On Numerical Solution of the Maximum Volume Ellipsoid Problem*

Yin Zhang† Liyan Gao‡

August, 2001

(CAAM Technical Report TR01-15)

Abstract

In this paper we study practical solution methods for finding the maximum-volume ellipsoid inscribing a given full-dimensional polytope in $\Re^n$ defined by a finite set of linear inequalities. Our goal is to design a general-purpose algorithmic framework that is reliable and efficient in practice. To evaluate the merit of a practical algorithm, we consider two key factors: the computational cost per iteration and the typical number of iterations required for convergence. In addition, numerical stability is also an important factor. We investigate some new formulations upon which we build primal-dual type, interior-point algorithms, and we provide theoretical justifications for the proposed formulations and algorithmic framework. Extensive numerical experiments have shown that one of the new algorithms should be the method of choice among the tested algorithms.

1 Introduction

The ellipsoidal approximation of polytopes is an important problem in its own right while it is also a basic subroutine in a number of algorithms for different problems. One example is that Lenstra's algorithm for the integer programming feasibility problem [11, 12] uses the ellipsoidal approximation of polytopes as a subroutine.

Consider a full-dimensional polytope $\mathcal{P} \in \Re^n$ defined by m linear inequalities. For brevity, we will call the problem of finding the maximum volume ellipsoid inscribing $\mathcal{P}$ the MaxVE problem. The MaxVE problem has its root in the rounding of convex bodies in $\Re^n$. One of the earliest studies was done by F. John [6]. In particular, John's results imply that once the maximum-volume, inscribing ellipsoid $\mathcal{E}$ is found in $\mathcal{P}$, then $\mathcal{E} \subset \mathcal{P} \subset n\mathcal{E}$, where $n\mathcal{E}$ is the ellipsoid resulting from dilating $\mathcal{E}$ by a factor n about its center. Such a pair of ellipsoids is also called a *Löwner-John pair* of $\mathcal{P}$. That is, $\mathcal{E}$ provides an n-bounding for $\mathcal{P}$.

*This research was supported in part by NSF Grants DMS-9973339 and DMS-9872009, DOE Grant DE-FG03-97ER25331 and DOE/LANL Contract 03891-99-23.

†Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005, USA. Email: yzhang@rice.edu.

‡Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005, USA. Email: gaoly@rice.edu.

Moreover, if $\mathcal{P}$ is centrally symmetric around the origin, then the rounding factor can be reduced to $\sqrt{n}$.

Ellipsoids have good geometric and computational properties that make them much easier to handle, both theoretically and computationally, than polytopes. For example, the global minimum of any quadratic in an ellipsoid can be located in polynomial time, while finding such a global minimum in a polytope is generally an NP-hard problem. For many problems a fruitful and effective approach is to use ellipsoids to approximate polytopes in various theoretic and algorithmic settings. A celebrated example is Khachiyan's ellipsoid method [8] – the first polynomial-time algorithm for linear programming. Other applications include optimal design [18, 20], computational geometry (for example, [22]) and algorithm construction (for example, [3] and [19]).

Recently, several randomized polynomial-time algorithms ([2, 7, 13], for example) have been proposed for approximating the volume of convex bodies (computing the volume itself is NP-hard). In the case of a polytope, these algorithms require approximating the polytope by an ellipsoid.

It is known that the rounding of a polytope can be accomplished by the (shallow-cut) ellipsoid method in polynomial time (see, for example, [17, 3]). It is also known, however, that the ellipsoid method is not a practically efficient algorithm. A number of interior-point algorithms have been proposed in recent years for the maximum volume ellipsoid problems, for example, by Nesterov and Nemirovskii [15], Khachiyan and Todd [10] (also see [9] for a related problem), Nemirovskii [14], and Anstreicher [1].

Nesterov and Nemirovskii [15] constructed a three-stage barrier method for finding an $\epsilon-$optimal ellipsoid $\mathcal{E}$ such that its volume $\mathrm{Vol}(\mathcal{E}) \geq \mathrm{Vol}(\mathcal{E}^*)e^{-\epsilon}$, where $\mathcal{E}^*$ is the maximum volume ellipsoid inscribing $\mathcal{P}$ and $\epsilon \in (0,1)$, with the complexity estimate $O(m^{2.5}(n^2 + m)\ln(\frac{mR}{\epsilon}))$ where m is the number of constraints and R is a priori known ratio of radius of two concentric balls, the larger ball containing the given polytope $\mathcal{P}$ and the smaller one being contained in $\mathcal{P}$. The term n^2 comes from the requirement of solving linear systems involving an $n \times n$ matrix-valued variable.

Khachiyan and Todd [10] proposed an algorithm that attains the complexity estimate of $O(m^{3.5}\ln(\frac{mR}{\epsilon})\ln(\frac{n\ln R}{\epsilon}))$. The algorithm applies the basic barrier method to a small number of subproblems and only requires solving linear systems of $n + m$ equations to compute the involved Newton directions. In their formulation the matrix-valued variable is explicitly treated as dependent on another vector-valued variable during the solution of Newton linear systems.

Nemirovskii [14] showed that the maximum volume ellipsoid problem can be reformulated as a saddle-point problem of $m+n$ variables and be solved by a path-following method for approximating saddle points of a sequence of self-concordant convex-concave functions as defined in [14]. Nemirovskii proved that the complexity of the algorithm is $O(m^{3.5}\ln(\frac{mR}{\epsilon}))$.

Most Recently, Anstreicher [1] proposed an algorithm that uses key ideas of Khachiyan and Todd [10] but avoids solving the subproblems required in the Khachiyan and Todd algorithm. This way, Anstreicher's algorithm attains the complexity estimate of $O(m^{3.5}\ln(\frac{mR}{\epsilon}))$, which is the same as in [14]. Anstreicher also showed that computing an approximate analytic center of the polytope can reduce the complexity to $O((mn^2 + m^{1.5}n)\ln(R) + m^{3.5}\ln(\frac{m}{\epsilon}))$.

In addition, Vandenberghe, Boyd and Wu [21] proposed an algorithm for the class of problems called MAXDET problems to which the MaxVE problem belongs. However, their algorithm does not take into account the special structure of the MaxVE problem.

All the aforementioned works are primarily concerned with the complexity issues and the proposed algorithms are theoretical in nature. On the contrary, the objective of the present study is to identify or construct a numerically efficient and stable algorithm for solving general MaxVE problems. Our study is not aimed at solving very large-scale problems, so we will not consider aspects of exploiting sparsity and other special structures that may be present in the polytope-defining inequalities.

Since for many convex programs, primal-dual interior-point algorithms have proven to be superior in practice than either primal or dual algorithms, we will mainly investigate primal-dual type algorithms, though we will also consider particular primal algorithms for the purpose of comparison.

Two features are common in all the known interior-point algorithms for solving the MaxVE problem. First, they are iterative in nature. Second, they require solving a linear system at each iteration to update the current iterate. Hence, in judging the practical efficiency of an algorithm, we must consider two key factors: (i) how many iterations the algorithm typically requires in practice for obtaining an approximate solution of a certain quality; and (ii) how expensive it is to solve the relevant linear system at each iteration. Besides efficiency, another important consideration is the robustness of the algorithm. The robustness of an iterative algorithm is often determined by the numerical stability of the solution procedure for linear systems that has to be invoked at every iteration.

In most primal-dual algorithms for linear programming or semidefinite programming, at each iteration one solves a large linear system by reducing it to a smaller Schur complement system through a block Gaussian elimination. Moreover, the coefficient matrix in the Schur complement system is often positive definite. This procedure has proven to be efficient and at the same time adequately stable. Likewise, in this paper we will try to identify primal-dual algorithms for which the corresponding linear systems can be reduced by block Gauss elimination to a well-behaved Schur complement system.

The paper is organized as follows. In section 2 we describe the formulation of the MaxVE problem. We introduce some primal-dual type interior-point algorithm in Section 3 and give related theoretical results in Section 4. We summarize the Khachiyan and Todd algorithm and our modification in Section 5. Numerical comparative results on these four algorithms are presented in Section 6 with detailed numerical data attached in the Appendix. Finally, we offer some concluding remarks in Section 7.

We now introduce some notation. For any given vector $v \in \Re^p$, we denote the $p \times p$ diagonal matrix with v on its diagonal either by $\mathrm{Diag}(v)$, or by its upper-case letter V whenever no confusion can occur. On the other hand, for a square matrix M, $\mathrm{diag}(M)$ is the vector formed by the diagonal of M. The Hadamard product is represented by the small circle "$\circ$". Unless specified otherwise, superscripts for vectors and subscripts for scalars, that are not elements of a matrix, are iteration counts. For a vector v, inequalities of the form $v > \alpha$ are interpreted as component-wise where α can be a scalar or a vector of the same size. For symmetric matrices, $A \succ B$, or equivalently $A - B \succ 0$, means that $A - B$ is positive definite. We use $\Re^m_+$ and $\Re^m_{++}$ to represent the nonnegative and positive orthants in $\Re^m$, respectively. The notation $\mathcal{S}^n_{++}$ represents the subspace of all symmetric positive definite matrices in $\Re^{n\times n}$. For a set $\mathcal{W}$ in $\Re^m$, we denote its closure by $\mathrm{col}(\mathcal{W})$. Finally, by default $\|\cdot\|$ represents the Euclidean norm unless otherwise specified.

2 The Maximum Volume Ellipsoid Problem

Consider a polytope $\mathcal{P}$ in $\Re^n$ given by

$$\mathcal{P} = \{v \in \Re^n : Av \le b\}, \tag{1}$$

where $A \in \Re^{m\times n}$, $m > n$ and $b \in \Re^m$. Recall that by definition, a polytope is a bounded polyhedron. For convenience of discussion, we will make the following two assumptions throughout the paper:

A1. The matrix A has full rank n and contains no zero-rows.

A2. There exists a strictly interior point $\bar{v} \in \mathcal{P}$ satisfying $A\bar{v} < b$.

Given a center $x \in \Re^n$ and a nonsingular scaling matrix $E \in \Re^{n\times n}$, an ellipsoid in $\Re^n$ centered at x can be defined as

$$\mathcal{E}(x, E) = \{v \in \Re^n : (v - x)^T (EE^T)^{-1}(v - x) \le 1\};$$

or equivalently,

$$\mathcal{E}(x, E) = \{v \in \Re^n : v = x + Es \text{ and } \|s\| \le 1\}, \tag{2}$$

where $\|\cdot\|$ is the Euclidean norm in $\Re^n$. Clearly, the shape of the ellipsoid is uniquely determined by the symmetric positive definite matrix EE^T, but not uniquely by E since the same ellipsoid can also be generated by EQ for any orthogonal matrix $Q \in \Re^{n\times n}$. Without loss of generality, we can assume that E itself is symmetric positive definite. Therefore, the ellipsoid $\mathcal{E}(x, E)$ is uniquely determined by the center x and the scaling matrix $E \in \mathcal{S}^n_{++}$.

It is easy to see that the ellipsoid $\mathcal{E}(x, E)$ is contained in $\mathcal{P}$ if and only if

$$\sup_{\|s\|=1} a_i^T (x + Es) \le b_i, \quad i = 1, \cdots, m$$

where a_i^T is the i-th row of A; or equivalently

$$a_i^T x + \|Ea_i\| \le b_i, \quad i = 1, \cdots, m.$$

Introducing the notation

$$h(E) = (\|Ea_1\|, \cdots, \|Ea_m\|)^T \in \Re^m, \tag{3}$$

we have

$$\mathcal{E}(x, E) \subset \mathcal{P} \Longleftrightarrow b - Ax - h(E) \ge 0. \tag{4}$$

Let V_n be the volume of the n-dimensional unit ball, then the volume of the ellipsoid $\mathcal{E}(x, E)$ defined in (2) is

$$\mathrm{Vol}(\mathcal{E}) \equiv V_n \det E.$$

It is evident that $\mathcal{E}(x^*, E^*)$ is the maximum-volume ellipsoid contained in $\mathcal{P}$ if and only if $(x^*, E^*) \in \Re^n \times \mathcal{S}^n_{++}$ solves the following optimization problem:

$$\begin{array}{cc} \min & -\log\det E \\ \text{s.t.} & b - Ax - h(E) \geq 0 \\ & (E \succ 0) \end{array} \tag{5}$$

where $E \succ 0$ means that E is symmetric positive definite. It is well known that the optimization problem (5) is a convex program with a unique pair of solution $(x^*, E^*) \in \Re^n \times \mathcal{S}^n_{++}$; and this solution is uniquely determined by the first-order optimality, or Karush-Kuhn-Tucker (KKT), conditions for the problem which can be derived as follows.

The Lagrangian function of the convex program (5) is

$$L(x, E, u) = -\log\det E - u^T(b - Ax - h(E))$$

where $u \in \Re^m$ is the vector of Lagrange multipliers. The KKT conditions consist of the equations $\nabla_x L = 0$, $\nabla_E L = 0$, feasibility and complementarity. Using the differentiation formulas

$$\nabla[\log\det E] = E^{-1} \quad \text{and} \quad \nabla h_i(E) = \frac{E a_i a_i^T + a_i a_i^T E}{2h_i(E)},$$

and introducing the notation $U := \mathrm{Diag}(u)$ and

$$Y \equiv Y(E, u) := \mathrm{Diag}(h(E))^{-1} U, \tag{6}$$

we can write the KKT conditions as

$$A^T u = 0, \tag{7a}$$
$$E^{-1} - [E(A^T Y A) + (A^T Y A)E]/2 = 0, \tag{7b}$$
$$z - (b - Ax - h(E)) = 0, \tag{7c}$$
$$Uz = 0, \tag{7d}$$
$$u, z \geq 0, \tag{7e}$$

where $E \succ 0$ and z is a slack variable.

3 Formulations and Primal-Dual Algorithms

In this section, we propose formulations and algorithms for effectively solving the MaxVE problem in practice. In constructing practically efficient algorithms, we consider the following three guidelines:

1. the algorithms should not carry the matrix-valued variable E as a completely independent variable because it would require too much computation;
2. the algorithms should be primal-dual algorithms because of their proven practical efficiency in numerous cases;
3. the algorithms should have theoretical guarantees to be well defined and well behaved.

The first objective above can be achieved by eliminating the matrix variable E. The elimination may occur either at the beginning of a formulation, or at the time of solving linear systems during iterations. In this paper, we will take the former approach.

3.1 Formulations without Matrix Variable

We now describe three new formulations, first proposed in [23], for the MaxVE problem which are free of the matrix variable E. The key idea in these formulations is to eliminate the matrix-valued variable E from the system by solving the equation (7b) for E. As can be verified easily, a solution to (7b) is

$$E(y) = (A^T Y A)^{-1/2}, \tag{8}$$

where $y = \mathrm{diag}(Y)$ and Y is defined in (6). We will later demonstrate that this solution is unique in $\mathcal{S}^n_{++}$. Upon the substitution of $E(y)$ into the definition of $h(y)$ (recall that $h_i(E) = \|Ea_i\|$), the vector $h(E)$ becomes a function of y that we will denote, with a slight abuse of notation, as $h(y)$; namely,

$$h(y) \equiv h(E(y)). \tag{9}$$

In [23], after substituting (8) and (9) into the KKT system, deleting (7b) and adding (6) written in a different form, i.e.,

$$u = g(y) := Y h(y), \tag{10}$$

the author obtained the following system:

$$F_0(x, y, u, z) = 0, \;\; y, u, z \geq 0, \tag{11}$$

where $x \in \Re^n$, $y, u, z \in \Re^m$, and the function $F_0 : \Re^{n+3m} \to \Re^{n+3m}$ is

$$F_0(x, y, u, z) = \begin{bmatrix} A^T u \\ Ax + h(y) + z - b \\ u - g(y) \\ Uz \end{bmatrix}. \tag{12}$$

Moreover, it is proposed in [23] to eliminate the variable u from the above system using the equation $u = g(y)$ in (10). The resulting system is

$$F_1(x, y, z) = 0, \;\; y, z \geq 0, \tag{13}$$

where the function $F_1 : \Re^{n+2m} \to \Re^{n+2m}$ is

$$F_1(x, y, z) = \begin{bmatrix} A^T g(y) \\ Ax + h(y) + z - b \\ Zg(y) \end{bmatrix}. \tag{14}$$

In (12) and (14), we have used the notation $U = \mathrm{Diag}(u)$ and $Z = \mathrm{Diag}(z)$, respectively.

In addition, the complementarity conditions $Uz = 0$ are clearly equivalent to the conditions $Yz = 0$ because $U = Y\mathrm{Diag}(h(y))$ and $h(y) > 0$ at the solution. Based on this observation, a third system is proposed in [23]:

$$F_2(x, y, z) = 0, \;\; y, z \geq 0, \tag{15}$$

where the function $F_2 : \Re^{n+2m} \to \Re^{n+2m}$ is

$$F_2(x,y,z) = \begin{bmatrix} A^T g(y) \\ Ax + h(y) + z - b \\ Yz \end{bmatrix}. \tag{16}$$

The three systems (11), (13) and (15) are all free of the matrix-valued variable E, which will form the bases for our algorithm construction[1]. However, in obtaining them we have applied nonlinear transformations whose properties need to be investigated. A most important question is whether or not these transformations preserve the uniqueness of solutions. We will answer this question and others in a subsequent section.

3.2 Primal-Dual Algorithmic Framework

The primal-dual algorithms to be proposed can be motivated from the view of the damped Newton's method applied to the so-called perturbed complementarity conditions. Another useful perspective is to view them as path-following algorithms. In this construction, one replaces the zero right-hand-side of relevant complementarity conditions by μw^0, where $\mu > 0$ and $w^0 \in \Re^m_{++}$, and applies the Newton method to the resulting "perturbed" system while decreasing the parameter μ to zero. Specifically, the perturbed systems for (13) and (15) have the form

$$F(x,y,z) = \begin{bmatrix} 0 \\ 0 \\ w \end{bmatrix}, \quad y, z > 0, \tag{17}$$

where F can be either F_1 or F_2, and for some $w^0 \in \Re^m_{++}$

$$w = \mu w^0, \quad \mu > 0.$$

Normally, one chooses $w^0 = e$ where e is vector of all ones.

We will prove later that each of the perturbed systems will have a unique solution for every $\mu > 0$, and as $\mu \to 0$ the corresponding solutions will converge to the (same) solution of the unperturbed systems from which the solution to the MaxVE problem can be easily constructed.

We now present our primal-dual interior-point algorithmic framework for the systems (13) and (15). The framework for the system (11) would be the same except that an extra variable $u \in \Re^m$ is present. In the rest of the paper, we will concentrate only on the formulations (13) and (15) but omit (11) because, being so closely related to (13), (11) shares almost identical theoretical properties with (13), while in our tests it seems to produce algorithms with inferior performance to that of their counterparts based on (13) and (15).

Algorithm 1 (Primal-Dual Interior-Point Algorithm).

Given $x^0 \in \mathcal{P}$ *and* $y^0, z^0 \in \Re^m_{++}$, *set* $k = 0$.

[1] In [23], some additional systems were also derived that we have found to be less satisfactory.

Step 1. *Choose* $\sigma_k \in (0,1)$*, set* μ_k *to* $\sigma_k \frac{g(y^k)^T z^k}{m}$ *for* $F = F_1$ *or to* $\sigma_k \frac{(y^k)^T z^k}{m}$ *for* $F = F_2$*.*

Step 2. *Solve for* (dx, dy, dz) *from*

$$F'(x^k, y^k, z^k) \begin{bmatrix} dx \\ dy \\ dz \end{bmatrix} = \mu_k \begin{bmatrix} 0 \\ 0 \\ e \end{bmatrix} - F(x^k, y^k, z^k). \tag{18}$$

Step 3. *Choose a step-length* $\alpha_k \in (0,1]$ *and update*

$$(x^{k+1}, y^{k+1}, z^{k+1}) = (x^k, y^k, z^k) + \alpha_k(dx, dy, dz),$$

such that $x^{k+1} \in \mathcal{P}$*,* $y^{k+1} > 0$ *and* $z^{k+1} > 0$*.*

Step 4. *If* $\|F(x^{k+1}, y^{k+1}, z^{k+1})\| \le \epsilon$*, stop; else increment* k *and go to Step 1.*

In addition to the initial guesses, this algorithmic framework has two essential parameters, σ_k and α_k, that need to be specified at each iteration. The main computation required is to solve the linear system (18) at every iteration.

When $F = F_1$, the coefficient matrix in the linear system (18), i.e., the Jacobian matrix of $F_1(x, y, z)$, is of the form

$$F_1'(x, y, z) = \begin{bmatrix} 0 & A^T g'(y) & 0 \\ A & h'(y) & I \\ 0 & Zg'(y) & \mathrm{Diag}(g(y)) \end{bmatrix}, \tag{19}$$

where $g'(y)$ and $h'(y)$ are the Jacobian matrices of $g(y)$ and $h(y)$, respectively. A direct differentiation shows that

$$g'(y) = \mathrm{Diag}(h(y)) + Yh'(y), \tag{20}$$

and (see also [23])

$$h'(y) = -\mathrm{Diag}(2h(y))^{-1}[Q(y) \circ Q(y)], \tag{21}$$

where

$$Q(y) = A(A^T Y A)^{-1} A^T. \tag{22}$$

It is worth noting that $Y^{1/2} Q(y) Y^{1/2}$ is an orthogonal projection matrix.

On the other hand, when $F = F_2$ we have

$$F_2'(x, y, z) = \begin{bmatrix} 0 & A^T g'(y) & 0 \\ A & h'(y) & I \\ 0 & Z & Y \end{bmatrix}. \tag{23}$$

An efficient way to solve the linear system (18) is the following block Gaussian elimination procedure: first eliminating dz and dy, then solving for dx, finally computing dy and

dz by back substitutions. We now formally describe the procedure for $F = F_1$. To simplify notation, we define three $m \times m$ matrices:

$$H \equiv H(y) := \mathrm{Diag}(h(y)), \quad N \equiv N(y) := g'(y), \tag{24}$$

and

$$M_1 \equiv M_1(y,z) := -h'(y) + [YH(y)]^{-1} ZN(y). \tag{25}$$

For now we will assume that M_1 is nonsingular, and we will prove this fact later.

The aforementioned block Gaussian elimination reduces $F_1'(x,y,z)$ into a lower triangular matrix, which is equivalent to, when $F = F_1$, pre-multiplying the equation (18) by the upper triangular elimination matrix

$$T_1(y,z) = \begin{bmatrix} I & A^T N M_1^{-1} & -A^T N M_1^{-1}(YH)^{-1} \\ 0 & I & -(YH)^{-1} \\ 0 & 0 & I \end{bmatrix}.$$

It is straightforward to verify that

$$T_1(y,z) F_1'(x,y,z) = \begin{bmatrix} A^T N M_1^{-1} A & 0 & 0 \\ A & -M_1 & 0 \\ 0 & ZN & YH \end{bmatrix}, \tag{26}$$

and for any vectors $r_1 \in \Re^n$ and $r_2, r_3 \in \Re^m$

$$T_1(y,z) \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \begin{pmatrix} r_1 + A^T N M_1^{-1} \left(r_2 - (YH)^{-1} r_3 \right) \\ r_2 - (YH)^{-1} r_3 \\ r_3 \end{pmatrix}. \tag{27}$$

Clearly, the linear system

$$F_1'(x,y,z) \begin{pmatrix} dx \\ dy \\ dz \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}$$

is equivalent to the linear system where the coefficient matrix is the one in (26) and the right-hand side is that of (27). This linear system can be formally solved by the procedure:

$$\begin{aligned} dx &= [A^T N M_1^{-1} A]^{-1} \left(r_1 + A^T N M_1^{-1} (r_2 - (YH)^{-1} r_3) \right), && \text{(28a)} \\ dy &= -M_1^{-1} \left(r_2 - (YH)^{-1} r_3 - A dx \right), && \text{(28b)} \\ dz &= (YH)^{-1} (r_3 - ZN dy). && \text{(28c)} \end{aligned}$$

This solution procedure requires $\mathcal{O}(m^3)$ operations (recall that $m > n$), with the bulk of the computation involving the $m \times m$ matrix M_1.

In a similar fashion, the linear system (18) corresponding to $F = F_2$ can be formally solved by the following procedure:

$$\begin{aligned} dx &= [A^T N M_2^{-1} A]^{-1} (r_1 + A^T N M_2^{-1} \left(r_2 - Y^{-1} r_3 \right)), && \text{(29a)} \\ dy &= -M_2^{-1} \left(r_2 - Y^{-1} r_3 - A dx \right), && \text{(29b)} \\ dz &= Y^{-1} (r_3 - Z dy), && \text{(29c)} \end{aligned}$$

where

$$M_2 \equiv M_2(y,z) := -h'(y) + Y^{-1}Z. \tag{30}$$

This procedure also requires $\mathcal{O}(m^3)$ operations in terms of the order, but less linear algebra computation than does the procedure (28a)-(28c).

Of course, we still need to establish in theory that the proposed primal-dual algorithms are well-defined. To this end, we need to show that the matrix $F_i'(x,y,z)$ are nonsingular for any $y, z > 0$, and the matrices M_i and $A^T N M_i^{-1} A$ are also nonsingular for both $i = 1$ and 2. These results will be presented next.

4 Theoretical Results

In this section, we give theoretical results regarding the well-definedness of the proposed algorithms, the uniqueness of solution in our formulations, as well as the existence and convergence of solution paths. We note that the formulations introduced in the last section are obtained by applying some nonlinear transformations. Therefore we need to show that these nonlinear transformations preserve the uniqueness of solution. We also mention that when $F = F_2$, the system in (17) is not equivalent to the optimality conditions of a convex program. Hence, it is not evident that solution paths defined by (17) should always exist for $F = F_2$.

4.1 Well-Definedness of Algorithms

We will show in this subsection that the proposed primal-dual algorithmic framework and the solution procedures (28a)-(28c) and (29a)-(29c) are well defined for both $F = F_1$ and $F = F_2$. (Following the same approach, one can also easily verify similar results for $F = F_0$.)

We recall that throughout the paper we have assumed that A has full rank with no zero rows. The main result of this subsection is the following theorem.

Theorem 1 (Non-singularity of Jacobian). *For any $y, z > 0$, the Jacobian matrices $F_i'(x,y,z)$ are nonsingular for $i = 1, 2$. Moreover, both the procedures (28a)-(28c) and (29a)-(29c) are well defined.*

Proof. The theorem follows directly from Lemma 3 below. □

Now we prove three technical results that will lead to the proof of Theorem 1.

Lemma 1. *Let $P \in \Re^{n\times n}$ be an orthogonal projection matrix, i.e., P satisfies $P^T = P$ and $P^2 = P$. Then the symmetric matrix*

$$G_\gamma = Diag(diag(P)) - \gamma P \circ P \tag{31}$$

is positive semidefinite for any $\gamma \le 1$. Moreover, if $diag(P) > 0$, then G_γ is positive definite for any $\gamma < 1$.

Proof. We note that since P is symmetric positive semidefinite, so is $P\circ P$ (see for example [4]). For the first statement, it suffices to only consider $\gamma = 1$.

Let (λ, x) be an eigenvalue and an associated eigenvector of G_1, respectively, such that

$$x_k = \max_i |x_i| \equiv 1,$$

which can always be achieved by a proper scaling. The k-th equation in $G_1 x = \lambda x$ is

$$P_{kk}x_k - \sum_{j=1}^{n} P_{kj}^2 x_j = \lambda x_k.$$

Since $P^2 = P$ and $P^T = P$, implying that $P_{kk} = \sum_{j=1}^{n} P_{kj}^2$, and $x_k = 1$, we have

$$\lambda = \sum_{j=1}^{n} P_{kj}^2 (1 - x_j) \geq 0,$$

where the last inequality follows from the fact that $|x_j| \leq 1$ for all j. Hence we have proved that G_1 is positive semidefinite. Together with the identity

$$G_\gamma = G_1 + (1-\gamma)P\circ P,$$

it implies that G_γ are positive semidefinite for all $\gamma \leq 1$ since both terms in the sum are positive semidefinite.

To prove the second statement, we assume that $diag(P) > 0$ and $\gamma < 1$. Then we rewrite

$$G_\gamma = (1-\gamma)\mathrm{Diag}(\mathrm{diag}(P)) + \gamma G_1,$$

which is clearly positive definite since the first term is positive definite and the second one is semidefinite. ☐

Lemma 2. *For any $y > 0$, the matrix $N(y) \equiv g'(y)$ is similar to a symmetric positive definite matrix, and thus is nonsingular.*

Proof. We first note $h(y) > 0$ whenever $y > 0$. In view of (24), (20) and (21),

$$\begin{aligned} N &= H - (2H)^{-1}Y[Q(y)\circ Q(y)] \\ &= H^{-1}(HYH - \frac{1}{2}Y[Q(y)\circ Q(y)]Y)Y^{-1} = H^{-1}GY^{-1} \\ &= [H^{-1/2}Y^{1/2}]\left([HY]^{-1/2}G[YH]^{-1/2}\right)[H^{-1/2}Y^{1/2}]^{-1} \end{aligned}$$

where

$$G := HYH - \frac{1}{2}Y[Q(y)\circ Q(y)]Y. \tag{32}$$

Therefore, N is similar to $[HY]^{-1/2}G[YH]^{-1/2}$, which is positive definite if and only if the matrix G is positive definite since both Y and H are positive diagonal matrices.

Recall that by our notation $Q(y) = A(A^TYA)^{-1}A^T$, $H = \mathrm{Diag}(h(y))$ and

$$h(y) \equiv h(E(y)) = (\mathrm{diag}(Q(y))^{1/2},$$

where the square root is taken element-wise. We have

$$HYH = \mathrm{Diag}(\mathrm{diag}(Q(y))Y = \mathrm{Diag}(\mathrm{diag}(Y^{1/2}Q(y)Y^{1/2})).$$

In addition, since $y_iQ_{ij}^2y_j = \left(\sqrt{y_i}Q_{ij}\sqrt{y_j}\right)^2$, we have

$$Y[Q(y)\circ Q(y)]Y = \left(Y^{1/2}Q(y)Y^{1/2}\right) \circ \left(Y^{1/2}Q(y)Y^{1/2}\right).$$

Therefore we can write

$$G = \mathrm{Diag}(\mathrm{diag}(P)) - \frac{1}{2}P\circ P,$$

where the matrix

$$P = Y^{1/2}Q(y)Y^{1/2} = Y^{1/2}A(A^TYA)^{-1}A^TY^{1/2}$$

is an orthogonal projection matrix. Since the vector y is positive and the matrix A has no zero rows, we have $\mathrm{diag}(P) > 0$. It follows from Lemma 1 with $\gamma = 1/2$ that G is indeed positive definite. This completes the proof. □

The relationships

$$N = H^{-1}GY^{-1} \quad \text{and} \quad N^{-1} = YG^{-1}H \tag{33}$$

that were used in the proof of Lemma 2 will be useful later.

Lemma 3. *For $y, z > 0$, there hold the following statements:*

1. *the matrix M_1 is similar to a symmetric positive definite matrix, and $A^TNM_1^{-1}A$ is symmetric positive definite;*

2. *the matrix M_2 is similar to a symmetric positive definite matrix, and $A^TNM_2^{-1}A$ is nonsingular.*

Proof. To prove the first statement, it suffices to prove that the matrix M_1N^{-1} is symmetric positive definite. Using the definitions of M_1, N and the formula for g', see (25), (24) and (20), respectively, and the relationships (33), we have $h' = Y^{-1}(N - H)$ and

$$\begin{aligned}
M_1N^{-1} &= ((YH)^{-1}ZN - h')N^{-1} \\
&= (YH)^{-1}Z - Y^{-1}(N-H)N^{-1} \\
&= (YH)^{-1}Z - Y^{-1} + Y^{-1}HN^{-1} \\
&= (YH)^{-1}Z - Y^{-1} + Y^{-1}H(YG^{-1}H) \\
&= (YH)^{-1}Z - Y^{-1} + HG^{-1}H \\
&= (YH)^{-1}Z + H\left(G^{-1} - (HYH)^{-1}\right)H.
\end{aligned}$$

Then it suffices to show that $G^{-1} - (HYH)^{-1}$ is symmetric positive definite since H, Y and Z are all positive diagonal matrices. While the symmetry is obvious, the positive definiteness follows from the fact that G equals HYH minus a positive semidefinite matrix, see (32); hence $G \prec YHY$ and $G^{-1} \succ (YHY)^{-1}$.

To prove the second statement, we use the formula for $h'(y)$ in (21), to obtain

$$M_2 = Y^{-1}Z - h' = H^{-1}(HY^{-1}Z + \frac{1}{2}Q \circ Q),$$

which is the product of two symmetric positive definite matrices, implying that M_2 is similar to a symmetric positive definite matrix. Since both M_2 and N are nonsingular, so is $ANM_2^{-1}A$. This completes the proof. $\square$

4.2 Uniqueness of Solution

Since we have utilized nonlinear transformations in the elimination of variables $E = E(y)$ and $u = g(y)$ from the KKT system (7a)-(7d), we need to establish a rigorous equivalence of our formulations (13) and (15) to the original KKT system. The main result is the following.

Theorem 2 (Uniqueness of Solution). *The systems (13) and (15) both have the same, unique solution (x^*, y^*, z^*) such that $y^*, z^* \geq 0$. Moreover, let $u^* = g(y^*)$ and $E^* = E(y^*)$. Then (x^*, E^*, u^*, z^*) is the unique solution of the KKT conditions (7a)-(7e).*

Proof. The conclusions follow directly from Lemmas 4 and 5, given below, and the uniqueness of the solution to the MaxVE problem. $\square$

We now prove the two technical lemmas.

Lemma 4. *Let $C \in \mathcal{S}^n_{++}$, then the matrix equation*

$$X^{-1} = \frac{1}{2}(CX + XC) \tag{34}$$

has a unique solution $X^* = C^{-1/2}$ in $\mathcal{S}^n_{++}$. Moreover, the mapping: $C \to X^*$ defined implicitly through (34) is homeomorphic between $\mathcal{S}^n_{++}$ and itself.

Proof. One can easily verify that both X^* and $-X^*$ are solutions to (34). This implies that the matrix equation (34) does not in general have a unique solution in $\Re^{n\times n}$.

Suppose that $\hat{X} \in \mathcal{S}^n_{++}$ is a solution to the equation (34) and U is an orthogonal matrix that diagonalizes $\hat{X}$, i.e., $U^T\hat{X}U = \Sigma$ where Σ is a positive diagonal matrix. Pre-multiplying both side of the equation (34) by U^T and post-multiplying them by U, we obtain

$$\Sigma^{-1} = \frac{1}{2}(D\Sigma + \Sigma D),$$

where $D = U^TCU$. Comparing the elements on both sides, we have

$$\frac{1}{2}D_{ij}(\Sigma_{ii} + \Sigma_{jj}) = \begin{cases} 0, & i \neq j, \\ 1/\Sigma_{ii}, & i = j. \end{cases}$$

Since $\mathrm{diag}(\Sigma) > 0$, we must have (i) $D_{ij} = 0$ for $i \neq j$ and (ii) $\Sigma_{ii} = D_{ii}^{-1/2}$. The first relationship says that $D = U^TCU$ is also diagonal. The second relationship says that $\Sigma = D^{-1/2}$, that is, $\hat{X} = C^{-1/2} \equiv X^*$. Consequently, X^* is the only solution of the equation (34) in $\mathcal{S}^n_{++}$.

The last statement of the lemma is evident in view of the explicit relationships $X^* = C^{-1/2}$ and $C = (X^*)^{-2}$. □

Lemma 5. *Let $g(y) \equiv Yh(y)$. Then the mapping $g : \Re^m_{++} \to \Re^m_{++}$ is homeomorphic between $\Re^m_{++}$ and its image under g, i.e., $g(\Re^m_{++}) \subset \Re^m_{++}$.*

Proof. It is straightforward to verify that the function $g(y)$ is continuously differentiable in $\Re^m_{++}$, whose derivative is represented by the matrix $g'(y) \equiv N(y)$. By Lemma 2, $N(y)$ is nonsingular in $\Re^m_{++}$. With these properties, the lemma is a direct consequence of the inverse function theorem. □

4.3 Existence and Convergence of Solution Paths

To justify our algorithms as the path-following type, we will show that (i) the perturbed system (17) with either $F = F_1$ or $F = F_2$ permits a unique solution for any given $w^0 \in \Re^m_{++}$ and each $\mu > 0$, hence the solution set forms a path; and (ii) as $\mu \to 0$ the path converges to the unique solution of the unperturbed system. Although it is straightforward to establish these results in the case of $F = F_1$, it is much more involved in the case of $F = F_2$ since the perturbed system (17) for $F = F_2$ does not correspond to the optimality conditions of a convex program.

Following the conventional terminology in the literature of interior-point methods, we will refer the collection of solution to the system (17) for $w^0 = e$ and $\mu > 0$ as the *central path* of the system, where $e \in \Re^m$ is the vector of all ones. Our analysis in this subsection applies to not only the central path but also to so-called weighted paths where $w^0 > 0$ is not equal to e.

The existence of paths for $F = F_1$ follows a standard argument as given below.

Proposition 1 (Existence and Convergence of Path for $F = F_1$).
For any $w^0 \in \Re^m_{++}$ and $\mu > 0$, the system (17) with $F = F_1$ have a unique solution $(x(\mu), y(\mu), z(\mu))$ such that $y(\mu), z(\mu) > 0$. Moreover,

$$\lim_{\mu \to 0} (x(\mu), y(\mu), z(\mu)) = (x^*, y^*, z^*),$$

where (x^*, y^*, z^*) is the solution of (13).

Proof. The proof follows from a standard argument which we will outline as follows. It is well-known that the system of the "perturbed" KKT (PKKT) conditions:

$$\begin{aligned} A^T u &= 0, && \text{(35a)} \\ E^{-1} - [E(A^TYA) + (A^TYA)E]/2 &= 0, && \text{(35b)} \\ z - (b - Ax - h(E)) &= 0, && \text{(35c)} \\ Uz &= w, && \text{(35d)} \\ u, z &> 0, && \text{(35e)} \end{aligned}$$

has a unique solution for any $w > 0$, where Y is defined as in (6), because it is equivalent to that the gradient of the following barrier function $B_w(x, E)$,

$$B_w(x, E) = -\log\det(E) - \sum_{i=1}^{m} w_i \log\left(b_i - a_i^T x - h_i(E)\right), \tag{36}$$

equals zero. This barrier function is strongly convex and has a unique stationary point $(x(\mu), E(\mu), u(\mu), z(\mu))$ corresponding to $w = \mu w^0$ for a fixed $w^0 \in \Re^m_{++}$ and any $\mu > 0$. It is well known that $(x(\mu), E(\mu), u(\mu), z(\mu))$ converges to the unique solution (x^*, E^*, u^*, z^*) of the (unperturbed) KKT system as $\mu \to 0$. Due to the homeomorphic relationships between the PKKT conditions and the conditions in (17) with $F = F_1$, we know that $(x(\mu), y(\mu), z(\mu))$, where $y(\mu) = \mathrm{Diag}(h(E(\mu)))^{-1}u(\mu)$, is also the unique solution of (17) with $F = F_1$. Moreover, the path $\{(x(\mu), y(\mu), z(\mu)) : \mu > 0\}$ converges to (x^*, y^*, z^*) where $y^* = \mathrm{Diag}(h(E^*))^{-1}u^*$. $\square$

We now consider the existence of solution to the system (17) when $F = F_2$; that is, the existence of solution to the system

$$\begin{aligned} A^T g(y) &= 0, && (37a)\\ Ax + h(y) + z - b &= 0, && (37b)\\ Yz &= w, && (37c)\\ y, z &> 0, && (37d) \end{aligned}$$

where $g(y)$ is defined as in (10). The situation here is more complicated because this system is no longer equivalent to the PKKT conditions (35a)-(35e) when $w > 0$, even though they are equivalent when $w = 0$. As such, we can no longer use the standard argument used in the proof of Proposition 1, in contrary to the case when $F = F_1$. The question is whether or not the following holds:

$$\{0 \in \Re^n\} \times \{0 \in \Re^m\} \times \Re^m_{++} \subset \mathcal{R}(F_2),$$

where

$$\mathcal{R}(F_2) := F_2(\Re^n \times \Re^m_{++} \times \Re^m_{++})$$

is the range of the function F_2 corresponding to the domain $\Re^n \times \Re^m_{++} \times \Re^m_{++}$. In particular, we want to know if the vectors $(0, 0, \mu e)$ for $\mu > 0$ are in the range of F_2; in other words, whether a central path exists for the system (17) in the case of $F = F_2$.

The answers to the above questions are affirmative and given in Theorem 3. To prove the theorem, it is necessary to establish a number of technical results. We start with the following proposition stating some useful facts.

Proposition 2. *The following facts hold:*

1. *Both F_1 and F_2 are locally homeomorphic at any point $(x, y, z) \in \Re^n \times \Re^m_{++} \times \Re^m_{++}$.*

2. *If $(\hat{x}, \hat{y}, \hat{z})$ is the solutions to the system (17) with $F = F_1$ and $w = \hat{w}$, then $(\hat{x}, \hat{y}, \hat{z})$ also satisfies (17) with $F = F_2$ (i.e., (37a)-(37d)) and $w = Diag(h(\hat{y}))^{-1}\hat{w}$.*

If one were able to choose $\hat{w}$ such that $\text{Diag}(h(\hat{y}))^{-1}\hat{w} = \mu e$, then he would find the point $(0, 0, \mu e)$ in the range of F_2. However, since $\hat{y}$ is dependent on $\hat{w}$, it is not clear whether or not such a vector $\hat{w}$ exists, let alone how to find it. Nevertheless, we do find a form of points $(0, 0, w)$ that are in the range of F_2.

Lemma 6. *Let $x \in \Re^n$, $E \in \mathcal{S}^n_{++}$ and $z \in \Re^m_+$ satisfy the equation*

$$Ax + h(E) + z = b. \tag{38}$$

Then there exists a constant $\gamma > 0$, independent of x, E and z, such that

$$\max(\|x\|, \|E\|, \|z\|) \leq \gamma.$$

Proof. The equation (38) implies that $x \in \mathcal{P}$ where $\mathcal{P}$ is the given polytope; hence such x's must be uniformly bounded above. Consequently, $b - Ax$ for $x \in \mathcal{P}$ is also uniformly bounded above, which in turn implies that both z and $h(E)$ are uniformly bounded above because they are both nonnegative and they sum up to $b - Ax$. Since $h_i(E) = (a_i^T E^2 a_i)^{1/2}$ and, by our assumption, the set $\{a_1, a_2, \cdots, a_m\}$ spans $\Re^n$, the uniform boundedness of $h(E)$ implies that of E. This completes the proof. $\square$

Lemma 7. *Let the barrier function $B_w(x, E)$ be defined as in (36), let $\mathcal{W}$ be a bounded set with its closure $col(\mathcal{W})$ in $\Re^m_{++} \cup \{0\}$. For any $w \in \Re^m_{++}$, define*

$$(x_w, E_w) := \arg\min B_w(x, E), \tag{39}$$

and for $w = 0 \in \Re^m$ define $(x_w, E_w) := (x^*, E^*)$ as the solution of the MaxVE problem (5). Then

$$\beta_{\mathcal{W}} := \inf_{w \in col(\mathcal{W})} \{\log\det(E_w)\} > -\infty.$$

Proof. Since the pair (x_w, E_w), $E_w \succ 0$, is the unique minimizer of $B_w(x, E)$, there exists some $(u_w, z_w) \in \Re^m_{++} \times \Re^m_{++}$ such that together they satisfies (35a)-(35e). It is well-known that the quadruple (x_w, E_w, u_w, z_w) is a continuous function of w in $\Re^m_{++}$, and that (x_w, E_w, u_w, z_w) converges to (x^*, E^*, u^*, z^*) as w converges to 0 from the interior of $\Re^m_{++}$. Hence, the composite function $\log\det(E_w)$ of w is a continuous function of w in $\Re^m_{++} \cup \{0\}$ and must attain its maximum on the compact set $\text{col}(\mathcal{W}) \subset \Re^m_{++} \cup \{0\}$. This proves the lemma. $\square$

Lemma 8. *Let $\mathcal{R}(F_2)$ be the range of the function F_2 corresponding to the domain $\Re^n \times \Re^m_{++} \times \Re^m_{++}$, and $\mathcal{W}$ be a bounded set in $\Re^m_{++}$ such that its closure $col(\mathcal{W}) \subset \Re^m_{++} \cup \{0\}$. Let*

$$\{0 \in \Re^n\} \times \{0 \in \Re^m_{++}\} \times \mathcal{W} \subset \mathcal{R}(F_2),$$

and $(x(w), y(w), z(w))$ be the solution to (37a)-(37c) corresponding to $w \in \mathcal{W}$. Then the set $\{y(w) : w \in \mathcal{W}\}$ is bounded.

Proof. The triple $(x(w), y(w), z(w))$ being the solution to (37a)-(37c) implies that the quadruple

$$(x_{w'}, E_{w'}, u_{w'}, z_{w'}) := (x(w), E(y(w)), g(y(w)), z(w))$$

is the solution to (35a)-(35e) with the right-hand side of the equation (35d) being replaced by $w' = \mathrm{Diag}(h(y(w)))w$. It is worth noting that the pair $(x_{w'}, E_{w'})$ also satisfies (39) with $w = w'$. Evidently, we have

$$E_{w'} \equiv E(y(w)).$$

Define the set

$$\mathcal{W}' := \{w' = \mathrm{Diag}(h(y_w))w : w \in \mathcal{W}\},$$

which is bounded because both $\mathcal{W}$ and the set of $\{h(y(w)) : w \in \mathcal{W}\}$ are bounded. It follows from Lemma 6 that the set

$$\{E_{w'} : w' \in \mathcal{W}'\} \equiv \{E(y(w)) : \ w \in \mathcal{W}\}$$

is bounded. Hence, the eigenvalues of $E(y(w))$ are uniformly bounded above. On the other hand, Lemma 7 implies that

$$\det(E(y(w))) \geq \exp(\beta_{\mathcal{W}}) > 0.$$

As a result, the eigenvalues of $E(y(w))$ are also uniformly bounded away from zero in the set $\mathcal{W}$. Consequently, the components of $h(y(w))$ are uniformly bounded above and away from zero in the set $\mathcal{W}$ because $h_i(y(w)) = (a_i^T E(y(w))a_i)^{1/2}$ and the rows a_i^T of A are all nonzero for $i = 1, \cdots, m$.

We note that the vector $\mathrm{Diag}[h(y(w))]^2 y(w)$ is the diagonal of the orthogonal projection matrix $Y(w)^{1/2}A[A^TY(w)A]^{-1}A^TY(w)^{1/2}$ and therefore is component-wise bounded above by the unity; namely,

$$y_i(w) \leq \frac{1}{h_i(y(w))^2}, \quad i = 1, 2, \cdots, m.$$

Since $h(y(w))$ is uniformly bounded away from zero $w \in \mathcal{W}$, we conclude that $y(w)$ is uniformly bounded above for $w \in \mathcal{W}$. This completes the proof. $\square$

Lemma 9. *Let $\mathcal{R}(F_2)$ be defined as in Lemma 8, then*

$$\{0 \in \Re^n\} \times \{0 \in \Re^m\} \times \Re^m_{++} \subset \ \mathcal{R}(F_2).$$

Proof. From the second statement of Proposition 2, we known that there exists a triple $(0, 0, w^\alpha) \in \mathcal{R}(F_2)$ for some $w^\alpha \in \Re^m_{++}$. Now for any given $w^\beta \in \Re^m_{++}$, we are to show that $(0, 0, w^\beta) \in \mathcal{R}(F_2)$.

Let us define the line segment between w^α and w^β

$$w(t) = (1 - t)w^\alpha + t\, w^\beta,$$

and the number

$$\hat{t} = \sup\{t \in [0,1] : \{(0,0,w(t')) : t' \in [0,t]\} \subset \mathcal{R}(F_2)\}.$$

Since $(0,0,w(0)) \in \mathcal{R}(F_2)$ and F_2 is homeomorphic between $\Re^n \times \Re^m_{++} \times \Re^m_{++}$ and $\mathcal{R}(F_2)$, we must have $\hat{t} > 0$. If $\hat{t} = 1$, we already have $w^\beta \in \mathcal{R}(F_2)$ and we are done.

Now suppose $\hat{t} < 1$. This implies that $(0,0,w(\hat{t})) \notin \mathcal{R}(F_2)$; otherwise by the local homeomorphism of F_2 the number $\hat{t}$ would not have been a supremum. Consider the set

$$\mathcal{W} := \{w(t) : t \in [0,\hat{t})\} \subset \mathcal{R}(F_2),$$

which is clearly bounded with its closure $\mathrm{col}(\mathcal{W})$ in $\Re^m_{++}$. It follows from Lemmas 6 and 8, the set

$$\{(x(w), y(w), z(w) : w \in \mathcal{W}\}$$

is also bounded. Let us denote $x(w(t))$ by $x(t)$, and so on. Then there must exist a sequence $\{t_k\}_{k=1}^\infty$ such that $t_k \to \hat{t}$ and $(x(t_k), y(t_k), z(t_k)) \to (\hat{x}, \hat{y}, \hat{z})$ for some $(\hat{x}, \hat{y}, \hat{z}) \in \Re^n \times \Re^m_+ \times \Re^m_+$ (otherwise, a convergent subsequence can be selected).

Since the function F_2 is continuous, we have $F_2(\hat{x}, \hat{y}, \hat{z}) = (0,0,w(\hat{t}))^T$, which means that $(0,0,w(\hat{t})) \in \mathcal{R}(F_2)$. This is a contradiction. Thus the assumption $\hat{t} < 1$ is false, and we have proved the lemma. $\square$

Finally we prove the existence and convergence of solution paths, including the central path, leading to the solution of the original MaxVE problem in the sense specified in the following theorem.

Theorem 3 (Existence and Convergence of Path for $F = F_2$).
For any $w^0 \in \Re^m_{++}$ and $\mu > 0$, the system (17) with $F = F_2$ and $w = \mu w^0$ has a unique solution $(x(\mu), y(\mu), z(\mu))$. Moreover,

$$\lim_{\mu \to 0} (x(\mu), y(\mu), z(\mu), u(\mu), E(\mu)) = (x^*, y^*, z^*, u^*, E^*),$$

where (x^*, y^*, z^*) satisfies the system (15), and (x^*, E^*, u^*, z^*) satisfies the KKT system (7a)-(7e). Consequently, (x^*, E^*) solves the MaxVE problem (5).

Proof. The first statement follows directly from Lemma 9 and the fact that F_2 is homeomorphic in $\Re^n \times \Re^m_{++} \times \Re^m_{++}$.

By Lemmas 6 and 8, the quantities $x(\mu), y(\mu), z(\mu), u(\mu)$ and $E(\mu)$ are all bounded as $\mu \to 0$. Hence, they must have accumulation points as $\mu \to 0$, say, x^*, y^*, z^*, u^* and E^*. Clearly, these accumulation points satisfy the two systems in the theorem. Since these systems only permit unique solutions, we conclude that all accumulation points of $x(\mu)$ as $\mu \to 0$ must coincide, and the same is true for other quantities as well; namely, accumulation points are actually the limit point. Obviously, x^* and E^* solve the optimization problem (5) because they, together with u^* and z^*, satisfy the optimality conditions (7a)-(7e). This proves the theorem. $\square$

4.4 Issues of Algorithmic Convergence

So far polynomial convergence theory for primal-dual interior point algorithms has been established only for convex conic programming in symmetric cones (see [16], for example). Given the highly nonlinear formulations upon which we build our primal-dual interior-point algorithms, it seems unlikely that polynomial convergence could be proven for our primal-dual algorithms unless some new paradigm is discovered.

On the other hand, performing some nonpolynomial, global convergence analysis for the proposed algorithmic framework appears to be a worthy task. Given the good theoretical properties we have already established for our formulations, we do not see fundamental difficulties in proving global and fast local convergence for some parameter choices in the proposed algorithmic framework. Such an analysis, however, would be rather lengthy and technical. To keep the current paper focused and within a reasonable length, we will not attempt a convergence analysis in this paper.

5 Khachiyan-Todd Algorithm and Modification

We will introduce two other algorithms, the Khachiyan and Todd algorithm [10] and a modification of it, and will later compare them with algorithms proposed in Section 3.

Given a set of inequalities $Ax \leq b$ and a strictly interior point x^0, using the change of variable $x = v + x^0$, we can rewrite the inequalities as $Av \leq b - Ax^0$. By multiplying both sides by the positive diagonal matrix $\mathrm{Diag}(b - Ax^0)^{-1}$, we obtain the following polytope

$$\mathcal{P} = \{v \in \Re^n : Cv \leq e\}, \tag{40}$$

where $C \equiv \mathrm{Diag}(b - Ax^0)^{-1}A \in \Re^{m\times n}$ and e is a vector of all ones in $\Re^m$. We will use this form of polytopes in this section as it was used by Khachiyan and Todd in [10].

In the formulation (5), the matrix-valued variable E appears in the constraints in a non-linear manner. In an alternative formulation given below, through the change of variables $B = E^2$ one can have the unknown matrix B to appear linearly in the constraints. Indeed, after substituting E^2 by B and using the form (40), we can rewrite the problem (5) into

$$\begin{array}{ll} \min & -\log\det B \\ \text{s.t.} & c_i^T B c_i \leq (1 - c_i^T x)^2, \quad i = 1, \ldots, m. \\ & (Cx < e, \quad B \succ 0) \end{array} \tag{41}$$

While the constraints of (41) are linear with respect to the matrix variable B, they are no longer linear or convex with respect to the vector variable x.

5.1 Khachiyan and Todd's algorithm

Khachiyan and Todd's algorithm [10] for the MaxVE problem has a good complexity bound and also takes the advantage of the special structure of the MaxVE problem. It is a suitable candidate for the purpose of performance comparison.

To make use of the simplicity of linear constraints, Khachiyan and Todd introduced the following subproblem, or auxiliary problem $AP(a)$, from (41):

$$\begin{array}{ll} \min & -\log\det B \\ \text{s.t.} & c_i^T B c_i \leq (1 - c_i^T x)(1 - c_i^T a), \quad i = 1, \ldots, m \\ & (B \succ 0) \end{array} \tag{42}$$

for a fixed $a \in \Re^n$ where $Ca < e$. Note that now the constraints are linear in both B and x. The key idea here is to solve subproblems $AP(a)$ iteratively until x and a become sufficiently close to each other so (42) becomes a good approximation of (41). Khachiyan and Todd use a primal barrier method to solve the subproblem $AP(a)$ where the barrier function has the form

$$F_t(x, B|\, a) = -\log\det B - t\sum_{i=1}^{m} \log\left((1 - c_i^T x)(1 - c_i^T a^k) - c_i^T B c_i\right),$$

where a is fixed and t is the barrier parameter. The Khachiyan and Todd (KT) Algorithm can be summarized as follows.

Algorithm 2 (Khachiyan and Todd's Algorithm).

Step 1. *Let a^0 be a strictly interior point of $\mathcal{P}$, $B^0 \succ 0$, $\epsilon > 0$, and $k = 0$.*

Step 2. *Solve the subproblem $AP(a^k)$ by using Newton's method to minimize the barrier function $F_t(x, B|\, a^k)$ for a sequence of $t \downarrow 0$. The solution of $AP(a^k)$ is (x^k, B^k).*

Step 3. *If $\|x^k - a^k\| \leq \epsilon$, then stop; else let $a^{k+1} = (a^k + x^k)/2$, increment k and go to Step 2.*

Khachiyan and Todd prove that to attain a sufficient accuracy only a small number of subproblems need to be solved, and derive a linear system of size $n + m$ for calculating the Newton direction. The updates to the matrix-valued variable B are not directly calculated as an independent variable, thus reducing the complexity of the algorithm. However, the drawback of their algorithm is that the barrier method used to solve the subproblem is not particular efficient in practice. Moreover, as we can see from the algorithmic framework, three layers of loops are involved in the KT algorithm: the loop for the subproblem parameter a, the loop for barrier parameter t, and the iterations for a fixed a and a fixed t.

5.2 A Modification of the KT Algorithm

Since primal barrier methods are generally less efficient than primal-dual, interior-point methods, in order to speed up the KT algorithm we modify it by applying a primal-dual interior-point method to the subproblems in the Step 2 of the KT algorithm, while keeping the outer iterations intact.

Following Khachiyan and Todd's approach, we transform the subproblem $AP(a)$ into the standard form $AP(0)$:

$$
\begin{array}{ll}
\min & -\log\det B \\
\text{s.t.} & c_i^T B c_i + c_i^T x \le 1, \quad i = 1, \ldots, m \\
 & (B \succ 0)
\end{array}
\tag{43}
$$

by the change of variables $x \Rightarrow x + a$ and the change of data $c_i/(1 - c_i^T a) \Rightarrow c_i$ for $i = 1, \cdots, m$.

The optimality conditions, or the Karush-Kuhn-Tucker (KKT) conditions, of problem $AP(0)$ are as follows:

$$
\begin{aligned}
C^T y &= 0, &\text{(44a)}\\
B^{-1} - C^T Y C &= 0, &\text{(44b)}\\
Cx + \operatorname{diag}(CBC^T) + z - e &= 0, &\text{(44c)}\\
Yz &= 0, &\text{(44d)}\\
y, z &\ge 0, &\text{(44e)}
\end{aligned}
$$

where $y \in \Re^m$ is the vector of Lagrangian multipliers, $z \in \Re^m$ consists of slack variables, and $C \in \Re^{m \times n}$ with c_i^T as its i-th row.

Following the same strategy used earlier, we eliminate the matrix variable B from the system using the substitution $B(y) = (C^T Y C)^{-1}$ that is the solution to (44b). We also replace the zero right-hand side of (44d) by μe. The resulting system that defines the central path is

$$
F_3(x, y, z) := \begin{pmatrix} C^T y \\ Cx + \operatorname{diag}(Q(y)) + z - e \\ Yz \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \mu e \end{pmatrix}
\tag{45}
$$

and $y, z > 0$, where $Q(y) = C(C^T Y C)^{-1} C^T$. Clearly, equation (45) is a square, nonlinear system of $n + 2m$ variables. The Jacobian matrix of $F_3(x, y, z)$ is

$$
F_3'(x, y, z) = \begin{bmatrix} 0 & C^T & 0 \\ C & -Q \circ Q & I \\ 0 & Z & Y \end{bmatrix}.
$$

To solve the Newton linear system

$$
F_3'(x, y, z) \begin{pmatrix} dx \\ dy \\ dz \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} := \begin{pmatrix} 0 \\ 0 \\ \mu e \end{pmatrix} - F_3(x, y, z),
$$

we use the following block Gaussian elimination procedure:

$$
\begin{aligned}
dx &= (C^T M^{-1} C)^{-1}(r_1 + C^T M^{-1}(r_2 - Y^{-1} r_3)),\\
dy &= M^{-1}(C dx - r_2 + Y^{-1} r_3),\\
dz &= Y^{-1}(r_3 - Z dy),
\end{aligned}
$$

where the matrix $M := Q \circ Q + Y^{-1} Z$ is symmetric positive definite.

The primal-dual algorithm for solving the subproblem $AP(0)$ falls into the same framework of Algorithm 1.

6 Numerical Results

In this section, we report our numerical results on the four algorithms: the KT algorithm, the modified KT, or MKT, algorithm, and the two direct primal-dual interior-point algorithms based on the systems (13) and (15) which we name F1PD and F2PD, respectively. The numerical tests were performed on three sets of test problems with a total of 200 problems. Our implementations of the four algorithms are in Matlab. All the experiments were run on an SGI Origin2000 computer with multiple 300-MZH R12000 processors. However, our programs use only a single processor at a time.

6.1 Implementation Details

In describing the implementation details, we first give some features common to all the algorithms and then other features specific to individual algorithms.

For all the algorithms, the input data for a polytope include the matrix A, the vector b and a strictly interior point point x^0 such that $Ax^0 < b$ which will serve as the starting point for the center of the initial ellipsoid. In our implementations, the point x_0 is selected to be the solution to an auxiliary linear program $\max\{\tau : Ax + \tau e \leq b\}$. Other choices are certainly possible such as the analytic center of the polytope. However, it was not our intention to use the best possible starting point.

Scaling is an important issue in numerical computation. In our implementations, we always first transform the inequality $Ax \leq b$ into the form $Cv \leq e$ using the change of variables and the row scaling as is described at the beginning of Section 5. After the transformation, the starting point x^0 is transformed into the origin, and the transformed polytope is better scaled.

In all the algorithms, the stopping tolerance is set to $\epsilon = 10^{-4}$. In the case of the KT and MKT algorithms, we stop the outer iterations whenever the relative change between the current and previous centers is less than or equal to ϵ. In the case of the F1PD and F2PD algorithms, we stop whenever the residual norm of F_i, $i = 1$ or 2, becomes less than or equal to ϵ.

We now describe some algorithm-specific features.

- The KT and MKT algorithms: Both algorithms have the same outer loop with the center varying. The initial center is the origin and the initial value for the matrix variable B is $B^0 = \rho I$ where I is the identity matrix and ρ is chosen such that the corresponding ball, centered at the origin with radius ρ, lies entirely inside the polytope. During the outer iterations, we use a warm-start strategy in which a later iteration always starts from the solution of the previous iteration.

- The KT algorithm: In the subproblems, the barrier parameter t is set to 0.5 initially and then decreased by a factor of 10 whenever the subproblem stopping criterion is met. For a fixed t value, the subproblem stopping criterion is that the gradient norm of the corresponding barrier function becomes less than or equal to t. This way, the stopping criterion becomes progressively more stringent as t approaches zero. We found that this adaptive strategy made the algorithm run significantly faster. To prevent the loss of symmetry during the computation, we set $B = (B + B^T)/2$ after B

is updated at every iteration. We update an iterate for (x, B) by a damped Newton step to ensure that the updated ellipsoid remains inside the polytope. Specifically, the step length is 0.75 times the largest allowable step that keeps the updated ellipsoid inside the polytope.

- The Primal-Dual algorithms: The primal-dual algorithmic framework (i.e., Algorithm 1) encompasses the F1PD and F2PD algorithms, and the subproblem solver of the MKT algorithm. The initial values for the primal-dual algorithms are set as follows: the initial center is $x = 0$; the initial multiplier value is $y = e$; and the initial slack variable z, say in the equation $z - g = 0$, is set as $z_i = \max(0.1, g_i)$. In addition to the initial values, there are two critical parameters in these algorithms: the so-called centering parameter σ^k and the step length α^k. In our implementations, we choose $\sigma^k = \min\{0.5, (y^k)^T z^k/m\}$ and $\alpha^k = \min(1, \tau\hat{\alpha})$ where $\tau \in (0, 1)$ and $\hat{\alpha}$ is the maximum length such that updated iterate for (x, y, z) reaches the boundary of the set $\mathcal{P} \times \Re^m_{++} \times \Re^m_{++}$. We use $\tau = 0.75$ for the F1PD and F2PD algorithms, and a more aggressive value $\tau = 0.9$ for the subproblem solver of the MKT algorithm because the subproblem (42) is not as nonlinear as its counterparts are in the F1PD and F2PD algorithms.

The parameter settings given above are rather generic and unsophisticated. For example, a line search scheme for determining step length could be a more effective and theoretically sound strategy. However, given our purpose of identifying the most robust and efficient algorithm, we consider our current settings to be appropriate and sufficient.

6.2 Test Problems

Three sets of test problems were used in our numerical experiments, consisting of 47, 143 and 10 problems, respectively. The total number of test problems is 200 and all the 200 test problems are made available at the web site:

`http://www.caam.rice.edu/~zhang/maxvep/.`

Test sets 1 and 2 are obtained from two integer programming feasibility problems through the search trees of an integer programming algorithm – the Lenstra algorithm for integer programming feasibility problem [11, 12]. This algorithm searches on a tree of subproblems and applies ellipsoidal approximation on each one of them. The polytopes in Sets 1 and 2 are taken from some branches of the search trees for two integer programming feasibility problems, respectively. The problem sizes in Sets 1 and 2 are relatively small with $m \leq 288$ and $n \leq 80$. Nevertheless, our numerical experience has indicated that some of the problems are non-trivial to solve.

In order to test the ability of our algorithms for solving larger problems, we generated a set of ten random problems that is called Set 3. The largest problem in this set has $m = 1200$ and $n = 500$. For each problem, we first use the Matlab function `sprandn` to generate a sparse random matrix B, and also use the `rand` function to generate a right-hand side vector $c > 0$, a upper-bound vector $ub > 0$ and a lower-bound vector $lb < 0$. Together, they form a polytope

$$\{x \in \Re^n : Bx \leq c, \;\; lb \leq x \leq ub\}$$

where $B \in \Re^{k \times n}$ and $c \in \Re^k$ and $lb, ub \in \Re^n$. By construction, the origin $x = 0$ is strictly interior to the polytope. Then we rewrite the polytope into the standard form

$$\{x \in \Re^n : Ax \leq b\},$$

where $A \in \Re^{m \times n}, b \in \Re^m$ with $m = k + 2n$. The matrix A is constructed, in an obvious manner, from the matrix B and the identity matrix in $\Re^n$, and the vector b from the vectors $c \in \Re^k$ and $lb, ub \in \Re^n$. The problems in Set 3 are sparse.

6.3 Test Results

Test results on Sets 1 and 2, totaling 190 problems are summarized in Table 1, while detailed results are given in Tables 3-6 in the Appendix. Four rows of numbers are presented in Table 1. In the first two rows, we list the test set number, the number of test problems in each set, the total number of iterations and the total amount of CPU time taken by each algorithm for solving the entire set of test problems. In the last two rows, we give the algebraic and geometric means for each category over the 190 test problems.

We note that the iteration numbers for the KT and the MKT algorithms are the numbers of inner-most, Newton iterations that involve solving systems of linear equations. These inner-most iterations are comparable with the iterations of the primal-dual algorithms because for a given problem they all require to solve linear systems of essentially the same size. The CPU time is given in seconds. As was mentioned earlier, detailed results on the test sets 1 and 2 are attached in the Appendix as Tables 3-6.

Table 1: Summary of Results on Tests 1 and 2

Test set	No. of probs	KT iter	KT time	MKT iter	MKT time	F1PD iter	F1PD time	F2PD iter	F2PD time
1	47	19416	3340	2655	240	692	124	694	77
2	143	56783	3567	9720	429	2448	168	2058	104
alg. mean		401.0	36.4	65.1	3.5	16.5	1.5	14.5	1.0
geo. mean		397.7	30.5	64.3	3.2	16.2	1.3	14.3	0.8

From Table 1, we observe that on average the KT algorithm took over 30 seconds per problem, the MKT algorithm is about 10 times faster than the KT algorithm; F1PD algorithm is over 2 times faster than the MKT algorithm, and the F2PD algorithm is about 1.5 times faster than the F1PD algorithm.

We mention that out of the 190 test problems in test sets 1 and 2 the KT algorithm failed to converge on two: problems 22 and 120 in the set 2. More conservative choices of parameters would make the KT algorithm converge on these two problems, but would also adversely affect the overall performance of the algorithm. We kept the current choices of parameters for the benefit of the KT algorithm.

The test results on the randomly generated test set 3 are presented in Table 2. Only the F1PD and F2PD algorithms were tested on this set of larger problems because the other two algorithms, noncompetitive in time, would require an excessively long time to run. Since these test problems are sparse, in addition to the matrix sizes m and n, we also include the number of nonzero entries, denoted as nnz, in the matrix A.

Table 2: Results on Test Set 3: Problems 1-10

Prob	Size			F1PD		F2PD	
number	m	n	nnz	iter	time	iter	time
1	600	100	7426	31	97	22	30
2	600	150	8408	30	107	23	39
3	600	200	7669	53	203	29	58
4	600	250	5022	60	249	31	73
5	800	100	5914	34	235	22	63
6	800	200	8029	34	271	24	91
7	800	300	8933	58	549	32	165
8	1000	300	11993	40	675	28	245
9	1000	400	8433	60	1134	31	330
10	1200	500	10518	73	2917	37	703
alg. mean	—	—	—	47.3	643.3	27.9	179.6
geo. mean	—	—	—	45.1	364.1	27.5	110.3

The results in Table 2 indicate that given the current choices of parameters, the F2PD algorithm clearly outperforms the F1PD algorithm by a considerable margin on the test set 3. Although the performance of the F1PD algorithm may be somewhat improved by selecting different parameters, we do not believe that it can in general outperform the F2PD algorithm because it requires more linear algebra calculation in each iteration for solving its version of the Newton linear system.

7 Concluding Remarks

The goal of this study is to find a practically efficient algorithmic framework for solving general MaxVE problems. Our extensive numerical results show that among the four tested algorithms, the method of choice is clearly the F2PD algorithm built on the formulation (15), which has been shown to have a sound theoretical foundation. We have established, among other things, the existence of a central path for this formulation even though, unlike in the conventional cases, this central path is not known to be directly connected to the optimality conditions of a barrier function.

The main advantage of the F2PD algorithm over the KT and the MKT algorithms is that, without the need for solving a number of subproblems either for fixed centers or fixed barrier parameter values, it requires less iterations (or linear system solutions) than the other two algorithms. We expect that the same advantage would still hold against some other untested algorithms like the one given in [1]. On the other hand, compared to the F1PD algorithm, the F2PD algorithm requires less linear algebra computation per iteration and seems to be more robust. These features make the F2PD algorithm particularly attractive.

The algorithms considered in this paper are all of the general-purpose type. For really large-scale problems with special structures, one will likely need special-purpose algorithms that can take full advantage of the problem-specific structures, in particular sparsity, in order to solve the problems efficiently. This should be a topic of further research.

References

[1] K. M. Anstreicher. Improved Complexity for Maximum Volume Inscribed Ellipsoids. Manuscript. June 2001.

[2] M. Dyer, A. Frieze, and R. Kannan. A random polynomial-time algorithm for estimating volumes of convex bodies. *Journal of the Association for Computing Machinery*, 38:1–17, 1991.

[3] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization.* Springer, Berlin, 1988.

[4] R.A. Horn and C.R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, 1991.

[5] R.A. Horn and C.R. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[6] F. John. Extreme problems with inequalities as subsidiary conditions. *Studies and Essays, presented to R. Courant on his 60th Birthday*, pages 187–204, 1948.

[7] R. Kannan and L. Lovász. Random walks and an $O^*(n^5)$ volume algorithm for convex bodies. Technical Report No. 1092, Dept. of Computer Science, Yale University, New Haven, CT, 1996.

[8] L. Khachiyan. A polynomial algorithm in linear programming. *Doklady Akademii Nauk SSSR*, 244:1093–1096, 1979.

[9] L. Khachiyan. Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research*, 21:307–320, 1996.

[10] L. Khachiyan and M. Todd. On the complexity of approximating the maximal inscribed ellipsoid for a polytope. *Mathematical Programming*, 61:137–159, 1993.

[11] H. W. Lenstra, Jr. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8:538–548, 1983.

[12] L. Lovász. *An Algorithmic Theory of Numbers, Graphs, and Convexity.* SIAM, Philadelphia, 1986

[13] L. Lovász and M. Simonovits. On the randomized complexity of volumes and diameters. In *Proceedings of the 33rd Annual Symposium on Foundation of Computer Science*, pages 482–491, 1992.

[14] A. Nemirovski. On self-concordant convex-concave functions. Technical report, Technion – Israel Institute of Technology, Technion, Israel, 1997.

[15] Y. E. Nesterov and A. S. Nemirovskii. *Interior Point Methods in Convex Programming –Theory and Applications.* The Society for Industrial and Applied Mathematics, Philadelphia, 1994.

[16] Y. Nesterov and M. Todd. Primal-dual interior-point methods for self-scaled cones.). SIAM Journal on Optimization 8 (1998) 324-364.

[17] A. Schrijver. *Theory of Linear and Integer Programming.* John Wiley & Sons Ltd, Chichester, 1986.

[18] S. Silvey and D. Titterington. A geometric approach to optimal design theory. *Biometrika*, 60:21–32, 1973.

[19] S. Tarasov, L. Khachiyan, and I. Erlich. The method of inscribed ellipsoid. *Soviet Mathematics Doklady*, 37:226–230, 1988.

[20] D. Titteringto. Optimal design: Some geometric aspects of d-optimality. *Biometrika*, 62:313–320, 1975.

[21] L. Vandenberghe, S. Boyd, and S. Wu. Determinant maximization with matrix inequality constraints. Technical report, EE Dept. Stanford University, 1996.

[22] E. Welzl. Smallest enclosing disks, balls and ellipsoids. In H. Maurer, editor, *New Results and New Trends in Computer Sciences*, volume 555, pages 359–370. Springer Lecture Notes in Computer Science, New York, 1991.

[23] Y. Zhang. An interior-point algorithm for the maximum-volume ellipsoid problem. Technical report TR98-15, CAAM Dept., Rice University, 1998 (Revised 1999).

Appendix: Detailed Data for Sets 1 and 2

The first three columns of the tables give problem number and size where m is the number of polytope-defining inequalities and n the number of variables. For the KT and MKT algorithms, the tables give both the number of outer-iterations (i.e., the number of subproblems solved) and the number of inner-iterations (i.e., the number of Newton iterations), separated by a slash. The time is measured by CPU seconds. On Problems 22 and 120 of the test set 2, the KT algorithm failed to solve the first subproblem (i.e., at the zero-th outer iteration). For these 2 problem, we set the outer iteration number to 0 and give the number of inner iterations the algorithm took before it stopped.

Table 3: Results on Test Set 1: Problems 1-47

Prob	Size		KT		MKT		F1PD		F2PD	
No.	m	n	Iter	Time (sec)	Iter	Time (sec)	Iter	Time (sec)	Iter	Time (sec)
1	288	80	8/369	155.83	8/43	10.14	21	8.82	18	4.40
2	286	79	9/416	194.64	9/79	19.18	20	9.00	21	5.80
3	272	72	9/375	137.47	9/48	9.30	15	5.33	15	3.22
4	260	66	10/431	145.55	10/58	9.81	17	5.54	17	3.38
5	256	64	10/410	132.96	10/49	8.04	14	4.39	14	2.69
6	246	59	10/414	117.01	10/53	7.44	15	4.09	15	2.50
7	244	58	10/411	116.80	10/52	7.11	14	3.79	15	2.52
8	236	54	10/405	98.98	10/57	7.05	15	3.55	16	2.33
9	234	53	10/414	101.27	10/54	6.51	15	3.54	15	2.21
10	232	52	10/425	101.98	10/53	6.17	15	3.54	16	2.37
11	230	51	9/386	87.61	9/52	5.98	15	3.31	15	2.11
12	228	50	9/384	85.17	9/54	6.08	15	3.26	15	2.01
13	226	49	10/441	96.21	10/56	6.06	15	3.20	16	2.19
14	224	48	10/433	88.96	10/58	6.19	16	3.33	16	2.07
15	222	47	10/439	88.64	10/55	5.57	16	3.18	16	1.95
16	220	46	10/432	83.21	10/55	5.41	16	3.08	16	1.96
17	218	45	10/433	81.07	10/54	5.16	16	2.94	16	1.81
18	216	44	10/439	80.37	10/55	5.10	16	2.95	16	1.76
19	212	42	9/405	69.66	9/52	4.58	15	2.54	15	1.55
20	210	41	9/391	65.11	9/53	4.57	13	2.14	13	1.36
21	208	40	10/423	67.76	10/63	5.26	16	2.57	15	1.47
22	206	39	10/434	66.50	10/61	4.91	16	2.53	15	1.49
23	206	39	10/434	66.50	10/63	5.10	15	2.30	15	1.46
24	204	38	10/431	64.24	10/60	4.67	16	2.43	15	1.35
25	202	37	9/393	55.93	9/63	4.84	15	2.11	15	1.31
26	200	36	9/393	53.98	10/59	4.29	15	2.04	15	1.26
27	198	35	10/429	57.06	10/60	4.24	15	1.99	15	1.22
28	196	34	10/428	55.18	10/67	4.64	15	1.92	15	1.20
29	194	33	10/430	53.54	10/66	4.45	16	1.99	15	1.16
30	192	32	10/424	50.88	10/58	3.83	15	1.83	15	1.15
31	192	32	10/434	52.16	10/61	4.05	15	1.85	14	1.05
32	190	31	10/411	46.93	10/56	3.46	15	1.75	15	1.10
33	188	30	10/410	44.56	10/57	3.39	15	1.69	15	1.04
34	186	29	10/406	42.55	10/56	3.21	14	1.51	14	0.97
35	184	28	10/410	41.42	10/60	3.34	13	1.34	13	0.83
36	184	28	10/406	40.98	10/60	3.37	14	1.45	14	0.91
37	182	27	10/417	40.33	10/56	3.03	13	1.29	14	0.86
38	180	26	10/413	37.95	10/58	3.03	13	1.25	14	0.81
39	180	26	10/413	37.95	10/57	2.93	14	1.35	14	0.84
40	178	25	10/403	35.61	10/56	2.76	13	1.19	13	0.72
41	176	24	10/406	34.18	10/51	2.40	13	1.21	13	0.69
42	174	23	10/418	33.77	10/54	2.46	12	0.99	13	0.68
43	172	22	10/404	31.10	10/50	2.17	12	0.95	13	0.66
44	172	22	10/405	31.17	10/52	2.28	12	0.97	13	0.66
45	168	20	10/404	28.50	10/57	2.31	13	0.96	14	0.63
46	164	18	10/384	24.35	10/56	2.10	13	0.89	12	0.51
47	162	17	10/400	15.91	10/48	1.61	10	0.42	10	0.39

Table 4: Results on Test Set 2: Problems 1-48

Prob	Size		KT		MKT		F1PD		F2PD	
No.	m	n	Iter	Time (sec)	Iter	Time (sec)	Iter	Time (sec)	Iter	Time (sec)
1	147	50	11/463	34.60	11/73	3.78	21	1.71	16	0.98
2	147	49	9/426	32.07	12/88	4.44	20	1.59	16	0.96
3	147	48	7/376	27.92	9/67	3.43	18	1.41	14	0.81
4	147	49	9/390	29.09	12/83	4.20	25	2.09	18	1.11
5	147	48	7/370	27.43	9/65	3.31	18	1.47	14	0.80
6	147	47	7/407	29.87	9/64	3.22	17	1.39	14	0.85
7	147	48	9/403	29.54	10/77	3.92	26	2.05	19	1.19
8	147	47	8/391	28.61	10/74	3.64	19	1.51	16	0.95
9	147	49	9/395	29.52	11/72	3.60	20	1.70	16	0.99
10	147	48	7/385	29.59	11/73	3.61	16	1.32	14	0.86
11	147	48	9/414	31.41	12/83	4.13	24	2.00	19	1.19
12	147	47	8/380	27.81	10/74	3.75	18	1.44	16	0.92
13	147	48	9/398	30.07	12/79	3.91	21	1.79	17	1.05
14	147	47	8/383	28.59	10/73	3.66	19	1.58	16	0.94
15	147	47	9/389	28.83	10/74	3.66	23	1.84	18	1.06
16	137	35	7/513	26.29	7/55	2.11	20	1.11	15	0.61
17	137	35	7/491	25.08	7/49	1.84	14	0.79	13	0.54
18	137	35	7/383	19.23	7/50	1.89	16	0.91	13	0.57
19	137	34	6/383	19.22	7/47	1.74	13	0.70	13	0.51
20	137	34	7/415	20.74	7/49	1.79	15	0.85	13	0.53
21	132	30	6/336	14.67	6/48	1.62	15	0.74	13	0.46
22	134	33	0/166	8.36	7/55	1.94	21	1.13	16	0.61
23	137	36	7/354	17.94	7/64	2.53	21	1.20	17	0.69
24	137	35	7/358	18.02	7/63	2.45	18	1.00	16	0.72
25	137	34	7/396	19.95	7/57	2.22	20	1.09	18	0.73
26	137	34	8/409	20.60	9/73	2.70	20	1.15	17	0.70
27	136	33	6/320	15.61	6/59	2.20	23	1.29	18	0.71
28	136	32	7/506	25.15	8/71	2.61	20	1.13	18	0.75
29	137	36	7/484	25.32	7/58	2.28	16	0.89	13	0.53
30	137	35	8/588	30.33	9/58	2.14	14	0.80	13	0.53
31	137	35	6/432	22.19	7/57	2.17	21	1.15	15	0.61
32	137	34	7/437	21.98	7/53	1.97	16	0.87	13	0.52
33	137	34	7/422	21.24	7/45	1.62	13	0.73	13	0.53
34	137	36	7/361	18.32	7/63	2.43	21	1.16	17	0.70
35	137	35	7/359	18.19	7/61	2.36	19	1.08	17	0.71
36	137	35	6/299	15.09	6/66	2.67	23	1.34	19	0.84
37	137	34	7/366	18.47	8/68	2.56	19	1.05	17	0.69
38	137	35	6/299	15.08	6/57	2.24	20	1.14	17	0.69
39	137	34	6/296	14.70	6/57	2.19	20	1.14	18	0.76
40	136	33	6/323	15.77	6/59	2.22	24	1.33	19	0.75
41	147	48	9/390	29.49	11/74	3.67	21	1.68	17	1.02
42	147	47	7/357	26.46	10/85	4.29	15	1.18	13	0.76
43	147	46	7/462	34.29	7/59	3.04	19	1.47	15	0.85
44	147	47	8/363	26.70	13/75	3.55	17	1.34	14	0.82
45	147	46	7/419	30.84	14/88	4.19	15	1.16	13	0.74
46	147	46	8/372	26.95	12/80	3.84	15	1.15	13	0.75
47	147	45	7/354	24.79	12/79	3.74	14	1.06	13	0.71
48	147	44	7/442	30.97	12/82	3.88	17	1.26	14	0.76

Table 5: Results on Test Set 2: Problems 49-96

Prob	Size		KT		MKT		F1PD		F2PD	
No.	m	n	Iter	Time (sec)	Iter	Time (sec)	Iter	Time (sec)	Iter	Time (sec)
49	147	46	8/368	26.21	10/65	3.14	17	1.30	14	0.79
50	147	45	7/371	26.21	13/82	3.84	15	1.13	13	0.72
51	147	45	8/370	25.95	11/71	3.40	16	1.21	14	0.78
52	147	43	7/370	24.92	12/75	3.44	15	1.08	13	0.68
53	147	46	7/359	25.71	11/72	3.46	15	1.14	12	0.68
54	147	47	11/479	35.21	11/84	4.22	19	1.49	17	0.99
55	147	46	8/401	27.98	12/80	3.83	15	1.12	14	0.76
56	147	46	10/459	31.92	12/80	3.81	17	1.28	16	0.87
57	147	45	9/430	29.70	13/74	3.40	15	1.12	14	0.75
58	147	45	8/396	26.77	9/67	3.23	13	0.94	12	0.65
59	135	39	7/389	20.67	7/55	2.16	14	0.81	13	0.56
60	135	38	7/403	21.29	9/61	2.31	14	0.80	12	0.52
61	135	38	7/461	24.80	8/59	2.30	15	0.87	12	0.51
62	135	38	7/390	20.77	9/61	2.30	14	0.81	12	0.51
63	135	37	8/383	19.70	11/67	2.42	15	0.84	14	0.58
64	135	39	8/381	20.03	11/64	2.32	12	0.70	11	0.47
65	135	38	7/340	17.95	10/58	2.12	11	0.63	11	0.49
66	135	38	7/350	18.56	10/60	2.20	11	0.63	11	0.48
67	135	38	6/345	18.53	10/70	2.60	14	0.81	12	0.52
68	135	37	7/455	23.12	8/53	1.98	13	0.71	12	0.49
69	135	38	7/381	19.89	9/62	2.34	14	0.80	12	0.50
70	135	36	7/389	19.86	10/65	2.31	14	0.79	13	0.53
71	135	35	7/472	23.91	7/55	2.07	15	0.82	13	0.52
72	135	35	7/357	17.73	9/55	1.93	11	0.61	11	0.44
73	135	34	7/393	19.29	9/62	2.20	15	0.81	13	0.50
74	135	35	7/399	20.11	11/61	2.12	13	0.71	12	0.49
75	135	35	7/472	23.92	7/55	2.07	15	0.82	13	0.52
76	135	35	7/356	17.67	10/59	2.06	13	0.71	12	0.48
77	135	36	7/459	23.39	9/63	2.30	14	0.76	12	0.49
78	135	37	6/379	19.85	7/55	2.10	13	0.73	12	0.51
79	137	39	7/425	23.02	13/83	3.19	15	0.87	13	0.56
80	137	40	7/415	22.99	10/65	2.63	14	0.84	12	0.53
81	147	45	8/398	26.92	9/66	3.17	13	0.95	12	0.63
82	135	38	7/381	19.86	10/66	2.43	14	0.80	12	0.50
83	134	37	7/365	18.55	10/58	2.07	13	0.72	12	0.50
84	134	37	7/376	19.15	7/55	2.07	14	0.77	12	0.51
85	134	37	7/418	21.52	8/58	2.15	14	0.77	12	0.50
86	135	39	7/410	21.69	8/59	2.30	14	0.80	12	0.52
87	137	37	7/373	19.62	9/65	2.45	16	0.92	13	0.54
88	128	36	7/385	17.27	7/57	1.99	16	0.79	13	0.51
89	128	36	7/395	17.23	10/60	1.98	13	0.63	12	0.46
90	137	39	7/338	18.27	10/61	2.34	11	0.65	11	0.48
91	128	37	6/316	14.37	11/71	2.40	14	0.72	13	0.50
92	137	38	6/399	21.73	13/79	2.96	13	0.76	12	0.51
93	137	38	8/370	19.65	12/65	2.35	11	0.64	11	0.48
94	137	37	7/372	19.61	11/74	2.80	15	0.85	13	0.55
95	137	37	6/384	20.61	11/67	2.48	14	0.81	12	0.51
96	137	39	7/389	21.29	10/68	2.62	13	0.76	12	0.53

Table 6: Results on Test Set 2: Problems 97-143

Prob	Size		KT		MKT		F1PD		F2PD	
No.	m	n	Iter	Time (sec)	Iter	Time (sec)	Iter	Time (sec)	Iter	Time (sec)
97	137	38	5/346	18.75	8/59	2.35	15	0.86	12	0.51
98	137	39	6/364	20.08	10/63	2.47	13	0.78	12	0.52
99	137	39	7/425	23.03	13/83	3.15	15	0.87	13	0.56
100	137	38	7/394	21.05	11/67	2.52	15	0.86	13	0.56
101	128	37	7/350	15.76	11/68	2.30	15	0.76	13	0.50
102	128	36	6/306	13.53	11/69	2.29	16	0.79	14	0.54
103	137	40	7/416	23.02	10/65	2.58	14	0.84	12	0.53
104	147	47	7/353	26.40	10/70	3.44	15	1.18	13	0.76
105	147	48	9/401	31.14	12/80	3.96	24	1.94	18	1.07
106	147	47	9/428	32.29	12/83	4.07	20	1.58	16	0.93
107	147	46	7/419	30.25	10/73	3.59	14	1.06	12	0.68
108	147	46	9/446	31.81	12/78	3.73	19	1.43	15	0.83
109	147	45	7/359	25.30	9/70	3.43	14	1.05	13	0.75
110	147	44	7/552	39.44	7/59	2.98	16	1.18	14	0.76
111	147	47	7/396	29.70	8/68	3.46	15	1.17	13	0.77
112	147	48	9/438	33.47	12/82	4.06	16	1.30	14	0.84
113	147	48	7/432	33.53	10/70	3.54	14	1.13	12	0.73
114	147	49	9/396	29.62	11/71	3.56	20	1.60	16	0.94
115	147	48	8/425	31.56	12/80	3.94	17	1.33	14	0.80
116	147	47	7/410	30.40	9/66	3.32	18	1.39	14	0.82
117	147	48	9/420	30.96	12/83	4.10	24	1.87	19	1.10
118	147	47	7/407	30.10	7/54	2.77	17	1.32	13	0.76
119	147	47	9/431	31.60	11/81	3.99	21	1.64	17	0.97
120	147	46	0/164	12.65	8/69	3.55	25	1.91	18	1.01
121	147	46	8/411	29.85	8/70	3.53	20	1.53	16	0.91
122	147	48	9/391	28.62	12/79	3.89	21	1.66	17	0.98
123	147	47	9/449	32.80	9/74	3.73	21	1.63	17	0.97
124	147	46	8/423	30.63	8/68	3.40	18	1.37	15	0.84
125	147	47	9/403	29.17	10/75	3.71	22	1.69	17	0.98
126	147	46	8/395	28.53	13/84	4.01	19	1.45	16	0.90
127	147	43	7/521	36.15	7/67	3.29	19	1.37	15	0.79
128	147	46	9/419	30.14	11/77	3.73	24	1.84	18	1.01
129	147	45	9/402	28.43	11/74	3.52	23	1.73	17	0.95
130	147	45	7/416	30.03	7/59	3.00	16	1.21	13	0.73
131	147	49	9/389	29.05	12/82	4.11	23	1.87	17	1.02
132	147	48	7/373	27.86	12/80	3.96	18	1.41	15	0.87
133	147	47	7/410	30.43	9/66	3.32	17	1.32	14	0.80
134	147	48	9/419	30.81	12/83	4.11	24	1.88	19	1.10
135	147	47	7/400	29.58	7/54	2.81	17	1.32	14	0.81
136	147	47	9/432	31.51	11/81	3.98	20	1.56	17	0.97
137	147	46	8/419	30.38	8/67	3.36	18	1.38	15	0.85
138	147	49	9/410	30.74	12/86	4.34	21	1.67	17	1.01
139	147	48	8/402	29.83	10/74	3.71	18	1.41	15	0.87
140	147	47	8/420	30.66	10/74	3.66	18	1.42	15	0.85
141	147	48	6/381	28.65	8/62	3.24	17	1.34	14	0.81
142	147	50	12/514	38.50	12/86	4.40	27	2.17	20	1.18
143	147	49	8/415	31.70	8/77	4.11	21	1.67	18	1.06