

Application of a Class of Preconditioners to Large Scale Linear Programming Problems

Venansius Baryamureeba, Trond Steihaug, and Yin Zhang*

Department of Informatics,
University of Bergen,
5020 Bergen, Norway

Abstract. In most interior point methods for linear programming, a sequence of weighted linear least squares problems are solved, where the only changes from one iteration to the next are the weights and the right hand side. The weighted least squares problems are usually solved as weighted normal equations by the direct method of Cholesky factorization. In this paper, we consider solving the weighted normal equations by a preconditioned conjugate gradient method at every other iteration. We use a class of preconditioners based on a low rank correction to a Cholesky factorization obtained from the previous iteration. Numerical results show that when properly implemented, the approach of combining direct and iterative methods is promising.

Key Words. Weighted linear least squares, Parallel processing, Preconditioners, Linear programming, Primal-dual infeasible interior point algorithms.

1 Introduction

The class of preconditioners we will consider is a low rank correction of a Cholesky factorization of a weighted normal equation coefficient matrix. The combination of a direct method and an iterative method was reported by Karmarkar and Ramakrishnan [8]. A low rank correction to the matrix was used by Goldfarb and Liu [6] to prove a low complexity bound for convex quadratic programming. A frequently used iterative method [3, 8, 9] is the preconditioned conjugate gradient method on the normal equations.

Throughout this paper we use the following notation: \min_i or \max_i is for all i for which the argument is defined. For any matrix A , A_{ij} is the element in the i -th row and j -th column, A_j is the j -th column, and $A_{j\bullet}$ is the j -th row. The symbol I is used to denote the identity matrix; its size will always be apparent from the context. For any square matrix X , $\lambda_i(X)$ are the eigenvalues of X arranged in non decreasing order. The letters L and R represent lower triangular factors or lower Cholesky factors, and the type of factors will always be apparent from the context.

* Department of Computational and Applied Mathematics, Rice University, Houston, Texas 77005, USA.

2 Problem

Consider a primal linear programming problem in standard form:

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to:} && Ax = b, \quad x \geq 0 \\ &&& A \in \mathfrak{R}^{m \times n}, \quad b \in \mathfrak{R}^m, \quad c, x \in \mathfrak{R}^n. \end{aligned} \tag{1}$$

It can be shown (see [2]) that the normal equation form for a primal-dual interior point method corresponding to problem (1) is of the form

$$AGA^T y = AGh \tag{2}$$

where G is a diagonal and positive definite matrix. At every interior point iteration, (2) is solved.

In the following, the matrix A will be a full rank m -by- n matrix, $m \leq n$.

3 The Class of Preconditioners

Let $G, H \in \mathfrak{R}^{n \times n}$ be positive definite and diagonal matrices. For a given index set $\mathcal{Q} \subseteq \{1 \leq j \leq n \mid G_{jj} \neq H_{jj}\}$, let the $n \times n$ diagonal matrices \bar{H} and \bar{G} be given by

$$\bar{H}_{jj} = \begin{cases} H_{jj} & \text{if } j \in \mathcal{Q} \\ 0 & \text{otherwise,} \end{cases} \quad \text{and} \quad \bar{G}_{jj} = \begin{cases} G_{jj} & \text{if } j \in \mathcal{Q} \\ 0 & \text{otherwise.} \end{cases}$$

Then, the diagonal matrix K defined in (3) is positive definite.

$$K = H + \bar{G} - \bar{H}. \tag{3}$$

Note that $K_{jj} = G_{jj}$ if $j \in \mathcal{Q}$, otherwise $K_{jj} = H_{jj}$. In the mixed interior point method H is a weight matrix at a previous iteration where a direct method is used to solve (2), and G is the weight matrix at the current interior point iteration. We can thus assume that we have a factorization of $AHA^T = LL^T$.

We denote the number of elements in \mathcal{Q} by $q = |\mathcal{Q}|$. Let $\bar{A} \in \mathfrak{R}^{m \times q}$ consist of all columns A_j such that $j \in \mathcal{Q}$. Further, let $\bar{D} \in \mathfrak{R}^{q \times q}$ be the nonzero diagonal submatrix of $\bar{G} - \bar{H}$ corresponding to the submatrix \bar{A} . Then

$$AKA^T = A(H + \bar{G} - \bar{H})A^T = AHA^T + \bar{A}\bar{D}\bar{A}^T. \tag{4}$$

An approximation to AGA^T is AKA^T . Observe that if $\mathcal{Q} = \{1 \leq j \leq n \mid G_{jj} \neq H_{jj}\}$ then $AKA^T = AGA^T$ so the approximation can be made arbitrary good.

Theorem 1. [2]: *Let $G, H \in \mathfrak{R}^{n \times n}$ be positive definite and diagonal. Let γ_j be the sorted elements of G_{jj}/H_{jj} in non-decreasing order*

$$\gamma_1 = \min_j \{G_{jj}/H_{jj}\} \leq \gamma_2 \leq \dots \leq \gamma_n = \max_j \{G_{jj}/H_{jj}\}.$$

Let \mathcal{Q} consist of the indices corresponding to the q_1 largest and the q_2 smallest diagonal elements of $H^{-1}G$. Then

$$\min\{\gamma_{q_2+1}, 1\} \leq \lambda_i((AKA^T)^{-1}AGA^T) \leq \max\{\gamma_{n-q_1}, 1\}.$$

Theorem 1 suggests choosing \mathcal{Q} to consist of indices j corresponding to q_1 largest and q_2 smallest diagonal elements of $H^{-1}G$. Further, the condition number of the preconditioned matrix is bounded above

$$\kappa((AKA^T)^{-1}AGA^T) \leq \frac{\max\{\gamma_{n-q_1}, 1\}}{\min\{\gamma_{q_2+1}, 1\}}.$$

We can keep the condition number of $(AKA^T)^{-1}AGA^T$ bounded by a prescribed constant if we carefully choose q_1 and q_2 sufficiently large. The next result from [2] gives a lower bound on the condition number.

Theorem 2. *Let $G, H \in \mathbb{R}^{n \times n}$ be positive definite and diagonal. Let $|\mathcal{Q}| = q_1 + q_2 < m$, where q_1 is the number of indices in \mathcal{Q} where $G_{jj}/H_{jj} > 1$. Let K be defined as in (3). Then*

$$\kappa((AKA^T)^{-1}AGA^T) \geq \frac{\lambda_{m-q_1}((AHA^T)^{-1}AGA^T)}{\lambda_{q_2+1}((AHA^T)^{-1}AGA^T)}.$$

Wang and O’Leary [10] discuss a strategy of preconditioning the normal equation system based on formulation (4). In their implementation, they choose the index set \mathcal{Q} to consist of the indices corresponding to the largest values of $|G_{jj} - H_{jj}|$.

3.1 Computing the Preconditioner

We consider two approaches of solving the linear system $AKA^T z = r$, namely the Sherman Morrison-Woodbury formula approach, and the approach based on updating the triangular factors.

Sherman Morrison-Woodbury Formula Let $AHA^T = LL^T$. Applying the Sherman Morrison-Woodbury formula ([7] pp. 50) to $AKA^T = LL^T + \bar{A}\bar{D}\bar{A}^T$ followed with some manipulation yields

$$(AKA^T)^{-1} = L^{-T}(I - VF^{-1}V^T)L^{-1}, \tag{5}$$

where $V = L^{-1}\bar{A}$ and $F = \bar{D}^{-1} + V^T V$. Solving $AKA^T z = r$ for z is equivalent to computing

$$z = (AKA^T)^{-1}r = L^{-T}[d - (V(F^{-1}(V^T d)))]], \tag{6}$$

where $d = L^{-1}r$. To compute z in (6) we need to store L, V, d , and the factors of F . Note that F is a $q \times q$ symmetric matrix that may be indefinite.

Algorithm 3.1 Dense case
 Define $V = \bar{A}$; $C = \bar{D}$

```

for  $i = 1, \dots, m$  do
     $p = (V_{i\bullet}C)^T$ 
     $T_{ii} = D_{ii} + V_{i\bullet}p$ 
     $u = (1/T_{ii})p$ 
     $C \leftarrow C - up^T$ 
    for  $j = i + 1, \dots, m$  do
         $V_{j\bullet} \leftarrow V_{j\bullet} - L_{ji}V_{i\bullet}$ 
         $R_{ji} = L_{ji} + V_{j\bullet}u$ 
    
```

Algorithm 3.2 Sparse case
 Define $V = \bar{A}$; $C = \bar{D}$; $R = L$; $T = D$

```

for  $i = 1, \dots, m$  do
    if  $V_{i\bullet} \neq 0$  then
         $p = (V_{i\bullet}C)^T$ 
         $T_{ii} = D_{ii} + V_{i\bullet}p$ 
         $u = (1/T_{ii})p$ 
         $C \leftarrow C - up^T$ 
        for  $j = i + 1, \dots, m$  do
            if  $L_{ji} \neq 0$  then
                 $V_{j\bullet} \leftarrow V_{j\bullet} - L_{ji}V_{i\bullet}$ 
                 $R_{ji} = L_{ji} + V_{j\bullet}u$ 
            
```

Fig. 1. Updating the triangular factors

Updating the Triangular Factors Suppose we have the factorization $AHA^T = LDL^T$. Now, we want to compute R and T so that $AKA^T \equiv LDL^T + \bar{A}\bar{D}\bar{A}^T = RTR^T$. Since the sparsity structure of AKA^T is the same as for AHA^T , sparsity structure of R is the same as for L . An algorithm for updating factors of dense and sparse matrices is given in Baryamureeba and Steihaug [1]. For algorithms updating factors of dense matrices see [1] and references therein. We state the algorithms in Figure 1. The sparse-case algorithm is very effective when there are either many zero rows in \bar{A} or few nonzero elements in the unit lower triangular factor L .

The major computation in an interior point algorithm is in solving linear systems. Demmel et al. [4] discuss parallel algorithms for conjugate gradients and computing Cholesky factors. The construction of the preconditioner (updating of the triangular factors) can be done in parallel.

4 Numerical Results

Here, we give some numerical results using the same implementation details as in [2]. The test problems are from the Netlib set [5] of linear programs. The test code is implemented in MATLAB

The variable ϵ is a proximity measure of the interior point iterations to a solution of the linear programming problem and τ is the number of iterations (or corrections) allowed for the preconditioned conjugate gradient method. The percentages in Table 1 are based on that approximately half of the direct solves are replaced by an iterative solution. The gain is therefor approximately the ratio of the difference between the two methods and half of the total time. The results in Table 1 show that the mixed primal-dual Newton (mixed PDN) interior-point method, which alternatively uses a direct (Cholesky factorization) method and a preconditioned (with the preconditioner described in Section 3) conjugate gradient method to solve (2), competes favourably with the primal-dual Newton (PDN) interior-point method on large-scale problems.

Problem name	Rows (m)	Columns (n)	PDN		Mixed PDN		Gain
			iter	cpu-time	iter	cpu-time	%
czprob	737	3141	56	89.01	59	59.58	66
d2q06c	2171	5831	54	544.85	57	421.33	45
d6cube	404	6184	42	317.58	45	239.16	51
stocfor2	2157	3045	41	113.26	44	98.05	27
scsd8	397	2750	16	21.41	19	17.49	37

Table 1. Comparison of two implementations of primal-dual Newton interior point method. If $\epsilon \geq 0.1$ then $\tau = 7$, otherwise $\tau = 40$. Fixed value of $q = 6$, $q_1 = 3$.

The numerical results show that the mixed PDN method is promising and merits further study. In this paper the mixed PDN method is based on the odd-even alternation technique. Other dynamic alternation schemes [10] which could be based on the easily computable bounds on condition numbers should be investigated.

References

- [1] V. Baryamureeba, and T. Steihaug, *Computational issues for a new class of preconditioners*, To appear in the Proceedings for the 2nd Workshop on Large-Scale Scientific Computations, 1999.
- [2] V. Baryamureeba, T. Steihaug and Y. Zhang, *A class of preconditioners for weighted least squares problems*, Technical Report No. 170, Department of Informatics, University of Bergen, 5020 Bergen, Norway, April 30, 1999.
- [3] T.J. Carpenter and D.F. Shanno, *An interior point method for quadratic programs based on conjugate projected gradients*, Computational Optimization and Applications, Vol. 2, pp. 5-28, 1993.
- [4] J. W. Demmel, M.T. Heath, and H.A. Van der Vorst, *Parallel numerical linear algebra*, Acta Numerica, pp. 111-197, 1993.
- [5] D.M. Gay, *Electronic mail distribution of linear programming test problems*, Mathematical Programming Society COAL Newsletter, No. 13, pp.10-12, 1985.
- [6] D. Goldfarb, and S. Liu, *An $O(n^3L)$ primal interior-point algorithm for convex quadratic programming*, Mathematical Programming, Vol. 49, pp. 325-340, 1991.
- [7] G. H. Golub, and Charles H. Van Loan, *Matrix computations*, Third Edition, 1996.
- [8] N.K. Karmarkar and K.G. Ramakrishnan, *Computational results of an interior point algorithm for large-scale linear programming*, Mathematical Programming, Vol. 52, pp. 555-586, 1991.
- [9] S. Mehrotra, *Implementations of affine scaling methods: Approximate solutions of systems of linear equations using preconditioned conjugate gradient methods*, ORSA J. Comput., 4(1992), pp. 103-118.
- [10] W. Wang, and D.P. O'Leary, *Adaptive use of iterative methods in predictor-corrector interior point methods for linear programming* Technical Report No. CS-TR-4011, Computer Science Department, University of Maryland, April 1999. (Revised version of TR-3560, November 1995)