SIAM J. OPTIM.
Vol. 17, No. 1, pp. 287–310
© 2006 Society for Industrial and Applied Mathematics

AN ITERATIVE SOLVER-BASED INFEASIBLE PRIMAL-DUAL PATH-FOLLOWING ALGORITHM FOR CONVEX QUADRATIC PROGRAMMING*

ZHAOSONG LU†, RENATO D. C. MONTEIRO‡, AND JEROME W. O'NEAL§

Abstract. In this paper we develop a long-step primal-dual infeasible path-following algorithm for convex quadratic programming (CQP) whose search directions are computed by means of a preconditioned iterative linear solver. We propose a new linear system, which we refer to as the *augmented normal equation* (ANE), to determine the primal-dual search directions. Since the condition number of the ANE coefficient matrix may become large for degenerate CQP problems, we use a maximum weight basis preconditioner introduced in [A. R. L. Oliveira and D. C. Sorensen, *Linear Algebra Appl.*, 394 (2005), pp. 1–24; M. G. C. Resende and G. Veiga, *SIAM J. Optim.*, 3 (1993), pp. 516–537; P. Vaida, *Solving Linear Equations with Symmetric Diagonally Dominant Matrices by Constructing Good Preconditioners*, Tech. report, Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, 1990] to precondition this matrix. Using a result obtained in [R. D. C. Monteiro, J. W. O'Neal, and T. Tsuchiya, *SIAM J. Optim.*, 15 (2004), pp. 96–100], we establish a uniform bound, depending only on the CQP data, for the number of iterations needed by the iterative linear solver to obtain a sufficiently accurate solution to the ANE. Since the iterative linear solver can generate only an approximate solution to the ANE, this solution does not yield a primal-dual search direction satisfying all equations of the primal-dual Newton system. We propose a way to compute an inexact primal-dual search direction so that the equation corresponding to the primal residual is satisfied exactly, while the one corresponding to the dual residual contains a manageable error which allows us to establish a polynomial bound on the number of iterations of our method.

Key words. convex quadratic programming, iterative linear solver, maximum weight basis preconditioner, primal-dual path-following methods, interior-point methods, augmented normal equation, inexact search directions, polynomial convergence

AMS subject classifications. 65F10, 65F35, 90C20, 90C25, 90C51

DOI. 10.1137/04060771X

1. Introduction.

In this paper we develop an interior-point long-step primal-dual infeasible path-following (PDIPF) algorithm for convex quadratic programming (CQP) whose search directions are computed by means of an iterative linear solver. We will refer to this algorithm as an *inexact* algorithm, in the sense that the Newton system which determines the search direction will be solved only approximately at each iteration. The problem we consider is

$$\min_x \left\{ \frac{1}{2} x^T Q x + c^T x : \quad Ax = b,\ x \geq 0 \right\}, \tag{1}$$

*Received by the editors May 3, 2004; accepted for publication (in revised form) December 5, 2005; published electronically May 19, 2006.
http://www.siam.org/journals/siopt/17-1/60771.html

†Department of Mathematical Sciences, Carnegie Mellon University, Pittsburgh, PA 15213-3890 (zhaolu@andrew.cmu.edu). This author was supported in part by NSF grant CCR-0203113 and ONR grant N00014-03-1-0401.

‡School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205 (monteiro@isye.gatech.edu). This author was supported in part by NSF grants CCR-0203113, CCF-0430644, and INT-9910084 and ONR grants N00014-03-1-0401 and N00014-05-1-0183.

§Research, Modelling, & Design Group, Delta Technology, 1001 International Boulevard, Department 709, Atlanta, GA 30354 (jerome.w.oneal@delta.com). This author was supported in part by the NDSEG Fellowship Program sponsored by the Department of Defense.

where the data are $Q \in \Re^{n\times n}$, $A \in \Re^{m\times n}$, $b \in \Re^m$, and $c \in \Re^n$, and the decision vector is $x \in \Re^n$. We also assume that Q is positive semidefinite and that a factorization $Q = VE^2V^T$ is explicitly given, where $V \in \Re^{n\times l}$ and E is an $l \times l$ positive diagonal matrix.

A similar algorithm for solving the special case of linear programming (LP), i.e., problem (1) with $Q = 0$, was developed by Monteiro and O'Neal in [16]. The algorithm studied in [16] is essentially the long-step PDIPF algorithm studied in [9, 28], the only difference being that the search directions are computed by means of an iterative linear solver. We refer to the iterations of the iterative linear solver as the *inner iterations* and to the ones performed by the interior-point method itself as the *outer iterations*. The main step of the algorithm studied in [9, 16, 28] is the computation of the primal-dual search direction $(\Delta x, \Delta s, \Delta y)$, whose Δy component can be found by solving a system of the form $AD^2A^T\Delta y = g$, referred to as the *normal equation*, where $g \in \Re^m$ and the positive diagonal matrix D depends on the current primal-dual iterate. In contrast to [9, 28], the algorithm studied in [16] uses an iterative linear solver to obtain an approximate solution to the normal equation. Since the condition number of the normal matrix AD^2A^T may become excessively large on degenerate LP problems (see e.g., [13]), the maximum weight basis (MWB) preconditioner T introduced in [19, 22, 25] is used to better condition this matrix, and an approximate solution of the resulting equivalent system with coefficient matrix $TAD^2A^TT^T$ is then computed. By using a result obtained in [17], which establishes that the condition number of $TAD^2A^TT^T$ is uniformly bounded by a quantity depending only on A, Monteiro and O'Neal [16] showed that the number of inner iterations of the algorithm in [16] can be uniformly bounded by a constant depending on n and A.

In the case of CQP, the standard normal equation takes the form

$$A(Q + X^{-1}S)^{-1}A^T\Delta y = g \tag{2}$$

for some vector g. When Q is not diagonal, the matrix $(Q+X^{-1}S)^{-1}$ is not diagonal, and hence the coefficient matrix of (2) does not have the form required for the result of [17] to hold. To remedy this difficulty, we develop in this paper a new linear system, referred to as the *augmented normal equation* (ANE), to determine a portion of the primal-dual search direction. This equation has the form $\tilde{A}\tilde{D}^2\tilde{A}^Tu = w$, where $w \in \Re^{m+l}$, $\tilde{D}$ is an $(n + l) \times (n + l)$ positive diagonal matrix, and $\tilde{A}$ is a 2×2 block matrix of dimension $(m+l)\times(n+l)$ whose blocks consist of A, V^T, the zero matrix, and the identity matrix (see (21)). As was done in [16], a MWB preconditioner $\tilde{T}$ for the ANE is computed and an approximate solution of the resulting preconditioned equation with coefficient matrix $\tilde{T}\tilde{A}\tilde{D}^2\tilde{A}^T\tilde{T}^T$ is generated using an iterative linear solver. Using the result of [17], which claims that the condition number of $\tilde{T}\tilde{A}\tilde{D}^2\tilde{A}^T\tilde{T}^T$ is uniformly bounded regardless of $\tilde{D}$, we obtain a uniform bound (depending only on $\tilde{A}$) on the number of inner iterations performed by the iterative linear solver to find a desirable approximate solution to the ANE (see Theorem 3.5).

Since the iterative linear solver can generate only an approximate solution to the ANE, it is clear that not all equations of the Newton system, which determines the primal-dual search direction, can be satisfied simultaneously. In the context of LP, Monteiro and O'Neal [16] proposed a recipe to compute an inexact primal-dual search direction so that the equations of the Newton system corresponding to the primal and dual residuals were both satisfied. In the context of CQP, such an approach is no longer possible. Instead, we propose a way to compute an inexact primal-dual search direction so that the equation corresponding to the primal residual is satisfied exactly, while the one corresponding to the dual residual contains a manageable error which allows us to establish a polynomial bound on the number of outer iterations of our method. Interestingly, the presence of this error on the dual residual equation implies that the primal and dual residuals go to zero at different rates. This is a unique feature of the convergence analysis of our algorithm in that it contrasts with the analysis of other interior-point PDIPF algorithms, where the primal and dual residuals are required to go to zero at the same rate.

The use of inexact search directions in interior-point methods has been extensively studied in the context of cone programming problems (see e.g., [1, 2, 7, 11, 12, 15, 18, 29]). Moreover, the use of iterative linear solvers to compute the primal-dual Newton search directions of interior-point path-following algorithms has also been extensively investigated in [1, 3, 4, 7, 12, 18, 19, 20, 22, 24]. For feasibility problems of the form $\{x \in \mathcal{H}_1 : \mathcal{A}x = b, x \in \mathcal{C}\}$, where $\mathcal{H}_1$ and $\mathcal{H}_2$ are Hilbert spaces, $\mathcal{C} \subseteq \mathcal{H}_1$ is a closed convex cone satisfying some mild assumptions, and $\mathcal{A} : \mathcal{H}_1 \to \mathcal{H}_2$ is a continuous linear operator, Renegar [21] has proposed an interior-point method where the Newton system that determines the search directions is approximately solved by performing a uniformly bounded number of iterations of the conjugate gradient (CG) method. To our knowledge, no one has used the ANE system in the context of CQP to obtain either an exact or inexact primal-dual search direction.

Our paper is organized as follows. In subsection 1.1, we give the terminology and notation which will be used throughout our paper. Section 2 describes the outer iteration framework for our algorithm and the complexity results we have obtained for it, along with presenting the ANE as a means to determine the search direction. In section 3, we discuss the use of iterative linear solvers to obtain a suitable approximate solution to the ANE and the construction of an inexact search direction based on this solution. Section 4 gives the proofs of the results presented in sections 2 and 3. Finally, we present some concluding remarks in section 5.

1.1. Terminology and notation. Throughout this paper, uppercase roman letters denote matrices, lowercase roman letters denote vectors, and lowercase Greek letters denote scalars. We let $\Re^n$, $\Re^n_+$, and $\Re^n_{++}$ denote the set of n-dimensional vectors having real, nonnegative real, and positive real components, respectively. Also, we let $\Re^{m\times n}$ denote the set of $m \times n$ matrices with real entries. For a vector $v \in \Re^n$, we let $|v|$ denote the vector whose ith component is $|v_i|$ for every $i = 1, \ldots, n$, and we let $\mathrm{Diag}(v)$ denote the diagonal matrix whose ith diagonal element is v_i for every $i = 1, \ldots, n$. In addition, given vectors $u \in \Re^m$ and $v \in \Re^n$, we denote by (u, v) the vector $(u^T, v^T)^T \in \Re^{m+n}$.

Certain matrices bear special notation, namely the matrices X, ΔX, S, D, and $\tilde{D}$. These matrices are the diagonal matrices corresponding to the vectors x, Δx, s, d, and $\tilde{d}$, respectively, as described in the previous paragraph. The symbol 0 will be used to denote a scalar, vector, or matrix of all zeros; its dimensions should be clear from the context. Also, we denote by e the vector of all 1's, and by I the identity matrix; their dimensions should be clear from the context.

For a symmetric positive definite matrix W, we denote its condition number by $\kappa(W)$, i.e., its maximum eigenvalue divided by its minimum eigenvalue. We will denote sets by uppercase calligraphic letters (e.g., $\mathcal{B}$, $\mathcal{N}$). For a finite set $\mathcal{B}$, we denote its cardinality by $|\mathcal{B}|$. Given a matrix $A \in \Re^{m\times n}$ and an ordered set $\mathcal{B} \subseteq \{1, \ldots, n\}$, we let $A_\mathcal{B}$ denote the submatrix whose columns are $\{A_i : i \in \mathcal{B}\}$ arranged in the same order as $\mathcal{B}$. Similarly, given a vector $v \in \Re^n$ and an ordered set $\mathcal{B} \subseteq \{1, \ldots, n\}$, we let $v_\mathcal{B}$ denote the subvector consisting of the elements $\{v_i : i \in \mathcal{B}\}$ arranged in the same order as $\mathcal{B}$.

We will use several different norms throughout the paper. For a vector $z \in \Re^n$, $\|z\| = \sqrt{z^T z}$ is the Euclidian norm, $\|z\|_1 = \sum_{i=1}^n |z_i|$ is the "1-norm," and $\|z\|_\infty = \max_{i=1,\dots,n} |z_i|$ is the "infinity norm." For a matrix $V \in \Re^{m\times n}$, $\|V\|$ denotes the operator norm associated with the Euclidian norm: $\|V\| = \max_{z:\|z\|=1} \|Vz\|$. Finally, $\|V\|_F$ denotes the Frobenius norm: $\|V\|_F = (\sum_{i=1}^m \sum_{j=1}^n V_{ij}^2)^{1/2}$.

2. Outer iteration framework.

In this section, we introduce our PDIPF algorithm based on a class of inexact search directions and discuss its iteration complexity. This section is divided into two subsections. In subsection 2.1, we discuss an exact PDIPF algorithm, which will serve as the basis for the inexact PDIPF algorithm given in subsection 2.2, and we give its iteration complexity result. We also present an approach based on the ANE to determine the Newton search direction for the exact algorithm. To motivate the class of inexact search directions used by our inexact PDIPF algorithm, we describe in subsection 2.2 a framework for computing an inexact search direction based on an approximate solution to the ANE. We then introduce the class of inexact search directions, state a PDIPF algorithm based on it, and give its iteration complexity result.

2.1. An exact PDIPF algorithm and the ANE.

Consider the following primal-dual pair of CQP problems:

$$\min_{x} \left\{ \frac{1}{2} x^T V E^2 V^T x + c^T x : \quad Ax = b,\ x \geq 0 \right\}, \tag{3}$$

$$\max_{(\hat{x},s,y)} \left\{ -\frac{1}{2} \hat{x}^T V E^2 V^T \hat{x} + b^T y : \quad A^T y + s - V E^2 V^T \hat{x} = c,\ s \geq 0 \right\}, \tag{4}$$

where the data are $V \in \Re^{n\times l}$, $E \in \mathrm{Diag}(\Re^l_{++})$, $A \in \Re^{m\times n}$, $b \in \Re^m$, and $c \in \Re^n$, and the decision variables are $x \in \Re^n$ and $(\hat{x}, s, y) \in \Re^n \times \Re^n \times \Re^m$. We observe that the Hessian matrix Q is already given in factored form $Q = VE^2V^T$.

It is well known that if x^* is an optimal solution for (3) and $(\hat{x}^*, s^*, y^*)$ is an optimal solution for (4), then (x^*, s^*, y^*) is also an optimal solution for (4). Now, let $\mathcal{S}$ denote the set of all vectors $w := (x, s, y, z) \in \Re^{2n+m+l}$ satisfying

$$Ax = b, \quad x \geq 0, \tag{5}$$

$$A^T y + s + Vz = c, \quad s \geq 0, \tag{6}$$

$$Xs = 0, \tag{7}$$

$$EV^T x + E^{-1} z = 0. \tag{8}$$

It is clear that $w \in \mathcal{S}$ if and only if x is optimal for (3), (x, s, y) is optimal for (4), and $z = -E^2 V^T x$. (Throughout this paper, the symbol w will always denote the quadruple (x, s, y, z), where the vectors lie in the appropriate dimensions; similarly, $\Delta w = (\Delta x, \Delta s, \Delta y, \Delta z)$, $w^k = (x^k, s^k, y^k, z^k)$, $\bar{w} = (\bar{x}, \bar{s}, \bar{y}, \bar{z})$, etc.)

We observe that the presentation of the PDIPF algorithm based on exact Newton search directions in this subsection differs from the classical way of presenting it in that we introduce an additional variable z as above. Clearly, it is easy to see that the variable z is completely redundant and can be eliminated, thereby reducing the method described below to the usual way of presenting it. The main reason for introducing the variable z is due to the development of the ANE presented at the end of this subsection.

We will make the following two assumptions throughout the paper.

Assumption 1. A has full row rank.

Assumption 2. The set $\mathcal{S}$ is nonempty.

For a point $w \in \Re^{2n}_{++} \times \Re^{m+l}$, let us define

$$\mu := \mu(w) \;=\; x^T s/n, \tag{9}$$
$$r_p := r_p(w) \;=\; Ax - b, \tag{10}$$
$$r_d := r_d(w) \;=\; A^T y + s + Vz - c, \tag{11}$$
$$r_V := r_V(w) \;=\; EV^T x + E^{-1} z, \tag{12}$$
$$r := r(w) \;=\; (r_p(w), r_d(w), r_V(w)). \tag{13}$$

Moreover, given $\gamma \in (0,1)$ and an initial point $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$, we define the following neighborhood of the central path:

$$\mathcal{N}_{w^0}(\gamma) := \left\{ w \in \Re^{2n}_{++} \times \Re^{m+l} : Xs \geq (1-\gamma)\mu e,\ r = \eta r^0 \right. \\ \left. \text{for some } 0 \leq \eta \leq \min\left[1, \frac{\mu}{\mu_0}\right] \right\}, \tag{14}$$

where $r := r(w)$, $r^0 := r(w^0)$, $\mu := \mu(w)$, and $\mu_0 := \mu(w^0)$.

We are now ready to state the PDIPF algorithm based on exact Newton search directions.

Exact PDIPF algorithm.

1. **Start:** Let $\epsilon > 0$ and $0 < \underline{\sigma} \leq \overline{\sigma} < 1$ be given. Let $\gamma \in (0,1)$ and $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ be such that $w^0 \in \mathcal{N}_{w^0}(\gamma)$. Set $k = 0$.
2. **While** $\mu_k := \mu(w^k) > \epsilon$ **do**
   (a) Let $w := w^k$ and $\mu := \mu_k$; choose $\sigma := \sigma_k \in [\underline{\sigma}, \overline{\sigma}]$.
   (b) Let $\Delta w = (\Delta x, \Delta s, \Delta y, \Delta z)$ denote the solution of the linear system

$$A\Delta x = -r_p, \tag{15}$$
$$A^T \Delta y + \Delta s + V\Delta z = -r_d, \tag{16}$$
$$X\Delta s + S\Delta x = -Xs + \sigma\mu e, \tag{17}$$
$$EV^T \Delta x + E^{-1}\Delta z = -r_V. \tag{18}$$

   (c) Let $\tilde{\alpha} = \operatorname{argmax}\{\alpha \in [0,1] : w + \alpha' \Delta w \in \mathcal{N}_{w^0}(\gamma),\ \forall \alpha' \in [0,\alpha]\}$.
   (d) Let $\bar{\alpha} = \operatorname{argmin}\left\{(x + \alpha\Delta x)^T (s + \alpha \Delta s) : \alpha \in [0, \tilde{\alpha}]\right\}$.
   (e) Let $w^{k+1} = w + \bar{\alpha}\Delta w$, and set $k \leftarrow k+1$.

   End (while)

A proof of the following result, under slightly different assumptions, can be found in [28].

Theorem 2.1. *Assume that the constants γ, $\underline{\sigma}$, and $\overline{\sigma}$ are such that*

$$\max\left\{\gamma^{-1}, (1-\gamma)^{-1}, \underline{\sigma}^{-1}, (1-\overline{\sigma})^{-1}\right\} = \mathcal{O}(1),$$

and that the initial point $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ satisfies $(x^0, s^0) \geq (x^*, s^*)$ for some $w^* \in \mathcal{S}$. Then, the exact PDIPF algorithm finds an iterate $w^k \in \Re^{2n}_{++} \times \Re^{m+l}$ satisfying $\mu_k \leq \epsilon\mu_0$ and $\|r^k\| \leq \epsilon \|r^0\|$ within $\mathcal{O}(n^2 \log(1/\epsilon))$ iterations.

A few approaches have been suggested in the literature for computing the Newton search direction (15)–(18). Instead of using one of them, we will discuss below a new approach, referred to in this paper as the ANE approach, that we believe to be suitable not only for direct solvers but especially for iterative linear solvers, as we will see in section 3.

Let us begin by defining the following matrices:

$$D := X^{1/2}S^{-1/2}, \tag{19}$$

$$\tilde{D} := \begin{pmatrix} D & 0 \\ 0 & E^{-1} \end{pmatrix} \in \Re^{(n+l)\times(n+l)}, \tag{20}$$

$$\tilde{A} := \begin{pmatrix} A & 0 \\ V^T & I \end{pmatrix} \in \Re^{(m+l)\times(n+l)}. \tag{21}$$

Suppose that we first solve the following system of equations for $(\Delta y, \Delta z)$:

$$\tilde{A}\tilde{D}^2\tilde{A}^T \begin{pmatrix} \Delta y \\ \Delta z \end{pmatrix} = \tilde{A} \begin{pmatrix} x - \sigma\mu S^{-1}e - D^2 r_d \\ 0 \end{pmatrix} + \begin{pmatrix} -r_p \\ -E^{-1}r_V \end{pmatrix} =: \ h. \tag{22}$$

This system is what we refer to as the ANE. Next, we obtain Δs and Δx according to

$$\Delta s = -r_d - A^T\Delta y - V\Delta z, \tag{23}$$

$$\Delta x = -D^2\Delta s - x + \sigma\mu S^{-1}e. \tag{24}$$

Clearly, the search direction $\Delta w = (\Delta x, \Delta s, \Delta y, \Delta z)$ computed as above satisfies (16) and (17) in view of (23) and (24). Moreover, it also satisfies (15) and (18) due to the fact that by (20)–(24), we have that

$$\begin{aligned}
\tilde{A}\begin{pmatrix} \Delta x \\ E^{-2}\Delta z \end{pmatrix} &= \tilde{A}\begin{pmatrix} -D^2\Delta s - x + \sigma\mu S^{-1}e \\ E^{-2}\Delta z \end{pmatrix} \\
&= \tilde{A}\begin{pmatrix} D^2 r_d + D^2A^T\Delta y + D^2V\Delta z - x + \sigma\mu S^{-1}e \\ E^{-2}\Delta z \end{pmatrix} \\
&= \tilde{A}\begin{pmatrix} D^2A^T\Delta y + D^2V\Delta z \\ E^{-2}\Delta z \end{pmatrix} + \tilde{A}\begin{pmatrix} D^2 r_d - x + \sigma\mu S^{-1}e \\ 0 \end{pmatrix} \\
&= \tilde{A}\tilde{D}^2\tilde{A}^T\begin{pmatrix} \Delta y \\ \Delta z \end{pmatrix} + \tilde{A}\begin{pmatrix} D^2 r_d - x + \sigma\mu S^{-1}e \\ 0 \end{pmatrix} \\
&= \begin{pmatrix} -r_p \\ -E^{-1}r_V \end{pmatrix}.
\end{aligned} \tag{25}$$

Theorem 2.1 assumes that Δw is the exact solution of (22), which is usually obtained by computing the Cholesky factorization of the coefficient matrix of the ANE. In this paper, we will consider a variant of the exact PDIPF algorithm whose search directions are approximate solutions of (22) and ways of determining these inexact search directions by means of a suitable preconditioned iterative linear solver.

2.2. An inexact PDIPF algorithm for CQP. In this subsection, we describe a PDIPF algorithm based on a family of search directions that are approximate solutions to (15)–(18) and discuss its iteration complexity properties.

Clearly, an approximate solution to the ANE can yield only an approximate solution to (15)–(18). In order to motivate the class of inexact search directions used by the PDIPF algorithm presented in this subsection, we present a framework for obtaining approximate solutions to (15)–(18) based on an approximate solution to the ANE.

Suppose that the ANE is solved only inexactly, i.e., that the vector $(\Delta y, \Delta z)$ satisfies

$$\tilde{A}\tilde{D}^2\tilde{A}^T \begin{pmatrix} \Delta y \\ \Delta z \end{pmatrix} = h + f \tag{26}$$

for some error vector f. If Δs and Δx were computed by (23) and (24), respectively, then it is clear that the search direction Δw would satisfy (16) and (17). However, (15) and (18) would not be satisfied, since by an argument similar to (25), we would have that

$$\begin{aligned} \tilde{A}\begin{pmatrix} \Delta x \\ E^{-2}\Delta z \end{pmatrix} &= \cdots = \tilde{A}\tilde{D}^2\tilde{A}^T \begin{pmatrix} \Delta y \\ \Delta z \end{pmatrix} + \tilde{A}\begin{pmatrix} D^2 r_d - x + \sigma\mu S^{-1}e \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} -r_p \\ -E^{-1}r_V \end{pmatrix} + f. \end{aligned}$$

Instead, suppose we use (23) to determine Δs as before, but now we determine Δx as

$$\Delta x = -D^2\Delta s - x + \sigma\mu S^{-1}e - S^{-1}p, \tag{27}$$

where the correction vector $p \in \Re^n$ will be required to satisfy some conditions which we will now describe.

To motivate the conditions on p, we note that (23), (26), and (27) imply that

$$\begin{aligned} &\tilde{A}\begin{pmatrix} \Delta x \\ E^{-2}\Delta z \end{pmatrix} + \begin{pmatrix} r_p \\ E^{-1}r_V \end{pmatrix} \\ &= \tilde{A}\begin{pmatrix} -D^2\Delta s - x + \sigma\mu S^{-1}e - S^{-1}p \\ E^{-2}\Delta z \end{pmatrix} + \begin{pmatrix} r_p \\ E^{-1}r_V \end{pmatrix} \\ &= \tilde{A}\begin{pmatrix} D^2 r_d + D^2A^T\Delta y + D^2V\Delta z - x + \sigma\mu S^{-1}e - S^{-1}p \\ E^{-2}\Delta z \end{pmatrix} + \begin{pmatrix} r_p \\ E^{-1}r_V \end{pmatrix} \\ &= \tilde{A}\tilde{D}^2\begin{pmatrix} A^T\Delta y + V\Delta z \\ \Delta z \end{pmatrix} + \tilde{A}\begin{pmatrix} D^2 r_d - x + \sigma\mu S^{-1}e \\ 0 \end{pmatrix} - \tilde{A}\begin{pmatrix} S^{-1}p \\ 0 \end{pmatrix} \\ &\qquad + \begin{pmatrix} r_p \\ E^{-1}r_V \end{pmatrix} \\ &= \tilde{A}\tilde{D}^2\tilde{A}^T\begin{pmatrix} \Delta y \\ \Delta z \end{pmatrix} + \tilde{A}\begin{pmatrix} D^2 r_d - x + \sigma\mu S^{-1}e \\ 0 \end{pmatrix} - \tilde{A}\begin{pmatrix} S^{-1}p \\ 0 \end{pmatrix} + \begin{pmatrix} r_p \\ E^{-1}r_V \end{pmatrix} \\ &= f - \tilde{A}\begin{pmatrix} S^{-1}p \\ 0 \end{pmatrix}. \end{aligned} \tag{28}$$

Based on the above equation, one is naturally tempted to choose p so that the right-hand side of (28) is zero, and consequently (15) and (18) are satisfied exactly. However, the existence of such p cannot be guaranteed and, even if it exists, its magnitude might not be sufficiently small to yield a search direction which is suitable for the development of a polynomially convergent algorithm. Instead, we consider an alternative approach where p is chosen so that the first component of (28) is zero and the second component is small. More specifically, by partitioning $f = (f_1, f_2) \in \Re^m \times \Re^l$, we choose $p \in \Re^n$ such that

$$AS^{-1}p = f_1. \tag{29}$$

It is clear that p is not uniquely defined. Note that (21) implies that (29) is equivalent to

$$f \; = \; \tilde{A} \begin{pmatrix} S^{-1}p \\ E^{-1}q \end{pmatrix}, \tag{30}$$

where $q := E(f_2 - V^T S^{-1}p)$. Then, using (21), (28), and (30), we conclude that

$$\begin{aligned} \tilde{A} \begin{pmatrix} \Delta x \\ E^{-2}\Delta z \end{pmatrix} + \begin{pmatrix} r_p \\ E^{-1}r_V \end{pmatrix} &= f - \tilde{A} \begin{pmatrix} S^{-1}p \\ E^{-1}q \end{pmatrix} + \tilde{A} \begin{pmatrix} 0 \\ E^{-1}q \end{pmatrix} \\ &= \tilde{A} \begin{pmatrix} 0 \\ E^{-1}q \end{pmatrix} = \begin{pmatrix} 0 \\ E^{-1}q \end{pmatrix}, \end{aligned} \tag{31}$$

from which we see that the first component of (28) is set to 0 and the second component is exactly $E^{-1}q$.

In view of (23), (27), and (31), the above construction yields a search direction Δw satisfying the following modified Newton system of equations:

$$A\Delta x = -r_p, \tag{32}$$

$$A^T \Delta y + \Delta s + V\Delta z = -r_d, \tag{33}$$

$$X\Delta s + S\Delta x = -Xs + \sigma\mu e - p, \tag{34}$$

$$EV^T \Delta x + E^{-1}\Delta z = -r_V + q. \tag{35}$$

As far as the outer iteration complexity analysis of our algorithm is concerned, all we require of our inexact search directions is that they satisfy (32)–(35) and that p and q be relatively small in the following sense.

DEFINITION 1. *Given a point $w \in \Re^{2n}_{++} \times \Re^{m+l}$ and positive scalars τ_p and τ_q, an inexact direction Δw is referred to as a (τ_p, τ_q)*-search direction *if it satisfies* (32)–(35) *for some p and q satisfying $\|p\|_\infty \le \tau_p \mu$ and $\|q\| \le \tau_q \sqrt{\mu}$, where μ is given by* (9).

We next define a generalized central path neighborhood which is used by our inexact PDIPF algorithm. Given a starting point $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ and parameters $\eta \ge 0$, $\gamma \in [0, 1]$, and $\theta > 0$, define the following set:

(36)

$$\mathcal{N}_{w^0}(\eta, \gamma, \theta) = \left\{ w \in \Re^{2n}_{++} \times \Re^{m+l} : \begin{array}{cc} Xs \ge (1-\gamma)\mu e, & (r_p, r_d) = \eta(r_p^0, r_d^0), \\ \|r_V - \eta r_V^0\| \le \theta\sqrt{\mu}, & \eta \le \mu/\mu_0 \end{array} \right\},$$

where $\mu = \mu(w)$, $\mu_0 = \mu(w^0)$, $r = r(w)$, and $r^0 = r(w^0)$. The generalized central path neighborhood is then given by

$$\mathcal{N}_{w^0}(\gamma, \theta) \; = \; \bigcup_{\eta \in [0,1]} \mathcal{N}_{w^0}(\eta, \gamma, \theta). \tag{37}$$

We observe that the neighborhood given by (37) agrees with the neighborhood given by (15) when $\theta = 0$.

We are now ready to state our inexact PDIPF algorithm.

Inexact PDIPF algorithm.

1. **Start:** Let $\epsilon > 0$ and $0 < \underline{\sigma} \le \overline{\sigma} < 4/5$ be given. Choose $\gamma \in (0,1)$, $\theta > 0$, and $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ such that $w^0 \in \mathcal{N}_{w^0}(\gamma,\theta)$. Set $k=0$.
2. **While** $\mu_k := \mu(w^k) > \epsilon$ **do**
   (a) Let $w := w^k$ and $\mu := \mu_k$; choose $\sigma \in [\underline{\sigma}, \overline{\sigma}]$.
   (b) Set

$$\tau_p = \gamma\sigma/4 \quad \text{and} \tag{38}$$

$$\tau_q = \left[\sqrt{1+(1-0.5\gamma)\,\sigma} - 1\right]\theta. \tag{39}$$

   (c) Set $r_p = Ax - b$, $r_d = A^T y + s + Vz - c$, $r_V = EV^T x + E^{-1} z$, and $\eta = \|r_p\|/\|r_p^0\|$.
   (d) Compute a (τ_p, τ_q)-search direction Δw.
   (e) Compute $\tilde{\alpha} := \operatorname{argmax}\{\alpha \in [0,1] : w + \alpha'\Delta w \in \mathcal{N}_{w^0}(\gamma,\theta),\ \forall \alpha' \in [0,\alpha]\}$.
   (f) Compute $\bar{\alpha} := \operatorname{argmin}\{(x+\alpha\Delta x)^T (s+\alpha\Delta s) : \alpha \in [0,\tilde{\alpha}]\}$.
   (g) Let $w^{k+1} = w + \bar{\alpha}\Delta w$, and set $k \leftarrow k+1$.

   End (while)

The following result gives a bound on the number of iterations needed by the inexact PDIPF algorithm to obtain an ϵ-solution to the KKT conditions (5)–(8). Its proof will be given in subsection 4.2.

Theorem 2.2. *Assume that the constants γ, $\underline{\sigma}$, $\overline{\sigma}$, and θ are such that*

$$\max\left\{\gamma^{-1}, (1-\gamma)^{-1}, \underline{\sigma}^{-1}, \left(1-\frac{5}{4}\overline{\sigma}\right)^{-1}\right\} = \mathcal{O}(1), \quad \theta = \mathcal{O}(\sqrt{n}), \tag{40}$$

and that the initial point $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ satisfies $(x^0, s^0) \ge (x^*, s^*)$ for some $w^* \in \mathcal{S}$. Then, the inexact PDIPF algorithm generates an iterate $w^k \in \Re^{2n}_{++} \times \Re^{m+l}$ satisfying $\mu_k \le \epsilon\mu_0$, $\|(r_p^k, r_d^k)\| \le \epsilon\|(r_p^0, r_d^0)\|$, and $\|r_V^k\| \le \epsilon\|r_V^0\| + \epsilon^{1/2}\theta\mu_0^{1/2}$ within $\mathcal{O}\left(n^2\log(1/\epsilon)\right)$ iterations.

3. Determining an inexact search direction via an iterative solver.

The results in subsection 2.2 assume we can obtain a (τ_p, τ_q)-search direction Δw, where τ_p and τ_q are given by (38) and (39), respectively. In this section, we will describe a way to obtain a (τ_p, τ_q)-search direction Δw using a uniformly bounded number of iterations of a suitable preconditioned iterative linear solver applied to the ANE. It turns out that the construction of this Δw is based on the recipe given at the beginning of subsection 2.2, together with a specific choice of the perturbation vector p.

This section is divided into two subsections. In subsection 3.1, we introduce the MWB preconditioner which will be used to precondition the ANE. In addition, we also introduce a family of iterative linear solvers used to solve the preconditioned ANE. Subsection 3.2 gives a specific approach for constructing a pair (p, q) satisfying (30), and an approximate solution to the ANE so that the recipe described at the beginning of subsection 2.2 yields a (τ_p, τ_q)-search direction Δw. It also provides a uniform bound on the number of iterations that any member of the family of iterative linear solvers needs to perform to obtain such a direction Δw when applied to the preconditioned ANE.

3.1. MWB preconditioner and a family of solvers.

In this subsection we introduce the MWB preconditioner, and we discuss its use as a preconditioner in solving the ANE via a family of iterative linear solvers. Since the condition number of the ANE matrix $\tilde{A}\tilde{D}^2\tilde{A}^T$ may "blow up" for points w near an optimal solution, the direct application of a generic iterative linear solver for solving the ANE without first preconditioning it is generally not effective. We discuss a natural remedy to this problem which consists of using a preconditioner $\tilde{T}$, namely the MWB preconditioner, such that $\kappa(\tilde{T}\tilde{A}\tilde{D}^2\tilde{A}^T\tilde{T}^T)$ remains uniformly bounded regardless of the iterate w. Finally, we analyze the complexity of the resulting approach to obtain a suitable approximate solution to the ANE.

We start by describing the MWB preconditioner. Its construction essentially consists of building a basis B of $\tilde{A}$ which gives higher priority to the columns of $\tilde{A}$ corresponding to larger diagonal elements of $\tilde{D}$. More specifically, the MWB preconditioner is determined by the following algorithm.

Maximum weight basis algorithm.

Start: Given $\tilde{d} \in \Re^{(n+l)}_{++}$, and $\tilde{A} \in \Re^{(m+l)\times(n+l)}$ such that $\text{rank}(\tilde{A}) = m + l$,

1. Order the elements of $\tilde{d}$ so that $\tilde{d}_1 \geq \cdots \geq \tilde{d}_{n+l}$; order the columns of $\tilde{A}$ accordingly.
2. Let $\mathcal{B} = \emptyset$, $j = 1$.
3. **While** $|\mathcal{B}| < m + l$ **do**
   (a) If $\tilde{A}_j$ is linearly independent of $\{\tilde{A}_i : i \in \mathcal{B}\}$, set $\mathcal{B} \leftarrow \mathcal{B} \cup \{j\}$.
   (b) $j \leftarrow j + 1$.
4. Return to the original ordering of $\tilde{A}$ and $\tilde{d}$; determine the set $\mathcal{B}$ according to this ordering and set $\mathcal{N} := \{1, \ldots, n + l\}\backslash\mathcal{B}$.
5. Set $B := \tilde{A}_{\mathcal{B}}$ and $\tilde{D}_{\mathcal{B}} := \text{Diag}(\tilde{d}_{\mathcal{B}})$.
6. Let $\tilde{T} = \tilde{T}(\tilde{A}, \tilde{d}) := \tilde{D}_{\mathcal{B}}^{-1}B^{-1}$.

end

Note that the above algorithm can be applied to the matrix $\tilde{A}$ defined in (21) since this matrix has full row rank due to Assumption 1. The MWB preconditioner was originally proposed by Vaidya [25] and Resende and Veiga [22] in the context of the minimum cost network flow problem. In this case, $\tilde{A} = A$ is the node-arc incidence matrix of a connected digraph (with one row deleted to ensure that $\tilde{A}$ has full row rank), the entries of $\tilde{d}$ are weights on the edges of the graph, and the set $\mathcal{B}$ generated by the above algorithm defines a maximum spanning tree on the digraph. Oliveira and Sorensen [19] later proposed the use of this preconditioner for general matrices $\tilde{A}$. Boman et al. [5] have proposed variants of the MWB preconditioner for diagonally dominant matrices, using the fact that they can be represented as $D_1 + AD_2A^T$, where D_1 and D_2 are nonnegative diagonal and positive diagonal matrices, respectively, and A is a node-arc incidence matrix.

For the purpose of stating the next result, we now introduce some notation. Let us define

$$\varphi_{\tilde{A}} := \max\{\|B^{-1}\tilde{A}\|_F : B \text{ is a basis of } \tilde{A}\}. \tag{41}$$

The constant $\varphi_{\tilde{A}}$ is related to the well-known condition number $\bar{\chi}_{\tilde{A}}$ (see [26]), defined as

$$\bar{\chi}_{\tilde{A}} := \sup\{\|\tilde{A}^T(\tilde{A}\tilde{E}\tilde{A}^T)^{-1}\tilde{A}\tilde{E}\| : \tilde{E} \in \text{Diag}(\Re^{(n+l)}_{++})\}.$$

Specifically, $\varphi_{\tilde{A}} \leq (n+l)^{1/2}\bar{\chi}_{\tilde{A}}$, in view of the facts that $\|C\|_F \leq (n+l)^{1/2}\,\|C\|$ for any matrix $C \in \Re^{(m+l)\times(n+l)}$ and, as shown in [23] and [26],

$$\bar{\chi}_{\tilde{A}} = \max\{\|B^{-1}\tilde{A}\| : B \text{ is a basis of } \tilde{A}\}.$$

The following result, which establishes the theoretical properties of the MWB preconditioner, follows as a consequence of Lemmas 2.1 and 2.2 of [17].

PROPOSITION 3.1. *Let $\tilde{T} = \tilde{T}(\tilde{A}, \tilde{d})$ be the preconditioner determined according to the maximum weight basis algorithm, and define $W := \tilde{T}\tilde{A}\tilde{D}^2\tilde{A}^T\tilde{T}^T$. Then, $\|\tilde{T}\tilde{A}\tilde{D}\| \leq \varphi_{\tilde{A}}$ and $\kappa(W) \leq \varphi^2_{\tilde{A}}$.*

Note that the bound $\varphi^2_{\tilde{A}}$ on $\kappa(W)$ is independent of the diagonal matrix $\tilde{D}$ and depends only on $\tilde{A}$. This will allow us to obtain a uniform bound on the number of iterations needed by any member of the family of iterative linear solvers described below to obtain a suitable approximate solution of (22). This topic is the subject of the remainder of this subsection.

Instead of dealing directly with (22), we consider the application of an iterative linear solver to the preconditioned ANE:

$$Wu = v, \tag{42}$$

where

$$W := \tilde{T}\tilde{A}\tilde{D}^2\tilde{A}^T\tilde{T}^T, \qquad v := \tilde{T}h. \tag{43}$$

For the purpose of our analysis below, the only thing we will assume regarding the iterative linear solver when applied to (42) is that it generates a sequence of iterates $\{u^j\}$ such that

$$\|v - Wu^j\| \;\leq\; c(\kappa)\left[1 - \frac{1}{\psi(\kappa)}\right]^j \|v - Wu^0\| \quad \forall\; j = 0, 1, 2, \ldots, \tag{44}$$

where c and ψ are positive, nondecreasing functions of $\kappa \equiv \kappa(W)$.

Examples of solvers which satisfy (44) include the steepest descent (SD) and CG methods, with the values for $c(\kappa)$ and $\psi(\kappa)$ given in Table 3.1.

TABLE 3.1

Solver	$c(\kappa)$	$\psi(\kappa)$
SD	$\sqrt{\kappa}$	$(\kappa + 1)/2$
CG	$2\sqrt{\kappa}$	$(\sqrt{\kappa} + 1)/2$

The justification for Table 3.1 follows from section 7.6 and Exercise 10 of section 8.8 of [14].

The following result gives an upper bound on the number of iterations that any iterative linear solver satisfying (44) needs to perform to obtain a ξ-*approximate solution of* (42), i.e., an iterate u^j such that $\|v - Wu^j\| \leq \xi\sqrt{\mu}$ for some constant $\xi > 0$.

PROPOSITION 3.2. *Let u^0 be an arbitrary starting point. Then, a generic iterative linear solver with a convergence rate given by* (44) *generates an iterate u^j satisfying $\|v - Wu^j\| \leq \xi\sqrt{\mu}$ in*

$$\mathcal{O}\left(\psi(\kappa)\log\left(\frac{c(\kappa)\|v - Wu^0\|}{\xi\sqrt{\mu}}\right)\right) \tag{45}$$

iterations, where $\kappa \equiv \kappa(W)$.

Proof. Let j be any iteration such that $\|v - Wu^j\| > \xi\sqrt{\mu}$. We use relation (44) and the fact that $1 + \omega \le e^\omega$ for all $\omega \in \Re$ to observe that

$$\xi\sqrt{\mu} < \|v - Wu^j\| \;\le\; c(\kappa)\left[1 - \frac{1}{\psi(\kappa)}\right]^j \|v - Wu^0\| \;\le c(\kappa)\exp\left\{\frac{-j}{\psi(\kappa)}\right\}\|v - Wu^0\|.$$

Rearranging the first and last terms of the inequality, it follows that

$$j \;<\; \psi(\kappa)\log\left(\frac{c(\kappa)\|v - Wu^0\|}{\xi\sqrt{\mu}}\right),$$

and the result is proven. ∎

From Proposition 3.2, it is clear that different choices of u^0 and ξ lead to different bounds on the number of iterations performed by the iterative linear solver. In subsection 3.2, we will describe a suitable way of selecting u^0 and ξ so that (i) the bound (45) is independent of the iterate w and (ii) the approximate solution $\tilde{T}^T u^j$ of the ANE, together with a suitable pair (p, q), yields a (τ_p, τ_q)-search direction Δw through the recipe described in subsection 2.2.

3.2. Computation of the inexact search direction Δw. In this subsection, we use the results of subsections 2.2 and 3.1 to build a (τ_p, τ_q)-search direction Δw, where τ_p and τ_q are given by (38) and (39), respectively. In addition, we describe a way of choosing u^0 and ξ which ensures that the number of iterations of an iterative linear solver satisfying (44) applied to the preconditioned ANE is uniformly bounded by a constant depending on n and $\varphi_{\tilde{A}}$.

Suppose that we solve (42) inexactly according to subsection 3.1. Then our final solution u^j satisfies $Wu^j - v = \tilde{f}$ for some vector $\tilde{f}$. Letting

$$\begin{pmatrix} \Delta y \\ \Delta z \end{pmatrix} = \tilde{T}^T u^j, \tag{46}$$

we easily see from (43) that (26) is satisfied with $f := \tilde{T}^{-1}\tilde{f}$. We can then apply the recipe of subsection 2.2 to this approximate solution, using the pair (p, q) which we will now describe.

First, note that (30) with f as defined above is equivalent to the system

$$\tilde{f} \;=\; \tilde{T}\tilde{A}\begin{pmatrix} S^{-1}p \\ E^{-1}q \end{pmatrix} \;=\; \tilde{T}\tilde{A}\tilde{D}\begin{pmatrix} (XS)^{-1/2} & 0 \\ 0 & I \end{pmatrix}\begin{pmatrix} p \\ q \end{pmatrix}. \tag{47}$$

Now, let $\mathcal{B} = (\mathcal{B}_1, \ldots, \mathcal{B}_{m+l})$ be the ordered set of basic indices computed by the MWB algorithm applied to the pair $(\tilde{A}, \tilde{d})$ and note that, by step 6 of this algorithm, the $\mathcal{B}_i$th column of $\tilde{T}\tilde{A}\tilde{D}$ is the ith unit vector for every $i = 1, \ldots, m+l$. Then, the vector $t \in \Re^{n+l}$ defined as $t_{\mathcal{B}_i} = \tilde{f}_i$ for $i = 1, \ldots, m+l$ and $t_j = 0$ for every $j \notin \{\mathcal{B}_1, \ldots, \mathcal{B}_{m+l}\}$ clearly satisfies

$$\tilde{f} \;=\; \tilde{T}\tilde{A}\tilde{D}\, t. \tag{48}$$

We then obtain a pair $(p, q) \in \Re^n \times \Re^l$ satisfying (30) by defining

$$\begin{pmatrix} p \\ q \end{pmatrix} \;:=\; \begin{pmatrix} (XS)^{1/2} & 0 \\ 0 & I \end{pmatrix} t. \tag{49}$$

It is clear from (49) and the fact that $\|t\| = \|\tilde{f}\|$ that

$$\|p\| \;\le\; \|XS\|^{1/2}\|\tilde{f}\|, \qquad \|q\| \;\le\; \|\tilde{f}\|. \tag{50}$$

As an immediate consequence of this relation, we obtain the following result.

LEMMA 3.3. *Suppose that* $w \in \Re^{2n}_{++} \times \Re^{m+l}$ *and positive scalars* τ_p *and* τ_q *are given. Assume that* u^j *is a* ξ*-approximate solution of* (42) *or, equivalently,* $\tilde{f} \le \xi\sqrt{\mu}$*, where* $\xi := \min\{n^{-1/2}\tau_p, \tau_q\}$*. Let* Δw *be determined according to the recipe given in subsection* 2.2 *using the approximate solution* (46) *and the pair* (p, q) *given by* (49)*. Then* Δw *is a* (τ_p, τ_q)*-search direction.*

Proof. It is clear from the previous discussion that Δw and the pair (p, q) satisfy (32)–(35). Next, relation (50) and the facts that $\xi \le n^{-1/2}\tau_p$ and $\|XS\|^{1/2} \le \sqrt{n\mu}$ imply that

$$\|p\|_\infty \;\le\; \|p\| \;\le\; \|XS\|^{1/2}\|\tilde{f}\| \;\le\; \sqrt{n\mu}\,\xi\sqrt{\mu} \;\le\; \tau_p\mu.$$

Similarly, (50) and the fact that $\xi \le \tau_q$ imply that $\|q\| \le \tau_q\sqrt{\mu}$. Thus, Δw is a (τ_p, τ_q)-search direction as desired. ◻

Lemma (3.3) implies that to construct a (τ_p, τ_q)-search direction Δw as in step 2(d) of the inexact PDIPF algorithm, it suffices to find a ξ-approximate solution to (42), where

$$\xi \;:=\; \min\left\{\frac{\gamma\sigma}{4\sqrt{n}}\,,\; \left[\sqrt{1+\left(1-\frac{\gamma}{2}\right)\sigma}-1\right]\theta\right\}. \tag{51}$$

We next describe a suitable way of selecting u^0 so that the number of iterations required by an iterative linear solver satisfying (44) to find a ξ-approximate solution of (42) can be uniformly bounded by a universal constant depending only on the quantities n and $\varphi_{\tilde{A}}$. First, compute a point $\bar{w} = (\bar{x}, \bar{s}, \bar{y}, \bar{z})$ such that

$$\tilde{A}\begin{pmatrix} \bar{x} \\ E^{-2}\bar{z} \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix}, \qquad A^T\bar{y} + \bar{s} + V\bar{z} = c. \tag{52}$$

Note that vectors $\bar{x}$ and $\bar{z}$ satisfying the first equation in (52) can be easily computed once a basis of $\tilde{A}$ is available (e.g., the one computed by the maximum weight basis algorithm in the first outer iteration of the inexact PDIPF algorithm). Once $\bar{y}$ is arbitrarily chosen, a vector $\bar{s}$ satisfying the second equation of (52) is immediately available. We then define

$$u^0 \;=\; -\eta\,\tilde{T}^{-T}\begin{pmatrix} y^0 - \bar{y} \\ z^0 - \bar{z} \end{pmatrix}. \tag{53}$$

The following lemma gives a bound on the size of the initial residual $\|Wu^0 - v\|$. Its proof will be given in subsection 4.1.

LEMMA 3.4. *Assume that* $\tilde{T} = \tilde{T}(\tilde{A}, \tilde{d})$ *is given and that* $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ *and* $\bar{w}$ *are such that* $(x^0, s^0) \ge |(\bar{x}, \bar{s})|$ *and* $(x^0, s^0) \ge (x^*, s^*)$ *for some* $w^* \in \mathcal{S}$*. Further, assume that* $w \in \mathcal{N}_{w^0}(\gamma, \theta)$ *for some* $\gamma \in [0, 1]$ *and* $\theta > 0$*, and that* W*,* v*, and* u^0 *are given by* (43) *and* (53)*, respectively. Then, the initial residual in* (44) *satisfies* $\|v - Wu^0\| \le \Psi\sqrt{\mu}$*, where*

$$\Psi \;:=\; \left[\frac{7n + \theta^2/2}{\sqrt{1-\gamma}} + \theta\right]\varphi_{\tilde{A}}. \tag{54}$$

As an immediate consequence of Proposition 3.2 and Lemmas 3.3 and 3.4, we can bound the number of inner iterations required by an iterative linear solver satisfying (44) to yield a (τ_p, τ_q)-search direction Δw.

THEOREM 3.5. *Assume that ξ is defined in (51), where σ, γ, θ are such that*

$$\max\{\sigma^{-1}, \gamma^{-1}, (1-\gamma)^{-1}, \theta, \theta^{-1}\}$$

is bounded by a polynomial of n. Assume also that $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ and $\bar{w}$ are such that $(x^0, s^0) \geq |(\bar{x}, \bar{s})|$ and $(x^0, s^0) \geq (x^*, s^*)$ for some $w^* \in \mathcal{S}$. Then, a generic iterative linear solver with a convergence rate given by (44) generates a ξ-approximate solution, which leads to a (τ_p, τ_q)-search direction Δw in

$$\mathcal{O}\left(\psi(\varphi^2_{\tilde{A}}) \log\left(c(\varphi^2_{\tilde{A}}) n \varphi_{\tilde{A}}\right)\right) \tag{55}$$

iterations. As a consequence, the SD and CG methods generate this approximate solution u^j in $\mathcal{O}(\varphi^2_{\tilde{A}} \log(n\varphi_{\tilde{A}}))$ and $\mathcal{O}(\varphi_{\tilde{A}} \log(n\varphi_{\tilde{A}}))$ iterations, respectively.

Proof. The proof of the first part of Theorem 3.5 immediately follows from Propositions 3.1 and 3.2 and Lemmas 3.3 and 3.4. The proof of the second part of Theorem 3.5 follows immediately from Table 3.1 and Proposition 3.1. □

Using the results of sections 2 and 3, we see that the number of "inner" iterations of an iterative linear solver satisfying (44) is uniformly bounded by a constant depending on n and $\varphi_{\tilde{A}}$, while the number of "outer" iterations in the inexact PDIPF algorithm is polynomially bounded by a constant depending on n and $\log \epsilon^{-1}$.

4. Technical results.

This section is devoted to the proofs of Lemma 3.4 and Theorem 2.2. Subsection 4.1 presents the proof of Lemma 3.4, and subsection 4.2 presents the proof of Theorem 2.2.

4.1. Proof of Lemma 3.4.

In this subsection, we will provide the proof of Lemma 3.4. We begin by establishing three technical lemmas.

LEMMA 4.1. *Suppose that $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$, $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$ for some $\eta \in [0,1]$, $\gamma \in [0,1]$, and $\theta > 0$, and $w^* \in \mathcal{S}$. Then*

$$(x - \eta x^0 - (1-\eta)x^*)^T (s - \eta s^0 - (1-\eta)s^*) \;\geq\; -\frac{\theta^2}{4}\mu. \tag{56}$$

Proof. Let us define $\tilde{w} := w - \eta w^0 - (1-\eta) w^*$. Using the definitions of $\mathcal{N}_{w^0}(\eta, \gamma, \theta)$, r, and $\mathcal{S}$, we have that

$$\begin{aligned} A\tilde{x} &= 0, \\ A^T\tilde{y} + \tilde{s} + V\tilde{z} &= 0, \\ V^T\tilde{x} + E^{-2}\tilde{z} &= E^{-1}(r_V - \eta r^0_V). \end{aligned}$$

Multiplying the second relation by $\tilde{x}^T$ on the left and using the first and third relations along with the fact that $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$, we see that

$$\begin{aligned} \tilde{x}^T\tilde{s} = -\tilde{x}^T V \tilde{z} \;&=\; [E^{-2}\tilde{z} - E^{-1}(r_V - \eta r^0_V)]^T \tilde{z} \;=\; \|E^{-1}\tilde{z}\|^2 - (E^{-1}\tilde{z})^T (r_V - \eta r^0_V) \\ &\geq \|E^{-1}\tilde{z}\|^2 - \|E^{-1}\tilde{z}\| \|r_V - \eta r^0_V\| \;=\; \left(\|E^{-1}\tilde{z}\| - \frac{\|r_V - \eta r^0_V\|}{2}\right)^2 - \frac{\|r_V - \eta r^0_V\|^2}{4} \\ &\geq -\frac{\|r_V - \eta r^0_V\|^2}{4} \;\geq\; -\frac{\theta^2}{4}\mu. \qquad \square \end{aligned}$$

LEMMA 4.2. *Suppose that $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ such that $(x^0, s^0) \geq (x^*, s^*)$ for some $w^* \in \mathcal{S}$. Then, for any $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$ with $\eta \in [0,1]$, $\gamma \in [0,1]$, and $\theta > 0$, we have*

$$\eta(x^T s^0 + s^T x^0) \leq \left(3n + \frac{\theta^2}{4}\right)\mu. \tag{57}$$

Proof. Using the fact $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$ and (56), we obtain

$$\begin{aligned} &x^T s - \eta(x^T s^0 + s^T x^0) + \eta^2 {x^0}^T s^0 - (1-\eta)(x^T s^* + s^T x^*) \\ &\quad + \eta(1-\eta)({x^*}^T s^0 + {s^*}^T x^0) + (1-\eta)^2 {x^*}^T s^* \;\geq\; -\frac{\theta^2}{4}\mu. \end{aligned}$$

Rearranging the terms in this equation and using the facts that $\eta \leq x^T s / {x^0}^T s^0$, ${x^*}^T s^* = 0$, $(x, s) \geq 0$, $(x^*, s^*) \geq 0$, $(x^0, s^0) > 0$, $\eta \in [0,1]$, $x^* \leq x^0$, and $s^* \leq s^0$, we conclude that

$$\begin{aligned} \eta(x^T s^0 + s^T x^0) &\leq \eta^2 {x^0}^T s^0 + x^T s + \eta(1-\eta)({x^*}^T s^0 + {s^*}^T x^0) + \frac{\theta^2}{4}\mu \\ &\leq \eta^2 {x^0}^T s^0 + x^T s + 2\eta(1-\eta){x^0}^T s^0 + \frac{\theta^2}{4}\mu \\ &\leq 2\eta {x^0}^T s^0 + x^T s + \frac{\theta^2}{4}\mu \\ &\leq 3x^T s + \frac{\theta^2}{4}\mu \;=\; \left(3n + \frac{\theta^2}{4}\right)\mu. \qquad \square \end{aligned}$$

LEMMA 4.3. *Suppose $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$, $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$ for some $\eta \in [0,1]$, $\gamma \in [0,1]$, and $\theta > 0$, and $\bar{w}$ satisfies* (52)*. Let W, v, and u^0 be given by* (43) *and* (53)*, respectively. Then,*

$$W u^0 - v \;=\; \tilde{T}\tilde{A}\begin{pmatrix} -x + \sigma\mu S^{-1}e + \eta(x^0 - \bar{x}) + \eta D^2(s^0 - \bar{s}) \\ E^{-1}(r_V - \eta r_V^0) \end{pmatrix}. \tag{58}$$

Proof. Using the fact that $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$ along with (21), (36), and (52), we easily obtain that

$$\begin{aligned} \begin{pmatrix} r_p \\ E^{-1} r_V \end{pmatrix} &= \begin{pmatrix} \eta r_p^0 \\ \eta E^{-1} r_V^0 + E^{-1}(r_V - \eta r_V^0) \end{pmatrix} \\ &= \eta\tilde{A}\begin{pmatrix} x^0 - \bar{x} \\ E^{-2}(z^0 - \bar{z}) \end{pmatrix} + \tilde{A}\begin{pmatrix} 0 \\ E^{-1}(r_V - \eta r_V^0) \end{pmatrix}, \end{aligned} \tag{59}$$

$$s^0 - \bar{s} = -A^T(y^0 - \bar{y}) - V(z^0 - \bar{z}) + r_d^0. \tag{60}$$

Using relations (20), (21), (43), (36), (53), (59), and (60), we obtain

$$
\begin{aligned}
Wu^0 - v &= \tilde{T}\tilde{A}\tilde{D}^2\tilde{A}^T\tilde{T}^T u^0 - \tilde{T}\tilde{A}\begin{pmatrix} x - \sigma\mu S^{-1}e - D^2 r_d \\ 0 \end{pmatrix} + \tilde{T}\begin{pmatrix} r_p \\ E^{-1} r_V \end{pmatrix} \\
&= -\eta\tilde{T}\tilde{A}\tilde{D}^2\tilde{A}^T\begin{pmatrix} y^0 - \bar{y} \\ z^0 - \bar{z} \end{pmatrix} - \tilde{T}\tilde{A}\begin{pmatrix} x - \sigma\mu S^{-1}e - \eta D^2 r_d^0 \\ 0 \end{pmatrix} + \tilde{T}\begin{pmatrix} r_p \\ E^{-1} r_V \end{pmatrix} \\
&= -\eta\tilde{T}\tilde{A}\begin{pmatrix} D^2A^T(y^0 - \bar{y}) + D^2V(z^0 - \bar{z}) - D^2 r_d^0 \\ E^{-2}(z^0 - \bar{z}) \end{pmatrix} \\
&\quad - \tilde{T}\tilde{A}\begin{pmatrix} x - \sigma\mu S^{-1}e \\ 0 \end{pmatrix} + \tilde{T}\begin{pmatrix} r_p \\ E^{-1} r_V \end{pmatrix}, \\
&= -\eta\tilde{T}\tilde{A}\begin{pmatrix} -D^2(s^0 - \bar{s}) \\ E^{-2}(z^0 - \bar{z}) \end{pmatrix} - \tilde{T}\tilde{A}\begin{pmatrix} x - \sigma\mu S^{-1}e \\ 0 \end{pmatrix} \\
&\quad + \eta\tilde{T}\tilde{A}\begin{pmatrix} x^0 - \bar{x} \\ E^{-2}(z^0 - \bar{z}) \end{pmatrix} + \tilde{T}\tilde{A}\begin{pmatrix} 0 \\ E^{-1}(r_V - \eta r_V^0) \end{pmatrix},
\end{aligned}
$$

which yields (58), as desired. ∎

We now turn to the proof of Lemma 3.4.

Proof. Since $w \in \mathcal{N}_{w^0}(\gamma, \theta)$, we have that $x_i s_i \geq (1-\gamma)\mu$ for all i, which implies

$$\|(XS)^{-1/2}\| \;\leq\; \frac{1}{\sqrt{(1-\gamma)\mu}}. \tag{61}$$

Note that $\|Xs - \sigma\mu e\|$, when viewed as a function of $\sigma \in [0,1]$, is convex. Hence, it is maximized at one of its endpoints, which, together with the facts $\|Xs - \mu e\| < \|Xs\|$ and $\sigma \in [\underline{\sigma}, \overline{\sigma}] \subset [0,1]$, immediately implies that

$$\|Xs - \sigma\mu e\| \;\leq\; \|Xs\| \;\leq\; \|Xs\|_1 \;=\; x^T s \;=\; n\mu. \tag{62}$$

Using the fact that $(x^0, s^0) \geq |(\bar{x}, \bar{s})|$ together with Lemma 4.2, we obtain that

$$
\begin{aligned}
\eta\|S(x^0 - \bar{x}) + X(s^0 - \bar{s})\| &\leq \eta\big\{\|S(x^0 - \bar{x})\| + \|X(s^0 - \bar{s})\|\big\} \;\leq\; 2\eta\big\{\|Sx^0\| + \|Xs^0\|\big\} \\
&\leq 2\eta(x^T s^0 + x^T s^0) \;\leq\; \left(6n + \frac{\theta^2}{2}\right)\mu. \qquad (63)
\end{aligned}
$$

Since $w \in \mathcal{N}_{w^0}(\gamma, \theta)$, there exists $\eta \in [0,1]$ such that $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$. It is clear that the requirements of Lemma 4.3 are met, so (58) holds. By (19), (20), and (58), we see that

$$
\begin{aligned}
\|v - Wu^0\| &= \left\|\tilde{T}\tilde{A}\tilde{D}\begin{pmatrix} (XS)^{-1/2}\{Xs - \sigma\mu e - \eta[S(x^0 - \bar{x}) + X(s^0 - \bar{s})]\} \\ r_V - \eta r_V^0 \end{pmatrix}\right\| \\
&\leq \|\tilde{T}\tilde{A}\tilde{D}\|\Big\{\|(XS)^{-1/2}\|\Big[\|Xs - \sigma\mu e\| + \eta\|X(s^0 - \bar{s}) + S(x^0 - \bar{x})\|\Big] \\
&\qquad + \|r_V - \eta r_V^0\|\Big\}, \\
&\leq \varphi_{\tilde{A}}\left\{\frac{1}{\sqrt{(1-\gamma)\mu}}\left[n\mu + \left(6n + \frac{\theta^2}{2}\right)\mu\right] + \theta\sqrt{\mu}\right\} \;=\; \Psi\sqrt{\mu},
\end{aligned}
$$

where the last inequality follows from Proposition 3.1, relations (61), (62), (63), and the assumption that $w \in \mathcal{N}_{w^0}(\gamma, \theta)$. ∎

4.2. "Outer" iteration results—Proof of Theorem 2.2. In this subsection, we will present the proof of Theorem 2.2. Specifically, we will show that the inexact PDIPF algorithm obtains an ϵ-approximate solution to (5)–(8) in $\mathcal{O}(n^2 \log(1/\epsilon))$ outer iterations.

Throughout this section, we use the following notation:

$$w(\alpha) := w + \alpha\Delta w, \quad \mu(\alpha) := \mu(w(\alpha)), \quad r(\alpha) := r(w(\alpha)).$$

LEMMA 4.4. *Assume that Δw satisfies (32)–(35) for some $\sigma \in \Re$, $w \in \Re^{2n+m+l}$, and $(p,q) \in \Re^n \times \Re^l$. Then, for every $\alpha \in \Re$, we have*
(a) $X(\alpha)s(\alpha) = (1-\alpha)Xs + \alpha\sigma\mu e - \alpha p + \alpha^2 \Delta X \Delta s$;
(b) $\mu(\alpha) = [1-\alpha(1-\sigma)]\mu - \alpha p^T e/n + \alpha^2 \Delta x^T \Delta s/n$;
(c) $(r_p(\alpha), r_d(\alpha)) = (1-\alpha)(r_p, r_d)$;
(d) $r_V(\alpha) = (1-\alpha)r_V + \alpha q$.

Proof. Using (34), we obtain

$$\begin{aligned} X(\alpha)s(\alpha) &= (X + \alpha\Delta X)(s + \alpha\Delta s) \\ &= Xs + \alpha(X\Delta s + S\Delta x) + \alpha^2 \Delta X \Delta s \\ &= Xs + \alpha(-Xs + \sigma\mu e - p) + \alpha^2 \Delta X \Delta s \\ &= (1-\alpha)Xs + \alpha\sigma\mu e - \alpha p + \alpha^2 \Delta X \Delta s, \end{aligned}$$

thereby showing that (a) holds. Left multiplying the above equality by e^T and dividing the resulting expression by n, we easily conclude that (b) holds. Statement (c) can be easily verified by means of (32) and (33), while statement (d) follows from (35). ∎

LEMMA 4.5. *Assume that Δw satisfies (32)–(35) for some $\sigma \in \Re$, $w \in \Re^{2n}_{++} \times \Re^{m+l}$, and $(p,q) \in \Re^n \times \Re^l$ such that $\|p\|_\infty \le \gamma\sigma\mu/4$. Then, for every scalar α satisfying*

$$0 \le \alpha \le \min\left\{1\,,\ \frac{\sigma\mu}{4\,\|\Delta X \Delta s\|_\infty}\right\}, \tag{64}$$

we have

$$\frac{\mu(\alpha)}{\mu} \ \ge\ 1-\alpha. \tag{65}$$

Proof. Since $\|p\|_\infty \le \gamma\sigma\mu/4$, we easily see that

$$|p^T e/n| \ \le\ \|p\|_\infty \ \le\ \sigma\mu/4. \tag{66}$$

Using this result and Lemma 4.4(b), we conclude for every α satisfying (64) that

$$\begin{aligned} \mu(\alpha) &= [1-\alpha(1-\sigma)]\mu - \alpha p^T e/n + \alpha^2 \Delta x^T \Delta s/n \\ &\ge [1-\alpha(1-\sigma)]\mu - \frac{1}{4}\,\alpha\sigma\mu + \alpha^2 \Delta x^T \Delta s/n \\ &\ge (1-\alpha)\mu + \frac{1}{4}\,\alpha\sigma\mu - \alpha^2 \|\Delta X \Delta s\|_\infty \\ &\ge (1-\alpha)\mu. \qquad \square \end{aligned}$$

LEMMA 4.6. *Assume that Δw is a (τ_p, τ_q)-search direction, where τ_p and τ_q are given by (38) and (39), respectively. Assume also that $\sigma > 0$ and that $w \in \mathcal{N}_{w^0}(\gamma, \theta)$*

with $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$, $\gamma \in [0,1]$, *and* $\theta \geq 0$. *Then,* $w(\alpha) \in \mathcal{N}_{w^0}(\gamma,\theta)$ *for every scalar* α *satisfying*

$$0 \leq \alpha \leq \min\left\{1\,,\ \frac{\gamma\sigma\mu}{4\,\|\Delta X \Delta s\|_\infty}\right\}. \tag{67}$$

Proof. Since $w \in \mathcal{N}_{w^0}(\gamma,\theta)$, there exists $\eta \in [0,1]$ such that $w \in \mathcal{N}_{w^0}(\eta,\gamma,\theta)$. We will show that $w(\alpha) \in \mathcal{N}_{w^0}((1-\alpha)\eta,\gamma,\theta) \subseteq \mathcal{N}_{w^0}(\gamma,\theta)$ for every α satisfying (67).

First, we note that $(r_p(\alpha), r_d(\alpha)) = (1-\alpha)\eta(r_p^0, r_d^0)$ by Lemma 4.4(c) and the definition of $\mathcal{N}_{w^0}(\eta,\gamma,\theta)$. Next, it follows from Lemma 4.5 that (65) holds for every α satisfying (64), and hence (67) due to $\gamma \in [0,1]$. Thus, for every α satisfying (67), we have

$$(1-\alpha)\eta \leq \frac{\mu(\alpha)}{\mu}\eta \ \leq\ \frac{\mu(\alpha)}{\mu}\,\frac{\mu}{\mu_0} \ =\ \frac{\mu(\alpha)}{\mu_0}. \tag{68}$$

Now, it is easy to see that for every $u \in \Re^n$ and $\tau \in [0,n]$, there holds $\|u - \tau(u^Te/n)e\|_\infty \leq (1+\tau)\|u\|_\infty$. Using this inequality twice, the fact that $w \in \mathcal{N}_{w^0}(\eta,\gamma,\theta)$, relation (38), and statements (a) and (b) of Lemma 4.4, we conclude for every α satisfying (67) that

$$\begin{aligned}
&X(\alpha)s(\alpha) - (1-\gamma)\,\mu(\alpha)e\\
&= (1-\alpha)\,[Xs - (1-\gamma)\mu e] + \alpha\gamma\sigma\mu e - \alpha\left[p - (1-\gamma)\left(\frac{p^Te}{n}\right)e\right]\\
&\quad + \alpha^2\left[\Delta X\Delta s - (1-\gamma)\left(\frac{\Delta x^T\Delta s}{n}\right)e\right]\\
&\geq \alpha\left[\gamma\sigma\mu - \left\|p - (1-\gamma)\frac{p^Te}{n}\,e\right\|_\infty - \alpha\left\|\Delta X\Delta s - (1-\gamma)\frac{\Delta x^T\Delta s}{n}\,e\right\|_\infty\right]e\\
&\geq \alpha\left(\gamma\sigma\mu - 2\|p\|_\infty - 2\alpha\|\Delta X\Delta s\|_\infty\right)e \ \geq\ \alpha\left(\gamma\sigma\mu - \frac{1}{2}\gamma\sigma\mu - \frac{1}{2}\gamma\sigma\mu\right)e \ =\ 0.
\end{aligned}$$

Next, by Lemma 4.4(d), we have that

$$r_V(\alpha) = (1-\alpha)r_V + \alpha q = \ (1-\alpha)\eta r_V^0 + \hat{a},$$

where $\hat{a} = (1-\alpha)(r_V - \eta r_V^0) + \alpha q$. To complete the proof, it suffices to show that $\|\hat{a}\| \leq \theta\sqrt{\mu(\alpha)}$ for every α satisfying (67). By using equation (39) and Lemma 4.4(b) along with the facts that $\|r_V - \eta r_V^0\| \leq \theta\sqrt{\mu}$ and $\alpha \in [0,1]$, we have

$$\begin{aligned}
\|\hat{a}\|^2 - \theta^2\mu(\alpha) &= (1-\alpha)^2\|r_V - \eta r_V^0\|^2 + 2\alpha(1-\alpha)[r_V - \eta r_V^0]^Tq + \alpha^2\|q\|^2 - \theta^2\mu(\alpha)\\
&\leq (1-\alpha)^2\theta^2\mu + 2\alpha(1-\alpha)\theta\sqrt{\mu}\|q\| + \alpha^2\|q\|^2\\
&\qquad - \theta^2\left\{[1-\alpha(1-\sigma)]\mu - \alpha\frac{p^Te}{n} + \alpha^2\frac{\Delta x^T\Delta s}{n}\right\}\\
&\leq \alpha^2\|q\|^2 + 2\alpha\theta\sqrt{\mu}\|q\| - \alpha\theta^2\sigma\mu + \alpha\theta^2\frac{p^Te}{n} - \alpha^2\theta^2\frac{\Delta x^T\Delta s}{n}\\
&\leq \alpha\left[\|q\|^2 + 2\theta\sqrt{\mu}\|q\| - \left(1 - \frac{\gamma}{4}\right)\theta^2\sigma\mu + \theta^2\alpha\|\Delta X\Delta s\|_\infty\right]\\
&\leq \alpha\left[\|q\|^2 + 2\theta\sqrt{\mu}\|q\| - \left(1 - \frac{\gamma}{2}\right)\theta^2\sigma\mu\right] \ \leq\ 0,
\end{aligned}$$

where the last inequality follows from the quadratic formula and the fact that $\|q\| \le \tau_q$, where τ_q is given by (39). ∎

Next, we derive a lower bound on the step size of the inexact PDIPF algorithm.

LEMMA 4.7. *In every iteration of the inexact PDIPF algorithm, the step length* $\bar{\alpha}$ *satisfies*

$$\bar{\alpha} \;\ge\; \min\left\{1, \frac{\min\{\gamma\sigma, 1-\frac{5}{4}\sigma\}\mu}{4\,\|\Delta X \Delta s\|_\infty}\right\} \tag{69}$$

and

$$\mu(\bar{\alpha}) \;\le\; \left[1 - \left(1 - \frac{5}{4}\,\sigma\right)\frac{\bar{\alpha}}{2}\right]\mu. \tag{70}$$

Proof. We know that Δw is a (τ_p, τ_q)-search direction in every iteration of the inexact PDIPF algorithm, where τ_p and τ_q are given by (38) and (39). Hence, by Lemma 4.6, the quantity $\tilde{\alpha}$ computed in step (g) of the inexact PDIPF algorithm satisfies

$$\tilde{\alpha} \;\ge\; \min\left\{1, \frac{\gamma\sigma\mu}{4\,\|\Delta X \Delta s\|_\infty}\right\}. \tag{71}$$

Moreover, by (66), it follows that the coefficient of α in the expression for $\mu(\alpha)$ in Lemma 4.4(b) satisfies

$$\begin{aligned} -(1-\sigma)\mu - \frac{p^T e}{n} \;&\le\; -(1-\sigma)\mu + \|p\|_\infty \;\le\; -(1-\sigma)\mu + \frac{1}{4}\,\gamma\sigma\mu \\ &=\; -\left(1 - \frac{5}{4}\,\sigma\right)\mu \;<\; 0, \end{aligned} \tag{72}$$

since $\sigma \in (0, 4/5)$. Hence, if $\Delta x^T \Delta s \le 0$, it is easy to see that $\bar{\alpha} = \tilde{\alpha}$ and hence that (69) holds in view of (71). Moreover, by Lemma 4.4(b) and (72), we have

$$\mu(\bar{\alpha}) \;\le\; [1 - \bar{\alpha}(1-\sigma)]\mu - \bar{\alpha}\frac{p^T e}{n} \;\le\; \left[1 - \left(1 - \frac{5}{4}\,\sigma\right)\bar{\alpha}\right]\mu \;\le\; \left[1 - \left(1 - \frac{5}{4}\,\sigma\right)\frac{\bar{\alpha}}{2}\right]\mu,$$

showing that (70) also holds. We now consider the case where $\Delta x^T \Delta s > 0$. In this case, we have $\bar{\alpha} = \min\{\alpha_{\min}\,, \tilde{\alpha}\}$, where $\alpha_{\min}$ is the unconstrained minimum of $\mu(\alpha)$. It is easy to see that

$$\alpha_{\min} = \frac{n\mu(1-\sigma) + p^T e}{2\Delta x^T \Delta s} \;\ge\; \frac{n[\mu(1-\sigma) - \frac{1}{4}\sigma\mu]}{2\Delta x^T \Delta s} \;\ge\; \frac{\mu(1-\frac{5}{4}\sigma)}{2\,\|\Delta X \Delta s\|_\infty}.$$

The last two observations together with (71) imply that (69) holds in this case too. Moreover, since the function $\mu(\alpha)$ is convex, it must lie below the function $\phi(\alpha)$ over the interval $[0, \alpha_{\min}]$, where $\phi(\alpha)$ is the affine function interpolating $\mu(\alpha)$ at $\alpha = 0$ and $\alpha = \alpha_{\min}$. Hence,

$$\mu(\bar{\alpha}) \;\le\; \phi(\bar{\alpha}) \;=\left[1 - (1-\sigma)\frac{\bar{\alpha}}{2}\right]\mu - \bar{\alpha}\frac{p^T e}{2n} \;\le\; \left[1 - \left(1 - \frac{5}{4}\,\sigma\right)\frac{\bar{\alpha}}{2}\right]\mu, \tag{73}$$

where the second inequality follows from (72). We have thus shown that $\bar{\alpha}$ satisfies (70). ∎

Our next task will be to show that the step size $\bar{\alpha}$ remains bounded away from zero. In view of (69), it suffices to show that the quantity $\|\Delta X \Delta s\|_\infty$ can be suitably bounded. The next lemma addresses this issue.

LEMMA 4.8. *Let $w^0 \in \Re^{2n}_{++} \times \Re^{m+l}$ be such that $(x^0, s^0) \geq (x^*, s^*)$ for some $w^* \in \mathcal{S}$, and let $w \in \mathcal{N}_{w^0}(\gamma, \theta)$ for some $\gamma \geq 0$ and $\theta \geq 0$. Then, the inexact search direction Δw used in the inexact PDIPF algorithm satisfies*

$$\max(\|D^{-1}\Delta x\|, \|D\Delta s\|) \leq \left(1 - 2\sigma + \frac{\sigma^2}{1-\gamma}\right)^{1/2} \sqrt{n\mu}$$

$$+ \frac{1}{\sqrt{1-\gamma}} \left(\frac{\gamma\sigma}{4}\sqrt{n} + 6n + \frac{\theta^2}{2}\right)\sqrt{\mu} + \theta\sqrt{\mu}. \tag{74}$$

Proof. Since $w \in \mathcal{N}_{w^0}(\gamma, \theta)$, there exists $\eta \in [0,1]$ such that $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$. Let $\widetilde{\Delta w} := \Delta w + \eta(w^0 - w^*)$. Using relations (32), (33), (35), and the fact that $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$, we easily see that

$$A\widetilde{\Delta x} = 0, \tag{75}$$

$$A^T\widetilde{\Delta y} + \widetilde{\Delta s} + V\widetilde{\Delta z} = 0, \tag{76}$$

$$V^T\widetilde{\Delta x} + E^{-2}\widetilde{\Delta z} = E^{-1}(q - r_V + \eta r_V^0). \tag{77}$$

Premultiplying (76) by $\widetilde{\Delta x}^T$ and using (75) and (77), we obtain

$$\begin{aligned}
\widetilde{\Delta x}^T\widetilde{\Delta s} = -\widetilde{\Delta x}^T V\widetilde{\Delta z} \;&=\; [E^{-2}\widetilde{\Delta z} - E^{-1}(q - r_V + \eta r_V^0)]^T\widetilde{\Delta z} \\
&= \|E^{-1}\widetilde{\Delta z}\|^2 - (q - r_V + \eta r_V^0)^T(E^{-1}\widetilde{\Delta z}) \\
&\geq \|E^{-1}\widetilde{\Delta z}\|^2 - \|q - r_V + \eta r_V^0\|\,\|E^{-1}\widetilde{\Delta z}\| \;\geq\; -\frac{\|q - r_V + \eta r_V^0\|^2}{4}.
\end{aligned} \tag{78}$$

Next, we multiply (34) by $(XS)^{-1/2}$ to obtain $D^{-1}\Delta x + D\Delta s = H(\sigma) - (XS)^{-1/2}p$, where $H(\sigma) := -(XS)^{1/2}e + \sigma\mu(XS)^{-1/2}e$. Equivalently, we have that

$$D^{-1}\widetilde{\Delta x} + D\widetilde{\Delta s} = H(\sigma) - (XS)^{-1/2}p + \eta\left[D(s^0 - s^*) + D^{-1}(x^0 - x^*)\right] \;=:\; g.$$

Taking the squared norm of both sides of the above equation and using (78), we obtain

$$\begin{aligned}
\|D^{-1}\widetilde{\Delta x}\|^2 + \|D\widetilde{\Delta s}\|^2 = \|g\|^2 - 2\widetilde{\Delta x}^T\widetilde{\Delta s} \;&\leq\; \|g\|^2 + \frac{\|q - r_V + \eta r_V^0\|^2}{2} \\
&\leq \left(\|g\| + \frac{\|q\| + \|r_V - \eta r_V^0\|}{\sqrt{2}}\right)^2 \;\leq\; \left(\|g\| + \theta\sqrt{\mu}\right)^2,
\end{aligned}$$

since $\|q\| + \|r_V - \eta r_V^0\| \leq \left[\sqrt{2} - 1\right]\theta\sqrt{\mu} + \theta\sqrt{\mu} = \sqrt{2}\theta\sqrt{\mu}$ by (36), (39), and the fact that $1 + (1 - \gamma/2)\sigma \leq 2$. Thus, we have

$$\begin{aligned}
&\max(\|D^{-1}\widetilde{\Delta x}\|\,,\,\|D\widetilde{\Delta s}\|) \;\leq\; \|g\| + \theta\sqrt{\mu} \\
&\leq\; \|H(\sigma)\| + \|(XS)^{-1/2}\|\,\|p\| + \eta\left[\|D(s^0 - s^*)\| + \|D^{-1}(x^0 - x^*)\|\right] + \theta\sqrt{\mu}.
\end{aligned}$$

This, together with the triangle inequality, the definitions of D and $\widetilde{\Delta w}$, and the fact that $w \in \mathcal{N}_{w^0}(\eta, \gamma, \theta)$, implies that

$$
\begin{aligned}
&\max(\|D^{-1}\Delta x\|, \|D\Delta s\|) \\
&\le \|H(\sigma)\| + \|(XS)^{-1/2}\|\,\|p\| + 2\eta\left[\|D(s^0 - s^*)\| + \|D^{-1}(x^0 - x^*)\|\right] + \theta\sqrt{\mu} \\
&\le \|H(\sigma)\| + \|(XS)^{-1/2}\|\,\|p\| + 2\eta\|(XS)^{-1/2}\|\left[\|X(s^0 - s^*)\| + \|S(x^0 - x^*)\|\right] + \theta\sqrt{\mu} \\
&\le \|H(\sigma)\| + \frac{1}{\sqrt{(1-\gamma)\mu}}\left[\|p\| + 2\eta\left(\|X(s^0 - s^*)\| + \|S(x^0 - x^*)\|\right)\right] + \theta\sqrt{\mu}.
\end{aligned} \tag{79}
$$

It is well known (see, e.g., [10]) that

$$\|H(\sigma)\| \le \left(1 - 2\sigma + \frac{\sigma^2}{1-\gamma}\right)^{1/2}\sqrt{n\mu}. \tag{80}$$

Moreover, using the fact that $s^* \le s^0$ and $x^* \le x^0$ along with Lemma 4.2, we obtain

$$\eta\left(\|X(s^0 - s^*)\| + \|S(x^0 - x^*)\|\right) \;\le\; \eta(s^T x^0 + x^T s^0) \;\le\; \left(3n + \frac{\theta^2}{4}\right)\mu. \tag{81}$$

The result now follows by noting that $\|p\| \le \sqrt{n}\|p\|_\infty$ and by incorporating inequalities (80), (81), and (38) into (79). ∎

We are now ready to prove Theorem 2.2.

Proof. Let Δw^k denote the search direction, and let $r^k = r(w^k)$ and $\mu_k = \mu(w^k)$ at the kth iteration of the inexact PDIPF algorithm. Clearly, $w^k \in \mathcal{N}_{w^0}(\gamma, \theta)$. Hence, using Lemma 4.8, assumption (40), and the inequality

$$\|\Delta X^k \Delta s^k\|_\infty \;\le\; \|\Delta X^k \Delta s^k\| \;\le\; \|(D^k)^{-1}\Delta x^k\|\,\|D^k \Delta s^k\|,$$

we easily see that $\|\Delta X^k \Delta s^k\|_\infty = \mathcal{O}(n^2)\mu_k$. Using this conclusion together with assumption (40) and Lemma 4.7, we see that, for some universal constant $\beta > 0$, we have

$$\mu_{k+1} \;\le\; \left(1 - \frac{\beta}{n^2}\right)\mu_k \quad \forall k \ge 0.$$

Using this observation and some standard arguments (see, for example, Theorem 3.2 of [27]), we easily see that the inexact PDIPF algorithm generates an iterate $w^k \in \mathcal{N}_{w^0}(\gamma, \theta)$ satisfying $\mu_k/\mu_0 \le \epsilon$ within $\mathcal{O}\left(n^2 \log(1/\epsilon)\right)$ iterations. The theorem now follows from this conclusion and the definition of $\mathcal{N}_{w^0}(\gamma, \theta)$. ∎

5. Concluding remarks. We have shown that the long-step PDIPF algorithm for LP based on an iterative linear solver presented in [16] can be extended to the context of CQP. This was not immediately obvious at first since the standard normal equation for CQP does not fit into the mold required for the results of [17] to hold. By considering the ANE, we were able to use the results about the MWB preconditioner developed in [17] in the context of CQP. Another difficulty we encountered was the proper choice of the starting iterate u^0 for the iterative linear solver. By choosing $u^0 = 0$ as in the LP case, we obtain $\|v - Wu^0\| = \|v\|$, which can only be shown to be $\mathcal{O}(\max\{\mu, \sqrt{\mu}\})$. In this case, for every $\mu > 1$, Proposition 3.2 would guarantee that the number of inner iterations of the iterative linear solver is

$$\mathcal{O}\left(\psi(\varphi_{\tilde{A}}^2)\max\left\{\log\left(c(\varphi_{\tilde{A}}^2)n\varphi_{\tilde{A}}\right), \log\mu\right\}\right),$$

a bound which depends on the logarithm of the current duality gap. On the other hand, Theorem 3.5 shows that choosing u^0 as in (53) results in a bound that does not depend on the current duality gap.

We observe that under exact arithmetic, the CG algorithm applied to $Wu = v$ generates an exact solution in at most $m + l$ iterations (since $W \in \Re^{(m+l)\times(m+l)}$). It is clear, then, that the bound (55) is generally worse than the well-known finite termination bound for CG. However, our results in section 3 were given for a family of iterative linear solvers, only one member of which is CG. Also, under finite precision arithmetic, the CG algorithm loses its finite termination property, and its convergence rate behavior in this case is still an active topic of research (see, e.g., [8]). Certainly, the impact of finite precision arithmetic on our results is an interesting open issue.

Clearly, the MWB preconditioner is not suitable for dense CQP problems since, in this case, the cost to construct the MWB is comparable to the cost to form and factorize $\tilde{A}\tilde{D}^2\tilde{A}^T$, and each inner iteration would require $\Theta((m+l)^2)$ arithmetic operations, the same cost as a forward and backward substitution. There are, however, some classes of CQP problems for which the method proposed in this paper might be useful. One class of problems for which PDIPF methods based on MWB preconditioners might be useful are those for which bases of $\tilde{A}$ are sparse, but the ANE coefficient matrices $\tilde{A}\tilde{D}^2\tilde{A}^T$ are dense; this situation generally occurs in sparse CQP problems for which n is much larger than $m + l$. Other classes of problems for which our method might be useful are network flow problems. The paper [22] developed interior-point methods for solving the minimum cost network flow problem based on iterative linear solvers with maximum spanning tree preconditioners. Related to this work, we believe that the following two issues could be investigated: (i) whether the incorporation of the correction term p defined in (29) in the algorithm implemented in [22] will improve the convergence of the method; (ii) whether our algorithm might be efficient for network flow problems where the costs associated with the arcs are quadratic functions of the arc flows. Identification of other classes of CQP problems which could be efficiently solved by the method proposed in this paper is another topic for future research.

Regarding the second question above, it is easy to see (after a suitable permutation of the variables) that $V^T = \begin{pmatrix} I & 0 \end{pmatrix}$ and E^2 is a positive diagonal matrix whose diagonal elements are the positive quadratic coefficients. In this case, it can be shown that $\tilde{A}$ is totally unimodular; hence $\varphi_{\tilde{A}}^2 \leq (m+l)(n-m+1)$ by Cramer's rule (see [17]).

The usual way of defining the dual residual is as the quantity

$$R_d := A^T y + s - VE^2V^T x - c,$$

which, in view of (11) and (12), can be written in terms of the residuals r_d and r_V as

$$R_d = r_d - VEr_V. \tag{82}$$

Note that, along the iterates generated by the inexact PDIPF algorithm, we have $r_d = \mathcal{O}(\mu)$ and $r_V = \mathcal{O}(\sqrt{\mu})$, implying that $R_d = \mathcal{O}(\sqrt{\mu})$. Hence, while the usual primal residual converges to 0 according to $\mathcal{O}(\mu)$, the usual dual residual does so according to $\mathcal{O}(\sqrt{\mu})$. This is a unique feature of the convergence analysis of our algorithm in that it contrasts with the analysis of other interior-point PDIPF algorithms, where the primal and dual residuals are required to go to zero at the same rate. The convergence analysis under these circumstances is possible due to the specific form of the $\mathcal{O}(\sqrt{\mu})$-term present in (82), i.e., one that lies in the range space of VE.

CQP problems where V is explicitly available arise frequently in the literature. One important example arises in portfolio optimization (see [6]), where the rank of V is often small. In such problems, l represents the number of observation periods used to estimate the data for the problem. We believe that the inexact PDIPF algorithm could be of particular use for this type of application.

REFERENCES

[1] K. M. Anstreicher, *Linear programming in $\mathcal{O}(n^3/(\ln n)\mathrm{L})$ operations*, SIAM J. Optim., 9 (1999), pp. 803–812.

[2] V. Baryamureeba and T. Steihaug, *On the convergence of an inexact primal-dual interior point method for linear programming*, in Large-Scale Scientific Computing, Lecture Notes in Comput. Sci. 3743, Springer-Verlag, Berlin, 2006, pp. 629–637.

[3] V. Baryamureeba, T. Steihaug, and Y. Zhang, *Properties of a Class of Preconditioners for Weighted Least Squares Problems*, Tech. report 99-16, Department of Computational and Applied Mathematics, Rice University, Houston, 1999.

[4] L. Bergamaschi, J. Gondzio, and G. Zilli, *Preconditioning indefinite systems in interior point methods for optimization*, Comput. Optim. Appl., 28 (2004), pp. 149–171.

[5] E. G. Boman, D. Chen, B. Hendrickson, and S. Toledo, *Maximum-weight-basis preconditioners*, Numer. Linear Algebra Appl., 11 (2004), pp. 695–721.

[6] T. J. Carpenter and R. J. Vanderbei, *Symmetric indefinite systems for interior-point methods*, Math. Programming, 58 (1993), pp. 1–32.

[7] R. W. Freund, F. Jarre, and S. Mizuno, *Convergence of a class of inexact interior-point algorithms for linear programs*, Math. Oper. Res., 24 (1999), pp. 50–71.

[8] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, Frontiers Appl. Math. 17, SIAM, Philadelphia, 1997.

[9] M. Kojima, N. Megiddo, and S. Mizuno, *A primal-dual infeasible-interior-point algorithm for linear programming*, Math. Programming, 61 (1993), pp. 263–280.

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

[11] M. Kojima, M. Shida, and M. Shindoh, *Search directions in the SDP and monotone SDLCP: Generalization and inexact computation*, Math. Program., 85 (1999), pp. 51–80.

[12] J. Korzak, *Convergence analysis of inexact infeasible-interior-point algorithms for solving linear programming problems*, SIAM J. Optim., 11 (2000), pp. 133–148.

[13] V. V. Kovacevic-Vujcic and M. D. Asic, *Stabilization of interior-point methods for linear programming*, Comput. Optim. Appl., 14 (1999), pp. 331–346.

[14] D. G. Luenberger, *Linear and Nonlinear Programming*, Addison–Wesley, Reading, MA, 1984.

[15] S. Mizuno and F. Jarre, *Global and polynomial-time convergence of an infeasible-interior-point algorithm using inexact computation*, Math. Program., 84 (1999), pp. 357–373.

[16] R. D. C. Monteiro and J. W. O'Neal, *Convergence Analysis of a Long-Step Primal-Dual Infeasible Interior-Point LP Algorithm Based on Iterative Linear Solvers*, Tech. report, Georgia Institute of Technology, Atlanta, 2003.

[17] R. D. C. Monteiro, J. W. O'Neal, and T. Tsuchiya, *Uniform boundedness of a preconditioned normal matrix used in interior point methods*, SIAM J. Optim., 15 (2004), pp. 96–100.

[18] Y. Nesterov and A. Nemirovskii, *Interior Point Polynomial Algorithms in Convex Programming*, Stud. Appl. Math. 13, SIAM, Philadelphia, 1995.

[19] A. R. L. Oliveira and D. C. Sorensen, *A new class of preconditioners for large-scale linear systems from interior point methods for linear programming*, Linear Algebra Appl., 394 (2005), pp. 1–24.

[20] L. F. Portugal, M. G. C. Resende, G. Veiga, and J. J. Judice, *A truncated primal-infeasible dual-feasible network interior point method*, Networks, 35 (2000), pp. 91–108.

[21] J. Renegar, *Condition numbers, the barrier method, and the conjugate-gradient method*, SIAM J. Optim., 6 (1996), pp. 879–912.

[22] M. G. C. Resende and G. Veiga, *An implementation of the dual affine scaling algorithm for minimum-cost flow on bipartite uncapacitated networks*, SIAM J. Optim., 3 (1993), pp. 516–537.

[23] M. J. Todd, L. Tunçel, and Y. Ye, *Characterizations, bounds, and probabilistic analysis of two complexity measures for linear programming problems*, Math. Program., 90 (2001), pp. 59–69.

[24] K.-C. Toh and M. Kojima, *Solving some large scale semidefinite programs via the conjugate residual method*, SIAM J. Optim., 12 (2002), pp. 669–691.
[25] P. Vaidya, *Solving Linear Equations with Symmetric Diagonally Dominant Matrices by Constructing Good Preconditioners*, Tech. report, Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, 1990.
[26] S. A. Vavasis and Y. Ye, *A primal-dual interior point method whose running time depends only on the constraint matrix*, Math. Program., 74 (1996), pp. 79–120.
[27] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1997.
[28] Y. Zhang, *On the convergence of a class of infeasible interior-point methods for the horizontal linear complementarity problem*, SIAM J. Optim., 4 (1994), pp. 208–227.
[29] G. Zhou and K.-C. Toh, *Polynomiality of an inexact infeasible interior point algorithm for semidefinite programming*, Math. Program., 99 (2004), pp. 261–282.