Research Article

Norm-Constrained Least-Squares Solutions to the Matrix Equation $AXB = C$

An-bao Xu and Zhenyun Peng

Guilin University of Electronic Technology, Guilin 541004, China

Correspondence should be addressed to Zhenyun Peng; yunzhenp@163.com

Received 16 March 2013; Accepted 9 May 2013

Academic Editor: Masoud Hajarian

Copyright © 2013 A.-b. Xu and Z. Peng. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An iterative method to compute the least-squares solutions of the matrix equation $AXB = C$ over the norm inequality constraint is proposed. For this method, without the error of calculation, a desired solution can be obtained with finitely iterative step. Numerical experiments are performed to illustrate the efficiency and real application of the algorithm.

1. Introduction

Throughout this paper, $R^{m\times n}$ denotes the set of all $m \times n$ real matrices. $I$ represents the identity matrix of size implied by context. $A^T$, $\|A\|$ denote, respectively, the transpose and the Frobenius norm of the matrix $A$. For the matrices $A = (a_{ij}) \in R^{m\times n}$, $B = (b_{ij}) \in R^{n\times q}$, $A \otimes B$ represents the Kronecker product of the matrices $A$ and $B$ defined as $A \otimes B = (a_{ij}B) \in R^{mp\times nq}$. The inner product in the matrix set space $R^{m\times n}$ is defined as $\langle A, B \rangle = \text{trace}(B^T A)$ for all the matrices $A, B \in R^{m\times n}$. Obviously, $R^{m\times n}$ is a Hilbert inner product space and the norm of a matrix generated by this inner product space is the Frobenius norm.

Solutions $X$ to the well-known linear matrix equation $AXB = C$ with special structures have been widely studied. See, for example, [1–5] for symmetric solutions, skew-symmetric solutions, centro-symmetric solutions, symmetric R-symmetric solutions, or (R,S)-symmetric solutions. To the best of our knowledge, the solutions to the matrix equation $AXB = C$ subject to the norm inequality constraint, however, have not been studied directly in the literature. In this paper we consider the solutions to the following least-squares problem over the norm inequality constraint:

$$
\min_{X \in R^{m\times n}} \frac{1}{2} \|AXB - C\|^2 \quad \text{subject to} \quad \|X\| \leq \Delta,
$$

(1)

where $A \in R^{p\times m}$, $B \in R^{n\times q}$, $C \in R^{p\times q}$ and $\Delta$ is a nonnegative real number.

Problem (1) can be regarded as a natural generalization of the unconstrained least-squares problem of the matrix equation $AXB = C$. In fact, when we let $\Delta$ be big enough, the problem will turn out to be the unconstrained least-squares problem of the matrix equation $AXB = C$. According to [6], moreover, the problem (1) is equivalent to the classical Tikhonov regulation approach of the matrix equation $AXB = C$

$$
\min_{X \in R^{m\times n}} \frac{1}{2} \|AXB - C\|^2 + \zeta \|X\|^2,
$$

(2)

where $\zeta > 0$ is the regularization parameter. While Tikhonov regularization involves the computation of a parameter that does not necessarily have a physical meaning in most problems, the problem (1) has the advantage that, in some applications, the physical properties of the problem either determine or make it easy to estimate an optimal value for the norm constraint $\Delta$. This is the case, for example, in image restoration where $\Delta$ represents the energy of the target image [7].

In this paper, an iterative method is proposed to compute the solutions of the problem (1). We will use the generalized Lanczos trust region algorithm (GLTR) [8], which is based on Steihaug-Toint algorithm [9, 10], as the frame method for deriving this iterative method. The basic idea is as follows. First, by using the Kronecker production of matrices, we transform the least-squares problem (1) into the trust-region subproblem in vector form which can be solved by the GLTR.
algorithm. Then, we transform the vector iterative method into matrix form. In the end, numerical experiments are given to illustrate the efficiency and real application of the proposed iteration algorithm.

2. Iteration Methods to Solve Problem (1)

In this section we first give the necessary and sufficient conditions for the problem (1) to have a solution. Then we propose an iteration method to compute the solution to the problem. And some properties of this algorithm are also given.

Obviously, problem (1) is equivalent to the following problem

\[
\min_{X \in \mathbb{R}^{m \times n}} \frac{1}{2} \langle AXB, AXB \rangle - \langle AXB, C \rangle \quad \text{subject to } \|X\| \leq \Delta.
\]

(3)

This equivalent form of the problem (1) makes us more convenient to prove the following theorem.

**Theorem 1.** Matrix \(X^*\) is a solution of the problem (3) if and only if there is a scalar \(\lambda^* \geq 0\) such that the following conditions are satisfied:

\[
A^TAX^*BB^T + \lambda^*X^* = A^TCB^T, \quad \lambda^* (\|X^*\| - \Delta) = 0.
\]

(4)

**Proof.** Assume that there is a scalar \(\lambda^* \geq 0\) such that the conditions (4) are satisfied. Let

\[
\varphi(X) = \frac{1}{2} \langle AXB, AXB \rangle - \langle AXB, C \rangle,
\]

\[
\hat{\varphi}(X) = \frac{1}{2} \langle AXB, AXB \rangle + \frac{1}{2} \lambda^* \langle X, X \rangle - \langle AXB, C \rangle
\]

(5)

= \varphi(X) + \frac{1}{2} \lambda^* \langle X, X \rangle.

For any matrix \(W \in \mathbb{R}^{m \times n}\), we have

\[
\hat{\varphi}(X^* + W) = \frac{1}{2} \langle A(X^* + W)B, A(X^* + W)B \rangle + \frac{1}{2} \lambda^* \langle (X^* + W), (X^* + W) \rangle - \langle A(X^* + W)B, C \rangle
\]

\[
= \left(\frac{1}{2} \langle AX^*B, AX^*B \rangle + \frac{1}{2} \lambda^* \langle X^*, X^* \rangle - \langle AX^*B, C \rangle\right)
\]

\[
+ \langle AWB, AX^*B \rangle + \lambda^* \langle W, X^* \rangle
\]

\[
- \langle AWB, C \rangle + \frac{1}{2} \langle AWB, AWB \rangle + \frac{1}{2} \lambda^* \langle W, W \rangle
\]

\[
= \hat{\varphi}(X^*) + \frac{1}{2} \langle AWB, AWB \rangle + \frac{1}{2} \lambda^* \langle W, W \rangle
\]

= \hat{\varphi}(X^*) + \frac{1}{2} \langle AWB, AWB \rangle + \frac{1}{2} \lambda^* \langle W, W \rangle
\]

(6)

This implies that \(X^*\) is a global minimizer of the function \(\hat{\varphi}(X)\). Since \(\hat{\varphi}(X) \geq \hat{\varphi}(X^*)\) for all \(X \in \mathbb{R}^{m \times n}\), we have

\[
\varphi(X) \geq \varphi(X^*) + \frac{1}{2} \lambda^* \langle (X^*, X^*) - (X, X) \rangle.
\]

(7)

The equality \(\lambda^* (\|X^*\| - \Delta) = 0\) implies that \(\lambda^* (\langle X^*, X^* \rangle - \Delta^2) = 0\). Consequently, the following inequality always holds:

\[
\varphi(X) \geq \varphi(X^*) + \frac{1}{2} \lambda^* (\Delta^2 - \langle X, X \rangle).
\]

(8)

Hence, from \(\lambda^* \geq 0\), we have \(\varphi(X) \geq \varphi(X^*)\) for all \(X \in \mathbb{R}^{m \times n}\) with \(\|X\| \leq \Delta\). And so \(X^*\) is a global minimizer of (3).

Conversely, assuming that \(X^*\) is a global solution of the problem (3), we show that there is a nonnegative \(\lambda^*\) such that satisfies conditions (4). For this purpose we consider two cases: \(\|X^*\| < \Delta\) and \(\|X^*\| = \Delta\).

In case \(\|X^*\| < \Delta\), \(X^*\) is certainly an unconstrained minimizer of \(\varphi(X)\). So \(X^*\) satisfies the stationary point condition \(\nabla \varphi(X^*) = 0\); that is, \(A^TAX^*BB^T - A^TCB^T = 0\). This implies that the properties (4) hold for \(\lambda^* = 0\). In the case \(\|X^*\| = \Delta\), the second equality is immediately satisfied, and \(X^*\) also solves the constrained problem

\[
\min_{X \in \mathbb{R}^{m \times n}} \varphi(X) \quad \text{subject to } \|X\| = \Delta.
\]

(9)

By applying optimality conditions for constrained optimization to this problem, we know that there exists a scalar \(\lambda^*\) such that the Lagrangian function defined by

\[
\xi(X, \lambda) = \varphi(X) + \frac{1}{2} \lambda (\langle X, X \rangle - \Delta^2)
\]

(10)

has a stationary point at \(X^*\). By setting \(\nabla_X \xi(X^*, \lambda^*)\) to zero, we obtain

\[
A^TAX^*BB^T - A^TCB^T + \lambda^*X^* = 0.
\]

(11)

Now the proof is concluded by showing that \(\lambda^* \geq 0\). Since the equality (11) holds, then \(X^*\) minimizes \(\hat{\varphi}(X)\), and so we have

\[
\varphi(X) \geq \varphi(X^*) + \frac{1}{2} \lambda^* (\langle X^*, X^* \rangle - \langle X, X \rangle)
\]

(12)

for all \(X \in \mathbb{R}^{m \times n}\). Suppose that there are only negative values of \(\lambda^*\) that satisfy (11). Then we have from (12) that

\[
\varphi(X) \geq \varphi(X^*) \quad \text{whenever } \|X\| \geq \|X^*\| = \Delta.
\]

(13)

Since we already know that \(X^*\) minimizes \(\varphi(X)\) for \(\|X\| \leq \Delta\), it follows that \(X^*\) is in fact a global, unconstrained minimize of \(\varphi(X)\). Therefore conditions (11) hold when \(\lambda^* = 0\), which contradicts our assumption that only negative values of \(\lambda^*\) can satisfy condition (11). The proof is completed. □
We give an iteration method to solve problem (1) as in Algorithm 2.

**Algorithm 2.** (i) Given matrices $X_0 = 0, Q_0 = 0$ and a small tolerance $\epsilon > 0$, compute:

$$R_0 = -A^T CB^T, \quad t_0 = -A^T CB^T, \quad y_0 = \|R_0\|, \quad P_0 = -R_0, \quad T_{-1} = [I].$$

Set $k \leftarrow 0$.

(ii) Computing $Q_k = t_k/y_k, \quad \delta_k = \|AQ_kB\|^2$, $t_{k+1} = A^T AQ_k BB^T - \delta_k Q_k - y_k Q_{k-1}, \quad y_{k+1} = \|t_{k+1}\|$

$$T_k = \begin{bmatrix} T_{k-1} & 0 \\ 0 & \Gamma_k \end{bmatrix},$$

where $\Gamma_k = (0, \ldots, 0, y_k)^T \in R^k$.

(iii) If $AP_kB \neq 0$, compute $\alpha_k = \|R_k\|^2/\|AP_kB\|^2.$

If $\|X_k + \alpha_k P_k\| \leq \Delta$, computing $R_{k+1} = R_k + \alpha_k A^T AP_k BB^T$, $\beta_k = \|R_k\|^2/\|R_k\|^2$, $X_{k+1} = X_k + \alpha_k P_k, P_{k+1} = -R_{k+1} + \beta_k P_k$, else, go to Step 4.

If $\|R_{k+1}\| < \epsilon$, stop, else set $k \leftarrow k + 1$ and go to Step 2.

(iv) Find the solution $h_k$ to the following optimization problem:

$$\min_{h \in R^{k+1}} \frac{1}{2} h^T T_k h + y_0 h^T e_1 \quad \text{subject to} \quad \|h\| \leq \Delta. \quad (14)$$

(v) If $y_{k+1}(e_{k+1}, h_k) < \epsilon$ (here $e_{k+1}$ represents the last column of identity matrix $I$), set

$$X_k = (Q_0, Q_1, \ldots, Q_k)(h_k \otimes I)$$

and then stop, else set $k \leftarrow k + 1$ and go to Step 2.

The basic iteration route of Algorithm 2 to solve problem (1) includes two cases: First, using CG method (Step 3) to compute the solution of problem (1) in feasible region. When the first case is failure, the solution of problem (1) in feasible region cannot be obtained by using CG method, and then the solution of problem (1) on the boundary can be obtained by solving the optimization problem (14). The properties about Algorithm 2 are given as follows.

**Theorem 3.** Assume that the sequences $\{R_i\}, \{P_i\},$ and $\{AP_iB\}$ are generated by Algorithm 2; then the following equalities hold for all $i \neq j, 0 \leq i, j \leq k:

$$\langle R_i, R_j \rangle = 0, \quad \langle P_i, R_j \rangle = 0, \quad \langle AP_iB, AP_jB \rangle = 0. \quad (15)$$

**Proof.** Since $\langle A, B \rangle = \langle B, A \rangle$ holds for all matrices $A$ and $B$, we only need to prove that the conclusion holds for all $0 \leq i < j \leq k$. Using induction and two steps are required.

**Step 1.** Show that $\langle R_1, R_{i+1} \rangle = 0, \langle P_i, R_{i+1} \rangle = 0$ and $\langle AP_iB, AP_{i+1}B \rangle = 0$ hold for all $i = 0, 1, 2, \ldots, k$. We also use the principle of mathematical induction to prove these conclusions. When $i = 0$, we have

$$\langle R_0, R_1 \rangle = \langle R_0, R_0 + \alpha_0 A^T AP_0 BB^T \rangle$$

$$= \langle R_0, R_0 \rangle + \frac{\langle R_0, R_0 \rangle}{\langle AP_0B, AP_0B \rangle} \langle R_0, A^T AP_0 BB^T \rangle$$

$$= \langle R_0, R_0 \rangle + \frac{\langle R_0, R_0 \rangle}{\langle AP_0B, AP_0B \rangle} \langle AR_0B, AP_0B \rangle = 0,$$

$$\langle P_0, R_1 \rangle = \langle P_0, R_0 + \alpha_0 A^T AP_0 BB^T \rangle$$

$$= \langle P_0, R_0 \rangle + \frac{\langle R_0, R_0 \rangle}{\langle AP_0B, AP_0B \rangle} \langle AP_0B, AP_0B \rangle = 0,$$

$$\langle AP_0B, AP_1B \rangle = \langle AP_0B, A (-R_1 + \beta_1 P_0) B \rangle$$

$$= - \langle AP_0B, AR_0B \rangle$$

$$+ \frac{\langle R_1, R_1 \rangle}{\langle R_0, R_0 \rangle} \langle AP_0B, AP_0B \rangle$$

$$= - \langle A^T AP_0 BB^T, R_1 \rangle$$

$$+ \frac{\langle R_1, R_1 \rangle}{\langle R_0, R_0 \rangle} \langle AP_0B, AP_0B \rangle$$

$$= - \langle AP_0B, AP_0B \rangle \langle R_1 - R_0, R_1 \rangle$$

$$+ \frac{\langle R_1, R_1 \rangle}{\langle R_0, R_0 \rangle} \langle AP_0B, AP_0B \rangle = 0.$$
\[
\langle R_s, R_s \rangle + \langle R_s, R_s \rangle + \langle P_i, A^T A P_i BB^T \rangle \\
\times \left( (-P_i, A^T A P_i BB^T) + \langle \beta_{i-1} P_i, A^T A P_i BB^T \rangle \right) \\
= \langle R_s, R_s \rangle + \langle R_s, R_s \rangle - \langle A Q_i B, A Q_i B \rangle \\
\times \langle Q_i, Q_i \rangle - \gamma_i \langle Q_i, Q_i-1 \rangle = 0.
\]

By the principle of induction, \( \langle Q_i, Q_i \rangle = 0 \) holds for all \( i = 0, 1, 2, \ldots, k \).

**Theorem 4.** Assume that the sequence \( \{Q_i\} \) is generated by Algorithm 2, then the following equalities hold:

\[
\langle Q_i, Q_i \rangle = \begin{cases} 
1, & i = j = 0, 1, 2, \ldots, k \\
0, & i \neq j, i, j = 0, 1, 2, \ldots.
\end{cases}
\]

**Proof.** By the definition of \( Q_i \), we immediately know that \( \langle Q_i, Q_i \rangle = 1 \) \( (i = 0, 1, 2, \ldots) \). Similar to the proof of Theorem 3, we also use the principle of mathematical induction to prove this conclusion with the two following cases.

**Step 1.** Show that \( \langle Q_i, Q_{i+1} \rangle = 0 \) for all \( i = 0, 1, 2, \ldots, k \).

When \( i = 0 \), we have

\[
\langle Q_0, Q_0 \rangle = \frac{1}{\gamma_0} \langle Q_0, A^T A Q_0 BB^T - \delta_0 Q_0 \rangle \\
= \frac{1}{\gamma_0} \left( \langle Q_0, A^T A Q_0 BB^T \rangle - \langle Q_0, \delta_0 Q_0 \rangle \right) \\
= \frac{1}{\gamma_0} \left( \langle A Q_0 B, A Q_0 B \rangle - \langle A Q_0 B, A Q_0 B \rangle \langle Q_0, Q_0 \rangle \right) \\
= 0.
\]

Assume that conclusion holds for all \( i \leq s \) \( (0 < s < k) \); then

\[
\langle Q_s, Q_{s+1} \rangle = \frac{1}{\gamma_{s+1}} \langle Q_s, A^T A Q_s BB^T - \delta_s Q_s - \gamma_s Q_{s-1} \rangle \\
= \frac{1}{\gamma_{s+1}} \left( \langle Q_s, A^T A Q_s BB^T \rangle - \langle Q_s, \delta_s Q_s \rangle - \langle Q_s, \gamma_s Q_{s-1} \rangle \right) \\
= \frac{1}{\gamma_{s+1}} \left( \langle A Q_s B, A Q_s B \rangle - \langle A Q_s B, A Q_s B \rangle \langle Q_s, Q_s \rangle \right) \\
\times \langle Q_s, Q_s \rangle - \gamma_s \langle Q_s, Q_{s-1} \rangle = 0.
\]

By the principle of induction, \( \langle Q_i, Q_{i+1} \rangle = 0 \) holds for all \( i = 0, 1, 2, \ldots, k \).
Step 2. Assume that \( \langle Q_i, Q_{i+l} \rangle = 0 \) for all \( 0 \leq i \leq k \) and \( 1 < l < k \) show that \( \langle Q_i, Q_{i+l+1} \rangle = 0 \).

\[
\langle Q_i, Q_{i+l+1} \rangle = \frac{1}{\gamma_{i+l+1}} \left( (\langle AQ_i B, AQ_{i+l} B \rangle - \langle Q_i, \delta_i \rangle Q_{i+l} - \gamma_i \gamma_{i+l} Q_{i+l-1} \right)
\]

\[
= \frac{1}{\gamma_{i+l+1}} \left( \langle A^T AQ_i BB^T, Q_{i+l} \rangle - 0 - 0 \right)
\]

\[
= \frac{1}{\gamma_{i+l+1}} \left( -\gamma_{i+l} Q_{i+l} - \gamma_i Q_{i+l-1}. Q_{i+l} \right) = 0.
\]

(22)

From steps 1 and 2, we have by the principle of mathematical induction that \( \langle Q_i, Q_j \rangle = 0 \) hold for all \( i, j = 0, 1, 2, \ldots, k \), \( i \neq j \).

**Theorem 5.** Assume that the sequences \( \{\gamma_k\}, \{T_k\}, \) and \( \{Q_i\} \) generated by Algorithm 2. Let

\[
\tilde{X} = \begin{bmatrix} Q_0 h^0 + Q_1 h^1 + \cdots + Q_k h^k \end{bmatrix} (h \otimes I),
\]

\[
h = (h^0, h^1, \ldots, h^k)^T \in \mathbb{R}^{k+1}.
\]

(23)

Then the following equality holds:

\[
\frac{1}{2} \langle A\tilde{X}B, A\tilde{X}B \rangle - \langle A\tilde{X}B, C \rangle = \frac{1}{2} h^T T_k h + \gamma_0 h^T e_1,
\]

where \( e_1 \) represents the first column of identity matrix \( I \).

**Proof.** By the definition of \( T_k \) and \( Q_k \) \( (k = 0, 1, 2, \ldots) \), we have

\[
\begin{align*}
& (A^T AQ_0 BB^T, A^T AQ_1 BB^T, \ldots, A^T AQ_k BB^T) \\
& = (Q_0, Q_1, \ldots, Q_k) (T_k \otimes I) + (0, \ldots, 0, \gamma_k Q_{k+1}).
\end{align*}
\]

(25)

Hence, we have

\[
\frac{1}{2} \langle A\tilde{X}B, A\tilde{X}B \rangle - \langle A\tilde{X}B, C \rangle
= \frac{1}{2} \langle \tilde{X}, A^T A\tilde{X}BB^T \rangle - \langle \tilde{X}, A^T CB^T \rangle
= \frac{1}{2} \left( \begin{bmatrix} Q_0, Q_1, \ldots, Q_k \end{bmatrix} (h \otimes I), A^T A \begin{bmatrix} Q_0 h^0 + Q_1 h^1 + \cdots + Q_k h^k \end{bmatrix} BB^T \right)
- \langle (Q_0, Q_1, \ldots, Q_k) (h \otimes I), \gamma_0 Q_0 \rangle
\]

\[
= \frac{1}{2} \left( (Q_0, Q_1, \ldots, Q_k) (h \otimes I), A^T A, q_0 BB^T h^0 \right)
+ A^T A Q_1 BB^T h^1 + \cdots + A^T A Q_k BB^T h^k
\]

\[
+ \gamma_0 \text{trace} (h^T 0 Q_0 h)
= \frac{1}{2} \left( (Q_0, Q_1, \ldots, Q_k) (h \otimes I), A^T A, q_0 BB^T, A^T A Q_1 BB^T, \ldots, \right.
\]

\[
\left. A^T A Q_k BB^T \right) 
\times (h \otimes I) + \gamma_0 h^T e_1
= \frac{1}{2} \left( (Q_0, Q_1, \ldots, Q_k) (h \otimes I), (Q_0, Q_1, \ldots, Q_k) (T_k \otimes I) (h \otimes I) - \gamma_0 h^T e_1
\]

\[
= \frac{1}{2} \text{trace} \left[ (h^T T_k h) (Q_0, Q_1, \ldots, Q_k)^T \right.
\]

\[
\left. + (Q_0, Q_1, \ldots, Q_k) (h \otimes I) + \gamma_0 h^T e_1
\]

\[
= \frac{1}{2} h^T T_k h + \gamma_0 h^T e_1.
\]

(26)

So the equality (24) holds. In addition, from above equality, we have \( h^T T_k h = \langle A\tilde{X}B, A\tilde{X}B \rangle \) for all \( h \in \mathbb{R}^{k+1} \). So \( T_k \) is positive semi-definite. The proof is completed. □

**Theorem 6.** Assume that the sequences \( \{Q_k\}, \{R_k\}, \{\gamma_k\}, \{\delta_k\}, \{\alpha_k\}, \) and \( \{\beta_k\} \) generated by Algorithm 2, then the following equalities hold for all \( k = 0, 1, 2, \ldots \):

\[
Q_k = (-1)^k \frac{R_k}{\lVert R_k \rVert}, \quad \delta_k = \begin{cases} 1, & k = 0, \\
\frac{1}{\alpha_k} + \frac{\beta_{k-1}}{\alpha_{k-1}}, & k > 0,
\end{cases}
\]

\[
\gamma_k = \frac{\sqrt{\beta_{k-1}}}{\alpha_{k-1}}.
\]

(27)

**Proof.** (the proof of the first equality in (27)). By the definition of \( Q_k \) and \( R_k \), we have

\[
Q_k = a_k (A^T A, k (A^T CB^T) (BB^T)^k
\]

\[
+ a_{k-1} (A^T A, k-1 (A^T CB^T) (BB^T)^{k-1}
\]

\[
+ \cdots + a_0 (A^T CB^T),
\]
\begin{align*}
R_k &= (-1)^k b_k (A^T A)^k (A^T C B^T)(B B^T)^k \\
&+ b_{k-1} (A^T A)^{k-1} (A^T C B^T)(B B^T)^{k-1} \\
&+ \cdots + b_0 (A^T C B^T),
\end{align*}

(28)

where $a_i, b_i$ $(i = 0, 1, 2, \ldots, k)$ are positive numbers. These equalities imply that $Q_k$ and $R_k$ belong to the same space

$$K_k = \text{span} \left\{ (A^T A)^k (A^T C B^T)(B B^T)^k, \right.$$ 

$$\left. (A^T A)^{k-1} (A^T C B^T)(B B^T)^{k-1}, \ldots, (A^T C B^T) \right\}. \tag{29}$$

And furthermore we can have

$$\text{span} \{Q_{k-1}, Q_{k-2}, \ldots, Q_0\} = K_{k-1} = \text{span} \{R_{k-1}, R_{k-2}, \ldots, R_0\}. \tag{30}$$

By Theorems 3 and 4, we have $Q_k \perp K_{k-1}$ and $R_k \perp K_{k-1}$. Hence $Q_k$ and $R_k$ must be linear correlation, so there exists a real number $c_k$ such that $Q_k = c_k R_k$. Noting that $\|Q_k\| = 1$, we have by (28) that $Q_k = (-1)^k R_k/\|R_k\|$.

(The proof of the second equality in (27)). Noting that the first equality in (27) holds, then, when $k = 0$, we have

$$\delta_0 = \langle AQ_0 B, AQ_0 B \rangle = \frac{\langle AR_0 B, AR_0 B \rangle}{\langle R_0, R_0 \rangle} = \frac{\langle AP_0 B, AP_0 B \rangle}{\langle R_0, R_0 \rangle} = \frac{1}{\alpha_0}. \tag{31}$$

When $k > 0$, we have

$$\delta_k = \langle AQ_k B, AQ_k B \rangle = \frac{\langle AR_k B, AR_k B \rangle}{\langle R_k, R_k \rangle} = \frac{\langle A (-P_k + \beta_{k-1} P_{k-1}) B, A (-P_k + \beta_{k-1} P_{k-1}) B \rangle}{\langle R_k, R_k \rangle} = \frac{1}{\alpha_k} + \frac{\beta_k^2}{\alpha_k} \frac{\langle AP_k B, AP_k B \rangle}{\langle R_k, R_k \rangle}. \tag{32}$$

(The proof of the third equality in (27)). By the definition of $\gamma_k$, we have

$$\gamma_k^2 = \langle t_k, t_k \rangle = \langle A^T A Q_{k-1} B B^T - \delta_{k-1} Q_{k-1} - \gamma_{k-1} Q_{k-2}, \gamma_{k-1} Q_{k} \rangle = \gamma_k \langle A^T A Q_{k-1} B B^T, Q_k \rangle$$

$$= \gamma_k \left( \frac{\langle A^T A R_{k-1} B B^T, R_k \rangle}{\|R_{k-1}\| \|R_k\|} \right)$$

$$= \gamma_k \left( \frac{\langle A^T A (-P_{k-1} + \beta_{k-2} P_{k-2}) B B^T, R_k \rangle}{\|R_{k-1}\| \|R_k\|} \right)$$

$$= \gamma_k \left( \frac{\langle R_{k-1} - R_k \rangle}{\alpha_{k-1}} \right)$$

$$= \gamma_k \frac{\sqrt{\beta_{k-1}}}{\alpha_{k-1}}. \tag{33}$$

Hence the third equality in (27) holds. The proof is completed.}

\textbf{Remark 7.} This theorem shows the relationship between the sequences $\{Q_k\}, \{R_k\}, \{\gamma_k\}, \{\delta_k\}, \{\alpha_k\}$, and $\{\beta_k\}$ to lower down the cost of calculation.

\section{3. The Main Results and Improvement of the Iteration Method}

We will show that the solution of the problem (1) can be obtained within finite iteration steps in the absence of round-off errors. And we give the detail to solve the problem (14) in order to complete Algorithm 2. By discussing the characterization of the proposed iteration method, the further optimization method for the proposed iteration method is given at the end of this section.

\textbf{Theorem 8.} Assume that the sequences $\{X_k\}, \{R_k\}$ are generated by Algorithm 2. Then the following equalities hold for all $k = 0, 1, 2, \ldots$:

$$A^T A X_k B B^T - A^T C B^T = R_k. \tag{34}$$

\textbf{Proof.} We use the principle of mathematical induction to prove this conclusion. When $k = 0$, obviously, the conclusion holds. Assume that the conclusion holds for $k - 1$; then

$$A^T A X_k B B^T - A^T C B^T = A^T A X_{k-1} B B^T - A^T C B^T + \alpha_k A X_{k-1} B B^T$$

$$= R_{k-1} + \alpha_k A X_{k-1} B B^T = R_k.$$

(35)

This implies that the conclusion holds for $k$. By the principle of mathematical induction, we know that the conclusion holds for all $k = 0, 1, 2, \ldots$. \hfill \qed
Remark 9. For Theorem 3, the sequences $R_0, R_1, R_2, \ldots$ are orthogonal to each other in the finite dimension matrix space $\mathbb{R}^{n^2}$; it is certain that there exists a positive number $k+1 \leq n^2$ such that $R_{k+1} = 0$. So without the error of calculation, the first stopping criterion in the algorithm will perform with finite steps. From Theorem 8, we get $A^T A X_{k+1} B B^T - A^T C B^T = 0$. According to Theorem 1, when we set $\lambda^* = 0$, $X_{k+1}$ is a solution of the problem (3).

Theorem 10. Assume that the sequences $\{Q_k\}$, $\{\gamma_k\}$, and $\{h_k\}$ are generated by Algorithm 2. Let

$$
\bar{X}_k = Q_0 h_k^0 + Q_1 h_k^1 + \cdots + Q_k h_k^k, \quad h_k = (h_k^0, h_k^1, \ldots, h_k^k).
$$

Then, for all $k = 0, 1, 2, \ldots$, there exists a nonnegative number $\lambda_k$ such that

$$
A^T A \bar{X}_k B B^T + \lambda_k \bar{X}_k - A^T C B^T = Q_{k+1} \gamma_k h_k^k, \quad \lambda_k (\| \bar{X}_k \| - \Delta) = 0.
$$

(36)

Proof. Assume that $h_k$ is the solution of optimization problem (14); then there exists a nonnegative number $\lambda_k$ such that the following optimality Karush-Kuhn-Tucker (KKT) conditions are satisfied:

$$
(T_k + \lambda_k I) h_k = -\gamma_0 e_1, \quad \lambda_k (\| h_k \| - \Delta) = 0.
$$

(38)

Noting that $\| \bar{X}_k \| = \| h_k \|$ and the second equality in (38) hold, we know that the second equality in (37) holds.

Since the first equality in (38) can be rewritten as

$$
(T_k \otimes I)(h_k^0 I, h_k^1 I, \ldots, h_k^k I)^T + \lambda_k (h_k^0 I, h_k^1 I, \ldots, h_k^k I)^T + (\gamma_0 I, 0, \ldots, 0)^T = 0,
$$

(39)

so we have

$$
(Q_0, Q_1, \ldots, Q_k)
\times \left[ (T_k \otimes I)(h_k^0 I, h_k^1 I, \ldots, h_k^k I)^T + \lambda_k (h_k^0 I, h_k^1 I, \ldots, h_k^k I)^T + (\gamma_0 I, 0, \ldots, 0)^T \right] = 0.
$$

(40)

Hence, we have

$$
A^T A \bar{X}_k B B^T + \lambda_k \bar{X}_k - A^T C B^T = Q_{k+1} \gamma_k h_k^k Q_{k+1} = 0.
$$

(41)

The proof is completed. □

Theorem 11. Assume that $\gamma_0, \gamma_1, \ldots, \gamma_k \neq 0$, and $\gamma_{k+1} = 0$. Then $\bar{X}_k = Q_0 h_k^0 + Q_1 h_k^1 + \cdots + Q_k h_k^k$ is the solution of the problem (1).

Proof. Since $\gamma_{k+1} = 0$ and $\bar{X}_k = Q_0 h_k^0 + Q_1 h_k^1 + \cdots + Q_k h_k^k$, we have by Theorem 10 that

$$
A^T A \bar{X}_k B B^T + \lambda_k \bar{X}_k = A^T C B^T, \quad \lambda_k (\| \bar{X}_k \| - \Delta) = 0,
$$

(42)

which implies that $\bar{X}_k$ is the solution of the problem (1). □

Remark 12. According to Theorem 4, the sequences $Q_0, Q_1, Q_2, \ldots$ are orthogonal each other in the finite dimension matrix space $\mathbb{R}^{n^2}$; it is certain that there exists a positive number $k \leq n^2$ such that $Q_k = 0$. Since $t_k = \gamma_k Q_k = 0$, then $\gamma_k = \sqrt{(t_k, t_k)} = 0$. So without the error of calculation, the second stopping criterion in the algorithm also performs with finite steps.

Remark 13. According to Remarks 9 and 12, we have that, without the error of calculation, a desired solution can be obtained with finitely iterative step by Algorithm 2.
Theorem 14. The solution $h_k$ of the problem (14) obtained by Algorithm 2 is on the boundary. In other words, $h_k$ is the solution of the following optimization problem:

$$
\min_{h \in \mathbb{R}^{n+1}} \frac{1}{2} h^T T_k h + h^T (\gamma_0 e_1) \quad \text{subject to} \quad \|h\| = \Delta.
$$

Proof. Assuming that the solution $h_k$ of the problem (14) obtained by Algorithm 2 is inside the boundary, we have by (38) that $T_k h_k = -\gamma_0 e_1$. By Theorem 5, we know $T_k$ is a positive semidefinite matrix. If $T_k$ is positive definite, then $h_k = -T_k^{-1}(\gamma_0 e_1)$ with $\|h_k\| < \Delta$ is a unique solution of the problem (14). Hence, we have by Theorem 5 that $X = (Q_0, Q_1, \ldots, Q_k)(h_k \otimes I)$ with $\|X\| = \|h_k\| < \Delta$ is a unique solution of the problem (1). In this case, the step of solving the problem (14) in Algorithm 2 cannot be implemented. If $T_k$ is positive semidefinite and not positive definite, then there exists a matrix $Z$ such that $T_k(h_k + Z) = -\gamma_0 e_1$ and $\|h_k + Z\| = \Delta$ which implies that $h_k + Z$ is a solution to the problem (1) on the boundary. This contradicts our assumption. \qed

Now we use the following Algorithm 15, which was proposed by More and Sorensen in paper [11], to solve the problem (43).

Algorithm 15. (I) Let a suitable starting value $\lambda_0^0$ and $\Delta > 0$ be given.

(II) For $i = 0, 1, \ldots$ until convergence.

(a) Factorize $T_k + \lambda_i^0 I = QAQ^T$, where $Q$ and $\Lambda$ are unit bidiagonal and diagonal matrices, respectively.

(b) Solve $QAQ^T h = -\gamma_0 e_1$.

(c) Solve $Qw = h$.

(d) Set $\lambda_i^{i+1} = \lambda_i^{i+1} + (\|h\| - \Delta)(\|h\|^2/\langle w, \Lambda^{-1} w \rangle)$.

In the implementation of Algorithm 15, the initial secular $\lambda_0^0$ can be chosen by the following principles: If $\|h_k(\lambda_{k-1})\| \geq \Delta$, let $\lambda_0^0 = \lambda_{k-1}$; else let $\lambda_0^0 = 0$, where $\lambda_{k-1}$ is obtained by the $(k-1)$th iterative steps of Algorithm 2. The stopping criteria can be used as $|\lambda_i^{i+1} - \lambda_i^0| \leq \varepsilon$, where $\varepsilon$ is a small tolerance.

By fully using the result of Theorem 6, Algorithm 2 can be optimized as in Algorithm 16.

Algorithm 16. (i) Given matrices $X_0 = 0, Q_{-1} = 0$ and a small tolerance $\varepsilon > 0$.

Computing $R_0 = -A^T C B^T$, $t_0 = -A^T C B^T$.

Set $y_0 = \|R_0\|$, $P_0 = -R_0$, $T_{-1} = [\cdot]$, and $k \leftarrow 0$.

(ii) If $AP^*_k B \neq 0$, compute

$$
Q_k = \frac{(-1)^k R_k}{\|R_k\|}, \quad \alpha_k = \frac{\|R_k\|^2}{\|AP^*_k B\|^2},
$$

$$
R_{k+1} = R_k + \alpha_k A^T AP^*_k B B^T,
$$

$$
\beta_k = \frac{\|R_{k+1}\|^2}{\|R_k\|^2}, \quad \delta_k = \begin{cases} 1, & k = 0, \\ \frac{1}{\alpha_k} + \frac{\beta_{k-1}}{\delta_{k-1}}, & k > 0, \end{cases}
$$

$$
y_{k+1} = \sqrt{\frac{R_k}{\alpha_k}}, \quad T_k = \left[ \begin{array}{c} T_{k-1} \\ \Gamma_k \end{array} \right], \quad \Gamma_k = \delta_k.
$$

(iii) If $\|X_{k+1} + \alpha_k P_{k+1}\| \leq \Delta$, computing $X_{k+1} = X_k + \alpha_k R_k, P_{k+1} = -R_k + \beta_k P_k$.

If $\|R_{k+1}\| \leq \varepsilon$, stop. Else, setting $k \leftarrow k + 1$ and go to Step 2.

(iv) Using Algorithm 15 to compute the solution $h_k$ of the problem (43).

(v) If $\|X_{k+1} \langle e_k, h_k \rangle \| < \varepsilon$, setting $\widetilde{X}_k = (Q_0, Q_1, \ldots, Q_k)(h_k \otimes I)$, then stop.

Else, setting $k \leftarrow k + 1$ and go to step 2.

4. Numerical Experiments

In this section, we present numerical examples to illustrate the availability and the real application of the proposed iteration method. All tests are performed using MATLAB 7.1 with a 32-bit Windows XP operating system. Our experiments are performed on an IBM computer of mode E520 with 2.8 GHz CPU and 3.25 G RAM. Because of the error of calculation, the iteration will not stop with finite steps. Hence, we regard the approximation solution $X_k$ as the solution of problem (1) if the $t(k) \leq 10^{-10}$, where

$$
t(k) = \left\{ \begin{array}{ll} \min \|X_{k+1}\|_F, & X_k + \alpha_k R_k \leq \Delta \\ y_{k+1} \langle e_k, h_k \rangle, & \|X_{k+1}\|_F = \Delta. \end{array} \right.
$$

Example 17. Given the matrices $A, B, C$ as follows:
When $\Delta = 40$, using Algorithm 16 and iterate 43 steps, we obtain the approximation solution


When $\Delta = 10$, using the Algorithm 16 and iterative 23 steps, we obtain the approximation solution

$$X_{23} = \begin{pmatrix} 0.2809 & 0.4183 & 0.9499 & 0.2019 & 0.8336 \\ -0.2275 & -0.1577 & -0.1726 & 0.0837 & 0.2434 \\ 0.1417 & 0.7532 & 2.2683 & 0.8422 & 3.1792 \\ -0.1560 & -0.2950 & -0.7358 & -0.1975 & -0.7844 \\ 1.0118 & 1.8727 & 5.1484 & 1.3685 & 5.1515 \\ 0.2703 & 0.7496 & 1.9622 & 0.6375 & 2.4602 \\ -0.1769 & 0.2915 & 1.0417 & 0.5770 & 2.0664 \end{pmatrix}.$$  

An example of such problem is image denoising with a Gaussian point spread function:

$$\kappa(t) = \omega(t) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{t}{\sigma}\right)^2\right),$$  

which is used as a model for out-of-focus as well as atmospheric turbulence blur [13]. In Figure 1(a), the original image is the standard test image of Lena with size $256 \times 256$. After the image was blurred by Gaussian kernel (53) with $\sigma = 0.01$, we get Figure 1(b); that is the matrix $G = A^T F'$ in (54). Image denoising, our target is to get the solution of $AF^T = G$. Tikhonov regularization is needed to treat this problem in order to control the effect of the noise on the solution. As we have said in Section I, Tikhonov regularization is equivalent to over the norm inequality constraint matrix equation

$$\min_{F \in \mathbb{R}^{m \times n}} \frac{1}{2} \left\| A F^T - G \right\|^2 \text{ subject to } \|F\| \leq \Delta.$$  

Example 18. We work with a 2D first-kind Fredholm integral equation of the generic form

$$\int_{0}^{1} \kappa(x-x') \omega(y-y') f(x',y') dx' dy' = g(x,y),$$  

where $\kappa$ and $\omega$ are function. Based on [12], we have that the discretization of the problem (51) leads to the linear relation

$$A F^T = G$$

between the discrete solution $F$ and the discrete data $G$, where

$$A_{ik} = m^{-1} \kappa(x_i-x_k'), \quad A_{jl} = n^{-1} \omega(y_j-y_l'),$$

$$F_{kl} = f(x_k',y_l'), \quad G_{ij} = g(x_i,y_j),$$

$$i,k = 1,2,\ldots,m, \quad j,l = 1,2,\ldots,n.$$
Based on [7], Δ represents the energy of the target image, so we get \( \Delta = |F^\dagger| \). Solving the above problem by Algorithm 16, we get the recovered image \( F^* \) in Figure 1(c).

It means our algorithm is suitable for image denoising.

**Acknowledgments**

The research was supported by National Natural Science Foundation of China (11261014) and Innovation Project of GUET Graduate Education (XJYC2012023).

**References**


Submit your manuscripts at http://www.hindawi.com