An Iterative Method for the Symmetric Arrowhead Solution of Matrix Equation

Minghui Wang, Luping Xu, Juntao Zhang

Abstract—In this paper, according to the classical algorithm LSQR for solving the least-squares problem, an iterative method is proposed for least-squares solution of constrained matrix equation. By using the Kronecker product, the form of LSQR is presented to obtain the like-minimum norm and minimum norm solutions in a constrained matrix set for the symmetric arrowhead matrices. Finally, numerical examples are also given to investigate the performance.

Keywords—Symmetric arrowhead matrix, iterative method, like-minimum norm, minimum norm, Algorithm LSQR.

I. INTRODUCTION

Let \( R^{m \times n} \) be the set of \( m \times n \) real matrices, \( SAR^{m \times n} \) be the set of \( n \times n \) real symmetric arrowhead matrices and \( I_n \) be the identity matrix of order \( n \). For any \( A \in R^{m \times n} \), \( A^T, A^* \), \( \|A\| \) and \( \|A\|^T \) denote the transpose, Moore-Penrose generalized inverse, Frobenius norm and Euclid norm, respectively.

For \( A = \{ x_i \} \in R^{m \times n} \) and \( B \in R^{p \times r} \), \( A \otimes B = (x_i B) \in R^{m \times n \times p \times r} \) represents the Kronecker product of \( A \) and \( B \).

For \( A = (x_1, \ldots, x_s) \in R^{m \times n} \), define \( \text{vec}(A) = (x_1^T, x_2^T, \ldots, x_s^T)^T \) and \( x_{i,j} \) as the sub-vector consisting of the elements form \( \alpha \)th component to \( \beta \)th component of \( x_i \). The inverse mapping of \( \text{vec}(\cdot) \) form \( R^{m \times n} \) to \( R^{mn} \) which is denoted by \( \text{mat}(\cdot) \) satisfies \( \text{mat}(\text{vec}(A)) = A \).

**Definition 1.** Let \( A \in SAR^{m \times n} \), \( A \) is called the symmetric arrowhead matrix if it has the following form:

\[
A = \begin{pmatrix}
x_{11} & x_{12} & \cdots & x_{1n} \\
x_{21} & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & x_{nn}
\end{pmatrix}
\]

and \( \text{vec}(A) \) stands for the corresponding vector

\[
(x_{11}, x_{12}, x_{13}, \ldots, x_{nn})^T.
\]

Symmetric arrowhead matrix has many applications in modern control theory which can represent the parameter matrix of nonlinear control systems. Such a matrix was described as radiationless transition in the isolated molecules and oscillator attached to a Fermi liquid \cite{1}. At present, their potential applications in electromagnetic compatibility have been more important such as the mathematical representation of interference factor.

Based on the study of \cite{2}, we consider the matrix equation

\[
AXB + CYD = E. \tag{1}
\]

Many people have studied the matrix equation above and other constrained matrix equations, see [3], [4], [6], [8], etc. Xu, Wei, and Zhang \cite{2} gave the solution of (1) by making use of the canonical correlation decomposition (CCD). Liao, Bai, and Lei \cite{5} studied the least-squares solution of (1) with the least norm by combining CCD and general singular value decomposition (GSVD).

In this paper, we discuss the least-squares solution of (1) for the symmetric arrowhead matrix. When \( A \in R^{m \times n}, B \in R^{p \times r}, C \in R^{s \times q}, D \in R^{l \times k} \) and \( E \in R^{m \times n} \) and

\[
S_n = \left\{ [X,Y] \in SAR^{m \times n}, Y \in SAR^{l \times k}, X \in SAR^{s \times q}, Y \in SAR^{p \times r} \right\}
\]

finding out \([X,Y] \in S_n\), such that

\[
\min \|4AXB + CYD - E\|.
\]

In [7], by using Moore-penrose inverse and the Kronecker product, it discussed the best approximation problem (2) and obtained a general expression of solutions.

For \( H = \text{diag}(H_1, H_2), P_1 = (B^T \otimes A)H_1, P_2 = (B^T \otimes A)H_2 \), and finding \( R = (I - P_1 P_1^*) P_2 \),

\[
G = R^T (I - R^T R) Z P_2^* (P_1^T)^T P_1^* (I - P_1 R^T),
\]

\[
Z = (I + (I - R^T R) P_1^T (P_1^T)^T P_1^* (I - R^T R))^3,
\]

\[
K_{11} = I - P_1 P_1^* P_1^T (I - R^T R) Z, K_{12} = (I - R^T R) Z,
\]

then the set of solutions \( S_n \) was expressed as

\[
\{x_1^*, x_2^*, \ldots, x_n^*\}.
\]
where $y$ is an arbitrary vector with the proper dimension. However, the huge computation cannot be easy to realize in the large scale system which motivates us to find an operable iterative method.

A matrix pair $[X, y]$ is referred to a minimum norm solution if it minimizes

$$
\|X\|_2^2 + \|y\|_2^2,
$$

and a like-minimum norm solution if it minimizes

$$
\|\text{tril}(X)\|_2^2 + \|\text{tril}(y)\|_2^2,
$$

where $\text{tril}(X)$ is denoted as lower triangular part of $X$, that is

$$
\text{tril}(X) = \begin{pmatrix}
x_1 & 0 & \cdots & 0 \\
x_2 & x_3 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
x_{n-1} & x_{n-2} & \cdots & x_n
\end{pmatrix}.
$$

II. PRELIMINARIES

To study the problem (2), we begin with the following lemma and the classical Algorithm LSQR presented for solving least-squares problem [3].

**Lemma 1.** Let $X \in \mathbb{R}^{n \times a}$, then $\text{vec}(X) = \text{Hvec}(X)$, where

$$
H = \begin{pmatrix}
e_1 & e_2 & e_3 & \cdots & e_a & e_0 & 0 & \cdots & 0 & 0 \\
0 & e_1 & 0 & \cdots & 0 & e_0 & 0 & \cdots & 0 & 0 \\
0 & 0 & e_2 & \cdots & 0 & e_0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & e_1 & 0 & \cdots & 0 & e_0
\end{pmatrix},
$$

$H_a \in \mathbb{R}^{a'(2a-1)}$ and $e_i = \left( \begin{array}{c} 0 \ldots 0 1 0 \ldots 0 \end{array} \right)$. First, let us review the Algorithm LSQR for solving the least-squares problem:

$$
\min_{\varphi \in \mathbb{R}^a} \| M \varphi - f \|_2.
$$

with given $M \in \mathbb{R}^{m \times n}$ and vector $f \in \mathbb{R}^m$, whose normal equation is

$$
M^T M \varphi = M^T f.
$$

The algorithm is summarized as follows.

**Algorithm LSQR**

1. Initialization

$$
\beta_{a_1} = f, a_1 v_1 = M^T u_1, h_1 = v_1, x_0 = 0, \tilde{z} = \beta_1. \tilde{P}_1 = a_1;
$$

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

   (i) bidiagonalization
   (a) $\beta_{a_i} u_{a_i} = M v_i - \alpha_i u_i,$
   (b) $a_{a_i} v_{a_i} = M^T u_{a_i} - \beta_{a_i} v_i$;

(ii) construct and use Givens rotation:
$$
\rho_i = \sqrt{\tilde{P}_i^2 + \tilde{P}_i},
$$
$$
c_i = \tilde{P}_i / \rho_i, s_i = \tilde{P}_i / \rho_i, \theta_i = s_i a_{a_i},
$$
$$
\tilde{P}_{i+1} = -c_i a_{a_i}, c_i = c_i s_i - \tilde{P}_i s_i, s_i = s_i, \tilde{P}_i
$$

(iii) update $x$ an $h$
$$
\varphi = \varphi + (\tilde{z} / \rho_i) h_i,
$$
$$
h_i = v_i - (\theta_i / \rho_i) h_i;
$$

(iv) Check convergence.

We can choose

$$
\| M^T (f - M \varphi) \|_2 = \| e_i, e_i, c_i \|_2 < \tau
$$

as convergence criteria, where $\tau > 0$ is a small tolerance. Note that if (5) has a solution $\varphi \in R(M^T M) \subseteq R(M^T)$, then $\varphi$ which is generated by Algorithm LSQR is the minimum norm solution of (5). Then we can have the symmetric arrowhead matrix solution generated by Algorithm LSQR which is the like-minimum norm and minimum norm solution of (2).

III. THE MATRIX-FORM ALGORITHM LSQR FOR (2)

**A. An Iterative Method for the Like-Minimum Norm Solution of (2)**

In this section, we will give some results of this paper and propose iterative methods based on Algorithm LSQR.

**Theorem 1.** Let $X \in \mathbb{R}^{n \times a}$, and $\text{vec}(X) = \text{Hvec}(X)$, then $H^T H = H^T$.

**Proof:** It follows Lemma 1,

$$
H^T H = \begin{pmatrix}
e_1^T & 0 & \cdots & 0 \\
e_2^T & e_1^T & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
e_a^T & \cdots & \cdots & 0 \\
0 & e_1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & e_a & \cdots & 0 \\
0 & 0 & \cdots & e_1^T \\
0 & 0 & \cdots & e_a^T
\end{pmatrix} \in \mathbb{R}^{2a-1 \times (2a-1)},
$$

where $I_{a-1} \in \mathbb{R}^{a \times a-1}$.

We can obtain matrix $H$ is full column rank, with

$$
(H^T H)^{\dagger} = \left( H H^T \right)^{\dagger} = H H^T = H.
$$
Theorem 2. Let $U \in \mathbb{R}^{m \times s}$ and $P = A^T U B^T$, $Q = \frac{P + P^T}{2}$. Then for the symmetric arrowhead constrained matrix $H$, we have

$$((B^T \otimes A)H^T) \vec{U} = \left((H^T H)^T \vec{E}_n Q + (E_n Q)^T + \text{diag}(Q) - 2 E_n QE_n \right)$$

where $E_n = e_i e_i^T$.

**Proof.** Notice that

$$((B^T \otimes A)H^T) \vec{U} = \left(H^T (B \otimes A^T) \vec{U} \right) = H^T \vec{AUB}\vec{P}^T$$

From Theorem 1, for any $P \in \mathbb{R}^{m \times s}$, we have

$$H^T \vec{E}_n Q + (E_n Q)^T + \text{diag}(Q) - 2 E_n QE_n$$

that is

$$H^T \vec{P} = H^T \vec{P}^T = H^T \vec{P} + \frac{P + P^T}{2}.$$ 

It is easy to obtain that

$$H^T \vec{P} + \frac{P + P^T}{2} = H^T \vec{E}_n Q + (E_n Q)^T + \text{diag}(Q) - 2 E_n QE_n.$$ 

From all above, we can have

$$((B^T \otimes A)H^T) \vec{U} = \left(H^T H \vec{P} + (E_n Q)^T + \text{diag}(Q) - 2 E_n QE_n \right).$$

Since

$$\begin{align*}
        |AXB + CD - E| & = \left\| (B^T \otimes A, D^T \otimes C) \right\|_{\vec{X} \rightarrow \vec{Y} \rightarrow \vec{Z}} - \left\| (B^T \otimes A, D^T \otimes C) \right\|_{\vec{X} \rightarrow \vec{Y} \rightarrow \vec{Z}}, \\
        (B^T \otimes A) H_1, (D^T \otimes C) H_2 & = \left\| \vec{X} \rightarrow \vec{Y} \rightarrow \vec{Z} \right\| - \left\| \vec{X} \rightarrow \vec{Y} \rightarrow \vec{Z} \right\|,
    \end{align*}$$

the symmetric arrowhead constrained problem is equivalent to (5) and

$$M = ((B^T \otimes A) H_1, (D^T \otimes C) H_2) \in \mathbb{R}^{n \times (2n^2 + 2k - 1)},$$

$$f = \vec{E} \in \mathbb{R}^{n^2}, \phi = \left\| \vec{X} \rightarrow \vec{Y} \rightarrow \vec{Z} \right\| \in \mathbb{R}^{2n^2 + 2k - 1},$$

where $H_1$ and $H_2$ are the symmetric arrowhead constrained matrices of degree $n$ and $k$, respectively. Therefore, the normal equation of (2) is

$$M^T M \phi = M^T f = M^T \vec{E}.$$ 

Now, we will apply Algorithm $LSQR$ to (2) and the iterative vector will be transformed into matrix so that Kronecker product and constrained matrix $H$ can be released. Then the vector $u$ and $v$ will be expressed by matrix $U$ and $V$ respectively so as to transform the matrix-vector product of $Mv$ and $M^T u$ to the matrix-matrix form.

Let $u = \vec{U} \in \mathbb{R}^m$ with $u \in \mathbb{R}^{m \times s}$, $v = \left\{ v_1, v_2 \right\} \in \mathbb{R}^{2s^2 - 1}$, where $v_1 = \vec{V}_1$ and $v_2 = \vec{V}_2$ with $V_i \in \mathbb{S}^{2n \times 2n}$, $V_j \in \mathbb{S}^{2n \times k}$.

Denoted by

$$W = E_n Q + (E_n Q)^T + \text{diag}(Q) - 2 E_n QE_n,$$

and according to Theorem 2, we have

$$\frac{M^T \vec{E}}{M \vec{E}} = \text{mat} \left( \frac{H^T H \vec{E}}{H^T H \vec{E}} \right) = \left( \frac{W}{W} \right) = 2W - \text{diag}(W).$$

Then

$$\frac{M \vec{E}}{M^T \vec{E}} = \text{mat} \left( \frac{H^T H \vec{E}}{H^T H \vec{E}} \right) = \text{mat} \left( \frac{H^T H \vec{E}}{H^T H \vec{E}} \right) = \left( \frac{W}{W} \right) = 2W - \text{diag}(W).$$

$$\text{mat}(M) = \frac{H^T H \vec{E}}{H^T H \vec{E}} = \text{mat} \left( \frac{H^T H \vec{E}}{H^T H \vec{E}} \right) = \left( \frac{W}{W} \right) = 2W - \text{diag}(W).$$

$$
\begin{align*}
    V_1 & = 2W - \text{diag}(W), \\
    V_2 & = 2W - \text{diag}(W).
\end{align*}
$$

Next, we will give the algorithm for the like-minimum norm solution of (2).

**Algorithm LSQR-W.1**

1. **Initialization**

   \begin{align*}
   X_0 & = 0 \in \mathbb{S}^{2n \times 2n}, \quad X_0 = 0 \in \mathbb{S}^{2n \times k}, \\
   \beta & = \frac{E}{E}, \quad U_i = \frac{E}{E}, \\
   R^{(0)} & = A^T U B^T, \quad R^{(0)} = C^T U D^T, \\
   Q^{(0)} & = \frac{P^{(0)} + P^{(0)^T}}{2}, \quad Q^{(0)} = \frac{P^{(0)} + P^{(0)^T}}{2}.
   \end{align*}

2. **Algorithm**

   \begin{align*}
   X_{i+1} & = X_i - \beta \left(U_i^T U_i - R^{(i)} \right)^{-1} U_i^T \left(R^{(i)} - Q^{(i)} \right), \\
   U_{i+1} & = \frac{U_i - \beta U_i^T \left(U_i^T U_i - R^{(i)} \right)^{-1} \left(U_i^T U_i - R^{(i)} \right) U_i}{\beta}, \\
   R^{(i+1)} & = A^T U_{i+1} B^T, \quad R^{(i+1)} = C^T U_{i+1} D^T, \\
   Q^{(i+1)} & = \frac{P^{(i+1)} + P^{(i+1)^T}}{2}, \quad Q^{(i+1)} = \frac{P^{(i+1)} + P^{(i+1)^T}}{2}.
   \end{align*}
\[ W^{(0)} = E_v Q^{(0)} + \left[ E_v Q^{(0)} \right]^T + \text{diag} \left( Q^{(0)} \right) - 2 E_v Q^{(0)} E_v, \]
\[ W^{(1)} = E_v Q^{(1)} + \left[ E_v Q^{(1)} \right]^T + \text{diag} \left( Q^{(1)} \right) - 2 E_v Q^{(1)} E_v, \]
\[ \mathbf{p}^{(0)} = 2 W^{(0)} - \text{diag} \left( W^{(0)} \right), \]
\[ \mathbf{p}^{(1)} = 2 W^{(1)} - \text{diag} \left( W^{(1)} \right), \]
\[ \alpha_i = \sqrt{\text{tril} \left( \mathbf{p}^{(i)} \right)^2 + \text{tril} \left( \mathbf{p}^{(i)} \right)^2}, \]
\[ v_i^{(0)} = \frac{p_i^{(0)}}{\alpha_i}, \quad v_i^{(1)} = \frac{p_i^{(1)}}{\alpha_i}, \]
\[ Z_i^{(0)} = v_i^{(0)} \left( v_i^{(0)} \right)^T + \left( \theta_i / \rho_i \right) Z_i^{(0)}, \]
\[ Z_i^{(1)} = v_i^{(1)} \left( v_i^{(1)} \right)^T + \left( \theta_i / \rho_i \right) Z_i^{(1)}. \]

(2) Iteration. For \( i = 1, 2, \cdots \)

(i) Compute \( U_i^{(i)} \):
\[ U_i^{(i)} = AV_i^{(i)} B + CV_i^{(i)} D - \alpha_i U_i, \]
\[ U_i = \frac{U_i^{(i)}}{\beta_i}, \]

(ii) Compute \( V_i^{(i)} \):
\[ P_i^{(0)} = A^T U_i^{(i)} B, \quad P_i^{(2)} = C^T U_i, \quad D^T, \]
\[ Q_i^{(0)} = P_i^{(0)} + P_i^{(2)}, \quad Q_i^{(1)} = \frac{P_i^{(1)} + P_i^{(2)}}{2}, \]
\[ W_i^{(0)} = E_v Q^{(0)} + \left[ E_v Q^{(0)} \right]^T + \text{diag} \left( Q^{(0)} \right) - 2 E_v Q^{(0)} E_v, \]
\[ W_i^{(1)} = E_v Q^{(1)} + \left[ E_v Q^{(1)} \right]^T + \text{diag} \left( Q^{(1)} \right) - 2 E_v Q^{(1)} E_v, \]
\[ \beta_i = \sqrt{\text{tril} \left( \mathbf{p}^{(i)} \right)^2 + \text{tril} \left( \mathbf{p}^{(i)} \right)^2}, \]
\[ v_i^{(0)} = \frac{p_i^{(0)}}{\beta_i}, \quad v_i^{(1)} = \frac{p_i^{(1)}}{\beta_i}, \]

(iii) Compute Givens rotation:
\[ \rho_i = \sqrt{p_i^{(0)} + p_i^{(2)}}, \]
\[ c_i = \frac{p_i^{(0)}}{\rho_i}, \quad s_i = \frac{p_i^{(2)}}{\rho_i}, \]
\[ \theta_i = c_i s_i, \]
\[ \beta_i = -s_i c_i, \]
\[ \xi_i = c_i, \quad \xi_i = s_i \]

(iv) Update \( X_i, Y_i, Z_i \):
\[ X_i = X_i + (\xi / \rho_i) Z_i^{(0)}, \]
\[ Y_i = Y_i + (\xi / \rho_i) Z_i^{(0)}, \]
\[ Z_i^{(0)} = v_i^{(0)} \left( v_i^{(0)} \right)^T + \left( \theta_i / \rho_i \right) Z_i^{(0)}, \]
\[ Z_i^{(1)} = v_i^{(1)} \left( v_i^{(1)} \right)^T + \left( \theta_i / \rho_i \right) Z_i^{(1)}. \]

(3) Check convergence.

Algorithm LSQR-W.1 can compute the like-minimum norm solution \( \varphi = \text{vec}(X) \) of (5). So we have the following result.

**Theorem 3.** The symmetric arrowhead solution generated by Algorithm LSQR-W.1 is the like-minimum norm solution of (2).

**B. An Iterative Method for the Minimum Norm Solution of (2)**

In Section A, the like-minimum solution generated by Algorithm LSQR-W.1 is not the best approximation result and non-unique. In this section, we will give an iterative method for the minimum norm solution of (2).

For \( X \in \mathbb{R}^{n \times n} \), define \( \text{vec}(X) = \text{vec}(X) \), and add the weight values to elements \( x_{ij} \), that is

\[
S = \begin{bmatrix}
\sqrt{2}/2 & 2 \cdot \sqrt{2}/2 & \cdots & 2 \cdot \sqrt{2}/2 \\
2 \cdot \sqrt{2}/2 & \sqrt{2}/2 & \cdots & \sqrt{2}/2 \\
\cdots & \cdots & \cdots & \cdots \\
2 \cdot \sqrt{2}/2 & \sqrt{2}/2 & \cdots & \sqrt{2}/2
\end{bmatrix}
\]

with \( S \in \mathbb{R}^{(2n+1)\times(2n+1)} \). Obviously, there is one to one linear mapping from \( \text{vec}(X) \) to \( \text{vec}(X) \). Let \( \mathbf{H} \) denote the minimum norm constrained matrix with

\[ \text{vec}(X) = \mathbf{H} \text{vec}(X). \]

It readily follows from Theorem 1 that

**Theorem 4.** Suppose \( \mathbf{H} \) is the symmetric arrowhead constrained matrix, then

\[ \mathbf{H} = \mathbf{S} \mathbf{R}^{-1} \]

and

\[ \mathbf{H}^T \mathbf{H} = 2 \mathbf{I}_{2n+1}. \]

Since

\[ \|Ax + CY - E\| = \left\| \left( B^T \otimes A \right) \Pi_1 \left( D^T \otimes C \right) \Pi_2 \left( \text{vec}(X) \right) \right\| - \| \text{vec}(E) \|, \]

problem (2) is equivalent to (5) and

\[ M = \left( (B^T \otimes A) \Pi_1 \left( D^T \otimes C \right) \Pi_2 \right) \in \mathbb{R}^{n(2n+2k-2)}, \]

\[ f = \text{vec}(E) \in \mathbb{R}^n, \quad x = \left( \text{vec}(X) \right) \in \mathbb{R}^{2n+2k-2}, \]

where \( \Pi_1 \) and \( \Pi_2 \) are the new symmetric arrowhead constrained matrices of degree \( n \) and \( k \), respectively.

For any \( v = \left( \begin{array}{c} v_1 \\ v_2 \end{array} \right) \in \mathbb{R}^{2n+2k-2}, \ v_1 = \text{vec}(V_1) \) and \( v_2 = \text{vec}(V_2) \) where \( V_1 \in \text{SAR}^{n \times n} \), \( V_2 \in \text{SAR}^{k \times k} \), we have

\[ \hat{v} = v + (\sqrt{2} - 1) \text{diag}(V), \]

and
\[ \text{mat}(Mv) = \text{mat}\left(\left( B^T \otimes A \right) H_v \text{vec}(V_i) + \left( D^T \otimes C \right) H_v \text{vec}(V_j) \right) \]
\[ = \text{mat}\left(\left( B^T \otimes A \right) H_v \text{vec}(V_i) + \left( D^T \otimes C \right) H_v \text{vec}(V_j) \right) \]
\[ = \text{mat}\left(\left( B^T \otimes A \right) H_v \text{vec}(V_i) + \left( D^T \otimes C \right) H_v \text{vec}(V_j) \right) \]
\[ = \text{mat}\left(\left( B^T \otimes A \right) \text{vec}(V_i) + \left( D^T \otimes C \right) \text{vec}(V_j) \right) \]
\[ = A \vec{V} B + C \vec{V} D, \]

For any \( u = \text{vec}(U) \in \mathbb{R}^m \) with \( U \in \mathbb{R}^{m \times n} \), denoted by
\[ W = E_iQ + \left( E_iQ \right)^T + \text{diag}(Q) - 2E_iQE_{ii}, \]
according to Theorem 2, we have
\[ \text{mat}\left( M^T u \right) = \text{mat}\left( H_i^T \left( B \otimes A^T \right) \text{vec}(U) \right) = \text{mat}\left( H_i^T H_i \text{vec}(P_i) \right) \]
\[ = \text{mat}\left( 2I, S, H_i \text{vec}(V_i) \right) \text{mat}\left( 2I, S, H_i \text{vec}(V_j) \right) \]
\[ = \begin{bmatrix} 2W_i - \left( 2 - \sqrt{2} \right) \text{diag}(W_i) \\ 2W_j - \left( 2 - \sqrt{2} \right) \text{diag}(W_j) \end{bmatrix}, \]

where \( V_i \) and \( V_j \) can be formally transformed to symmetric matrices
\[ V_i = 2W_i - \left( 2 - \sqrt{2} \right) \text{diag}(W_i), \]
\[ V_j = 2W_j - \left( 2 - \sqrt{2} \right) \text{diag}(W_j). \]

Next, we give the algorithm for the minimum norm solution of (2).

**Algorithm LSQR-W.2**

1. Initialization
\[ X_0 = 0, U_0 \in \mathbb{S} \mathbb{A} \mathbb{R}^{m \times n}, \]
\[ \beta_i = \left\| \frac{e_i}{\beta_i} \right\|, \]
\[ P_i^{(0)} = A^T U_i B, \quad P_i^{(1)} = C^T U_i D^T, \]
\[ Q_i^{(0)} = \frac{P_i^{(0)} + P_i^{(1)}}{2}, \quad Q_i^{(1)} = \frac{P_i^{(1)} + P_i^{(2)}}{2}, \]
\[ W_i^{(0)} = E_i Q_i^{(0)} + \left( E_i Q_i^{(1)} \right)^T + \text{diag}(Q_i^{(0)}) - 2E_i Q_i^{(0)} E_{ii}, \]
\[ W_i^{(1)} = E_i Q_i^{(1)} + \left( E_i Q_i^{(2)} \right)^T + \text{diag}(Q_i^{(1)}) - 2E_i Q_i^{(1)} E_{ii}, \]
\[ F_i^{(0)} = 2W_i^{(0)} - \left( 2 - \sqrt{2} \right) \text{diag}(W_i^{(0)}), \]
\[ F_i^{(1)} = 2W_i^{(1)} - \left( 2 - \sqrt{2} \right) \text{diag}(W_i^{(1)}), \]
\[ \alpha_i = \sqrt{\left\| F_i^{(0)} \right\| + \left\| F_i^{(1)} \right\|}, \]
\[ V_i^{(0)} = \frac{F_i^{(0)}}{\alpha_i}, \quad V_i^{(1)} = \frac{F_i^{(1)}}{\alpha_i}, \]
\[ Z_i^{(0)} = V_i^{(0)}, \quad Z_i^{(1)} = V_i^{(1)}, \]
\[ \bar{X_i} = \beta_i, \quad \bar{P}_i = \alpha_i. \]

2. Iteration. For \( i = 1, 2, \cdots \)
Theorem 5. The symmetric arrowhead solution generated by Algorithm LSQR-W.2 is the minimum norm solution of (2).

IV. NUMERICAL EXAMPLES

In this section, we reported three numerical examples to illustrate the efficiency of the algorithms we proposed. First, we present the Example 1 in [7] with the results generated by Algorithm LSQR-W.1 and Algorithm LSQR-W.2.

Example 1. Considering the following matrix equation

\[ AXB + CYD = E, \]

with \( A = B = C = D = I_4 \) and

\[
E = \begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0
\end{bmatrix}
\]

i.e.,

\[ I_4X_{i} + I_4Y_{i} = E, \]

where \( I_4 \) is the unit matrix of order 4. We can also obtain the minimum solution by Algorithm LSQR-W.1 and Algorithm LSQR-W.2 which should be

\[
X = \begin{bmatrix}
0.5 & 0.5 & 0.5 & 0.5 \\
0.5 & 0 & 0 & 0.5 \\
0.5 & 0 & 0 & 0.5
\end{bmatrix}, \quad Y = \begin{bmatrix}
0.5 & 0.5 & 0.5 & 0.5 \\
0.5 & 0 & 0 & 0 \\
0.5 & 0 & 0 & 0.5
\end{bmatrix}
\]

The exact result could be computed by our methods in no more than two iterations. Fig. 1 plots the relation between error

\[ \gamma_k = \log_{10}\left( \|AX_{i}B + CY_{i}D - E\| \right) \]

and iterative number \( K \).

Next we present two other matrix equations which show the methods numerically reliable in various circumstances.

Example 2. Given

\[
A = \begin{bmatrix}
\text{hilb}(5) & \text{zeros}(5,3) \\
\text{eye}(5) & \text{ones}(5,3)
\end{bmatrix}, \quad B = \begin{bmatrix}
\text{ones}(3,7) & \text{zeros}(3,5) \\
\text{zeros}(5,7) & \text{pascal}(5)
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
\text{magic}(6) \\
\text{ones}(4,6)
\end{bmatrix}, \quad D = \begin{bmatrix}
\text{hankel}(1:4, 0) & \text{zeros}(4,8) \\
\text{zeros}(2,4) & \text{ones}(2,8)
\end{bmatrix}
\]

\[ E = \begin{bmatrix}
\text{toeplitz}(1:10) & \text{ones}(10,2)
\end{bmatrix}. \]

Thus, \( X = \text{ones}(8,8), \quad Y = \text{ones}(6,6) \) and \( E = AXB + CYD \).

Notice that the problem (1) is not consistent. For \( M \) and \( \varphi \) defined by (8) and (9), we choose residual error

\[ \|M^T Mx - M^T f\| = \|\varphi_1, \varphi_2, \varphi_3\|. \]

Then Figs. 4 and 5 plot the relation between error

\[ \eta_k, \delta_k = \log_{10}\left( \|M^T (Mx - f)\| \right) \]

and iterative number \( K \) by Algorithm LSQR-W.1 and Algorithm LSQR-W.2, respectively.

Fig. 1 The relation between error \( \gamma_k \) and iterative number \( K \)
Fig. 2 The relation between error $\varepsilon_k$ and iterative number $K$

Fig. 3 The relation between error $\mu_k$ and iterative number $K$

Fig. 4 The relation between error $\eta_k$ and iterative number $K$

Fig. 5 The relation between error $\delta_k$ and iterative number $K$

REFERENCES