, $M_5 > 0$ depending on A, B, b, c and d only such that $\theta_\lambda(t) < (M_3 + M_4/\omega + M_5\omega)^2$. Q.E.D.

3.3. Main Convergence Result

By combining Lemmas 1 to 3, we obtain the following global convergence result for the QLPPF algorithm:

Theorem 1. Suppose that (P) is primal non-degenerate in the sense of Lemma 3 (b) and let $\psi(\cdot)$ be given by (3.18). If in the QLPPF algorithm (t^1,p^1) together with some u^1 satisfies

$$t^1 > 0, \qquad Bt^1 = d, \tag{3.33}$$
$$\theta_{\varepsilon^1\gamma^1}(t^1) \le \psi(\varepsilon^1/\gamma^1), \tag{3.34}$$
$$\rho_{\varepsilon^1,\gamma^1}(t^1,p^1,u^1) \le \beta, \tag{3.35}$$

for some scalar

$$0 < \beta < 1/\psi(\varepsilon^1/\gamma^1), \tag{3.36}$$

and if we choose

$$\alpha^k = \max\{ \sqrt{(\theta_{\varepsilon^k\gamma^k}(t^k)\beta^2 + \sqrt{m})/(\beta + \sqrt{m})}\, , 1/(1 + \beta\sqrt{\gamma^1/\varepsilon^1}/\|b\|)\}, \quad \forall\, k, \tag{3.37}$$

then $\{\varepsilon^k\} \downarrow 0$, $\{\gamma^k\} \downarrow 0$ linearly, and $\{((t^k + A^Tp^k - c)/\varepsilon^k, u/\varepsilon^k)\}$, $\{(t^k,p^k)\}$ approach the optimal solution set of, respectively, (P) and (D).

Proof: First notice from $\psi(\omega) > 1$ for all $\omega > 0$ [cf. (3.18)] and (3.36) that $\beta < 1$, so $(\alpha^k)^2$ is lower bounded by $\sqrt{m}/(1 + \sqrt{m})$ for all k [cf. (3.37)]. Also notice from (2.7) that

$$\varepsilon^k/\gamma^k = \varepsilon^1/\gamma^1, \qquad \forall\, k. \tag{3.38}$$

We claim that

$$(t^k,p^k) \in \mathcal{Y}_{\varepsilon^{k-1}}, \qquad \rho_{\varepsilon^{k-1},\gamma^{k-1}}(t^k,p^k,u^k) \le \beta, \qquad \rho_{\varepsilon^k,\gamma^k}(t^k,p^k,u^k) \le \beta, \tag{3.39}$$

for all $k \ge 2$. It is easily seen by using (3.33)-(3.37), (2.5)-(2.7) and Lemma 2, that (3.39) holds for $k = 2$. Suppose that (3.39) holds for all $k \le h$, for some $h \ge 2$. Then, $(t^h,p^h) \in \mathcal{Y}_{\varepsilon^{h-1}}$ and

$\rho_{\varepsilon^{h-1},\gamma^{h-1}}(t^h,p^h,u^h) \le \beta < 1$. Since $\varepsilon^h = \alpha^{h-1}\varepsilon^{h-1}$, $\gamma^h = \alpha^{h-1}\gamma^{h-1}$ [cf. (2.7)] and $(\alpha^{h-1})^2 \ge \sqrt{m}/(1 + \sqrt{m})$, we can apply Lemma 3 (b) to conclude

$$\theta_{\varepsilon^h\gamma^h}(t^h) \quad \le \psi(\varepsilon^{h-1}/\gamma^{h-1}) \quad = \psi(\varepsilon^1/\gamma^1), \tag{3.40}$$

where the equality follows from (3.38). Then, by (3.36), $\beta < 1/\theta_{\varepsilon^h\gamma^h}(t^h)$. Since (3.39) holds for k = h, we also have $\rho_{\varepsilon^h,\gamma^h}(t^h,p^h,u^h) \le \beta$, so Lemma 2 together with (2.5)-(2.7) and (3.37) yields

$$(t^{h+1},p^{h+1}) \in \mathcal{Y}_{\varepsilon^h}, \qquad \rho_{\varepsilon^h,\gamma^h}(t^{h+1},p^{h+1},u^{h+1}) \le \beta, \qquad \rho_{\varepsilon^{h+1},\gamma^{h+1}}(t^{h+1},p^{h+1},u^{h+1}) \le \beta.$$

Hence, (3.39) holds for k = h+1.

Since (3.39) holds for all $k \ge 2$, we see that (3.40) holds for all $h \ge 2$. Then, by (3.36), $\theta_{\varepsilon^h\gamma^h}(t^h)\beta$ is less than 1 and bounded away from 1 for all $h \ge 2$, so that [cf. (3.37)] α^k is less than 1 and bounded away from 1. Hence $\{\varepsilon^k\} \downarrow 0$, $\{\gamma^k\} \downarrow 0$ at the rate of a geometric progression. The remaining proof follows from (3.39) and Lemma 1. Q.E.D.

Notice that instead of $\theta_{\varepsilon^k\gamma^k}(t^k)$ we can use, for example, the upper bound $1/\psi(\varepsilon^1/\gamma^1)$ in (3.37), and linear convergence would be preserved. However, this bound is typically loose and difficult to compute. There is also the issue of finding ε^1, γ^1, β, (t^1,p^1) and u^1 satisfying (3.33)-(3.36), which we address in §3.4.

3.4. Algorithm Initialization

By Theorem 1, if the primal non-degeneracy assumption therein holds and if we can find ε, γ, (t,p) and u satisfying

$$t > 0, \qquad Bt = d, \tag{3.41}$$
$$\theta_{\varepsilon\gamma}(t) \le \psi(\varepsilon/\gamma), \tag{3.42}$$
$$\rho_{\varepsilon,\gamma}(t,p,u) < 1/\psi(\varepsilon/\gamma), \tag{3.43}$$

then we can set $\beta = \rho_{\varepsilon,\gamma}(t,p,u)$ (assuming $\rho_{\varepsilon,\gamma}(t,p,u) \neq 0$) and start the QLPPF algorithm with ε, γ, (t,p), and we would obtain linear convergence. How do we find such ε, γ, (t,p) and u?

One obvious way is to fix any $\varepsilon > 0$, any $\gamma > 0$, and then solve the penalized problem $(D_{\varepsilon,\gamma})$. The solution (t,p) obtained satisfies $t > 0$, $Bt = d$, $\nabla f_{\varepsilon,\gamma}(t,p) + \begin{bmatrix} B^T u \\ 0 \end{bmatrix} = 0$ for some u [cf. (2.2)] so, by (2.3) and (3.1)-(3.2), $\rho_{\varepsilon,\gamma}(t,p,u) = 0$ and $(t,p) \in \mathcal{Y}_\varepsilon$. Hence (3.41), (3.43) hold and, by Lemma 3 (b), $\theta_{\varepsilon\gamma}(t) \le \psi(\varepsilon/\gamma)$, so (3.42) also holds. [Of course $(D_{\varepsilon,\gamma})$ needs not be solved exactly.] To solve the problem $(D_{\varepsilon,\gamma})$, we can use any method for convex differentiable minimization (e.g., gradient descent, coordinate descent), and we would typically want ε small and γ large so that $(D_{\varepsilon,\gamma})$ is well-conditioned.

Suppose that there holds $Be = 0$ and $Ae = b$. [This holds, for example, when B is the zero matrix (which corresponds to the case when all equality constraints are penalized), and a change of variable $x' = (\bar{X})^{-1}x$, where $\bar{x}$ is any interior feasible solution of (P) (i.e., $A\bar{x} = b$, $\bar{x} > 0$) and $\bar{X} = \mathrm{Diag}(\bar{x})$, has been made in (P).] Then we can find a usable ε, γ, (t,p), u immediately: Fix any $\varepsilon > 2\psi(1)(\|c\| + \|B^T(BB^T)^{-1}d\|)$ and let $\gamma = \varepsilon$. Also let $w = B^T(BB^T)^{-1}d$ and

$$p = (AA^T)^{-1}A(c - w),$$
$$t = \varepsilon e + w,$$
$$u = -(BB^T)^{-1}d.$$

Then $Bw = d$, $A(c - A^T p) = Aw$, $At = \varepsilon b + Aw$ and $B^T u = \varepsilon e - t$, so that

$$Bt \quad = Bw \quad = d,$$

$$
\begin{aligned}
A(c - t - A^Tp) &= -\varepsilon b, \\
T(c - t - A^Tp - B^Tu) + \varepsilon\gamma e &= (\varepsilon I + W)(c - \varepsilon e - A^Tp) + (\varepsilon)^2 e \\
&= (\varepsilon I + W)(c - A^Tp) - \varepsilon w,
\end{aligned}
$$

where $T = \text{Diag}(t)$ and $W = \text{Diag}(w)$. Also from $\psi(1) > 1$ and our choice of ε we have $\varepsilon > \|w\|$, so $t > 0$. Hence $(t,p) \in \mathcal{Y}_\varepsilon$ [cf. (3.2)] and [cf. (3.1) and $\varepsilon = \gamma$]

$$
\begin{aligned}
\rho_{\varepsilon,\gamma}(t,p,u) &= \|(\varepsilon I + W)(c - A^Tp) - \varepsilon w\|/(\varepsilon)^2 \\
&\leq \|c - w - A^Tp\|/\varepsilon + \|W\|\,\|c - A^Tp)\|/(\varepsilon)^2 \\
&= \|(I - A^T(AA^T)^{-1}A)(c - w)\|/\varepsilon + \|W\|\,\|(I - A^T(AA^T)^{-1}A)c + A^T(AA^T)^{-1}Aw\|/(\varepsilon)^2 \\
&\leq \|c - w\|/\varepsilon + \|w\|\,(\|c\| + \|w\|)/(\varepsilon)^2. \qquad (3.44)
\end{aligned}
$$

where the last inequality follows from the triangle inequality and the nonexpansive property of projection matrices. From our choice of ε we see that $(\|c\| + \|w\|)/\varepsilon < .5/\psi(1)$, so the right hand side of (3.44) is bounded by $.5/\psi(1) + (.5/\psi(1))^2 \leq 1/\psi(1) = 1/\psi(\varepsilon/\gamma)$, where the inequality follows from $\psi(1) \geq 1$ and the equality follows from $\gamma = \varepsilon$. Hence (3.41), (3.43) hold. Also, since $(t,p) \in \mathcal{Y}_\varepsilon$, then (3.43) together with Lemma 3 (b) shows that (3.42) holds.

4. Numerical Results

In order to study the performance of the QLPPF algorithm in practice, we have implemented the algorithm to solve the special case of (P) and (D) in which B is the zero matrix, i.e., (P) is of the form

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b, \quad x \geq 0. \end{array} \tag{4.1}$$

[This corresponds to penalizing all equality constraints in the corresponding dual problem.] Below we describe our implementation and present our preliminary numerical experience.

1. Initialization. In our implementation, we set for all problems

$$\varepsilon^1 = 10^{-7}\|c\|_1/m, \qquad \gamma^1 = 10^4\|c\|_1/m,$$

and set (arbitrarily)

$$p^1 = 0, \qquad t^1 = e.$$

[Notice that since B is the zero matrix, t^1 can be set to any positive vector.] Care must be exercised in choosing ε^1 and γ^1: if their values are set too low, then the QLPPF algorithm may fail to converge; if their values are set too high, then the QLPPF algorithm may require many iterations to converge. [Notice that we set ε^1 and γ^1 directly proportional to the average cost $\|c\|_1/m$ so their values scale with c.]

2. Steplength Selection. To ensure that the t^k's remain inside the positive orthant, we employ a backtracking scheme similar to that used by Setiono: whenever $t^k + \Delta t^k$ is outside the positive orthant, we replace the formula for t^{k+1} in (2.6) by

$$t^{k+1} = t^k + .98\lambda^k \Delta t^k, \tag{4.2}$$

where

$$\lambda^k = \min_{\Delta t_j^k<0} -\frac{t_j^k}{\Delta t_j^k}. \tag{4.3}$$

However, this raises a difficulty, namely, for λ^k much smaller than 1, the vector $(.98\lambda^k\Delta t^k, \Delta p^k)$ may be far from the Newton direction $(\Delta t^k, \Delta p^k)$ and, as a consequence, the iterates may fail to converge. To remedy this, we replace [analogous to (4.2)] the formula for p^{k+1} in (2.6) by

$$p^{k+1} = p^k + .98\lambda^k \Delta p^k,$$

whenever non–convergence is detected. [The parameter value .98 is chosen somewhat arbitrarily, but it works well in our tests.]

The proper choice of the α^k's is very important for the QLPPF algorthm: if the α^k's are too near 1 (so the penalty parameters decrease slowly), then the algorithm would converges slowly; if the α^k's are too near 0 (so the penalty parameters decrease rapidly), then the algorithm may fail to converge. In our implementation we adjust the α^k's dynamically according to the following rule:

$$\alpha^k = \begin{cases} \max\{.3, .95\alpha^{k-1}\} & \text{if } \lambda^k = 1; \\ .6 & \text{if } \lambda^k \le .2; \\ \alpha^{k-1} & \text{otherwise,} \end{cases} \qquad \forall\, k \ge 2,$$

with α^1 set to .5. The rational for this adjustment rule is that, if $\lambda^k = 1$, then the current iterate is closely following the solution trajectory (so we can decrease the penalty parameters at a faster rate and still retain convergence) and, if $\lambda^k \le .2$, then the current iterate is unable to follow the solution trajectory (so we must decrease the penalty parameters at a slower rate).

3. Termination. To avoid numerical problems, we stop decreasing the penalty parameters ε and γ when they reach some prespecified tolerances ε_{min} and γ_{min}, respectively. In our tests we set

$$\varepsilon_{min} = 10^{-12}\|c\|_1/m, \qquad \gamma_{min} = 10^{-9}\|c\|_1/m.$$

We terminate the QLPPF algorithm when the relative duality gap and the violation of primal and dual feasibility are small. More specifically, we terminate whenever the current iterate, denoted by (t,p), satisfies

$$\frac{\|X(A^Tp - c)\|_1}{|c^Tx|} \leq 10^{-7}, \tag{4.4}$$

$$\max\{ \|Ax - b\|_\infty, \|[-x]^+\|_\infty \} \leq 10^{-7}, \tag{4.5}$$

$$\|[A^Tp - c]^+\|_\infty \leq 10^{-7}, \tag{4.6}$$

where $x = (t + A^Tp - c)/\varepsilon$ [cf. Lemma 1], $X = \text{Diag}(x)$, and $[\cdot]^+$ denotes the orthogonal projection onto the nonnegative orthant. Only for three of our test problems could the above termination criterion not be met (owing to violation of (4.4) and (4.6)) in which case the algorithm is terminated whenever primal feasibility (4.5) is met and $|c^Tx - v^*|/|v^*|$ is less than 5×10^{-7}, where v^* denotes the optimal value of (4.1).

4. Solving for the Direction. The most expensive computation at each iteration of the QLPPF algorithm lies in solving the system of linear equations (2.5). This can be seen to entail solving a single linear system of the form

$$AQA^Tw = z, \tag{4.7}$$

for w, where z is some n–vector and Q is some n×n diagonal matrix whose j–th diagonal entry is

$$\frac{\varepsilon\gamma}{\varepsilon\gamma + (t_j)^2}, \tag{4.8}$$

with $\varepsilon > 0$, $\gamma > 0$, and t some positive m–vector. [Linear system of the form (4.7) also arise in interior point algorithms, but (4.7) has the nice property that the condition number of Q can be controlled (by adjusting the penalty parameters ε and γ).] In our implementation, (4.7) is solved using YSMP, a sparse matrix package for symmetric positive semi–definite systems developed at Yale University (see [EGSS79], [EGSS82]) and a precursor to the commercial package SMPAK (Scientific Computing Associates, 1985). YSMP comprises a set of Fortran routines implementing the Cholesky decomposition scheme and, as a preprocessor, the minimum-degree ordering algorithm (see, e.g., [GeL81]). In our implementation, the minimum–degree ordering routine ORDER is called first to obtain a permutation of the rows and columns of the matrix AA^T so that fill-in is reduced during factorization. Then, AA^T is symbolically factored using the routine SSF. (SSF is called only once since the nonzero pattern of AQA^T does not change with Q.) At each iteration, the matrix AQA^T is numerically factored by the routine SNF (taking advantage of information generated by SSF concerning the location of the nonzeros in the factorization), and the two triangular systems thus generated are solved by the routine SNS to obtain a solution of (4.7). (We also experimented with the public domain version of the sparse matrix package SPARSPAK [GeL81], presently avaliable from Netlib. We found SPARSPAK to be comparable to YSMP in solution time but somewhat inferior in solution accuracy.)

5. Data Structure. The data structure used in our implementation is similar to that described in, e.g., [AKRV89], [MoM87]. Each matrix is stored in sparse format by row. In order to compute the nonzero entries of the matrix AQA^T efficiently for any Q, we also store the nonzero entries of the outer products $A_j(A_j)^T$, where A_j denotes the j–th column of A. AQA^T is then computed using the formula

$$AQA^T = \sum_j q_j A_j(A_j)^T,$$

where q_j denotes the j–th diagonal entry of Q and the product of q_j is taken with each nonzero entry of $A_j(A_j)^T$.

6. Test Problems. Our test problems comprise twenty of the linear programming problems distributed publicly through Netlib (see [Gay86]). These problems range in size from 27 rows and 51 columns up to 712 rows and 2467 columns and, for some of them, slack columns must be added and null rows must be removed to transform them into the form (4.1). [We also wrote a routine to convert these problems from their original MPS format to that used by our implementation.] The statistics for the test problems (after problem transformation) are summarized in Table 4.1.

7. Computing Environment. Our implementation was written in Fortran and was compiled and ran on a single–processor Ardent TITAN (a graphics supercomputer) under the AT&T System V Release 3 UNIX operating system with Berkeley 4.3 extension. The optimization option for the compiler is –O1, i.e., optimization without vectorization/parallelization.

Table 4.2 summarizes the computational results obtained with our implementation of the QLPPF algorithm. Columns 2 and 3 show, respectively, the total number of iterations and the CPU time. Columns 4 and 6 show, respectively, the cost and the accuracy of the final primal solution [the latter is measured by the left hand side quantity in (4.6)]. Analogously, columns 5 and 7 show, respectively, the cost and the accuracy of the final dual solution [the latter is measured by the left hand side quantity in (4.7)]. For most of the problems, the primal cost agrees with the optimal value in the first 7 digits and the accuracy of the primal solution is between 10^{-7} and 10^{-14}. Thus the quality of the computed solutions compares favorably with that of solutions generated by interior point algorithms. The number of iterations varies between 29 and 44 and the CPU time varies between 0.3 and 83 seconds, depending on the problem size and the sparsity of the constraint matrix. For most of the problems, over half of the CPU time is devoted to solving the linear system (4.8) at every iteration. (We also performed tests on a μVAX–2000 Work Station under the operating system VMS 4.1. The resulting number of iterations is roughly the same; the accuracy of the final solutions improves slightly; and the CPU times are from 6 to 7 times that on the TITAN.)

The number of iterations for the QLPPF algorithm is comparable to that for the projected Newton barrier method of Gill et. al. [GMSWT86], but is typically more than that for the affine–scaling algorithm or for Setiono's algorithm. Specifically, by comparing column 3 of [MoM87, Table 5] (also see [BDDW89], [MSSPB88]) with column 2 of Table 4.2, we see that the number of iterations for the QLPPF algorithm can be up to 3/2 times that for the affine–scaling algorithm. Similarly, the number of iterations for the QLPPF algorithm can be up to 5/3 times that for Setiono's algorithm (compare column 3 of [Set89, Table 3] with column 2 of Table 4.2). On the other hand, there are some problems on which the number of iterations is less for the QLPPF algorithm than for the other algorithms.

In conclusion, our computational results indicate that, for linear programming, a mixed interior point–exterior point penalty method, as exemplified by the QLPPF algorithm, can perform near the level of interior point algorithms. On the other hand, we caution that these results are very preliminary and thus should be viewed only as encouraging. In particular, some fine tuning of the initial penalty parameters are necessary for the QLPPF algorithm to attain the performance shown in Table 4.2. As a case in point, for the Netlib problem Scagr7 (see [Gay86]), the total number of iterations is 52 if ε^1 is set according to (4.2) and drops down to 42 if ε^1 is raised by a factor of ten.

Finally, we remark it is typically beneficial to operate the QLPPF algorithm with a large ratio of γ^k/ε^k. An intuitive explanation for this is that, if γ^k/ε^k is small, then γ^k is not a sufficiently large penalty (relative to ε^k) to maintain $t^k + \Delta t^k$ within the positive orthant. This results in small stepsizes λ^k [cf. (4.4)] and hence slow convergence.

Problem Name	Rows	Cols	Constraint Nonzeros [1]	Hessian Nonzeros [2]	Optimal Value [3]
Afiro	27	51	102	90	−4.6475314E+2
Adlittle	56	138	424	384	2.2549496E+5
Scsd1	77	760	2388	1133	8.6666666E+0
Share2B	96	162	777	871	−4.1573224E+2
Share1B	117	253	1179	1001	−7.6589319E+4
Scsd6	147	1350	4316	2099	5.0500000E+1
Lotfi	153	366	1136	1196	−2.5264706E+1
Beaconfd	173	295	3408	2842	3.3592486E+4
Israel	174	316	2443	11227	−8.9664482E+5
BrandY	193	303	2202	2734	1.5185099E+3
Sc205	205	317	665	656	−5.2202061E+1
E226	223	472	2768	2823	−1.8751929E+1
ScTap1	300	660	1872	1686	1.4122500E+3
BandM	305	472	2494	3724	−1.5862801E−2
Scfxm1	330	600	2732	3233	1.8416759E+4
Ship04s	360	1506	4400	3272	1.7987147E+6
Ship04l	360	2166	6380	4588	1.7933245E+6
Scrs8	490	1275	3288	2198	9.0429695E+2
Scfxm2	660	1200	5469	6486	3.6660261E+4
Ship08s	712	2467	7194	5440	1.9200982E+6

Table 4.1. Test Problem Characteristics.

[1] The number of nonzero entries in A.

[2] The number of nonzero entries in AA^T.

[3] Cited from [Gay86].

Problem Name	Iters.	CPU (sec.) [1]	Primal Cost [2]	Dual Cost [2]	Primal Feas. [3]	Dual Feas. [3]
Afiro	29	0.26	−4.6475313E+2	−4.6475316E+2	2E−08	6E−11
Adlittle	35	1.12	2.2549498E+5	2.2549497E+5	4E−09	3E−08
Scsd1	37	5.56	8.6666685E+0	8.6666668E+0	2E−14	0
Share2B	33	2.26	−4.1573226E+2	−4.1573218E+2	7E−10	0
Share1B	37	3.49	−7.6589327E+4	−7.6585942E+4	8E−08	2E−06
Scsd6	37	10.46	5.0500016E+1	5.0499999E+1	6E−13	0
Lotfi	41	5.22	−2.5264705E+1	−2.5264679E+1	4E−11	3E−10
Beaconfd	28	10.00	3.3592487E+4	3.3592486E+4	1E−09	2E−09
Israel	42	83.56	−8.9664481E+5	−8.9659622E+5	3E−08	5E−05
BrandY	35	12.08	1.5185099E+3	1.5185098E+3	5E−08	1E−11
Sc205	34	2.32	−5.2202054E+1	−5.2202059E+1	1E−10	7E−12
E226	44	14.09	−1.8751921E+1	−1.8751929E+1	1E−11	2E−10
ScTap1	40	7.60	1.4122500E+3	1.4122499E+3	4E−12	0
BandM	40	14.45	−1.5862803E+2	−1.5862803E+2	9E−11	9E−10
Scfxm1	36	13.93	1.8416759E+4	1.8416758E+4	2E−08	2E−09
Ship04s	41	14.49	1.7987148E+6	1.7987147E+6	3E−10	0
Ship04l	42	20.70	1.7933246E+6	1.7933245E+6	3E−10	0
Scrs8	44	23.52	9.0429723E+2	9.0429692E+2	7E−11	1E−08
Scfxm2	37	34.01	3.6660262E+4	3.6660260E+4	1E−08	3E−09
Ship08s	43	29.97	1.9200984E+6	1.9200981E+6	6E−10	0

Table 4.2. Computational Results for the QLPPF Algorithm.

[1] Obtained using the intrinsic function SECNDS on the TITAN.

[2] Shown first 8 digits only.

[3] Shown first digit only.

5. Some Extensions

We have thusfar assumed that the parameters ε and γ are decreased at the same rate in the QLPPF algorithm. Alternatively we can decrease them at different rates. For example, Setiono's algorithm employs the strategy whereby ε is first decreased with γ held fixed and, once ε reaches a prescribed tolerance, then γ is decreased with ε held fixed. (In [Set89], the product $\varepsilon\gamma$ is what is referred to as γ.) We can also use different penalty parameters for different coordinates.

Our convergence results very possibly also extend to linear complementarity problems with positive semi-definite matrices – in the same manner that the results in [Tse89a] can be extended to these problems (see [Tse89b]). This is a topic for further study.

References

[AKRV89] Adler, I., Karmarkar, N., Resende, M. G. C., and Veiga, G., "Data Structures and Programming Techniques for the Implementation of Karmarkar's Algorithm," *ORSA Journal on Computing*, 1 (1989), 84-106.

[Ber82] Bertsekas, D. P., *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, NY (1982).

[BDDW89] Boggs, P. T., Domich, P. D., Donaldson, J. R., and Witzgall, C., "Algorithmic Enhancements to the Method of Centers for Linear Programming Problems," Technical Report, National Institute of Standards and Technology, Gaithersburg, MD (February, 1989).

[EGSS79] Eisenstat, S. C., Gursky, M. C., Schultz, M. H., and Sherman, A. H., "Yale Sparse Matrix Package I. The Symmetric Codes," Research Report #112, Department of Computer Science, Yale University, New Haven, CT.

[EGSS82] Eisenstat, S. C., Gursky, M. C., Schultz, M. H., and Sherman, A. H., "Yale Sparse Matrix Package I. The Symmetric Codes," *International Journal for Numerical Methods in Engineering*, 18 (1982), 1145–1151.

[FiM68] Fiacco, A. V., and McCormick, G. P., *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley and Sons, New York, NY (1968).

[Fri57] Frisch, K. R., "The Logarithmic Potential Method of Convex Programming," University Institute of Economics memorandom, Oslo, Norway (May 1955).

[GaZ81] Garcia, C. B. and Zangwill, W. I., *Pathways to Solutions, Fixed Points, and Equilibria*, Prentice-Hall, Englewood Cliffs, NJ (1981).

[Gay86] Gay, D. M., "Electronic Mail Distribution of Linear Programming Test Problems," Numerical Analysis Manuscript 86–0, AT&T Bell Laboratories, Murray Hill, NJ (August 1986).

[GeL81] George, J. A. and Liu, J. W.-H., *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, Englewood Cliffs, NJ (1981).

[GMSTW86] Gill, P. E., Murray, W., Saunders, M. A., Tomlin, J. A., and Wright, M. H., "On Projected Newton Barrier Methods for Linear Programming and an Equivalence to Karmarkar's Projective Method," *Mathematical Programming*, 36 (1986), 183-209.

[Gon89] Gonzaga, C. C., "An Algorithm for Solving Linear Programming Problems in $O(n^3L)$ Operations," in *Progress in Mathematical Programming: Interior-Point and Related Methods* (N. Megiddo, Editor) Springer-Verlag, New York, NY (1989).

[HaB70] Haarhoff, P. C., and Buys, J. D., "A New Method for the Optimization of a Nonlinear Function Subject to Nonlinear Constraints," *Computer Journal*, 13 (1970), 178-184.

[Hes69] Hestenes, M. R., "Multiplier and Gradient Methods," *Journal of Optimization Theory and Applications*, 4 (1969), 303-320.

[Hua67] Huard, P., "Resolution of Mathematical Programming with Nonlinear Constraints by the Method of Centers," in *Nonlinear Programming* (Abadie, J., Editor), North-Holland, Amsterdam (1967), 207-219.

[JiO78] Jittorntrum, K., and Osborn, M. R., "Trajectory Analysis and Extrapolation in Barrier Function Methods," *Journal of Australian Mathematical Society: Series B*, 20 (1978), 352–369.

[Kar84] Karmarkar, N., "A New Polynomial-Time Algorithm for Linear Programming," *Combinatorica*, 4 (1984), 373-395.

[KMY87] Kojima, M., Mizuno, S. and Yoshise, A., "A Polynomial-Time Algorithm for a Class of Linear Complementarity Problems," *Mathematical Programming*, 44 (1989), 1-26.

[Lus89] Luss, H. (Editor), Special Issue on Optimization, *AT&T Technical Journal*, 68 (1989).

[Man84] Mangasarian, O. L., "Normal Solutions of Linear Programs," *Mathematical Programming Study*, 22 (1984), 206-216.

[MSSPB88] Marsten, R. E., Saltzman, M. J., Shanno, D. F., Pierce, G. S., and Ballintijn, J. F., "Implementation of a Dual Affine Interior Point Algorithm for Linear Programming," Working Paper, Center for the Management of Information, University of Arizona, Tucson, AZ (November, 1988).

[MMS88] McShane, K. A., Monma, C. L., and Shanno, D., "An Implementation of a Primal-Dual Interior Point Method for Linear Programming," Technical Report, Mathematics and Operations Research, Bell Communications Research, Morristown, NJ (March 1988).

[MeS88] Mehrotra, S. and Sun, J., "An Algorithm for Convex Quadratic Programming that Requires $O(n^{3.5}L)$ Arithmetic Operations," Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL (February 1988).

[Meg89] Megiddo, N. (Editor), *Progress in Mathematical Programming: Interior-Point and Related Methods*, Springer-Verlag, New York, NY (1989)

[MoM87] Monma, C. L. and Morton, A. J., "Computational Experience with a Dual Affine Variant of Karmarkar's Method for Linear Programming," *Operations Research Letters*, 6 (1987), 261-267.

[MoA87] Monteiro, R. C., and Adler, I., "Interior Path Following Primal-Dual Algorithms - Part I: Linear Programming," *Mathematical Programming*, 44 (1989), 27-41.

[PaS82] Papadimitriou, C. H. and Steiglitz, K., *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, NJ (1982).

[Pow69] Powell, M. J. D., "A Method for Nonlinear Constraints in Minimization Problems," In *Optimization* (R. Fletcher, ed.), Academic Press, New York, NY (1969), 283-298.

[Ren88] Renegar, J., "A Polynomial-Time Algorithm, based on Newton's Method, for Linear Programming," *Mathematical Programming*, 40 (1988), 59-93.

[Roc70] Rockafellar, R. T., *Convex Analysis*, Princeton University Press, Princeton, NJ (1970).

[Roc76] Rockafellar, R. T., "Augmented Lagrangians and Applications for the Proximal Point Algorithm in Convex Programming," *Mathematics of Operations Research*, 1 (1976), 97-116.

[Set89] Setiono, R., "An Interior Dual Proximal Point Algorithm for Linear Programs," Technical report, Computer Sciences Department, University of Wisconsin, Madison, WI (September, 1989).

[Tod88] Todd, M. J., "Recent Developments and New Directions in Linear Programming," School of Operations Research and Industrial Engineering, Technical Report No. 827, Cornell University, Ithaca, NY (1988).

[Tse89a] Tseng, P., "A Simple Complexity Proof for a Polynomial-Time Linear Programming Algorithm," *Operations Research Letters*, 8 (1989), 155-159.

[Tse89b] Tseng, P., "Complexity Analysis of a Linear Complementarity Algorithm Based on a Lyapunov Function," Laboratory for Information and Decision Systems, Report No. P-1884, M.I.T., Cambridge, MA (June, 1989).

[WBD88] Witzgall, C., Boggs, P. T., and Domich, P. D., "On the Convergence Behavior of Trajectories for Linear Programming," Technical Report, National Bureau of Standards, Boulder, CO (August, 1988).