

A Simple Polynomial-Time Algorithm for Convex Quadratic Programming¹

by

Paul Tseng²

Abstract

In this note we propose a polynomial-time algorithm for convex quadratic programming. This algorithm augments the objective by a logarithmic penalty function and then solves a sequence of quadratic approximations of this program. This algorithm has a complexity of $O(m^{1/2} \cdot L)$ iterations and $O(m^{3.5} \cdot L)$ arithmetic operations, where m is the number of variables and L is the size of the problem encoding in binary. The novel feature of this algorithm is that it admits a very simple proof of its complexity, which makes it valuable both as a teaching and as a research tool. The proof uses a new Lyapunov function to measure the duality gap, which has itself interesting properties that can be used in a line search procedure to accelerate convergence. If the cost is separable, the line search is particularly simple to implement and, if the cost is linear, the line search stepsize is obtainable in a closed form. This algorithm maintains both primal and dual feasibility at all iterations.

KEY WORDS: quadratic program, Karmarkar method, quadratic approximation, logarithmic penalty functions

¹ This research is partially supported by the U.S. Army Research Office, contract DAAL03-86-K-0171 (Center for Intelligent Control Systems), and by the National Science Foundation, grant NSF-ECS-8519058.

² Center for Intelligent Control Systems, Room 35-205, Massachusetts Institute of Technology, Cambridge, MA 02139.

Acknowledgement: Thanks are due to Robert Freund and Tom Luo for their helpful comments.

1. Introduction

Consider quadratic programming problems of the form

$$\begin{aligned} \text{Minimize} \quad & \langle x, Qx \rangle / 2 + \langle c, x \rangle & (P) \\ \text{subject to} \quad & Ax = b, \quad x \geq 0, \end{aligned}$$

where Q is an $m \times m$ symmetric, positive semi-definite matrix, c is an m -vector, A is an $n \times m$ matrix, b is an n -vector, and $\langle \cdot, \cdot \rangle$ denotes the usual Euclidean inner product. In our notation, all vectors are column vectors and superscript T denotes the transpose. We will denote by \mathfrak{R}^m (\mathfrak{R}^n) the m -dimensional (n -dimensional) Euclidean space.

For any vector x in \mathfrak{R}^m , we will denote by x_j the j th component of x . For any positive vector x in \mathfrak{R}^m , we will denote by D_x the $m \times m$ positive diagonal matrix whose j th diagonal entry is the j th component of x . Let S denote the relative interior of the feasible set for (P), i.e.

$$S = \{ x \in \mathfrak{R}^m \mid Ax = b, \quad x > 0 \}.$$

We will also denote by e the vector in \mathfrak{R}^m all of whose components are 1's. "Log" will denote the natural log and $\|\cdot\|_1$, $\|\cdot\|_2$ will denote, respectively the L_1 -norm and the L_2 -norm. We make the following standing assumption about (P):

Assumption A:

- (a) Both S and $\{ u \in \mathfrak{R}^n \mid A^T u < c \}$ are nonempty.
- (b) A has full row rank.

Assumption A (b) is made only to simplify the analysis and can be removed without affecting either the algorithm or the convergence results. Note that Assumption A (a) implies (cf. [3], Corollary 29.1.5) that the set of optimal solutions for (P) is nonempty and bounded. For any $\epsilon > 0$, consider the following approximation of (P):

$$\begin{aligned} \text{Minimize} \quad & f_\epsilon(x) & (P_\epsilon) \\ \text{subject to} \quad & Ax = b, \end{aligned}$$

where we define $f_\varepsilon: (0, \infty)^m \rightarrow \mathfrak{R}$ to be the penalized function:

$$f_\varepsilon(\mathbf{x}) = \langle \mathbf{x}, \mathbf{Q}\mathbf{x} \rangle / 2 + \langle \mathbf{c}, \mathbf{x} \rangle - \varepsilon \sum_j \log(x_j). \quad (1.1)$$

Note that

$$\nabla f_\varepsilon(\mathbf{x}) = \mathbf{Q}\mathbf{x} + \mathbf{c} - \varepsilon \begin{bmatrix} 1/x_1 \\ \vdots \\ 1/x_m \end{bmatrix}, \quad \nabla^2 f_\varepsilon(\mathbf{x}) = \mathbf{Q} + \varepsilon \begin{bmatrix} 1/(x_1)^2 & \dots & 0 \\ & \vdots & \\ 0 & \dots & 1/(x_m)^2 \end{bmatrix}. \quad (1.2)$$

The first polynomial-time algorithm for (P) was given by Kozlov, Tarasov and Khachiyan [16] and was based on the ellipsoid method [17], [18]. In this note we propose another polynomial-time algorithm for (P) that is motivated by Karmarkar's method [1] and its interpretation as a projected Newton method based on the logarithmic barrier function [19]. Our approach, which is similar to that taken in [10]-[13], is to solve (approximately) a sequence of problems $\{(P_{\varepsilon^r})\}$, where $\{\varepsilon^r\}$ is a geometrically decreasing sequence of positive scalars. The approximate solution of (P_{ε^r}) , denoted by \mathbf{x}^r , is obtained by solving the quadratic approximation of $(P_{\varepsilon^{r-1}})$ around \mathbf{x}^{r-1} . The novel feature of our algorithm is the simplicity of its analysis, and yet it has an excellent complexity. Our algorithm scales using only primal solutions and, in this respect, it is closely related to the algorithms of [11], [13]. However, it differs from the latter in its choice of starting points and the choice of parameters. This difference, as we shall see, gives rise to a much different (but much simpler and sharper) analysis and reveals more of the algorithmic structure. In particular, convergence is based on the use of a certain Lyapunov function that measures the amount by which the complementary slackness condition is violated. This function has itself interesting properties that can be used in a line search procedure to accelerate convergence. For linear programming problems ($\mathbf{Q} = 0$), this line search procedure is particularly simple. Such line search feature is not previously known for this class of methods.

This note proceeds as follows: in §2 we show that, given an approximately-optimal primal dual pair of (P_ε) , an approximately-optimal primal dual pair of $(P_{\alpha\varepsilon})$, for some $\alpha \in (0, 1)$, can be obtained by solving a quadratic approximation of (P_ε) . In §3 and §4 we present our algorithm and analyze its convergence. In §5 we discuss the initialization of our algorithm. In §6 we discuss extensions.

2. Technical Preliminaries

Fix any $\bar{\epsilon} > 0$ and consider the problem (P $_{\bar{\epsilon}}$)

$$\begin{aligned} \text{Minimize} \quad & \langle x, Qx \rangle / 2 + \langle c, x \rangle - \bar{\epsilon} \sum_j \log(x_j) \\ \text{subject to} \quad & Ax = b. \end{aligned}$$

Let \bar{x} be any element of S and let \bar{u} be any element of \mathfrak{R}^n . We replace the objective function by its quadratic approximation around \bar{x} . This gives (cf. (1.2))

$$\begin{aligned} \text{Minimize} \quad & \langle Qx + c - \bar{\epsilon}(\bar{X})^{-1}e, z \rangle + \langle z, (Q + \bar{\epsilon}(\bar{X})^{-2})z \rangle / 2 \\ \text{subject to} \quad & Az = 0, \end{aligned}$$

where we let $x = \bar{x} + z$ and $\bar{X} = D_{\bar{x}}$. The Karush-Kuhn-Tucker point for this problem, say (z, u) , satisfies

$$Q\bar{x} + c - \bar{\epsilon}(\bar{X})^{-1}e + (Q + \bar{\epsilon}(\bar{X})^{-2})z - A^T u = 0, \quad (2.1a)$$

$$Az = 0. \quad (2.1b)$$

Let $d = (\bar{X})^{-1}z$. Then Eqs. (2.1a)-(2.1b) can be rewritten as

$$(\bar{\epsilon}I + \bar{X}Q\bar{X})d - (A\bar{X})^T(u - \bar{u}) = \bar{r},$$

$$A\bar{X}d = 0,$$

where we let $\bar{r} = \bar{\epsilon}e - \bar{X}(Q\bar{x} + c - A^T\bar{u})$. Solving for d gives

$$\bar{\Gamma}^{1/2}d = [I - \bar{\Gamma}^{-1/2}\bar{X}A^T[A\bar{X}\bar{\Gamma}^{-1}\bar{X}A^T]^{-1}A\bar{X}\bar{\Gamma}^{-1/2}]\bar{\Gamma}^{-1/2}\bar{r}, \quad (2.2)$$

where we let $\bar{\Gamma} = \bar{\epsilon}I + \bar{X}Q\bar{X}$. Since the orthogonal projection is a nonexpansive mapping (with respect to the L_2 -norm), we have from (2.2) that

$$\|\bar{\Gamma}^{1/2}d\|_2 \leq \|\bar{\Gamma}^{-1/2}\bar{r}\|_2. \quad (2.3)$$

Let

$$x = \bar{x} + z, \quad (2.4)$$

and denote $X = D_x$, $\Delta = D_d$. Then $X = \bar{X} + \Delta\bar{X}$ and hence

$$\begin{aligned} \bar{\epsilon}e - X(Qx + c - A^T u) &= \bar{\epsilon}e - \bar{X}Q\bar{x} - \bar{X}Qz - \bar{X}c + \bar{X}A^T u \\ &\quad - \Delta\bar{X}Q\bar{x} - \Delta\bar{X}Qz + \Delta[-\bar{X}c + \bar{X}A^T u] \\ &= \bar{\epsilon}d - \Delta\bar{X}Q\bar{x} - \Delta\bar{X}Qz + \Delta[-\bar{X}c + \bar{X}A^T u] \\ &= \Delta[\bar{\epsilon}e - \bar{X}Q\bar{x} - \bar{X}Qz - \bar{X}c + \bar{X}A^T u] \\ &= \bar{\epsilon}\Delta d, \end{aligned}$$

where the second and the fourth equality follow from (2.1a). This implies that

$$\begin{aligned} \|\bar{\epsilon}e - X(Qx + c - A^T u)\|_2 &= \bar{\epsilon}\|\Delta d\|_2 \\ &\leq \bar{\epsilon}\|\Delta d\|_1 \\ &= \bar{\epsilon}(\|d\|_2)^2 \\ &\leq (\|\bar{\Gamma}^{1/2}d\|_2)^2 \\ &\leq (\|\bar{\Gamma}^{-1/2}r\|_2)^2, \end{aligned} \quad (2.5)$$

where the first inequality follows from properties of the L_1 -norm and the L_2 -norm, the second inequality follows from the fact that the eigenvalues of $\bar{\Gamma}^{-1}$ do not exceed $1/\bar{\epsilon}$, and the third inequality follows from (2.3).

Consider any $\beta \in (0,1)$ and any scalar α satisfying

$$(\beta^2 + m^{1/2})/(\beta + m^{1/2}) \leq \alpha < 1. \quad (2.6)$$

Let $\epsilon = \alpha\bar{\epsilon}$, $r = \epsilon e - X(Qx + c - A^T u)$, and $\Gamma = \epsilon I + XQX$. Then

$$\|\Gamma^{-1/2}r\|_2/\epsilon^{1/2} \leq \|r\|_2/\epsilon$$

$$\begin{aligned}
&= \|\alpha \bar{\epsilon} e - X(Qx + c - A^T u)\|_2 / (\alpha \bar{\epsilon}) \\
&\leq \|\bar{\epsilon} e - X(Qx + c - A^T u)\|_2 / (\alpha \bar{\epsilon}) + (1-\alpha) \cdot m^{1/2} / \alpha \\
&\leq (\|\Gamma^{-1/2} r\|_2)^2 / (\alpha \bar{\epsilon}) + (1/\alpha - 1) \cdot m^{1/2},
\end{aligned}$$

where the first inequality follows from the fact that the eigenvalues of Γ^{-1} do not exceed $1/\epsilon$, the second inequality follows from the triangle inequality, and the third inequality follows from (2.5). Hence, by (2.6),

$$\|\Gamma^{-1/2} r\|_2 \bar{\epsilon}^{1/2} \leq \beta \quad \Rightarrow \quad \|\Gamma^{-1/2} r\|_2 / \epsilon^{1/2} \leq \beta. \quad (2.7)$$

From (2.3) we also have $\|d\|_2 \leq \beta < 1$. Hence $e+d > 0$ and (cf. (2.4)) $x > 0$. Also, by (2.1b) and (2.4), $Ax = A(\bar{x}+z) = b$. Furthermore, from (2.1a) we have that $Qx + c - A^T u = \bar{\epsilon} \bar{X}^{-1}(e - d)$. Since (cf. $\|d\|_2 \leq \beta < 1$) $e - d > 0$, this implies that

$$0 < Qx + c - A^T u = \bar{\epsilon} \cdot X^{-1}(I+\Delta)(e - d) \leq \bar{\epsilon} \cdot X^{-1}e, \quad (2.8)$$

where the equality follows from the fact $\bar{X}^{-1} = X^{-1}(I+\Delta)$ and the second inequality follows from the observation that $(I+\Delta)(e - d) = e - \Delta d$. Hence u is dual feasible. [If the dual cost of u , i.e. $\langle b, u \rangle + \min\{ \langle y, Qy \rangle / 2 + \langle c - A^T u, y \rangle \mid y \in \mathfrak{R}^m, y \geq 0 \}$, is not finite, then there exists $w \in \mathfrak{R}^m$ such that $w \geq 0$, $Qw = 0$, and $\langle c - A^T u, w \rangle < 0$. Multiplying by w gives $0 < \langle Qx + c - A^T u, w \rangle = \langle c - A^T u, w \rangle < 0$, a contradiction.]

For any $\epsilon > 0$, let $\rho_\epsilon: (0, \infty)^m \times \mathfrak{R}^n \rightarrow [0, \infty)$ denote the function

$$\rho_\epsilon(y, p) = \|(\epsilon I + D_y Q D_y)^{-1/2} (\epsilon e - D_y(Qy + c - A^T p))\|_2 / \epsilon^{1/2}, \quad \forall y \in (0, \infty)^m, \forall p \in \mathfrak{R}^n.$$

We have then just proved the following important lemma (cf. (2.7)-(2.8)):

Lemma 1 For any $\beta \in (0, 1)$, any $\bar{\epsilon} > 0$ and any $(\bar{x}, \bar{u}) \in S \times \mathfrak{R}^n$ such that $\rho_{\bar{\epsilon}}(\bar{x}, \bar{u}) \leq \beta$, we have

$$(x,u) \in S \times \mathcal{R}^n, \quad \rho_{\alpha \bar{\epsilon}}(x,u) \leq \beta,$$

$$0 \leq Qx + c - A^T u \leq \bar{\epsilon} \cdot (D_x)^{-1} e,$$

where $\alpha = (\beta^2 + m^{1/2}) / (\beta + m^{1/2})$ and (x,u) is defined as in (2.1a), (2.1b), (2.4).

[The function $\rho_\epsilon(y,p)$ measures the amount by which the complementary slackness condition $D_y(Qy + c - A^T p) = 0$ is violated. It also has some nice properties which we will discuss in §6.]

3. The Homotopy Algorithm

Choose any $\beta \in (0,1)$ and let $\alpha = (\beta^2 + m^{1/2}) / (\beta + m^{1/2})$. Lemma 1 and (2.1a)-(2.1b), (2.4) motivate the following algorithm for solving (P), parameterized by two positive scalars $\gamma \leq \eta$:

Homotopy Algorithm

Step 0: Choose any $(x^1, u^1) \in S \times \mathcal{R}^n$ such that $\rho_\eta(x^1, u^1) \leq \beta$. Let $\epsilon^1 = \eta$.

Step r: Compute (z^{r+1}, u^{r+1}) to be a solution of

$$\begin{bmatrix} Q + \epsilon^r (D_{x^r})^{-2} & -A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} z \\ u \end{bmatrix} = \begin{bmatrix} \epsilon^r (D_{x^r})^{-1} e - Qx^r - c \\ 0 \end{bmatrix}.$$

Set $x^{r+1} = x^r + z^{r+1}$, $\epsilon^{r+1} = \alpha \epsilon^r$.

If $\epsilon^{r+1} \leq \gamma$, terminate.

We gave the above algorithm the name "homotopy" (or "path-following") because it solves (approximately) a sequence of problems $\{(P_{\epsilon^r})\}$ that approaches (P) (see [2]). Note that u^r is dual feasible for all r (cf. Lemma 1).

4. Convergence Analysis

By Lemma 1, the homotopy algorithm generates, in at most $(\log(\gamma) - \log(\eta)) / \log(\alpha)$ steps, an $(x, u) \in S \times \mathcal{R}^n$ satisfying

$$\begin{aligned} 0 &\leq Qx + c - A^T u \leq \gamma (D_x)^{-1} e, \\ Ax &= b. \end{aligned} \tag{4.1}$$

Eq. (4.1) implies that

$$\begin{aligned} 0 &\leq \langle Q_j, x \rangle + c_j - \langle A_j, u \rangle && \text{if } x_j \leq \gamma^{1/2}, \\ 0 &\leq \langle Q_j, x \rangle + c_j - \langle A_j, u \rangle \leq \gamma^{1/2} && \text{otherwise,} \end{aligned}$$

where Q_j denotes the j th column of Q , c_j denotes the j th component of c , and A_j denotes the j th column of A . Also, since \log is a concave function and its slope at 1 is 1, we have that $\log(1 - \delta) \leq -\delta$, for any $\delta \in (0, 1)$. Therefore

$$\begin{aligned} \log(\alpha) &= \log(1 - \beta(1 - \beta) / (\beta + m^{1/2})) \\ &\leq -\beta(1 - \beta) / (\beta + m^{1/2}). \end{aligned}$$

Hence we have just proved following:

Lemma 2 For any $\beta \in (0, 1)$ and any positive scalars $\gamma \leq \eta$, the homotopy algorithm generates, in at most $(\log(\eta) - \log(\gamma)) \cdot (\beta + m^{1/2}) / \beta(1 - \beta)$ steps, a pair of optimal primal and dual solutions to a perturbed problem of (P), where the linear cost coefficients are perturbed by at most $\gamma^{1/2}$ and the lower bounds are perturbed by at most $\gamma^{1/2}$.

Hence if we choose $\beta = 1/2$, $\eta = 2^{\lambda L}$ and $\gamma = 2^{-\lambda L}$, where L denotes the size of the problem encoding in binary (defined as in [10]-[14]), for some constant λ sufficiently large, the homotopy algorithm would terminate in $O(m^{1/2} \cdot L)$ steps with an optimal primal dual solution pair to a

perturbed problem of (P) and the size of the perturbation is $2^{-\Theta(L)}$. An optimal primal dual solution pair for (P) can be recovered by using, say, the techniques described in [10] (also see [1]). Since the amount of computation per step is at most $O(m^3)$ arithmetic operations (not counting Step 0), the homotopy algorithm has a complexity of $O(m^{3.5}L)$ arithmetic operations. [We assume for the moment that Step 0 can be done very "fast". See §5 for justification.] It may be possible to reduce the complexity to $O(m^3L)$ by using the rank-one update technique described in [1], [5], [11].

5. Algorithm Initialization

In this section we show that, for η sufficiently large, Step 0 of the homotopy algorithm (i.e. to generate a primal dual pair $(x,u) \in S \times \mathcal{R}^n$ satisfying $\rho_\eta(x,u) \leq \beta$) can be done very "fast".

Suppose that (P) is in the canonical form considered by Karmarkar (see §5 of [1] for details on how to transform general convex quadratic programs into this canonical form). We claim that, for $\eta = \|Qe + c\|_2 / \beta$, a point $(x,u) \in S \times \mathcal{R}^n$ satisfying $\rho_\eta(x,u) \leq \beta$ can be found immediately. To see this, note that in Karmarkar's canonical form, A and b have the form

$$A = \begin{bmatrix} A' \\ e^T \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ m \end{bmatrix},$$

where A' is some $(n-1) \times m$ matrix, and the point e is assumed to satisfy $Ae = b$. Let $x = e$ and $p = (0, \dots, 0, -1)^T$. Then

$$\begin{aligned} e + (AD_x)^T p &= e + A^T p \\ &= e - e = 0. \end{aligned} \tag{5.1}$$

Hence, by the triangle inequality, we have that

$$\begin{aligned} \rho_\eta(e, \eta p) &= \|(\eta I + Q)^{-1/2} (-Qe - c + \eta e + \eta A^T p)\|_2 / \eta^{1/2} \\ &\leq \|Qe + c\|_2 / \eta + \|e + A^T p\|_2 \end{aligned}$$

$$= \beta.$$

Alternatively, we can solve the problem (P_η) , whose Karush-Kuhn-Tucker point (x,p) can be seen to satisfy $\rho_\eta(x,p) = 0$ (such a point exists since the optimal solution set for (P) is bounded). If the feasible set for (P) is bounded, we can instead solve the following problem

$$\begin{aligned} &\text{Maximize } \sum_j \log(x_j) && (5.2) \\ &\text{subject to } Ax = b, \end{aligned}$$

whose Karush-Kuhn-Tucker point (x,p) can be seen to satisfy (5.1). Then, for $\eta = \beta \|D_x Qx + D_x c\|_2$, we also have $\rho_\eta(x,\eta p) \leq \beta$. Polynomial-time algorithms for solving (5.2) are described in [7] and [8]. [Note that neither (P_η) nor (5.2) need to be solved exactly.]

6. Conclusion and Extensions

In this note we have proposed an algorithm for convex quadratic programming and have provided a short proof of its complexity. This algorithm solves a sequence of approximations to the original problem, each augmented by a logarithmic penalty function. This algorithm maintains primal and dual feasibility and has a complexity (in terms of the number of steps) that is comparable to existing interior point methods for convex quadratic programming (see [10]-[14]).

There are many directions in which our results can be extended. For example, we can accelerate the rate of convergence of the homotopy algorithm by setting ϵ^r to be the smallest positive ϵ for which $\rho_\epsilon(x^r, u^r) \leq \beta$. This minimization is difficult in general, but if Q is diagonal, it can be verified that the quantity $\epsilon \rho_\epsilon(x^r, u^r)^2 = \sum_j (\epsilon - v_j)^2 / (\epsilon + q_j)$ is convex in ϵ (the second derivative is nonnegative), where v_j denotes the j th component of $D_{x^r}(Qx^r + c - A^T u^r)$ and q_j denotes the j th diagonal entry of $D_{x^r} Q D_{x^r}$. Hence in this case the above minimization reduces to finding a solution ϵ of the equation

$$\sum_j (\epsilon - v_j)^2 / (\epsilon + q_j) = \epsilon \beta^2. \quad (6.1)$$

Because the lefthand side is convex in ϵ , such a solution can be found using simple line search techniques (see [15]). For linear programming (i.e. $Q = 0$) (6.1) further reduces to the scalar quadratic equation

$$\sum_j (\epsilon - v_j)^2 = \epsilon^2 \beta^2, \quad (6.2)$$

whose solution is unique and is obtainable in a closed form. In fact, even for general Q , the solution of (6.2) is at least as good as $\alpha \epsilon^{r-1}$. [This follows from the observation that $\rho_\epsilon(x^r, u^r) \leq \|e - v\|_2 / \epsilon$ for all $\epsilon > 0$, where v denotes the m -vector whose j th component is v_j , and that (cf. proof of Lemma 1) $\| \alpha \epsilon^{r-1} e - v \|_2 / (\alpha \epsilon^{r-1}) \leq \beta$.]

We can also choose β to minimize α (this gives $\alpha = 2(m+m^{1/2})^{1/2} - 2m^{1/2}$). Also, Freund [9] noted that, at the r th step, one can take a quadratic approximation for $(P_{\alpha \epsilon^r})$ instead of for (P_{ϵ^r}) . The resulting analysis is slightly different, but achieves the same complexity. Other possible extensions include complexity reduction and the extension to problems with upper bound constraints or with general convex costs.

References

- [1] Karmarkar, N., "A New Polynomial-Time Algorithm for Linear Programming," *Combinatorica*, 4 (1984), 373-395.
- [2] Garcia, C. B. and Zangwill, W. I., *Pathways to Solutions, Fixed Points, and Equilibria*, Prentice-Hall, Inc., N. J. (1981).
- [3] Rockafellar, R. T., *Convex Analysis*, Princeton University Press, Princeton, N.J. (1970).
- [4] Bertsekas, D. P., *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press (1982).
- [5] Gonzaga, C. C., "An Algorithm for Solving Linear Programming Problems in $O(n^3L)$ Operations," Memo. UCB/ERL M87/10, Electronics Research Laboratory, University of California, Berkeley, CA (March 1987).
- [6] Strassen, V., "Gaussian Elimination is not Optimal," *Numerische Mathematik*, 13 (1969), 354-356.
- [7] Sonnevend, G., "An 'Analytical Center' for Polyhedrons and New Classes of Global algorithms for Linear (Smooth, Convex) Programming," Preprint, Dept. of Numerical Analysis, Institute of Mathematics Eotvos University, 1088, Budapest, Muzeum Korut, 6-8 (1985).
- [8] Freund, R. M., "Projective Transformations for Interior Point Methods, Part II: Analysis of an Algorithm for finding the Weighted Center of a Polyhedral System," OR 180-88, Operations Research Center, M.I.T., Cambridge, MA (June 1988).
- [9] Freund, R. M., Private Communication (September 1988).
- [10] Monteiro, R. C., and Adler, I., "An $O(n^3L)$ Interior Point Algorithm for Convex Quadratic Programming," Technical Report No. ORC 87-15, Department of Industrial Engineering and Operations Research, University of California, Berkeley, CA (June 1987).
- [11] Goldfarb, D. and Liu, S., "An $O(n^3L)$ Primal Interior Point Algorithm for Convex Quadratic Programming," Technical Report, Dept. of IEOR, Columbia University, N.Y. (1988).
- [12] Kojima, M., Mizuno, S. and Yoshise, A., "A Polynomial-Time Algorithm for a Class of

Linear Complementarity Problems," Report No. B-193, Dept. of Information Sciences, Tokyo Institute of Technology, Tokyo, Japan (1987).

- [13] Ye, Y., "Further Development on the Interior Algorithm for Convex Quadratic Programming," Manuscript, Dept. of Engineering-Economic Systems, Stanford University, CA (1987).
- [14] Mehrotra, S. and Sun, J., "An Algorithm for Convex Quadratic Programming that Requires $O(n^{3.5}L)$ Arithmetic Operations," Dept. of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL (February 1988).
- [15] Luenberger, D. G., *Introduction to Linear and Nonlinear Programming*," Addison-Wesley, MA (1973).
- [16] Kozlov, M. K., Tarasov, S. P., and Khachiyan, L. G., "Polynomial Solvability of Convex Quadratic Programming," *Doklady Akademii Nauk SSSR*, 248 (1979) (translated in *Soviet Mathematics Doklady*, 20 (1979), 1108-111).
- [17] Shor, N. Z., "Utilization of the Operation of Space Dilation in the Minimization of Convex Functions," *Kibernetika*, 6 (1970), 6-12 (translated in *Cybernetics*, 13 (1970), 94-96).
- [18] Yudin, D. B. and Nemirovskii, A. S., "Informational Complexity and Effective Methods of Solution for Convex Extremal Problems," *Ekonomika i Matematicheskie Metody*, 12 (1976), 357-369 (translated in *Matekon*, 13 (1977), 25-45).
- [19] 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.