Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

ITERATIVE METHOD FOR THE LEAST SQUARES PROBLEM OF A MATRIX EQUATION WITH TRIDIAGONAL MATRIX CONSTRAINT*

HUAWEI PAN† AND YUAN LEI‡

Abstract. The matrix-form LSQR method is presented in this paper for solving the least squares problem of the matrix equation $AXB = C$ with tridiagonal matrix constraint. Based on a matrix-form bidiagonalization procedure, the least squares problem associated with the tridiagonal constrained matrix equation $AXB = C$ reduces to a unconstrained least squares problem of linear system, which can be solved by using the classical LSQR algorithm. Furthermore, the preconditioned matrix-form LSQR method is adopted for solving the corresponding least squares problem.

Key words. Matrix iterative method, Matrix Krylov subspace, Least squares tridiagonal solution, Preconditioner.

AMS subject classifications. 15A24, 65F20, 65F22, 65K10.

1. Introduction.

Throughout this paper, we denote the set of all $m \times n$ real matrices by $\mathbb{R}^{m\times n}$, the set of all $n \times n$ real tridiagonal matrices by $\mathbb{TR}^{n\times n}$ and the $k \times k$ identity matrix by I_k. For a matrix $A \in \mathbb{R}^{m\times n}$, we denote its transpose and trace by A^T and $\mathrm{tr}(A)$, respectively. The symbol $\|\cdot\|_F$ denotes the Frobenius norm associated with the inner product $\langle A, B\rangle = \mathrm{tr}(B^T A)$ for all $A, B \in \mathbb{R}^{n\times m}$. In this paper, we call two matrices A and B orthogonal if $\langle A, B\rangle = 0$, and use $A \otimes B$ to stand for the Kronecker product of matrices A and B.

The least squares problem of the linear matrix equation

$$AXB = C \tag{1.1}$$

usually arises in the structural dynamic design and finite element model updating problem [5, 21], and the constrained least squares solution $\widehat{X}$ is used to update the the preliminary estimation matrix $\bar{X}$. Various matrix decomposition methods and iterative methods for the least squares problem with symmetric or symmetric

*Received by the editors on February 22, 2011. Accepted for publication on December 16, 2012. Handling Editor: Bryan L. Shader.

†College of information science and engineering, Hunan University, Changsha, Hunan 410082, China (hw_pan@163.com).

‡College of Mathematics and Economics, Changsha, Hunan 410082, China (yleimath@hnu.edu.cn). The work was supported by National Natural Science Foundations of China (no. 11201136) and Fundamental Research Funds for the Central Universities (no. 531107040014).

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

positive semidefinite matrix constraints have been widely discussed in the literature [3, 4, 13, 17, 20]. However, we should point out that the preliminary estimation matrix $\bar{X}$ is derived by the finite element discretization method in finite element model updating problems, so the preliminary estimation matrix $\bar{X}$ usually possesses the sparse tridiagonal matrix structure [2]. In order to improve the updating precision, the updated matrix $\widehat{X}$ should preserve the same sparse matrix structure. Hence, it is necessary to consider the following least squares problem:

Problem A. Given matrices $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{n\times p}$ and $C \in \mathbb{R}^{m\times p}$, find $\widehat{X} \in \mathbb{TR}^{n\times n}$ such that

$$\|A\widetilde{X}B - C\|_F = \min_{X\in\mathbb{TR}^{n\times n}} \|AXB - C\|_F. \tag{1.2}$$

The updated matrix $\widehat{X}$ is an optimal approximation solution of Problem A to the preliminary estimation matrix $\bar{X} \in \mathbb{TR}^{n\times n}$, i.e.,

$$\|\widehat{X} - \bar{X}\|_F = \min_{\widetilde{X}\in S_E} \|\widetilde{X} - \bar{X}\|_F, \tag{1.3}$$

where S_E denotes the solution set of Problem A. Obviously, the optimal approximation solution $\widehat{X}$ is unique to the given matrix $\bar{X}$ due to the fact that S_E is a closed convex set. If there does not exist a preliminary estimation matrix [11], i.e., the matrix $\bar{X} = 0$, then the updated matrix $\widehat{X}$ is referred to as the minimum norm solution of Problem A. Similar to [8, 12], the optimal approximation problem (1.3) is equivalent to the problem of finding the least-squares tridiagonal solution of a new matrix equation $A\ddot{X}B = \ddot{C}$ with minimum norm. Therefore, without loss of generality, we only consider the minimum norm solution of Problem A in this paper.

By using the Kronecker product of matrices, the least squares problem (1.2) can be equivalently rewritten as a unconstrained least squares problem in vector-form

$$\min_x \|(B^T \otimes A)Kx - \mathrm{vec}(C)\|_2,$$

where K is the basis-matrix of the linear space $\mathrm{vec}(\mathbb{TR}^{n\times n})$, see Magnus [15] for details. But its use is restricted to the case when n is small, and the difficulty in solving the large order linear system makes it impractical. In addition, the matrix decomposition methods developed in [3, 14, 20] cannot be used to solve the least squares problems (1.2), and the difficulty lies in the fact that the matrix product usually does not preserve the sparse matrix structure. Therefore, the aim of this paper is to present iteration method for solving the least squares problem (1.2) without the employment of the Kronecker product.

In this paper, a matrix-form bidiagonalization procedure is given to compute a set of orthonormal basis of a matrix Krylov subspace, which includes the unique minimum

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

norm solution of Problem A. Based on the matrix-form bidiagonalization procedure, the least squares problem (1.2) reduces to a unconstrained least squares problem of a linear system, and then the classical LSQR algorithm are used as the framework for deriving the matrix-form iteration method. Next, the preconditioned matrix-form LSQR method is adopted for solving the corresponding least squares problem, and the CIMGS preconditioner proposed in [1, 19] is applied to accelerate the convergence of the matrix-form LSQR method presented in this paper.

This paper is organized as follows. In Section 2, we first introduce some preliminary results and then prove that the minimum norm solution of Problem A belongs to a matrix Krylov subspace. In Sections 3 and 4, the matrix-form LSQR algorithms and the corresponding preconditioned method are proposed for solving the least squares problem (1.2), and a simple stopping criterion is determined. In Section 5, we present several numerical examples and use some brief concluding remarks in Section 6 to end our paper.

2. Some preliminary results. We denote the orthogonal projection of $\mathbb{R}^{n\times n}$ onto $\mathbb{TR}^{n\times n}$ by τ. Hence, it is easy to verify that $\langle X, Y\rangle = \langle \tau(X), Y\rangle$ for any $X \in \mathbb{R}^{n\times n}$ and $Y \in \mathbb{TR}^{n\times n}$. Let $\varphi : \mathbb{TR}^{n\times n} \to \mathbb{R}^{m\times p}$ be the linear map with $\varphi(X) = AXB$. The map $\varphi^* : \mathbb{R}^{m\times p} \to \mathbb{TR}^{n\times n}$ by $\varphi^*(Y) = \tau(A^TYB^T)$ is called the adjoint operator of φ because the following equality

$$\langle \varphi(X), Y\rangle = \langle AXB, Y\rangle = \langle X, A^TYB^T\rangle = \langle X, \tau(A^TYB^T)\rangle = \langle X, \varphi^*(Y)\rangle$$

holds for any $X \in \mathbb{TR}^{n\times n}$ and $Y \in \mathbb{R}^{m\times p}$.

Let the product of the operators φ and φ^* be $\psi = \varphi^* \circ \varphi$, which is a linear operator on $\mathbb{TR}^{n\times n}$ given by $\psi(X) = \tau(A^TAXBB^T)$. We call

$$\mathcal{K}(\psi, V) = \operatorname{span}\{V, \psi(V), \ldots, \psi^{k-1}(V), \ldots\} \subseteq \mathbb{TR}^{n\times n} \tag{2.1}$$

the matrix Krylov subspace generated by the linear operator ψ and the tridiagonal matrix V, where

$$V = \tau(A^TCB^T) \in \mathbb{TR}^{n\times n}.$$

In the sequel, $\psi^k(V)$ is defined recursively as $\psi^k(V) = \psi(\psi^{k-1}(V))$ for $k \geq 1$ and $\psi^0(V) = V$.

In this section, we will show that the unique minimum norm solution of Problem A belongs to the matrix Krylov subspace $\mathcal{K}(\psi, V)$. To this end, we give some theoretical results about the least squares tridiagonal solution of the matrix equation (1.1) by using the projection theorem in finite-dimension inner product space [18].

LEMMA 2.1. *The matrix $\widetilde{X}$ is a solution of Problem A if and only if*

$$\psi(\widetilde{X}) = V. \tag{2.2}$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

Moreover, any least squares tridiagonal solution $\ddot{X}$ *of (1.1) can be expressed as* $\ddot{X} = \widetilde{X} + \bar{X}$*, where the matrix* $\bar{X} \in \mathbb{TR}^{n\times n}$ *satisfies the linear homogeneous matrix equation* $A\bar{X}B = 0$*.*

Proof. Let

$$\mathcal{L} = \{Z \mid Z = AXB,\ X \in \mathbb{TR}^{n\times n}\}.$$

Then $\mathcal{L}$ is obviously a linear subspace of $\mathbb{R}^{m\times p}$. By the projection theorem, we know that the matrix $\widetilde{X}$ is a solution of Problem A if and only if the corresponding residual $\widetilde{R} = C - A\widetilde{X}B$ satisfies $\widetilde{R} \perp \mathcal{L}$, i.e.,

$$\begin{aligned}\langle AXB, C - A\widetilde{X}B\rangle &= \langle X, A^TCB^T - A^TA\widetilde{X}BB^T\rangle \\ &= \langle X, \tau(A^TCB^T) - \tau(A^TA\widetilde{X}BB^T)\rangle \\ &= 0\end{aligned}$$

for all $X \in \mathbb{TR}^{n\times n}$, which implies that (2.2) holds.

If $\widetilde{X} + \bar{X}$ is a least squares tridiagonal solution of (1.1), then we have

$$\|A\widetilde{X}B - C\|_F^2 = \|A(\widetilde{X} + \bar{X})B - C\|_F^2 = \|A\bar{X}B - \widetilde{R}\|_F^2 = \|A\bar{X}B\|_F^2 + \|\widetilde{R}\|_F^2$$

due to the fact that $A\bar{X}B \in \mathcal{L}$, and equality means that the tridiagonal matrix $\bar{X}$ satisfies the linear homogeneous equation $A\bar{X}B = 0$. Conversely, if the matrix $\ddot{X}$ is of the form

$$\ddot{X} = \widetilde{X} + \bar{X},$$

where the matrix $\bar{X} \in \mathbb{TR}^{n\times n}$ satisfies $A\bar{X}B = 0$, then we have

$$\|A\ddot{X}B - C\|_F = \|A\bar{X}B - \widetilde{R}\|_F = \|A\widetilde{X}B - C\|_F.$$

Hence, the matrix $\ddot{X}$ is a solution of Problem A. □

THEOREM 2.2. *Let the matrix* $\widehat{X}$ *satisfy the minimization problem*

$$\|A\widehat{X}B - C\|_F = \min_{X\in\mathcal{K}(\psi,V)} \|AXB - C\|_F. \tag{2.3}$$

Then $\widehat{X}$ *is the unique minimum norm solution of Problem A, where* $\mathcal{K}(\psi, V)$ *is the matrix Krylov subspace defined in (2.1).*

Proof. Because $f(X) = \|AXB - C\|_F$ is a convex, continuous and differentiable function in finite-dimension subspace $\mathcal{K}(\psi, V)$, there exists unique solution to the the minimization problem (2.3). Let the sequence of tridiagonal matrices $\{V_i\}_{i=1}^s$ be a set of linear independent basis of the matrix Krylov subspace $\mathcal{K}(\psi, V)$ with dimension s.

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

If the matrix $\widehat{X} \in \mathcal{K}(\psi, V)$ satisfies (2.3), then

$$\|A\widehat{X}B - C\|_F \leq \|A(\widehat{X} + tV_i)B - C\|_F \tag{2.4}$$

holds for all real numbers t, as well as the basis matrices V_i $(i = 1, 2, \ldots, s)$, and we have

$$\begin{aligned} \|A(\widehat{X} + tV_i)B - C\|_F^2 &= \langle A(\widehat{X} + tV_i)B - C, A(\widehat{X} + tV_i)B - C\rangle \\ &= \|A\widehat{X}B - C\|_F^2 + 2t\langle AV_iB, A\widehat{X}B - C\rangle \\ &\quad + t^2\langle AV_iB, AV_iB\rangle. \end{aligned}$$

Combining this equality with (2.4), we have

$$(2\langle AV_iB, A\widehat{X}B - C\rangle + t\langle AV_iB, AV_iB\rangle)t \geq 0$$

for all real numbers t and the basis matrices V_i $(i = 1, 2, \ldots, s)$. By taking $t \to 0^+$ and $t \to 0^-$, respectively, we have

$$\langle AV_iB, A\widehat{X}B - C\rangle \geq 0 \quad \text{and} \quad \langle AV_iB, A\widehat{X}B - C\rangle \leq 0.$$

These two inequalities indicate that

$$\langle AV_iB, A\widehat{X}B - C\rangle = 0$$

holds for all basis matrices V_i $(i = 1, 2, \ldots, s)$. As a consequence, we have

$$\langle V_i, \tau(A^TA\widehat{X}BB^T) - \tau(A^TCB^T)\rangle = 0, \quad \text{i.e.,} \quad \psi(\widehat{X}) = V$$

due to the fact that

$$\widehat{X} \in \mathcal{K}(\psi, V) \quad \text{and} \quad \psi(\widehat{X}) - V \in \mathcal{K}(\psi, V).$$

Hence, the matrix $\widehat{X}$ is a solution of Problem A.

The least squares tridiagonal matrix $\widehat{X} \in \mathcal{K}(\psi, V)$ implies that there exists a matrix $\widehat{H} \in \mathbb{R}^{m\times p}$ such that $\widehat{X} = \tau(A^T\widehat{H}B^T)$. From Lemma 1, we know that any least squares tridiagonal solution of (1.1) can be expressed as $\widehat{X} + \bar{X}$, where the tridiagonal matrix $\bar{X}$ satisfies $A\bar{X}B = 0$, then we have

$$\langle \widehat{X}, \bar{X}\rangle = \langle \tau(A^T\widehat{H}B^T), \bar{X}\rangle = \langle A^T\widehat{H}B^T, \bar{X}\rangle = \langle \widehat{H}, A\bar{X}B\rangle = 0.$$

It follows that

$$\|\widehat{X} + \bar{X}\|_F^2 = \|\widehat{X}\|_F^2 + \|\bar{X}\|_F^2,$$

which implies that the least squares tridiagonal solution $\widehat{X}$ is the unique minimum norm solution. $\square$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

Theorem 2.2 indicates that the minimum norm solution, $\widehat{X}$, of Problem A belongs to the matrix Krylov subspace $\in \mathcal{K}(\psi, V)$. Hence, the solution $\widehat{X}_k \in \mathcal{K}_k(\psi, V)$ of the following minimization problem

$$\|A\widehat{X}_kB - C\|_F = \min_{X\in\mathcal{K}_k(\psi,V)} \|AXB - C\|_F \tag{2.5}$$

is an approximate solution to the minimum norm solution $\widehat{X}$ of Problem A, and $\widehat{X}_k \to \widehat{X}$ with k enough large. If the sequence of tridiagonal matrices $\{V_1, V_2, \ldots, V_k\}$ is a set of orthonormal basis of the k-dimension matrix Krylov subspace $\mathcal{K}_k(\psi, V)$, then the matrix $\widehat{X}$ can be expressed as

$$\widehat{X} = \sum_{i=1}^{s} y_i V_i.$$

Hence, the tridiagonal solution $\widehat{X}$ can be obtained once $\{V_i\}_{i=1}^k$ and $\{y_i\}_{i=1}^k$ have been computed.

In the following section, we will first construct a set of orthonormal basis of the k-dimension Krylov subspace $\mathcal{K}_k(\psi, V)$ by applying the matrix-form bidiagonalization procedure, and then derive a matrix iteration method for solving Problem A based on the classical LSQR algorithm.

3. The iteration method for solving Problem A. The matrix-form bidiagonalization algorithm for constructing a set of orthonormal basis of the matrix Krylov subspace $\mathcal{K}_k(\psi, V)$ is described as follows:

Algorithm 3.1. Matrix-form bidiagonalization algorithm.

1. Given matrices A, B and C;

2. Compute $U_1 = C/\|C\|_F$, $W_1 = \varphi^*(U_1) = \tau(A^TU_1B^T)$, $\alpha_1 = \|W_1\|_F$ and $V_1 = W_1/\alpha_1$;

3. For $i = 1, 2, \ldots$, compute

   $S_{i+1} = \varphi(V_i) - \alpha_i U_i = AV_iB - \alpha_i U_i$,

   $\beta_{i+1} = \|S_{i+1}\|_F$, if $\beta_{i+1} = 0$, then stop,

   $U_{i+1} = S_{i+1}/\beta_{i+1}$,

   $W_{i+1} = \varphi^*(U_{i+1}) - \beta_{i+1}V_i = \tau(A^TU_{i+1}B^T) - \beta_{i+1}V_i$,

   $\alpha_{i+1} = \|W_{i+1}\|_F$, if $\alpha_{i+1} = 0$, then stop,

   $V_{i+1} = W_{i+1}/\alpha_{i+1}$.

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

From Algorithm 3.1, we know that $V_i \in \mathbb{TR}^{n\times n}$ and $\|U_i\|_F = \|V_i\|_F = 1$, for $i = 1, 2, \ldots$ Moreover, we can further show the following proposition by induction.

THEOREM 3.1. *If the matrix-form bidiagonalization algorithm does not stop before the kth step, then we have*

$$\langle U_i, U_j\rangle = 0 \quad \textit{and} \quad \langle V_i, V_j\rangle = 0, \quad i \neq j,\ 1 \leq i, j \leq k.$$

Proof. For $i = 1,\ j = 2$, it follows that

$$\begin{aligned}
\langle U_1, U_2\rangle &= \langle U_1, \tfrac{AV_1B-\alpha_1U_1}{\beta_2}\rangle \\
&= \tfrac{1}{\beta_2}\langle A^TU_1B^T, V_1\rangle - \tfrac{\alpha_1}{\beta_2}\langle U_1, U_1\rangle \\
&= \tfrac{1}{\alpha_1\beta_2}\langle \tau(A^TU_1B^T), W_1\rangle - \tfrac{\alpha_1}{\beta_2} \\
&= 0,
\end{aligned}$$

$$\begin{aligned}
\langle V_1, V_2\rangle &= \tfrac{1}{\alpha_2}\langle V_1, \tau(A^TU_2B^T) - \beta_2V_1\rangle \\
&= \tfrac{1}{\alpha_2}\langle V_1, A^T\tfrac{AV_1B-\alpha_1U_1}{\beta_2}B^T\rangle - \tfrac{\beta_2}{\alpha_2}\langle V_1, V_1\rangle \\
&= \tfrac{1}{\alpha_2\beta_2}[\langle \beta_2U_2 + \alpha_1U_1, \beta_2U_2 + \alpha_1U_1\rangle \\
&\quad -\alpha_1\langle V_1, \tau(A^TU_1B^T)\rangle] - \tfrac{\beta_2}{\alpha_2} \\
&= \tfrac{1}{\alpha_2\beta_2}[\beta_2^2 + \alpha_1^2 - \langle W_1, W_1\rangle] - \tfrac{\beta_2}{\alpha_2} \\
&= 0.
\end{aligned}$$

Assume that the conclusions $\langle U_i, U_j\rangle = 0$ and $\langle V_i, V_j\rangle = 0$ hold for all $1 \leq i \leq j-1 (1 < j < k)$. Then

$$\begin{aligned}
\langle U_i, U_{j+1}\rangle &= \tfrac{1}{\beta_{j+1}}\langle U_i, AV_jB - \alpha_jU_j\rangle \\
&= \tfrac{1}{\beta_{j+1}}\langle \tau(A^TU_iB^T), V_j\rangle \\
&= \tfrac{1}{\beta_{j+1}}\langle W_i + \beta_jV_{i-1}, V_j\rangle \\
&= \tfrac{1}{\beta_{j+1}}\langle W_i, V_j\rangle \\
&= 0,
\end{aligned}$$

$$\begin{aligned}
\langle U_j, U_{j+1}\rangle &= \tfrac{1}{\beta_{j+1}}\langle U_j, AV_jB - \alpha_jU_j\rangle \\
&= \tfrac{1}{\beta_{j+1}}\langle W_j + \beta_jV_{j-1}, V_j\rangle - \tfrac{\alpha_j}{\beta_{j+1}}\langle U_j, U_j\rangle \\
&= \tfrac{1}{\alpha_j\beta_{j+1}}\langle W_j, W_j\rangle - \tfrac{\alpha_j}{\beta_{j+1}} \\
&= 0,
\end{aligned}$$

$$\begin{aligned}
\langle V_i, V_{j+1}\rangle &= \tfrac{1}{\alpha_{j+1}}\langle V_i, \tau(A^TU_{j+1}B^T) - \beta_{j+1}V_j\rangle \\
&= \tfrac{1}{\alpha_{j+1}\beta_{j+1}}\langle AV_iB, AV_jB - \alpha_jU_j\rangle \\
&= \tfrac{1}{\alpha_{j+1}\beta_{j+1}}\langle \beta_{i+1}U_{i+1} + \alpha_iU_i, \beta_{j+1}U_{j+1} + \alpha_jU_j\rangle \\
&\quad -\tfrac{\alpha_j}{\alpha_{j+1}\beta_{j+1}}\langle \beta_{i+1}U_{i+1} + \alpha_iU_i, U_j\rangle.
\end{aligned}$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

If $i+1<j$, then

$$\langle V_i, V_{j+1}\rangle = 0.$$

If $i+1=j$, then

$$\langle V_i, V_{j+1}\rangle = \frac{1}{\alpha_{j+1}\beta_{j+1}}\langle \beta_j U_j, \alpha_j U_j\rangle - \frac{\alpha_j}{\alpha_{j+1}\beta_{j+1}}\langle \beta_j U_j, U_j\rangle = 0.$$

If $i=j$, then

$$\begin{array}{rcl} \langle V_j, V_{j+1}\rangle & = & \frac{1}{\alpha_{j+1}}\langle V_j, \tau(A^T U_{j+1} B^T)\rangle - \frac{\beta_{j+1}}{\alpha_{j+1}}\langle V_j, V_j\rangle \\ & = & \frac{1}{\alpha_{j+1}\beta_{j+1}}\langle AV_jB, AV_jB - \alpha_j U_j\rangle - \frac{\beta_{j+1}}{\alpha_{j+1}} \\ & = & \frac{1}{\alpha_{j+1}\beta_{j+1}}\langle \beta_{j+1}U_{j+1} + \alpha_j U_j, \beta_{j+1}U_{j+1} + \alpha_j U_j\rangle \\ & & -\frac{\alpha_j}{\alpha_{j+1}\beta_{j+1}}\langle AV_jB, U_j\rangle - \frac{\beta_{j+1}}{\alpha_{j+1}} \\ & = & \frac{\alpha_j^2+\beta_{j+1}^2}{\alpha_{j+1}\beta_{j+1}} - \frac{\alpha_j}{\alpha_{j+1}\beta_{j+1}}\langle V_j, W_j + \beta_j V_{j-1}\rangle - \frac{\beta_{j+1}}{\alpha_{j+1}} \\ & = & 0. \end{array}$$

By the principle of induction and the fact that $\langle A, B\rangle = \langle B, A\rangle$ holds for all matrices A and B in $\mathbb{R}^{m\times n}$, we know that $\langle U_i, U_j\rangle = 0$ and $\langle V_i, V_j\rangle = 0$ hold for all $1 \le i,\ j \le k,\ i \ne j$. $\square$

If the tridiagonal matrix V_i $(i \le k)$ can be computed without breakdown, it then follows directly from the matrix-form bidiagonalization procedure that the tridiagonal matrices $V_1, V_2, \ldots, V_k$ form a set of orthonormal basis of the k-dimension matrix Krylov subspace $\mathcal{K}_k(\psi, V)$. Moreover, Theorem 3.1 shows that the sequence of tridiagonal matrices $V_1, V_2, \ldots$ generated by Algorithm 3.1 in exact arithmetic are orthonormal to each other in the finite dimension matrix space $\mathbb{TR}^{n\times n}$. Hence, the iteration must be terminated at most $3n-2$ steps in the absence of roundoff errors.

Similar to the classic Krylov subspace method, the recurrence relations of the matrix-from bidiagonalization procedure can be rewritten in another matrix form, which will be useful for deriving the approximate solution $\widehat{X}_k$. Firstly, let us introduce some notation as in [9, 6]. $\mathbb{V}_k$ and $\mathbb{U}_k$ denote the $n \times kn$ and $m \times kp$ block matrices respectively:

$$\mathbb{V}_k = [V_1, V_2, \ldots, V_k] \quad \text{and} \quad \mathbb{U}_k = [U_1, U_2, \ldots, U_k].$$

$\widetilde{L_k}$ denotes the $(k+1)\times k$ lower bidiagonal matrix whose nonzeros entries are defined by Algorithm 3.1, and L_k is the $k \times k$ matrix obtained from $\widetilde{L_k}$ by deleting its last row:

$$\widetilde{L_k} = \begin{pmatrix} L_k \\ \beta_{k+1} e_k^{(k)T} \end{pmatrix} \quad \text{and} \quad L_k = \begin{pmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & \beta_k & \alpha_k \end{pmatrix},$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

where $e_k^{(k)}$ is the last column of the identity matrix I_k.

For the matrix $H = (h_{ij}) \in \mathbb{R}^{k\times s}$, the notation $*$ denotes the following product:

$$
\begin{aligned}
\mathbb{V}_k * H &= [\mathbb{V}_k * H_{\cdot,1}, \mathbb{V}_k * H_{\cdot,2}, \ldots, \mathbb{V}_k * H_{\cdot,s}] \\
&= [\textstyle\sum_{i=1}^k h_{i1}V_i, \sum_{i=1}^k h_{i2}V_i, \ldots, \sum_{i=1}^k h_{is}V_i].
\end{aligned} \tag{3.1}
$$

It is easy to see that the following relations hold for the matrices H_1 and H_2 with compatible dimensions:

$$\mathbb{V}_k * (H_1 + H_2) = \mathbb{V}_k * H_1 + \mathbb{V}_k * H_2 \quad \text{and} \quad \mathbb{V}_k * (H_1 H_2) = (\mathbb{V}_k * H_1) * H_2.$$

If the matrix H_2 is invertible, then from

$$\mathbb{V}_k * (H_1 H_2) = \mathbb{U}_k$$

it follows that

$$\mathbb{V}_k * H_1 = \mathbb{U}_k * H_2^{-1}.$$

For a real vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_k)^T \in \mathbb{R}^k$, the product (3.1) reduces to

$$\mathbb{V}_k * \alpha = \sum_{i=1}^k \alpha_i V_i,$$

and since the matrices $V_i, i = 1, 2, \ldots, k$, generated in Algorithm 3.1, are orthonormal with respect to the matrix inner product $\langle\cdot,\cdot\rangle$, the following result

$$\|\mathbb{V}_k * \alpha\|_F = \|\alpha\|_2 \tag{3.2}$$

holds for all $\alpha \in \mathbb{R}^k$, where $\|\cdot\|_2$ is vector 2-norm. See [9] for details.

If the matrix-form bidiagonalization algorithm does not stop before the k-th step, then the tridiagonal matrices $V_1, V_2, \ldots, V_k$ are the othonormal basis of the matrix Krylov subspace $\mathcal{K}_k(\psi, V)$ and the recurrence relations of Algorithm 3.1 can be rewritten as

$$[\varphi(V_1), \varphi(V_2), \ldots, \varphi(V_k)] = \mathbb{U}_{k+1} * \widetilde{L_k}, \tag{3.3}$$

$$[\varphi^*(U_1), \varphi^*(U_2), \ldots, \varphi^*(U_k)] = \mathbb{V}_k * {L_k}^T. \tag{3.4}$$

The approximation solution $\widehat{X}_k$ can be expressed as $\widehat{X}_k = \mathbb{V}_k * y_k$, and the corresponding residual

$$
\begin{aligned}
R_k &= C - A\widehat{X}_k B \\
&= C - \varphi(\widehat{X}_k) \\
&= C - [\varphi(V_1), \varphi(V_2), \ldots, \varphi(V_k)] * y_k \\
&= C - \mathbb{U}_{k+1} * (\widetilde{L_k} y_k) \\
&= \mathbb{U}_{k+1} * (\|C\|_F e_k^{(1)} - \widetilde{L_k} y_k),
\end{aligned} \tag{3.5}
$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

where $e_k^{(1)}$ is the first column of the identity matrix I_k. Using the properties (3.2) of the product $*$, the Frobenius norm of the residual R_k can be expressed as

$$\|R_k\|_F = \| \|C\|_F e_k^{(1)} - \widetilde{L_k} y_k \|_2.$$

Hence, the minimization problem (2.5) is equivalent to

$$\| \|C\|_F e_k^{(1)} - \widetilde{L_k} y_k \|_2 = \min_{y \in \mathbb{R}^k} \| \|C\|_F e_k^{(1)} - \widetilde{L_k} y \|_2, \tag{3.6}$$

which indicates the least squares problem of the matrix equation (1.1) with tridagonal matrix constraint in the sense of Frobenuis norm can be equivalently transformed to an unconstrained least squares problem with the vector 2-norm $\|\cdot\|_2$. Because the lower bidiagonal matrix $\widetilde{L_k}$ is full column rank, this unconstrained least squares problem can be solved according to the classical LSQR algorithm.

Using the QR decomposition of the lower bidiagonal matrix $\widetilde{L_k}$ and the vector $\|C\|_F e_k^{(1)}$ simultaneously, we have

$$\mathcal{Q}_k \widetilde{L_k} = \begin{pmatrix} \Omega_k \\ 0 \end{pmatrix} = \begin{pmatrix} \rho_1 & \theta_2 & & & \\ & \rho_2 & \theta_3 & & \\ & & \ddots & \ddots & \\ & & & \rho_{k-1} & \theta_k \\ & & & & \rho_k \\ & & & & 0 \end{pmatrix},$$

$$\mathcal{Q}_k (\|C\|_F e_k^{(1)}) = \begin{pmatrix} \mathcal{Z}_k \\ \tilde{\zeta}_{k+1} \end{pmatrix} = \begin{pmatrix} \zeta_1 \\ \zeta_2 \\ \vdots \\ \zeta_{k-1} \\ \zeta_k \\ \tilde{\zeta}_{k+1} \end{pmatrix}.$$

The $(k+1) \times (k+1)$ orthogonal matrix $\mathcal{Q}_k$ is a product of Givens rotations $\mathcal{Q}_k = \mathcal{Q}_{k,k+1} \mathcal{Q}_{k-1,k} \cdots \mathcal{Q}_{12}$, which is chosen to eliminate the subdiagonal elements $\beta_2, \ldots, \beta_{k+1}$ of $\widetilde{L_k}$. If set $\tilde{\rho}_1 = \alpha_1$, $\tilde{\zeta}_1 = \beta_1 = \|C\|_F$, then for $i = 1, 2, \ldots, k$, we can construct the Givens rotation $\mathcal{Q}_{i,i+1}$ such that

$$\begin{aligned} \mathcal{Q}_{i,i+1} \begin{pmatrix} \tilde{\rho}_i & 0 & \tilde{\zeta}_i \\ \beta_{i+1} & \alpha_{i+1} & 0 \end{pmatrix} &= \begin{pmatrix} c_i & s_i \\ -s_i & c_i \end{pmatrix} \begin{pmatrix} \tilde{\rho}_i & 0 & \tilde{\zeta}_i \\ \beta_{i+1} & \alpha_{i+1} & 0 \end{pmatrix} \\ &= \begin{pmatrix} \rho_i & \theta_{i+1} & \zeta_i \\ 0 & \tilde{\alpha}_{i+1} & \tilde{\zeta}_{i+1} \end{pmatrix}, \end{aligned}$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

where

$$c_i = \frac{\tilde{\rho}_i}{\sqrt{\tilde{\rho}_i^2 + \beta_{i+1}^2}} \quad \text{and} \quad s_i = \frac{\beta_{i+1}}{\sqrt{\tilde{\rho}_i^2 + \beta_{i+1}^2}}.$$

Substituting the QR decomposition of the lower bidiagonal matrix into the unconstrained least squares problem (3.6) yields $y_k = \Omega_k^{-1}\mathcal{Z}_k$. Consequently, the approximation solution X_k can be formed as

$$X_k = \mathbb{V}_k * y_k = (\mathbb{V}_k * \Omega_k^{-1}) * \mathcal{Z}_k,$$

and the corresponding residual (3.5) can be rewritten as

$$(3.7)\quad R_k = \mathbb{U}_{k+1} * \left[\mathcal{Q}_k^T \begin{pmatrix} \mathcal{Z}_k \\ \tilde{\zeta}_{k+1} \end{pmatrix} - \mathcal{Q}_k^T \begin{pmatrix} \Omega_k \\ 0 \end{pmatrix} y_k \right] = \mathbb{U}_{k+1} * (\mathcal{Q}_k^T \tilde{\zeta}_{k+1} e_{k+1}^{(k+1)}).$$

Let

$$\mathbb{H}_k = [H_1, H_2, \ldots, H_k] = \mathbb{V}_k * \Omega_k^{-1} = [V_1, V_2, \ldots, V_k] * \Omega_k^{-1}.$$

By using the properties of the product $*$ as well as the structure of upper bidiagonal matrix Ω_k, we have

$$(3.8)\quad H_k = \frac{1}{\rho_k}(V_k - \theta_k H_{k-1}) \quad \text{and} \quad X_k = \mathbb{H}_k * \begin{pmatrix} \mathcal{Z}_{k-1} \\ \zeta_k \end{pmatrix} = X_{k-1} + \zeta_k H_k.$$

The recursions indicate that the matrix H_k and the approximation solution X_k can be obtained from H_{k-1} and X_{k-1} respectively, and there only one extra product of a scalar and a matrix need to be computed. Denote $G_k = \rho_k H_k$, for $k = 1, 2, \ldots$, the recursions (3.8) can be rewritten as

$$X_k = X_{k-1} + \frac{\zeta_k}{\rho_k} G_k \quad \text{and} \quad G_k = V_k - \frac{\theta_k}{\rho_{k-1}} G_{k-1}.$$

where $G_1 = V_1$.

Obviously, we can regard $\|\psi(\widehat{X}_k) - V\|_F < \epsilon$ as the stopping criteria, where $\epsilon > 0$ is a small tolerance. However, the stopping criteria need not to be computed explicitly. Notice that the residual R_k is of the form (3.7), then by (3.4) we have

$$\begin{aligned}
\|\psi(\widehat{X}_k) - V\|_F &= \|\tau(A^T R_k B^T)\|_F \\
&= \|\tau(A^T [\mathbb{U}_{k+1} * (\mathcal{Q}_k^T \tilde{\zeta}_{k+1} e_{k+1}^{(k+1)})] B^T)\|_F \\
&= \|[\tau(A^T U_1 B^T), \ldots, \tau(A^T U_k B^T), \tau(A^T U_{k+1} B^T)] * (\mathcal{Q}_k^T \tilde{\zeta}_{k+1} e_{k+1}^{(k+1)})\|_F \\
&= \|[V_1, \ldots, V_k, V_{k+1}] * (L_{k+1}^T \mathcal{Q}_k^T \tilde{\zeta}_{k+1} e_{k+1}^{(k+1)})\|_F \\
&= |\tilde{\zeta}_{k+1}| \|(\mathcal{Q}_k L_{k+1})^T e_{k+1}^{(k+1)}\|_2 \\
&= |c_k \alpha_{k+1} \tilde{\zeta}_{k+1}|.
\end{aligned}$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

Hence, the stopping criterion is easily determined by the product of three scalars, instead of computing the residual directly. Summarizing the formulas developed above, we have the following matrix iteration method.

Algorithm 3.2. Matrix-form LSQR algorithm.

1. Given matrices A, B, C and the tolerance ϵ;

2. Compute $\beta_1 = \|C\|_F$, $U_1 = C/\beta_1$, $V_1 = \varphi^*(U_1)$, $\alpha_1 = \|V_1\|_F$ and $V_1 := V_1/\alpha_1$;

3. Set $X_0 = 0$, $\tilde{\rho}_1 = \alpha_1$, $\tilde{\zeta}_1 = \beta_1$, $G_1 = V_1$;

4. For $i = 1, 2, \ldots$,

   4.1 Compute U_{i+1}: $U_{i+1} = \varphi(V_i) - \alpha_i U_i$, $\beta_{i+1} = \|U_{i+1}\|_F$, $U_{i+1} := U_{i+1}/\beta_{i+1}$;

   4.2 Compute $\mathcal{Q}_{i,i+1}$: $\rho_i = \sqrt{\tilde{\rho}_i^2 + \beta_{i+1}^2}$, $c_i = \tilde{\rho}_i/\rho_i$, $s_i = \beta_{i+1}/\rho_i$;

   4.3 Compute V_{i+1}: $V_{i+1} = \varphi^*(U_{i+1}) - \beta_{i+1} V_i$, $\alpha_{i+1} = \|V_{i+1}\|_F$, $V_{i+1} := V_{i+1}/\alpha_{i+1}$;

   4.4 Compute ζ_i, $\tilde{\zeta}_{i+1}$: $\theta_{i+1} = s_i \alpha_{i+1}$, $\tilde{\rho}_{i+1} = c_i \alpha_{i+1}$, $\zeta_i = c_i \tilde{\zeta}_i$, $\tilde{\zeta}_{i+1} = -s_i \tilde{\zeta}_i$;

   4.5 Compute X_i and G_{i+1}: $X_i = X_{i-1} + \frac{\zeta_i}{\rho_i} G_i$, $G_{i+1} = V_{i+1} - \frac{\theta_{i+1}}{\rho_i} G_i$;

   4.6 Check convergence: if $|c_i \alpha_{i+1} \tilde{\zeta}_{i+1}| < \epsilon$, then stop.

Theoretically, the minimum norm solution of Problem A can be obtained within at most $3n - 2$ steps by the matrix-form LSQR algorithm in exact arithmetic. If the coefficient matrices A, B and C are n-th order square matrices, the matrix-form LSQR algorithm requires about $4n^3 + 8n^2$ multiplications in each iteration.

4. Preconditioned matrix-form LSQR algorithm. Due to the slow convergence for large condition numbers of the coefficient matrices (see Example 5.2 in Section 5), it is essential to use preconditioning in association with the matrix iteration method. The general precondition method for the least squares problem (1.2) is to construct two nonsingular matrices S_1 and S_2 with special sparse structures, and

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

the least squares problem (1.2) is equivalent to the following problem

$$\text{minimize } \|AS_1^{-1}YS_2^{-1}B - C\|_F \tag{4.1}$$

subject to

$$S_1XS_2 = Y, \quad X \in \mathbb{TR}^{n\times n}. \tag{4.2}$$

But until now, it seems that no one has studied the preconditioning method for the least squares problem of constrained matrix equation (1.1), and the difficulty lies in the fact that the nonsingular preconditioning matrices S_1 and S_2 are not easy to choose so that the constrained matrix equation (4.2) is consistent in given constrained matrix set for any $Y \in \mathbb{R}^{n\times n}$. To this end, we have attempted to apply a feasible preconditioned method to improve convergence of the matrix-form LSQR method in this section.

4.1. Preconditioned matrix-form LSQR algorithm. Denote $e_k^{(i)}$ the i-th column of the identity matrix I_k, then the matrices

$$F_{ij} = e_m^{(i)}e_p^{(j)T}, \quad \text{for} \quad i = 1, 2, \ldots, m, \ j = 1, 2, \ldots, p$$

are a set of orthonormal basis of the linear space $\mathbb{R}^{m\times p}$, and

$$E_{ij} = e_n^{(i)}e_n^{(j)T}, \quad \text{for} \quad i, j = 1, 2, \ldots, n, \ j-1 \le i \le j+1$$

are a set of orthonormal basis of the subspace $\mathbb{TR}^{n\times n}$, respectively. For simplicity, we denote

$$\mathbb{F} = [F_{11}, \ F_{12}, \ \ldots, \ F_{1p}, \ F_{21}, \ \ldots, \ F_{mp}] \tag{4.3}$$

and

$$\mathbb{E} = [E_{11}, \ E_{12}, \ E_{21}, \ \ldots, \ E_{23}, \ \ldots, \ E_{nn}]. \tag{4.4}$$

For the linear operators φ and φ^* defined in section 2, we have

$$\varphi\ \mathbb{E} = \mathbb{F} * H \quad \text{and} \quad \varphi^*\ \mathbb{F} = \mathbb{E} * H^T,$$

where the matrix $H \in \mathbb{R}^{mp\times(3n-2)}$ can be expressed as

$$H = (a_1 \otimes b_1^T, a_1 \otimes b_2^T, a_2 \otimes b_1^T, \ldots, a_2 \otimes b_3^T, \ldots, a_n \otimes b_{n-1}^T, a_n \otimes b_n^T), \tag{4.5}$$

and a_i denotes the i-th column of the matrix A and b_j the j-th row of the matrix B. Obviously, the singular value of the linear operator φ is determined by the matrix H.

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

From (4.5), we notice that the matrix H can be reformulated as a more simplified form

$$H = (A \otimes B^T)P \begin{pmatrix} I_{3n-2} \\ 0 \end{pmatrix} \in \mathbb{R}^{mp\times(3n-2)}, \tag{4.6}$$

where $P \in \mathbb{R}^{mp\times mp}$ is a certain permutation matrix.

We denote $\widetilde{H} = HS^{-1}$, where the matrix $S \in \mathbb{R}^{(3n-2)\times(3n-2)}$ is nonsingular, and let

$$\widetilde{\varphi}\ \mathbb{E} = \mathbb{F} * \widetilde{H}.$$

For the basis matrices (4.3) and (4.4), the linear operator $\widetilde{\varphi}$ is uniquely determined by the matrix $\widetilde{H}$ due to the fact that the linear operator space $\mathcal{L}(\mathbb{TR}^{n\times n}, \mathbb{R}^{m\times p})$ is isomorphic to the matrix space $\mathbb{R}^{mp\times(3n-2)}$. Furthermore, the linear operator $\widetilde{\varphi}^*$, determined by the matrix $\widetilde{H}^T = S^{-T}H^T$:

$$\widetilde{\varphi}^*\ \mathbb{F} = \mathbb{E} * \widetilde{H}^T,$$

is the adjoint operator of $\widetilde{\varphi}$.

Let us denote by $\mathcal{F}_X$ the coordinate vector of the matrix $X \in \mathbb{R}^{m\times p}$ under the set of orthonormal basis (4.3) and $\mathcal{E}_Y$ the coordinate vector of the tridiagonal matrix $Y \in \mathbb{TR}^{n\times n}$ under the set of orthonormal basis (4.4). Suppose the vector $x_i \in \mathbb{R}^n$ is the i-th row of the matrix X of order $m \times n$, and denote the vector consisting of from the α-th component to β-th component of x_i by $x_{i,\alpha:\beta}$, then the coordinate vectors $\mathcal{F}_X$ and $\mathcal{E}_Y$ are of the following forms

$$\mathcal{F}_X = \begin{pmatrix} x_{1,1:p}^T \\ x_{2,1:p}^T \\ x_{3,1:p}^T \\ \vdots \\ x_{m,1:p}^T \end{pmatrix} \in \mathbb{R}^{mp} \quad \text{and} \quad \mathcal{E}_Y = \begin{pmatrix} y_{1,1:2}^T \\ y_{2,1:3}^T \\ y_{3,2:4}^T \\ \vdots \\ y_{n,n-1:n}^T \end{pmatrix} \in \mathbb{R}^{3n-2}.$$

Consequently, for all $X \in \mathbb{TR}^{n\times n}$, we have

$$\widetilde{\varphi}(X) = \widetilde{\varphi}\ (\mathbb{E} * \mathcal{E}_X) = \mathbb{F} * (\widetilde{H}\mathcal{E}_X),$$

and there exists an unique tridiagonal matrix Y such that

$$\mathcal{E}_Y = S^{-1}\mathcal{E}_X, \tag{4.7}$$

which means that

$$\widetilde{\varphi}(X) = \mathbb{F} * (H\mathcal{E}_Y) = \varphi(\mathbb{E} * \mathcal{E}_Y) = \varphi(Y).$$

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

For all $X \in \mathbb{R}^{m\times p}$, we have

$$\widetilde{\varphi}^*(X) = \widetilde{\varphi}^*(\mathbb{F} * \mathcal{F}_X) = \mathbb{E} * (\widetilde{H}^T \mathcal{F}_X)$$

and

$$\varphi^*(X) = \varphi^*(\mathbb{F} * \mathcal{F}_X) = \mathbb{E} * (H^T \mathcal{F}_X).$$

Obviously, there exists an unique tridiagonal matrix $Z \in \mathbb{TR}^{n\times n}$ for an arbitrary given matrix $X \in \mathbb{R}^{m\times p}$, which coordinate vector $\mathcal{E}_Z$ satisfies $\mathcal{E}_Z = \widetilde{H}^T \mathcal{F}_X = S^{-T} H^T \mathcal{F}_X$. It then follows that

$$\varphi^*(X) = \mathbb{E} * (S^T \mathcal{E}_Z).$$

Since $\mathcal{E}_{\varphi^*(X)}$ is the coordinate vector of the $n \times n$ tridiagonal matrix $\varphi^*(X)$, we can further obtain that

$$\varphi^*(X) = \mathbb{E} * \mathcal{E}_{\varphi^*(X)} = \mathbb{E} * (S^T \mathcal{E}_Z),$$

which means that the coordinate vector $\mathcal{E}_Z$ can be computed by

$$\mathcal{E}_Z = S^{-T} \mathcal{E}_{\varphi^*(X)} \tag{4.8}$$

and

$$\widetilde{\varphi}^*(X) = Z = \mathbb{E} * \mathcal{E}_Z.$$

From the discussions above, we know that the linear operators $\widetilde{\varphi}$ and $\widetilde{\varphi}^*$ do not need to be constructed explicitly, and $\widetilde{\varphi}(X)$ and $\widetilde{\varphi}^*(X)$ can be computed by solving the linear systems (4.7) and (4.8), respectively. Hence, we can choose the appropriate nonsingular matrix S such that the linear operator $\widetilde{\varphi}$ has a more favorable spectrum than the original linear operator φ. Replace the linear operator φ by $\widetilde{\varphi}$ in Algorithm 3.2, and we can derive the preconditioned version of the matrix-form LSQR.

Algorithm 4.1. Preconditioned matrix-form LSQR algorithm.

1. Given matrices A, B, C and the tolerance ϵ;
2. Compute $\beta_1 = \|C\|_F$, $\widetilde{U}_1 = C/\beta_1$, denote $\widetilde{V}_1 = \widetilde{\varphi}^*(\widetilde{U}_1)$;
   2.1 compute $\widetilde{V}_1$: $V_1 = \varphi^*(\widetilde{U}_1)$, $\mathcal{E}_{\widetilde{V}_1} = S^{-T}\mathcal{E}_{V_1}$, $\widetilde{V}_1 = \mathbb{E} * \mathcal{E}_{\widetilde{V}_1}$;
   2.2 normalization: $\alpha_1 = \|\widetilde{V}_1\|_F$ and $\widetilde{V}_1 := \widetilde{V}_1/\alpha_1$;
3. Set $X_0 = 0$, $\tilde{\rho}_1 = \alpha_1$, $\tilde{\zeta}_1 = \beta_1$, $G_1 = \widetilde{V}_1$;
4. For $i = 1, 2, \ldots$,

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

4.1 Compute $\widetilde{\varphi}(\widetilde{V}_i)$: $\mathcal{E}_{Y_i} = S^{-1}\mathcal{E}_{\widetilde{V}_i}$, $Y_i = \mathbb{E} * \mathcal{E}_{Y_i}$, $\widetilde{\varphi}(\widetilde{V}_i) = \varphi(Y_i) = AY_iB$;

4.2 Compute $\widetilde{U}_{i+1}$: $\widetilde{U}_{i+1} = \widetilde{\varphi}(\widetilde{V}_i) - \alpha_i\widetilde{U}_i = \varphi(Y_i) - \alpha_i\widetilde{U}_i = AY_iB - \alpha_i\widetilde{U}_i$;

4.3 Normalization: $\beta_{i+1} = \|\widetilde{U}_{i+1}\|_F$, $\widetilde{U}_{i+1} := \widetilde{U}_{i+1}/\beta_{i+1}$;

4.4 Compute $\mathcal{Q}_{i,i+1}$: $\rho_i = \sqrt{\tilde{\rho}_i^2 + \beta_{i+1}^2}$, $c_i = \tilde{\rho}_i/\rho_i$, $s_i = \beta_{i+1}/\rho_i$;

4.5 Compute $\widetilde{\varphi}^*(\widetilde{U}_{i+1})$: $V_{i+1} = \varphi^*(\widetilde{U}_{i+1})$, $\mathcal{E}_{Z_{i+1}} = S^{-T}\mathcal{E}_{V_{i+1}}$, $\widetilde{\varphi}^*(\widetilde{U}_{i+1}) = Z_{i+1}$;

4.6 compute $\widetilde{V}_{i+1}$: $\widetilde{V}_{i+1} = \widetilde{\varphi}^*(\widetilde{U}_{i+1}) - \beta_{i+1}\widetilde{V}_i = Z_{i+1} - \beta_{i+1}\widetilde{V}_i$;

4.7 Normalization: $\alpha_{i+1} = \|\widetilde{V}_{i+1}\|_F$, $\widetilde{V}_{i+1} := \widetilde{V}_{i+1}/\alpha_{i+1}$;

4.8 Compute ζ_i, $\tilde{\zeta}_{i+1}$: $\theta_{i+1} = s_i\alpha_{i+1}$, $\tilde{\rho}_{i+1} = c_i\alpha_{i+1}$, $\zeta_i = c_i\tilde{\zeta}_i$, $\tilde{\zeta}_{i+1} = -s_i\tilde{\zeta}_i$;

4.9 Compute X_i and G_{i+1}: $X_i = X_{i-1} + \frac{\zeta_i}{\rho_i}G_i$, $G_{i+1} = \widetilde{V}_{i+1} - \frac{\theta_{i+1}}{\rho_i}G_i$;

4.10 Check convergence: if $|c_i\alpha_{i+1}\tilde{\zeta}_{i+1}| < \epsilon$, then stop.

The extra cost of the preconditioning method will be in solving linear systems of the forms $Sx = y$ and $S^Tp = s$. Hence, S has to be chosen so that such systems can be easily solved. In the following, we will apply the CIMGS preconditioner proposed in [1, 19] to accelerate the convergence of the matrix-form LSQR method.

4.2. CIMGS preconditioner. We note that if the matrix H defined in (4.6) is full column rank, then it is possible to achieve more faster convergence by using the incomplete QR decomposition of H as a preconditioner. Hence, if we assume the matrix A is of full column rank and the matrix B is of full row rank, then the matrix H is a full column rank matrix.

It is well known that the incomplete QR decomposition of the matrix H can be determined by the incomplete modified Gram-Schmidt (IMGS) algorithm proposed by Jennings and Ajiz [10], and this algorithm never breaks down for the matrix with full column rank. However, it needs to computing directly and storing the column vectors of the matrix $Q \in \mathbb{R}^{mp\times(3n-2)}$ at each stage in the IMGS algorithm. To avoid this, an alternative one given by Wang [19] is a more compressed algorithm (CIMGS) for computing the IMGS preconditioner. In exact arithmetic, the CIMGS algorithm produces the same incomplete factor $R \in \mathbb{R}^{(3n-2)\times(3n-2)}$ of H as IMGS, and therefore, it inherits the robustness of IMGS.

Let the symmetric positive definite matrix $D = (d_{ij}) \in \mathbb{R}^{(3n-2)\times(3n-2)}$ be given by $D = H^TH$, P be a nonzero position set, and for simplicity, we denote the largest integer number not to exceed x by $[x]$, the i-th column of the matrix A by a_i, the j-th row of the matrix B by b_j and the i-th element of the vector a by $a(i)$, then the CIMGS algorithm for generating an upper triangular matrix $\mathbb{R}^{(3n-2)\times(3n-2)}$ can be described as follows:

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

Algorithm 4.2. CIMGS algorithm.

1. Given the matrices A, B and a nonzero position set P;
2. Compute the elements of R in the first row:

   $a = a_1^T A$, $b = b_1 B^T$, $r_{11} = \sqrt{a(1)}\sqrt{b(1)}$;

   for $t = 2 : (3n-2)$

   $i = \left[\frac{t}{3}\right] + 1$, $j = t - 2(i-1)$;

   if $(1,t) \notin P$, then $r_{1t} = 0$ else $r_{1t} = \frac{a(i)b(j)}{r_{11}}$;

   end
3. Update the elements $d_{kt},\ k,t = 2,3,\ldots,(3n-2)$:

   for $k = 2 : (3n-2)$

   $i = \left[\frac{k}{3}\right] + 1$, $j = k - 2(i-1)$;

   $a = a_i^T A$, $b = b_j B^T$;

   for $t = 2 : (3n-2)$

   $i = \left[\frac{t}{3}\right] + 1$, $j = t - 2(i-1)$;

   $d_{kt} = d_{kt} - r_{1k} r_{1t} = a(i)b(j) - r_{1k} r_{1t}$;

   end

   end
4. Compute the elements $r_{kt},\ k = 2,\ldots,(3n-2),\ t = k,\ldots,(3n-2)$:

   for $k = 2 : (3n-2)$

   $r_{kk} = \sqrt{d_{kk}}$;

   for $t = k+1 : (3n-2)$

   $d_{kt} = \frac{d_{kt}}{r_{kk}}$;

   if $(k,t) \notin P$ then $r_{kt} = 0$ else $r_{kt} = d_{kt}$; end

   end

   for $t = k+1 : (3n-2)$

   for $s = k+1 : (3n-2)$

   if $(k,t) \in P$ or $(k,s) \in P$ then $d_{st} = d_{st} - d_{ks} d_{kt}$ end

   end

   end

   end

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

In this algorithm, the elements of R are computed row by row, and it is no need to explicitly form the matrix H and the symmetric positive definite matrix $D = H^T H$. All that is required is to be able to access one row of the matrices $A^T A$ and BB^T at a time, respectively, and then the rows can be discarded after one row of the upper triangular matrix R has been computed. If we take the precondition matrix $S = R$, where R is a nonsingular upper triangular matrix generated by Algorihtm 4.2, then the corresponding linear systems of the forms $Sx = y$ and $S^T p = s$ can be solved by back substituting and forward substituting procedures respectively in each iteration.

5. Numerical examples. In this section, we give some numerical examples to illustrate the efficiency of the matrix-form LSQR algorithm and the corresponding precondition method. All the tests are performed using MATLAB 7.0 on a personal computer. Because of the influence of the error of calculation, the iteration will not stop within a finite number of steps. Hence, we regard the approximate solution $\widehat{X}_k$ as the minimum norm least squares tridiagonal solution of the matrix equation (1.1) if

$$\|\psi(\widehat{X}_k) - V\|_F = |c_k \alpha_{k+1} \tilde{\zeta}_{k+1}| < 1.0e - 08.$$

EXAMPLE 5.1. This small example is used to examine the theoretical results of this paper. Let matrices

$$A = \begin{pmatrix} \text{zeros}(4) & \text{zeros}(4) \\ \text{hankel}(1:4) & \text{ones}(4) \end{pmatrix}, \ B = \begin{pmatrix} \text{toeplitz}(1:4) & \text{ones}(4) \\ \text{zeros}(4) & \text{ones}(4) \end{pmatrix},$$

$$\Delta C = \begin{pmatrix} \text{pascal}(4) & \text{zeros}(4) \\ \text{zeros}(4) & \text{zeros}(4) \end{pmatrix},$$

where pascal(n) denotes the n-th order Pascal matrix, toeplitz(: n) and hankel(1: n) denote the n-th order Toeplitz matrix and Hankel matrix with first row $(1, 2, \ldots, n)$, respectively, and ones(n) and zeros(n) denote the $n \times n$ matrices whose all elements are one and zero, respectively.

We take $C = AXB + \Delta C$, where

$$X = \begin{pmatrix} 1 & -2 & & & \\ -1 & 2 & -2 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -2 \\ & & & -1 & 1 \end{pmatrix} \in \mathbb{R}^{8\times 8}.$$

Then we can theoretically show that X is a least squares tridiagonal solution of equation (1.1) by the fact that $\tau(A^T \Delta C B^T) = 0$. We obtain the approximate solution

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

as follows at the 20-th step by using Algorithm 3.2:

$$\widehat{X}_{20} = \begin{pmatrix} 1 & -2 & & & & & & \\ -1 & 2 & -2 & & & & & \\ & -1 & 2 & -2 & & & & \\ & & -1 & 2 & -2 & & & \\ & & & -1 & -0.2 & -0.2 & & \\ & & & & -0.2 & -0.2 & -0.2 & \\ & & & & & -0.2 & -0.2 & -0.2 \\ & & & & & & -0.2 & -0.2 \end{pmatrix}.$$

By concrete computations, we have

$$\|\varphi(\widehat{X}_{20}) - V\|_F = 1.2724e - 012 \quad \text{and}$$
$$\|A\widehat{X}_{20}B - C\|_F = \|AXB - C\|_F = \|\Delta C\|_F = 26.4008,$$

which indicate that both the matrices $\widehat{X}_{20}$ and X are the least squares tridiagonal solutions. We can further verify that

$$\|\widehat{X}_{20}\|_F = 5.7793 < \|X\|_F = 7.8102.$$

Hence, the computation results are in accordance with the theories established in this paper.

Example 5.2. In this example, we test the matrix-form LSQR algorithm when the coefficient matrices A and B have variant condition numbers. The test matrices are randomly constructed by using the singular value decomposition:

$$A = U_a D_a V_a^T \quad \text{and} \quad B = U_b D_b V_b^T,$$

where the orthogonal matrices U_a, V_a, U_b and V_b are constructed as follows (in MATLAB notation)

$$[U_a, V_a] = \text{svd}(\text{toeplitz}(1:n)), \quad [U_b, V_b] = \text{svd}(\text{hankel}(1:n)),$$

and the diagonal matrices D_a and D_b are formed as

$$D_a = \text{diag}([\text{rand}(n/2,1) + \text{ones}(n/2,1); 10^{(-a)} * \text{rand}(n/2,1)]),$$

$$D_b = \text{diag}([10^{(-b)} * \text{rand}(n/2,1); 2 * \text{rand}(n/2,1) - \text{ones}(n/2,1)]).$$

Hence, the magnitudes of the condition numbers of the matrices A and B can be respectively determined by the positive constants a and b. The righthand side matrix C is fixed by $C = \text{ones}(n) + 2 * \text{rand}(n)$.

We list our numerical results with dimensions from $n = 50$ to $n = 300$ in Table 1. In this table, we list the condition numbers of the matrices A and B respectively, the iteration numbers k, the CPU times for the different dimensions n with the error ERR=$\|\psi(\widehat{X}_k) - V\| < 1.0e - 08$.

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

Table 1
Convergence for variant matrix sizes and condition numbers

n	cond(A)	cond(B)	k	CPU(s)	ERR
50	67.2089	28.1342	38	0.016	9.2609e-9
100	4.71e+2	1.39e+2	63	0.23	7.7307e-9
200	3.16e+2	2.16e+2	72	1.812	7.1797e-9
300	5.54e+2	9.45e+2	66	5.422	7.9218e-9
50	3.98e+3	2.67e+3	712	0.47	9.2000e-9
100	7.33e+3	8.74e+3	1159	4.297	8.9088e-9
200	2.09e+4	1.04e+4	1594	41.89	9.9531e-9
300	8.38e+4	5.89e+3	1836	155.657	8.6112e-9
50	2.13e+6	7.21e+5	2233	1.375	9.7300e-9
100	5.04e+6	3.65e+7	12015	44.625	8.0757e-9
200	2.32e+7	1.37e+7	47242	1255.719	5.6999e-9
300	1.34e+7	1.25e+7	105620	8486.312	9.0697e-9

The results in Table 1 show that the CPU time grows quickly as n increases for the roughly same condition number, and the first group of data indicates that the iteration number seems not to depend very much on the matrix size n. However, we see from this table that the convergence speed of the matrix-form LSQR algorithm is affected by the condition numbers of the coefficient matrices A and B, and the iteration number k for the smaller condition number is much less than that for the larger one with the same precision requirement.

EXAMPLE 5.3. The last example is used to compare the preconditioning matrix-form LSQR algorithm with the unpreconditioning one presented in this paper for the coefficient matrices with large condition numbers in aspects of iteration numbers and the elapsed CPU times.

Table 2
Numerical results for the coefficient matrices with the same large condition numbers

Methods	n (Cond)	50 (3.52e+9)	100 (1.01e+9)	150 (2.92e+9)	200 (5.41e+8)
Unpreconditioning	k	2608	12365	32047	65868
	CPU(s)	1.41	41.5	359.6	1624.2
	ERR	9.52e-9	8.76e-9	8.67e-9	8.45e-9
CIMGS	k	1158	4861	12855	23489
	CPU(s)	1.34	8.48	63.2	287.0
	ERR	1.79e-9	3.74e-9	9.58e-9	6.84e-9

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

In this example, the matrices A, B and C with dimension n are randomly constructed as in Example 5.2, and the approximate solution $\widehat{X}$ is been computed with the precision ERR$=\|\psi(\widehat{X}) - V\|_F < 1.0e-08$. In our implementations, we give the nonzero position set $P = \{(i,j)|i,j = 1,2,\ldots,(3n-2)$ and $i \le j \le i+n\}$ for the CIMGS preconditioner. The numerical results with dimensions from $n = 50$ to $n = 200$ are listed in Table 2.

Obviously, the matrix-form LSQR algorithm with CIMGS preconditioner yields better performance than the unpreconditioning algorithm in aspects of iteration steps and CPU times.

6. Concluding remarks. In this paper, we have constructed a matrix-form LSQR algorithm (Algorithm 3.2) for solving the least squares problem of the matrix equation $AXB = C$ for unknown $n \times n$ tridiagonal matrix X. We have shown that the approximate solution $\widehat{X}_k$, generated by the matrix-form LSQR algorithm at the k-th step, minimizes the residual norm $\|AXB - C\|_F$ in the matrix Krylov subspace $\mathcal{K}_k(\psi, V)$ and the least-squares tridiagonal solution of equation $AXB = C$ with minimum norm can be obtained within at most $3n - 2$ iteration steps by the matrix-form LSQR algorithm in exact arithmetic.

In this paper, we have made an attempt to construct a matrix-form CIMGS preconditioner to accelerate the convergence of the matrix-form LSQR method. The algorithm with CIMGS preconditioner becomes superior to the unpreconditioned algorithm particularly for the coefficient matrices with large condition numbers, which has been confirmed through the numerical experiments. Although, several problems need to be further considered. For example, a more efficient algorithm with the appropriate preconditioning techniques should be constructed to avoid solving two linear systems simultaneously in each iteration. In addition, we should further consider the incomplete QR decomposition for the case that the coefficient matrices are rank deficient in future work.

REFERENCES

[1] Å. Björck. *Numerical Methods for Least Squares Problems.* SIAM, Philadelphia, 1996.
[2] S.H. Chen. *Matrix Perturbation Theory in Structural Dynamical Design.* Science Press, Beijing, 2007.
[3] K.E. Chu. Symmetric solution of linear matrix equations by matrix decomposition. *Linear Algebra Appl.*, 119:35–50, 1989.
[4] Y.B. Deng, X.Y. Hu, and L. Zhang. Least squares solution of $BXA^T = T$ over symmetric, skew-symmetric, and positive semidefinite X^*. *SIAM J. Matrix Anal. Appl.*, 25:486–494, 2003.
[5] M.I. Friswell and J.E. Mottershead. *Finite Element Model Updating in Structural Dynamics.* Kluwer Academic Publication, Boston, 1995.

Electronic Journal of Linear Algebra ISSN 1081-3810
A publication of the International Linear Algebra Society
Volume 23, pp. 1001-1022, December 2012

ELA

[6] M. Heyouni and A. Essai. Matrix Krylov subspace methods for linear systems with multiple right-hand sides. *Numer. Algorithms*, 40:137–156, 2005.

[7] N.J. Higham. Matrix nearness problems and applications. In M.J.C. Gover and S. Barnett (editors), *Applications of Matrix Theory.* Oxford University Press, 1–27, 1989.

[8] J.J. Hou, Z.Y. Peng, and X.L. Zhang. An iterative method for the least squares symmetric solution of matrix equation $AXB = C$. *Numer. Algorithms*, 42:181–192, 2006.

[9] K. Jbilou, A. Messaoudi, and H. Sadok. Global FOM and GMRES algorithms for matrix equations. *Appl. Numer. Math.*, 31:49–63, 1999.

[10] A. Jennings and M.A. Ajiz. Incomplete methods for solving $A^T Ax = b$. *SIAM J. Sci. Comput.*, 5:978–987, 1984.

[11] C.F. Kong and H. Dai. Updating of the viscous damping matrix from incomplete modal experimental data. *J. Vib. Eng.*, 18(1):85–90, 2005 (in Chinese).

[12] Y. Lei and A.P. Liao. A minimal residual algorithm for the inconsistent matrix equation $AXB = C$ over symmetric matrices. *Appl. Math. Comput.*, 188:499–513, 2007.

[13] A.P. Liao and Z.Z. Bai. Least-squares solution of $AXB = D$ over symmetric positive semidefinite matrices X. *J. Comput. Math.*, 21:175–182, 2003.

[14] A.P. Liao and Y. Lei. Optimal approximate solution of the matrix equation $AXB = C$ over symmetric matrices. *J. Comput. Math.*, 25:543–552, 2007.

[15] J.R. Magnus. L-structured matrices and linear matrix equations. *Linear Multilinear Algebra*, 14:67–88, 1983.

[16] C.C. Paige and A. Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Software*, 8(1):43–71, 1982.

[17] Y.Y. Qiu, Z.Y. Zhang, and J.F. Lu. Matrix iterative solutions to the least squares problem of $BXA^T = F$ with some linear constraints. *Appl. Math. Comput.*, 185:284–300, 2007.

[18] R.C. Wang. *Functional Analysis and Optimization Theory.* Beijing University of Aeronautics and Astronautics Press, Beijing, 2003 (in Chinese).

[19] X. Wang, K.A. Gallivan, and R. Bramley. CIMGS: An incomplete orhtogonal factorization preconditionier. *SIAM J. Sci. Comput.*, 18:516–536, 1997.

[20] Y.X. Yuan. The optimal solution of linear matrix equation by using matrix decompositions. *Math. Numer. Sin.*, 24:165–176, 2002 (in Chinese).

[21] J.H. Zhang. *Modeling of Dynamical Systems.* National Defence and Industry Press, Beijing, 2000 (in Chinese).