Preconditioning and iterative solution of symmetric indefinite linear systems arising from interior point methods for linear programming

Joo-Siong Chai* and Kim-Chuan Toh†
*High Performance Computing for Engineered Systems (HPCES)
Singapore-MIT Alliance, 4 Engineering Drive 3, Singapore 117576
Email: smap0059@nus.edu.sg
†Department of Mathematics
National University of Singapore, 2 Science Drive 2, Singapore 117543
Email: mattohkc@math.nus.edu.sg
Research supported in part by NUS Academic Research Grant C-146-000-302-471.

***Abstract*— We study the preconditioning of symmetric indefinite linear systems of equations that arise in interior point solution of linear optimization problems. The preconditioning method that we study exploits the block structure of the augmented matrix to design a similar block structure preconditioner to improve the spectral properties of the resulting preconditioned matrix so as to improve the convergence rate of the iterative solution of the system. We also propose a two-phase algorithm that takes advantage of the spectral properties of the transformed matrix to solve for the Newton directions in the interior-point method. Numerical experiments have been performed on some LP test problems in the NETLIB suite to demonstrate the potential of the preconditioning method discussed.**

I. INTRODUCTION

The standard approach to solve the linear system of equations in interior-point algorithm for linear programming (LP) uses direct methods based on sparse Cholesky factorization of the symmetric positive definite normal matrix, or Bunch-Parlett decomposition of the symmetric indefinite augmented matrix. However, as the size of the matrix becomes large, the computational effort of direct methods grows in the order of m^3, if the LP data is dense.

Iterative methods such as Krylov subspace methods have the potential to reduce the computation time by working with approximate directions rather than exact directions. The amount of reduction achievable by any iterative method is determined by the spectral properties of the coefficient matrix, which determine the convergence rate of the iterative method. Hence, it is important to precondition the coefficient matrix to achieve a small condition number or good clustering of the eigenvalues.

It is known that the spectral properties of both the normal matrix and the augmented matrix deteriorate as the interior-point iterates converge to a solution. In general, however, the normal matrix is more ill-conditioned than the augmented matrix, and it is also harder to design a good preconditioner for the former than the latter [2]. Hence, we focus our study on the augmented system of equations.

In this paper, we present our findings in our study on one preconditioning approach that exploits the block structure of the symmetric indefinite augmented matrix that arises in an interior-point method for LP. We first transform the augmented system into an equivalent reduced 2×2 block system. Based on the transformed system, we design the preconditioning matrix by approximating the block structure of the inverse of the transformed matrix. We also propose a two-phase algorithm that takes advantage of the spectral properties of the transformed matrix to compute the search directions in the interior-point method. Phase one employs existing direct or iterative method to compute the search directions via the normal equations, while phase two will compute the search directions based on the transformed equations by some preconditioned iterative method.

In the ensuing sections, we will discuss the construction of preconditioners, the two-phase algorithm, implementation issues, and highlight some preliminary experimental results achieved so far. Finally, we conclude with possible future direction on this work.

II. PROBLEM FORMULATION

Consider the following LP problem:

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b \\ & x + s = u \\ & x, s \geq 0 \end{array} \tag{1}$$

where $c, x, s, u \in I\!R^n$, $b \in I\!R^m$, $A \in I\!R^{m \times n}$. The dual problem of (1) has the form:

$$\begin{array}{ll} \min & b^T y - u^T w \\ \text{s.t.} & A^T y - w + z = c \\ & z, w \geq 0 \end{array} \tag{2}$$

where $z, w \in I\!R^n$, $y \in I\!R^m$.

The corresponding first order optimality conditions are:

$$\begin{aligned} Ax &= b \\ x + s &= u \\ A^T y - w + z &= c \\ XZe &= \mu e \\ SWe &= \mu e \end{aligned} \tag{3}$$

where X, S, Z and W are diagonal matrices whose diagonals are formed from the vectors x, s, z and w respectively, e is the n-vector of all ones, $\mu > 0$ is a barrier parameter.

To solve for the search directions, we apply Newton's method to (3) to obtain the following Newton system of equations:

$$\begin{bmatrix} A & 0 & 0 & 0 & 0 \\ I & 0 & I & 0 & 0 \\ 0 & A^T & 0 & I & -I \\ Z & 0 & 0 & X & 0 \\ 0 & 0 & W & 0 & S \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \\ \Delta z \\ \Delta w \end{bmatrix} = \begin{bmatrix} r_p \\ r_u \\ r_d \\ \mu e - XZe \\ \mu e - SWe \end{bmatrix} \tag{4}$$

where

$$\begin{aligned} r_p &= b - Ax \\ r_u &= u - x - s \\ r_d &= c - A^T y - z + w \end{aligned} \tag{5}$$

By substituting

$$\begin{aligned} \Delta z &= X^{-1}(\mu e - XZe - Z\Delta x) \\ \Delta s &= r_u - \Delta x \\ \Delta w &= S^{-1}(\mu e - SWe - W\Delta s) \end{aligned} \tag{6}$$

into (4), we can reduce the set of Newton equations to the following augmented system of equations:

$$\begin{bmatrix} -D & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} g \\ r_p \end{bmatrix} \tag{7}$$

where

$$\begin{aligned} D &= X^{-1}Z + S^{-1}W \\ g &= r_d - X^{-1}(\mu e - XZe) + S^{-1}(\mu e - SWe) - S^{-1}Wr_u \end{aligned}$$

This can further be reduced to the normal equations

$$AD^{-1}A^T\Delta y = AD^{-1}g + r_p. \tag{8}$$

The normal matrix $AD^{-1}A^T$ in (8) is the Schur complement of the augmented matrix in (7). After solving for Δy in (8), we may compute Δx through

$$\Delta x = D^{-1}(A^T\Delta y - g). \tag{9}$$

In each iteration of the primal-dual interior-point algorithm, solution of either (7) or (8) have to be computed numerically. Very often, however, the matrix D is highly ill-conditioned, especially when the interior point iterate approaches optimality. This typically causes the coefficient matrix to be highly ill-conditioned even if A is well-conditioned. Applying iterative solutions such as Krylov subspace method to such systems often encounter exceedingly slow convergence. Furthermore, constructing an effective preconditioner for such a system is extremely difficult when D is ill-conditioned.

In general, it is more difficult to design a suitable preconditioner for the Schur complement matrix than the augmented matrix, and the former is likely to be more ill-conditioned than the latter at a given interior-point iterate [2]. In the next section, we study a transformation of the augmented system to an equivalent reduced system that we believe is more amenable to the construction of good preconditioners.

We use $\|\cdot\|$ to denote either the vector or matrix 2-norm. For any two non-negative numbers α and β, we write $\alpha = O(\beta)$ if there is a moderate constant c such that $\alpha \leq c\beta$. We write $\alpha = \Omega(\beta)$ to indicate that $\alpha = O(\beta)$ and $\beta = O(\alpha)$. For a matrix M, we write $M = O(\beta)$ to denote $\|M\| = O(\beta)$. For a vector x, we use x^{-1} to denote the vector whose ith component is x_i^{-1}.

III. Conditioning of the normal equations

It is well known that the ill-conditioning of the matrix D in (7) is often due to the separation of its diagonal elements into two clusters with different orders of magnitude, where one cluster corresponds to non-active constraints while another corresponds to active constraints. Suppose D can be partitioned as

$$D = \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix}$$

where $\text{diag}(D_1) = \mu\tilde{D}_1$, and $\text{diag}(D_2) = \tilde{D}_2/\mu$ with $\text{diag}(\tilde{D}_1), \text{diag}(\tilde{D}_2) = \Omega(1)$. Let the corresponding partition in A and g be

$$A = [A_1, \quad A_2], \;\; \Delta x = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \end{bmatrix}, \;\; g = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}. \tag{10}$$

Let M be the coefficient matrix of the normal equation (8). When μ is small, strict complementarity of the iterates (x, z) and (s, w) imply that we have

$$D_1 = \mu\tilde{D}_1, \qquad D_2 = \frac{1}{\mu}\tilde{D}_2$$

where $\text{diag}(\tilde{D}_i) = \Omega(1)$ for $i = 1, 2$. In this section, we shall assume that $\mu \ll 1$.

For the analysis below, we consider the SVD $A_1\tilde{D}_1^{-1/2} = U\Sigma V^T = U_1\Sigma_1 V_1^T$, where Σ_1 is the square diagonal matrix of positive singular values. Let U_2 be the matrix whose columns form an orthonormal basis of $\mathcal{N}(\tilde{D}_1^{-1/2}A_1^T) = \mathcal{N}(A_1^T)$. Note that U_2 is a null matrix iff $\mathcal{N}(A_1^T) = \{0\}$, i.e., iff A_1 has full row rank. Let, for $i, j = 1, 2$,

$$W_{ij} = U_i^T(A_2\tilde{D}_2^{-1}A_2^T)U_j. \tag{11}$$

Lemma 3.1: Assuming that A has full row rank, then W_{22} is nonsingular.

Proof. Since A has full row rank, it is easily shown that $\mathcal{N}(A_1^T) \bigcap \mathcal{N}(A_2^T) = \{0\}$. We show that $A_2^T U_2$ has full column rank by contradiction. Suppose $A_2^T U_2 v = 0$ for some $v \neq 0$. Then $U_2 v \in \mathcal{N}(A_2^T)$. Since we also have $U_1 v \in \mathcal{N}(A_1^T)$, this leads to a contradiction. Thus $U_2^T A_2$ has full row rank and W_{22} is nonsingular. □

Proposition 3.1: (a) If A_1 has full row rank, then $\kappa(M) \approx \kappa(A_1\tilde{D}_1^{-1}A_1^T)$.

(b) If A_1 does not have full row rank, then

$$\kappa(M) \approx \frac{1}{\mu^2}\|A_1\tilde{D}_1^{-1}A_1^T\|\|W_{22}^{-1}\|. \tag{12}$$

Note that $\|W_{22}^{-1}\| = O(1)$ since W_{22} is nonsingular.

Proof. (a) The result is obvious and we shall omit the proof.

(b) The matrix M can be written as

$$\begin{aligned} M &= A_1D_1^{-1}A_1^T + A_2D_2^{-1}A_2^T \\ &= \frac{1}{\mu}\left(A_1\tilde{D}_1^{-1}A_1^T + \mu^2 A_2\tilde{D}_2^{-1}A_2^T\right). \end{aligned}$$

Thus

$$M = \frac{1}{\mu}U\begin{bmatrix} \Sigma_1^2+\mu^2W_{11} & \mu^2W_{12} \\ \mu^2W_{12}^T & \mu^2W_{22}\end{bmatrix}U^T,$$

and we have

$$M^{-1} = \mu U\begin{bmatrix} G^{-1}+\mu^2G^{-1}W_{12}Q^{-1}W_{12}^TG^{-1} & -G^{-1}W_{12}Q^{-1} \\ -Q^{-1}W_{12}^TG^{-1} & \frac{1}{\mu^2}Q^{-1}\end{bmatrix}U^T$$

where $Q = W_{22} - \mu^2W_{12}^TG^{-1}W_{12}$ with $G = \Sigma_1^2 + \mu^2W_{11}$. It is clear that $\|M\| \approx \|A_1\tilde{D}_1^{-1}A_1^T\|/\mu$ and $\|M^{-1}\| \approx \|Q^{-1}\|/\mu \approx \|W_{22}^{-1}\|/\mu$, and the required result (12) follows readily. □

IV. Reduced augmented system

Let E_1 be a given positive definite diagonal matrix with the same dimension as D_1. (Usually we choose E_1 to be a positive multiple of the identity matrix.) Instead of computing Δx and Δy from (7), we propose to compute them by solving a smaller augmented system given in the next lemma.

Lemma 4.1: The solution of (7) can be computed from the following reduced augmented equation (RAE):

$$\underbrace{\begin{bmatrix} H & B \\ B^T & -\Psi\end{bmatrix}}_{K}\begin{bmatrix}\Delta y \\ \Delta\tilde{x}_1\end{bmatrix} = \begin{bmatrix} h \\ F_1^{-1/2}g_1\end{bmatrix} \tag{13}$$

where $F_1 = E_1 + D_1$, and

$$\begin{aligned} \Delta\tilde{x}_1 &= F_1^{-1/2}E_1\Delta x_1 \\ \Psi &= D_1E_1^{-1} \\ H &= A\mathrm{diag}(F_1^{-1},\, D_2^{-1})A^T \\ B &= A_1F_1^{-1/2} \\ h &= r_p + A\mathrm{diag}(F_1^{-1},\, D_2^{-1})g \end{aligned} \tag{14}$$

Once Δy has been computed, Δx_2 can be readily computed from the equation

$$\Delta x_2 = D_2^{-1}(A_2^T\Delta y - g_2). \tag{15}$$

Proof. By substituting the partitions of D, A and g into (7) and using $\Delta x_2 = D_2^{-1}(A_2^T\Delta y - g_2)$, we get

$$-D_1\Delta x_1 + A_1^T\Delta y = g_1 \tag{16}$$

$$A_1\Delta x_1 + A_2D_2^{-1}A_2^T\Delta y = r_p + A_2D_2^{-1}g_2. \tag{17}$$

By adding $A_1F_1^{-1}$ times (16) to (17), we have

$$A_1F_1^{-1}E_1\Delta x_1 + A\mathrm{diag}(F_1^{-1},\, D_2^{-1})A^T\Delta y = r_p + A\mathrm{diag}(F_1^{-1}, D_2^{-1})g.$$

The above equation, together with (16) scaled by $F^{-1/2}$ gives (13). □

Lemma 4.2:

$$\|K\| \le 2\max\Big\{\|H\|,\, \|B\|,\, \|\Psi\|\Big\}.$$

Proof. The proof is easy and we shall omit it. □

Lemma 4.3: The inverse of the reduced augmented matrix in (7) is given by

$$K^{-1} = \begin{bmatrix} H^{-1/2}(I-P)H^{-1/2} & H^{-1}BS^{-1} \\ S^{-1}B^TH^{-1} & -S^{-1}\end{bmatrix}, \tag{18}$$

where $S = B^TH^{-1}B + \Psi$, and $P = H^{-1/2}BS^{-1}B^TH^{-1/2}$ satisfies the condition $0 \preceq P \preceq I$, i.e., P and $I-P$ are positive semidefinite. Furthermore,

$$\|K^{-1}\| \le 2\max\Big\{\|H^{-1}\|, \|S^{-1}\|\Big\}$$

Proof. The proof of (18) can be deduced from [3, p. 389]. By the definition of S, we have $0 \preceq S^{-1/2}B^TH^{-1}BS^{-1/2} \preceq I$, and thus $\|H^{-1/2}BS^{-1/2}\| \le 1$. This implies that

$$\begin{aligned} \|H^{-1}BS^{-1}\| &\le \|H^{-1/2}\|\,\|H^{-1/2}BS^{-1/2}\|\,\|S^{-1/2}\| \\ &\le \|H^{-1/2}\|\|S^{-1/2}\| \\ &\le \max\Big(\|H^{-1}\|\,,\, \|S^{-1}\|\Big). \end{aligned}$$

It is easy to see that

$$\begin{aligned} \|K^{-1}\| &\le 2\max\Big\{\|H^{-1/2}(I-P)H^{-1/2}\|,\, \|H^{-1}BS^{-1}\|,\, \|S^{-1}\|\Big\} \\ &\le 2\max\Big\{\|H^{-1}\|,\, \|S^{-1}\|\Big\} \end{aligned}$$

□

Let the SVD of B be $B = U\Sigma V^T = U_1\Sigma_1V_1^T$, where Σ_1 is the diagonal matrix of positive singular values. Here U_1 and V_1 are the matrices whose columns form an orthonormal basis of $\mathcal{R}(B)$ and $\mathcal{R}(B^T)$, respectively. Let U_2 and V_2 be the columns of U and V other than those in U_1 and V_1, respectively. Then the columns of U_2 and V_2 form an orthonormal basis of $\mathcal{N}(B^T) = \mathcal{N}(A_1^T)$ and $\mathcal{N}(B)$, respectively.

Lemma 4.4: The following results hold.

(a) Consider the matrix $H := BB^T + \mu\tilde{W}$. We have

$$\mu H^{-1} = U_2W_{22}^{-1}U_2^T + O(\mu), \tag{19}$$

where $W_{22} = U_2^T\tilde{W}U_2$. Thus $\|H^{-1}\| = \|W_{22}^{-1}\|/\mu$ if $\mathcal{N}(A_1^T) \neq \{0\}$.

(b)

$$B^TH^{-1}B = V_1(I + O(\mu))V_1^T. \tag{20}$$

If B has full column rank, then $B^TH^{-1}B = I + V_1O(\mu)V_1^T$.

(c) Consider $S = B^TH^{-1}B + \Psi$. We have

$$\mu S^{-1} = V_2\Big(V_2^T\tilde{D}_1E_1^{-1}V_2\Big)^{-1}V_2^T + O(\mu).$$

Proof. (a) Let $W := U^T \tilde{W} U$ be written as

$$W \;=\; \begin{bmatrix} W_{11} & W_{12} \\ W_{12}^T & W_{22} \end{bmatrix},$$

where $W_{ij} = U_i^T \tilde{W} U_j$ for $i, j = 1, 2$. We have

$$H \;=\; U \begin{bmatrix} \Sigma_1^2 + \mu W_{11} & \mu W_{12} \\ \mu W_{12}^T & \mu W_{22} \end{bmatrix} U^T.$$

Thus

$$H^{-1} \;=\; U \begin{bmatrix} G^{-1} - \mu G^{-1} W_{12} Q^{-1} W_{12}^T G^{-1} & G^{-1} W_{12} Q^{-1} \\ Q^{-1} W_{12}^T G^{-1} & -\frac{1}{\mu} Q^{-1} \end{bmatrix} U^T$$

where $G = \Sigma_1^2 + \mu W_{11}$ and $Q = -W_{22} + \mu W_{12}^T G^{-1} W_{12}$. Thus

$$\mu H^{-1} \;=\; -U_2 Q^{-1} U_2^T + O(\mu) \;=\; U_2 W_{22}^{-1} U_2^T + O(\mu).$$

This completes the proof of (19).

(b) Note that we have

$$B^T H^{-1} B \;=\; V_1 \Sigma_1 \left(G^{-1} - \mu G^{-1} W_{12} Q^{-1} W_{12}^T G^{-1} \right) \Sigma_1 V_1^T.$$

Since $G^{-1} = \Sigma_1^{-2} + O(\mu)$, we get the result in (20) readily. When B has full column rank, V_1 is an orthogonal matrix, and hence $V_1 V_1^T = I$.

(c) It is clear that S can be written in the form

$$S \;=\; V \begin{bmatrix} I + O(\mu) + \mu \Psi_{11} & \mu \Psi_{12} \\ \mu \Psi_{12}^T & \mu \Psi_{22} \end{bmatrix} V^T,$$

where $\Psi_{ij} = V_i^T \tilde{D}_1 E_1^{-1} V_j$, for $i, j = 1, 2$. Using the same proof as in (a), the required result is easily shown. □

Proposition 4.1: Assume that $\mu \ll 1$ so that $\|\Psi\| \leq \|B\|$.

(a) If A_1 has full row rank, then BB^T is nonsingular and if A_1 has full column rank

$$\kappa(K) = O(1) \max\{1, \|B\|^2\} \max\{\|(BB^T)^{-1}\|, 1\};$$

Otherwise,

$$\kappa(K) = \frac{O(1)}{\mu} \max\{1, \|B\|^2\} \max\{\mu \|(BB^T)^{-1}\|, \|(V_2^T \tilde{D}_1 E_1^{-1} V_2)^{-1}\|\}.$$

(b) If A_1 does not have full row rank, then

$$\kappa(K) = \frac{O(1)}{\mu} \max\{1, \|B\|^2\} \max\{\|W_{22}^{-1}\|, \|(V_2^T \tilde{D}_1 E_1^{-1} V_2)^{-1}\|\}. \quad (21)$$

Proof. From lemma 4.2, and noting that $\|H\| \approx \|B\|^2$, it is easy to see that $\|K\| = O\Big(\max\{1, \|B\|^2\} \Big)$. (a) Since BB^T is nonsingular, we have $\|H^{-1}\| = \|(BB^T)^{-1}\| + O(\mu)$. If A_1 also has full column rank, then $S = B^T (BB^T)^{-1} B + \Psi = I + \mu \tilde{D}_1 E_1^{-1}$, implying that $\|S^{-1}\| = 1 + O(\mu)$. Thus $\|K^{-1}\| = O(1) \max\{\|(BB^T)^{-1}\|, 1\}$.

(b) From lemma 4.3, we have $\|H^{-1}\| = \|W_{22}^{-1}\|/\mu$, and $\|S^{-1}\| = \|(V_2^T \tilde{D}_1 E_1^{-1} V_2)^{-1}\|/\mu$. Thus $\|K^{-1}\| = O\Big(\max\{\|W_{22}^{-1}\|, \|(V_2^T \tilde{D}_1 E_1^{-1} V_2)^{-1}\|\} \Big)/\mu$, and the required result follows readily. □

A. Residual vectors

Lemma 4.5: Suppose Δy is computed approximately from the normal equation (8) with residual vector $\gamma = r_p + AD^{-1} g - AD^{-1} A^T \Delta y$. Assume that once Δy is given, Δx can be computed without rounding errors from (9), and $\Delta z, \Delta s, \Delta w$ are also computed exactly from (6). Then the residual vector associated with the Newton equation (4) is given by

$$[\gamma;\; 0;\; 0;\; 0;\; 0]\,.$$

Proof. The proof is straightforward and is thus omitted. □

Lemma 4.6: Suppose Δy and Δx_1 is computed approximately from the RAE (13) with residual vector

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} \;=\; \begin{bmatrix} h \\ F_1^{-1/2} g_1 \end{bmatrix} - \begin{bmatrix} H & B \\ B^T & -\Psi \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta \tilde{x}_1 \end{bmatrix}$$

Assume that once Δy and $\Delta \tilde{x}_1$ is given, Δx_2 can be computed without rounding errors from (15), and $\Delta z, \Delta s, \Delta w$ are also computed exactly from (6). Then the residual vector associated with the Newton equation (4) is given by

$$\Big[\xi - A_1 F_1^{-1/2} \eta;\; 0;\; (F_1^{1/2} \eta\;;\; 0);\; 0;\; 0\Big]\,.$$

Proof. It is easily shown that

$$\begin{aligned} -D\Delta x + A^T \Delta y \;&=\; g - [F_1^{1/2} \eta\;;\; 0] \\ A \Delta x \;&=\; r_p - \Big(\xi - A_1 F_1^{-1/2} \eta \Big), \end{aligned}$$

and the required result follows readily from the above equations. □

V. Preconditioning Approach

For the reduced augmented matrix in (13), the effect of an ill-conditioned D is less prominent, thus we expect the construction of an effective preconditioner to be easier. We shall study an approach of designing the preconditioner that exploits the block structure of the inverse of the reduced augmented matrix in this section.

The two general guidelines to design a preconditioning matrix P for a given matrix K are:

1. P should approximate K such that $P^{-1}K$ has good spectral properties.
2. Linear system with P as the coefficient matrix should be much easier to solve than the original system.

In this section, we shall attempt to construct a preconditioner for the reduced augmented matrix by approximating the block structure of its inverse, as proposed in [4].

Consider the block structure of the inverse of K:

$$K^{-1} \;=\; \begin{bmatrix} H^{-1} - H^{-1} B S^{-1} B^T H^{-1} & H^{-1} B S^{-1} \\ S^{-1} B^T H^{-1} & -S^{-1} \end{bmatrix} \quad (22)$$

where S is the Schur complement matrix of K, that is,

$$S \;=\; B^T H^{-1} B + \Psi. \quad (23)$$

This naturally leads us to consider a preconditioner P with the following block structure

$$P_c^{-1} \;=\; \begin{bmatrix} \widehat{H}^{-1} - \widehat{H}^{-1}B\widehat{S}^{-1}B^T\widehat{H}^{-1} & \widehat{H}^{-1}B\widehat{S}^{-1} \\ \widehat{S}^{-1}B^T\widehat{H}^{-1} & -\widehat{S}^{-1} \end{bmatrix} \quad (24)$$

where $\widehat{H}$ and $\widehat{S}$ are positive definite approximations of H and S, respectively.

To apply the preconditioner P_c in a Krylov subspace method, the preconditioning step $P_c^{-1}[u;v]$ can be computed efficiently as follows:

$$\begin{aligned} &\text{Compute} \quad w = \widehat{H}^{-1}u; \\ &\text{Compute} \quad z = \widehat{S}^{-1}(B^Tw - v); \\ &\text{Compute} \quad P_c^{-1}\,[u\,;\,v] = \left[\widehat{H}^{-1}(u - Bz)\,;\,z\right]. \end{aligned}$$

We avoid explicitly computing $\widehat{S}^{-1}$ by pre-computing the sparse Cholesky factorization of $\widehat{S}$, and then solve the resulting linear system each time we need to evaluate $z = \widehat{S}^{-1}(B^Tw - v)$ in the preconditioning step. Similar remark holds for $\widehat{H}^{-1}$.

Thus if $\widehat{H}$ and $\widehat{S}$ are effective approximations of H and S which are relatively simple to compute, the resulting preconditioned system is expected to perform better than the original system under an iterative solution method.

Theorem 1: (a) Suppose $\widehat{S} = B^T\widehat{H}^{-1}B + \Psi$ is used in P_c. Then $P_c^{-1}K$ has p eigenvalues clustered at 1. The remaining m real eigenvalues are those of the matrix

$$Y := G \,+\, (I - G)\widehat{H}^{-1/2}H\widehat{H}^{-1/2}, \quad (25)$$

where

$$G \;\;=\;\; \widehat{H}^{-1/2}B\,\widehat{S}^{-1}\,B^T\widehat{H}^{-1/2}.$$

(b) Suppose $\widehat{H} = H$ is used in the preconditioner P_c. Then $P_c^{-1}K$ has 1 as an eigenvalue with multiplicity m. The remaining p eigenvalues are those of the matrix $\widehat{S}^{-1}S$.

Proof. It is easily verified that

$$P_c^{-1}K \;=\; \begin{bmatrix} \widehat{H}^{-1}H - \widehat{H}^{-1}B\widehat{S}^{-1}B^T(\widehat{H}^{-1}H - I) & \widehat{H}^{-1}B\widehat{S}^{-1}(\widehat{S} - B^T\widehat{H}^{-1}B - \Psi) \\ \widehat{S}^{-1}B^T(\widehat{H}^{-1}H - I) & \widehat{S}^{-1}(B^T\widehat{H}^{-1}B + \Psi) \end{bmatrix}. \quad (26)$$

(a) Since $\widehat{S} = B^T\widehat{H}^{-1}B + \Psi$, the (2,2) block in (26) reduces to I_p. Thus

$$P_c^{-1}K \;=\; \begin{bmatrix} \widehat{H}^{-1}H - \widehat{H}^{-1}B\widehat{S}^{-1}B^T(\widehat{H}^{-1}H - I) & 0 \\ \widehat{S}^{-1}B^T(\widehat{H}^{-1}H - I) & I_p \end{bmatrix}. \quad (27)$$

It is clear that $P_c^{-1}K$ has 1 as an eigenvalue with multiplicity p, and the remaining eigenvalues are determined by its $(1,1)$ block. Note that the $(1,1)$ block of above preconditioned matrix is similar to

$$G \,+\, (I - G)\widehat{H}^{-1/2}H\widehat{H}^{-1/2}. \quad (28)$$

(b) When $\widehat{K} = K$, the preconditioned matrix in (26) becomes

$$P_c^{-1}A \;\;=\;\; \begin{bmatrix} I_m & K^{-1}B(I - \widehat{S}^{-1}S) \\ 0 & \widehat{S}^{-1}S \end{bmatrix}. \quad (29)$$

With the above expression, it is easy to see that the result stated in the theorem holds true. □

Proposition 5.1: Suppose we take $\widehat{H} = \text{diag}(H)$. The matrix

$$Y = G \,+\, (I - G)\widehat{H}^{-1/2}H\widehat{H}^{-1/2} \quad (30)$$

has p positive real eigenvalues clustered at $1 + O(\mu)$, and the remaining $m - p$ positive real eigenvalues are $O(\mu)$.

Proof. Let $J = B^T\widehat{H}^{-1/2}$. Then

$$G \;=\; J^T(JJ^T + \Psi)^{-1}J,$$

which has the same form as the matrix in part (b) of lemma 4.4. Thus $G = P_1(I + O(\mu))P_1^T$, where the columns of P_1 form an orthonormal basis of $\mathcal{R}(J^T)$. It is easily deduced that

$$P^TYP \;\;=\;\; \begin{bmatrix} I + O(\mu) & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} O(\mu) & 0 \\ 0 & I \end{bmatrix} P^T\widehat{H}^{-1/2}H\widehat{H}^{-1/2}P.$$

Now

$$\begin{aligned} P^T\widehat{H}^{-1/2}H\widehat{H}^{-1/2}P \;\;&=\;\; P^T(J^TJ)P + \mu P^T(\widehat{H}^{-1/2}\tilde{W}\widehat{H}^{-1/2})P \\ &=\;\; \begin{bmatrix} \Sigma_1 + O(\mu) & O(\mu) \\ O(\mu) & O(\mu) \end{bmatrix}, \end{aligned}$$

where Σ_1 is the diagonal matrix of positive singular values of J. □

Remark 5.1: The advantage of factorizing $B^TB = F^{-1/2}A_1^TA_1F^{-1/2}$ compared to $BB^T = A_1F^{-1}A_1^T$ is the following. If the partition of D does not change from one IPM iteration to the next, we can make use of the factorization $A_1^TA_1 = L\Lambda L^T$ to factorize $B^TB = \tilde{L}\Lambda\tilde{L}^T$ in the next IPM iteration, where $\tilde{L} = F^{-1/2}L$. But the same cannot be done for BB^T.

VI. Two-Phase Algorithm Design

From the discussion in the previous sections, we can deduce the following features of the preconditioner:

1. Spectral properties of the reduced augmented matrix K in (13) improves as elements in D_1 get smaller and D_2 get larger. In the context of interior-point algorithm, we expect K to become better-conditioned as the optimal solution is approached.
2. Certain amount of computational work is required to compute $\widehat{H}$, $\widehat{S}$ and the preconditioning step described in the previous section. Hence, $\widehat{H}$ and $\widehat{S}$ should be simple enough for efficient computation, yet sophisticated enough for effective preconditioning.

Note that point 1 is in contrast to the situation commonly encountered in applying preconditioned conjugate gradient method to the normal equations, where the linear system usually becomes more ill-conditioned as the interior-point

algorithm progresses, thus slowing the convergence rate of the iterative solution towards the end.

Hence, we propose a two-phase algorithm to solve for the Newton step directions in the interior-point iterations. In the initial phase of the interior-point algorithm, we apply some existing iterative method to solve for the step directions. Toward the end of the Newton iteration, when it is advantageous to employ the proposed reduced augmented system, we switch over to this system of equations.

By now we can identify a few issues in the design of this two-phase algorithm, namely, the criterion used to switch from phase 1 to phase 2, and partitioning of the matrix D into D_1 and D_2.

A. Switching criterion

There are a few switching criteria that seem to work fine:

1. Switch when the duality gap falls below a certain threshold.
2. Switch when the elements of matrix D form 2 distinct clusters.
3. If an iterative method is employed in phase 1, switch when it takes excessive number of iterations to converge.

We are still in the process of fine tuning the switching criterion.

B. Partitioning of matrix D

When the elements of D form two distinct clusters of different orders of magnitude, a convenient partition is to assign the cluster with smaller value to D_1 and the other cluster to D_2. However, sometimes no obvious cluster can be determined. In that case, one strategy is to assign elements in D that are less than 1 to D_1, and assign the rest to D_2. This simple strategy looks reasonable on most of the NETLIB LP problems. We are still fine tuning the strategy to determine the clusters in D.

VII. NUMERICAL EXPERIMENTS

In this section we present some experimental results of applying the proposed two-phase algorithm and the preconditioned reduced augmented system to solve some of the LP problems in the NETLIB suite. The purpose is to demonstrate that the proposed preconditioning approach and the idea of the two-phase algorithm have the potential to solve such problems efficiently. All the numerical experiments were carried out in MATLAB on a Pentium III 1 GHz PC with 256MB of RAM.

A. Implementation

We implemented our two-phase algorithm on the MATLAB-based software package LIPSOL [5], which employs the predictor-corrector primal-dual interior-point method to solve linear programs. The numerical experiment was run by replacing the direct solver in LIPSOL with our two-phase algorithm. In phase one, we use the preconditioned conjugate gradient (PCG) method to solve the normal equations (8) iteratively. When the elements in D_1 become much smaller than those in D_2, the algorithm switches to phase two, where we solve the reduced augmented equations (13) iteratively with symmetric quasi-minimal residual (SQMR) method [1]. The implementation of our algorithm can thus be summarized as follows:

S1. Initialize the interior-point algorithm in LIPSOL.
S2. If elements in D_1 are not much smaller than those in D_2, then
(*Phase 1*) Solve for the search directions using PCG on the normal equations;
else
(*Phase 2*) Solve the resulting reduced augmented system of equations for the search directions by SQMR method.
S3. If the interior-point iterate converges, stop the iteration. Otherwise, solve for the next search directions.

In the experiment, we used incomplete Cholesky factorization of the normal matrix as the preconditioner in PCG. The drop tolerance for the incomplete Cholesky factorization was set to 10^{-3}.

We assigned elements in D that are less than 1 to D_1 and the rest to D_2. When the geometric mean of D_2 is more than 10^8 times larger than that of D_1, we switch to phase 2 of the algorithm.

To perform the preconditioning step in SQMR, we chose $\hat{H}$ to be the diagonal matrix formed by taking the diagonal of H, and chose $\hat{S}$ to be

$$\widehat{S} \;=\; B^T \widehat{H}^{-1} B + \Psi.$$

B. Stopping criteria

The interior-point algorithm in LIPSOL settles on a solution when the feasibility and duality gaps are small enough. Specifically, a solution is considered to have converged when the following condition is satisfied:

$$\tau = \max\left(\frac{\|r_p\|}{1+\|b\|}, \frac{\|r_d\|}{1+\|c\|}, \frac{\|r_u\|}{1+\|u\|}, \frac{\|x^T z + s^T w\|}{n+n_u}\right) < 10^{-8}.$$

where n_u is the number of nonzeros in u, and the rest of the variables are taken from the residual equations in (5).

By Lemma 4.5, the stopping criterion used in PCG for solving the normal equations is

$$\|\gamma\| < \kappa_P \tau$$

where κ_P is a constant parameter to be set in the experiment.

Similarly, by Lemma 4.6, the stopping criterion used in SQMR for solving the reduced augmented equations is

$$\max\left(\frac{\|\xi - A_1 F_1^{-1/2}\eta\|}{1+\|b\|}, \frac{\|F_1^{1/2}\eta\|}{1+\|c\|}\right) < \kappa_Q \tau$$

where κ_Q is a constant parameter to be set in the experiment.

The values we set for κ_P and κ_Q were both equal to 10^{-2}.

C. Experimental results

In our numerical experiment, we tested our two-phase algorithm on several problems from the NETLIB collection of LP test problems. The vital statistics of these problems are summarized in Table I.

We have generated three sets of results from the experiment as presented in Table II. The first column displays the running time taken by direct solution of the normal equations, that is, the original LIPSOL was used to solve the problems. The second column shows the results of applying PCG on the normal equations, while the last column contains the results generated by our two-phase algorithm, which is a hybrid of PCG on the normal equations and SQMR on the reduced augmented equations. A cross '$\times$' in the table under a particular method indicates failure to converge to a solution for the corresponding problem using that method, while a dash '-' under the hybrid column means that the algorithm did not switch phase. In the experiment, the number of PCG iterations was capped at $\max(500, m)$, where m is the size of the normal matrix.

Tables III and IV show a detailed breakdown of the number of iterations taken by the iterative methods on the problems `pds-06` and `maros-r7`, respectively. In each table, the leftmost column indicates the interior-point iterates, the next two columns show the number of PCG iterations in the predictor and corrector step of each interior-point iterate, while the last two columns show the number of iterations used by the two-phase algorithm.

D. Discussion

We can observe from the experimental results that the number of PCG iterations on the normal equations generally increases as the interior-point progresses, while that of the reduced augmented equations generally decreases as the interior-point approaches a solution. The results also indicate that the PCG method fails to converge to a solution for some of the problems.

With the two-phase algorithm, the success rate is much higher, and it converges to a solution faster than using the PCG method alone for most of the problems. Judging from the running times, the algorithm still cannot measure up to the direct method. One reason is that LIPSOL has been highly optimized, while our implementation has not yet been refined. However, it does show some potential in solving larger problems such as `maros-r7` and `pds-20`.

In each predictor or corrector step of an interior-point iterate, the SQMR algorithm needs to solve the Schur complement system (23) for as many times as its number of iterations, whereas the direct algorithm of LIPSOL solves the Schur complement system (8) exactly once. Hence, the relative size and structure of the two Schur complement matrices play an important role in determining the relative performance of the two algorithms. For instance, some of the problems have denser rows than columns in the constraint matrices. Proper handling of these dense rows might be needed to solve the reduced augmented system more efficiently.

VIII. CONCLUSION

In this report we have presented a preconditioning approach that transforms the augmented system into a reduced system that is likely to become better-conditioned toward the end of the interior-point algorithm. A preconditioner is then designed by approximating the block structure of the inverse of the transformed matrix to further improve the spectral properties of the transformed system. Capitalizing on the special spectral properties of the transformed matrix, we also proposed a two-phase algorithm that solves the linear system using an existing technique such as PCG in the beginning, and then switches to solve the reduced augmented system when the interior-point iterates approach a solution.

The experimental results presented in the last section have demonstrated the potential of our proposed method in solving large scale LP problems. In particular, we have seen that it greatly enhances the performance of iterative solution of the linear equations. The results have also highlighted several areas of the method that need to be further developed and refined.

1) Fine tune the switching mechanism to make optimum use of the complementary methods in phase one and two.
2) Improve the preconditioner design to minimize the computational effort and yet remain effective in improving the spectral properties.
3) Handling of dense columns and rows in the constraint matrix in PCG and SQMR, respectively.

Finally, the next step that we can explore to take this iterative approach to the next level may include the following areas:

1) Reduce the computational effort in the preconditioning step of the reduced augmented system by using incomplete Cholesky factor of $\hat{S}$ rather than the exact factor.
2) Experiment with another form of block preconditioning by letting $\hat{H}$ be incomplete factorization of H and $\hat{S}$ be $diag(S)$.

REFERENCES

[1] R.W. Freund and N.M. Nachtigal, *A new Krylov-subspace method for symmetric indefinite linear systems*, Proceedings of the 14th IMACS World Congress on Computational and Applied Mathematics, W.F. Ames (Ed.), IMACS, 1253-1256, 1994.

[2] 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*, Technical Report TR97-27, Department of Computational and Applied Mathematics, Rice University, 1997.

[3] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996.

[4] K.C. Toh, K.K. Phoon and S.H. Chan, *Block Preconditioners for Symmetric Indefinite Linear Systems*, International J. Numerical Methods in Engineering, in press.

[5] Y. Zhang, *User's Guide to LIPSOL*, http://www.caam.rice.edu/ zhang/lipsol, Department of Computational and Applied Mathematics, Rice University, 1998.

Problem	Rows	Columns	Nonzeros
afiro	27	51	102
adlittle	55	137	417
israel	174	316	2443
fffff800	501	1005	6283
ship12s	466	2293	6556
ganges	1137	1534	6593
sctap3	1480	3340	9734
bnl2	2268	4430	14914
pds-02	2788	7551	16230
degen3	1503	2604	25432
d2q06c	2171	5831	33081
pilot	1441	4657	42300
pds-06	9617	29087	62582
pilot87	2030	6460	72479
pds-10	16239	49613	106802
fit2d	25	10524	129042
osa-07	1118	25067	144812
maros-r7	3136	9408	144848
pds-20	33250	107627	231155

TABLE I

PROBLEM STATISTICS.

Problem	Direct method	PCG	Hybrid method
afiro	0.24	0.26	
adlittle	0.37	0.50	0.51
israel	1.33	×	2.98
fffff800	1.96	28.97	10.74
ship12s	0.92	×	3.61
ganges	1.82	×	11.85
sctap3	1.81	5.11	5.04
bnl2	8.95	×	256.36
pds-02	9.59	48.31	34.07
degen3	12.88	×	90.71
d2q06c	19.50	×	390.10
pilot	31.28	×	258.55
pds-06	163.41	520.21	345.67
pilot87	144.80	×	1025.68
pds-10	884.67	×	891.10
fit2d	20.29	20.00	-
osa-07	33.08	×	73.29
maros-r7	152.19	×	118.33
pds-20	10839.64	×	5996.89

TABLE II

SOLUTION TIME IN SECONDS, '×' INDICATES FAILURE TO CONVERGE, '-' INDICATES NO SWITCHING HAS TAKEN PLACE.

IP	PCG		Hybrid		IP	PCG		Hybrid	
step	P	C	P	C	step	P	C	P	C
1	2	1	2	1	24	67	62	67	62
2	1	2	1	2	25	62	59	62	59
3	2	2	2	2	26	69	64	69	64
4	3	3	3	3	27	67	63	67	63
5	3	3	3	3	28	72	67	72	67
6	3	4	3	4	29	72	67	72	67
7	3	2	3	2	30	77	72	77	72
8	3	3	3	3	31	78	75	78	75
9	2	3	2	3	32	86	76	86	76
10	2	3	2	3	33	98	90	74	52
11	2	3	2	3	34	90	91	67	60
12	3	3	3	3	35	111	113	77	46
13	5	5	5	5	36	125	122	67	49
14	6	7	6	7	37	158	146	68	41
15	8	7	8	7	38	181	178	52	37
16	9	8	9	8	39	175	160	48	35
17	12	8	12	8	40	237	237	50	37
18	18	17	18	17	41	249	242	36	29
19	20	19	20	19	42	358	292	48	9
20	32	29	32	29	43	500	500	2	0
21	53	53	53	53	44	358	190	2	0
22	73	71	73	71					
23	56	48	56	48					

TABLE III

PROBLEM pds-06: NUMBER OF ITERATIONS TAKEN BY PCG AND HYBRID METHOD FOR PREDICTION (P) AND CORRECTION (C) IN EACH INTERIOR-POINT (IP) STEP. THE HYBRID METHOD SWITCHES PHASE AFTER 32 IP STEPS.

IP	PCG		Hybrid	
step	P	C	P	C
1	1	0	1	0
2	1	1	1	1
3	2	1	2	1
4	2	1	2	1
5	6	5	6	5
6	4	3	4	3
7	6	5	6	5
8	10	9	10	9
9	36	32	36	32
10	210	206	2	2
11	500	500	3	3
12	500	500	3	3
13	500	500	4	4
14	×	×	4	4
15			4	3
16			2	2
17			2	2

TABLE IV

PROBLEM maros-r7: NUMBER OF ITERATIONS TAKEN BY PCG AND HYBRID METHOD FOR PREDICTION (P) AND CORRECTION (C) IN EACH INTERIOR-POINT (IP) STEP. THE HYBRID METHOD SWITCHES PHASE AFTER 9 IP STEPS.