Research Article

Iterative Algorithms for Symmetric Positive Semidefinite Solutions of the Lyapunov Matrix Equations

Min Sun and Jing Liu

1School of Mathematics and Statistics, Zaozhuang University, Zaozhuang, Shandong 277160, China
2School of Date Sciences, Zhejiang University of Finance and Economics, Hangzhou, Zhejiang, China

Correspondence should be addressed to Min Sun; zyouxiaodou@163.com

Received 27 May 2019; Revised 21 October 2019; Accepted 7 November 2019; Published 10 February 2020

Academic Editor: Kishin Sadarangani

Copyright © 2020 Min Sun and Jing Liu. 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.

It is well-known that the stability of a first-order autonomous system can be determined by testing the symmetric positive definite solutions of associated Lyapunov matrix equations. However, the research on the constrained solutions of the Lyapunov matrix equations is quite few. In this paper, we present three iterative algorithms for symmetric positive semidefinite solutions of the Lyapunov matrix equations. The first and second iterative algorithms are based on the relaxed proximal point algorithm (RPPA) and the Peaceman–Rachford splitting method (PRSM), respectively, and their global convergence can be ensured by corresponding results in the literature. The third iterative algorithm is based on the famous alternating direction method of multipliers (ADMM), and its convergence is subsequently discussed in detail. Finally, numerical simulation results illustrate the effectiveness of the proposed iterative algorithms.

1. Introduction

Linear matrix equations are often encountered in various engineering disciplines, such as control theory [1, 2], system theory [3, 4] and stability analysis [5, 6]. For example, the stability of the first-order autonomous system \( \dot{x}(t) = Ax(t) \) is determined by whether the associated continuous-time Lyapunov equation \( AX + XA^T = -M \) exists a positive definite solution or not, and the stability of the discrete-time Lyapunov matrix equation \( X - A^T X A = M \), where \( M \) is a given positive definite matrix with approximate size [7].

In this paper, we consider the following constrained (continuous-time) Lyapunov matrix equations (termed as CLMEs): finding a matrix \( X^* \in S^m_+ \) such that

\[
AX^* + X^* A^T = C,
\]

where \( S^m_+ \) denotes the symmetric positive semidefinite (SPSD) matrix cone, \( A \in \mathbb{R}^{m \times m} \) and \( C \in \mathbb{R}^{m \times m} \) are two given constant matrices, and \( X^* \in \mathbb{R}^{m \times m} \) is an unknown matrix to be determined. To solve problem (1), many iterative methods have been developed in the literature. For example, based on the hierarchical identification principle in parameter estimation, Ding et al. [8] designed a gradient-based iterative algorithm and a least-squares iterative algorithm for problem (1) without \( X^* \in S^m_+ \). Niu et al. [9] and Xie and Ma [10] extended the method in [8] by designing some accelerated gradient-based iterative algorithms for problem (1) without \( X^* \in S^m_+ \), which often have more efficient numerical performance by choosing suitable parameters. Based on the hierarchical identification principle and a new fixed point equation, Sun et al. [7] designed two least-squares iterative algorithms for problem (1) without \( X^* \in S^m_+ \). Conjugate gradient method is an influential numerical method for nonlinear optimization programming, which has also been applied to solve problem (1) without \( X^* \in S^m_+ \). For example, Huang and Ma [11] proposed a conjugate gradient method to solve problem (1) whether it is consistent or not. An appealing characteristic of the method proposed by Huang and Ma [11] is that: (i) it has finite termination property in the absence of round-off errors; (ii) it can obtain least
Frobenius norm solution of problem (1) when it adopts some special kind of initial matrix. Furthermore, Ke and Ma [12] extended the alternating direction method of multipliers (ADMM), which is a famous numerical method in separable optimization programming, to solve problem (1) with the nonnegative constraint $X^* \geq 0$, which is obviously motivated by the design principle proposed by Xu et al. [13].

Though the above iterative methods often have high efficient numerical performance, they only solve problem (1) without the constraint $X^* \in S^n_{+}$. However, we often need to solve problem (1) with the constraint $X^* \in S^n_{+}$ in practice. In this paper, we are going to deal with this issue and propose three iterative algorithms, which are based on the famous relaxed proximal point algorithm (RPPA), the augmented Lagrangian method (ALM) and the alternating direction method of multipliers (ADMM), respectively. By transforming CLMEs into some optimization programming, we can determine the existence of solutions to CLMEs by judging that the optimal value of the transformed optimization programming equals to zero or not.

Before ending this section, let us introduce the notations used in this paper. For any matrix $N \in \mathbb{R}^{mn \times n}$, we use $N^\top$ and $\text{tr}(N)$ to denote its transpose and trace, respectively. The symbol $I_n$ stands for an identity matrix of order $n$. Let $\langle A, B \rangle = \text{tr}(B^\top A) = \sum_{ij}a_{ij}b_{ij}$, where the matrices $A$ and $B$ has the same size. The Frobenius norm $\|N\|$ is defined as $\|N\| = \sqrt{\text{tr}(N^\top N)}$. The symbol $A \otimes B$ is defined as $A \otimes B = (a_jB)$, which represents the Kronecker product of matrices $A$ and $B$. For a matrix $A \in \mathbb{R}^{mn \times n}$, the vectorization operator $\text{vec}(A)$ is defined by $\text{vec}(A) = (a_1^\top, a_2^\top, \ldots, a_m^\top)^\top$, where $a_k$ is the $k$-th column of the matrix $A$. According to the property of the Kronecker product, for any matrices $M, N$ and $X$ with approximate size, it holds that

$$\text{vec}(MXN) = (N^\top \otimes M)\text{vec}(X).$$

(2)

The Kronecker product satisfies the following properties [14]:

Property 1. Let $A \in \mathbb{R}^{mn \times n}$ and $B \in \mathbb{R}^{p \times d}$. Then,

$$A \otimes B = (A \otimes I_p)(I_n \otimes B) = (I_m \otimes B)(A \otimes I_q).$$

(3)

Property 2. Let $A \in \mathbb{R}^{mn \times n}, C \in \mathbb{R}^{mp \times p}, B \in \mathbb{R}^{pq \times q}$, and $D \in \mathbb{R}^{rs \times s}$. Then,

$$(A \otimes B)(C \otimes D) = (AC) \otimes (BD).$$

(4)

Let $P_{S^n_{+}}(\cdot)$ denote the projection operator of a matrix onto the convex set $S^n_{+}$, i.e.,

$$P_{S^n_{+}}(A) = \arg \min_{B \in S^n_{+}} \|A - B\|^2,$$

(5)

which satisfies the following property [15]:

$$\langle P_{S^n_{+}}(A) - A, B - P_{S^n_{+}}(A) \rangle \geq 0, \quad \forall B \in S^n_{+}.$$  

(6)

Furthermore, $P_{S^n_{+}}(A)$ has a closed-form expression, i.e.,

$$P_{S^n_{+}}(A) = U \Lambda^+ U^\top,$$

$$\|U \Lambda^+ U^\top - \Lambda\|_F = \min_{\Lambda \in S^n_{+}} \|U \Lambda^+ U^\top - \Lambda\|_F,$$

(7)

and the computational load of the above procedure is $10m^3$ flops.

The remainder of this paper is organized as follows. In Sections 2 and 3, we present the first and second iterative algorithm for problem (1) and analyze their global convergence. Section 4 discusses the third iterative algorithm for problem (1) and presents its global convergence. Section 5 presents some numerical results to illustrate the efficiency of the proposed algorithms. Finally, Section 6 concludes the entire paper and discusses future research opportunities.

2. The First Iterative Algorithm and Its Convergence

In this section, motivated by the relaxed proximal point algorithm (RPPA) in [16], we shall present an iterative algorithm for the constrained Lyapunov matrix equations and discuss its convergence.

Firstly, let us transform the CLMEs into the following constrained matrix optimization (termed as CMO):

$$\min_{X \in S^n_{+}} \frac{1}{2}\|AX + XA^\top - C\|^2.$$  

(8)

Remark 1. Obviously, the objective function of CMO (8) has lower bound zero, therefore, CMO (8) must have optimal solution, and from its optimal value, we can determine the existence of solutions of CLMEs. In fact, if the optimal value of CMO (8) equals to zero, CLMEs has at least one solution; otherwise, CLMEs does not have solution.

Since CMO (8) can be viewed as the matrix form of the traditionary convex optimization with vector variables, we apply the relaxed proximal point algorithm (termed as RPPA) proposed by Gol’shtein and Tre’tyakov in to solve CMO (8), and get the following RPPA.

Algorithm 1. The RPPA for problem (8)

Step 1. Input a parameter $\gamma \in (0, 2)$, a symmetric positive semidefinite matrix $G \in \mathbb{R}^{m \times m}$, a tolerance $\varepsilon > 0$. Initialize $X^0 \in S^n_{+}$. Set $k = 0$.

Step 2. Generate a temporal iterate $X^k$ by

$$X^k = \arg \min_{X \in S^n_{+}} \left\{ \frac{1}{2} \|I_n \otimes A + A \otimes I_m \|_F^2 \right\}$$

$$+ \frac{1}{2} \|\text{vec}(X) - \text{vec}(X^k)\|_G^2,$$

in which $\|x\|_G^2$ denotes $x^\top Gx$ for some vector $x \in \mathbb{R}^m$. 

$$\frac{1}{2} \|\text{vec}(X) - \text{vec}(X^k)\|_G^2.$$  

(9)
Step 3. If
\[ \|X^k - X^*\| < \varepsilon, \]  
then, stop and return an approximate solution \(X^{k+1}\) of (8).

Step 4. Compute the new iterate by
\[ X^{k+1} = X^k - \gamma (X^k - \bar{X}^k). \]  

Set \( k = k + 1 \), and goto Step 2.

In the following, we set the matrix \( G \) in RPPA as
\[ G = \tau I_m - \Delta^T \Delta, \quad \Delta = I_m \otimes A + A \otimes I_m, \quad \tau \geq \|\Delta^T \Delta\|. \]  

Now, let us elaborate on the optimal subproblem in (9). For the given \( X^k \), by some simple manipulations, the \( X \) subproblem in (9) is amount to solving the following problem:

\[ \bar{X}^k = \arg \min_{X \in S^m_{++}} \frac{1}{2} \| \text{vec}(X) - \left( I_m - \frac{\Delta^T \Delta}{\tau} \right) \text{vec}(X^k) - \frac{\Delta^T \text{vec}(C)}{\tau} \|^2 \]

\[ = \arg \min_{X \in S^m_{++}} \| X - \left( X^k - \frac{(A^T A X^k + A^T X^k A^T + A X^k + X^k A^T A)}{\tau} + \frac{(AC + CA^T)}{\tau} \) \|^2 \]  

\[ = P_{S^m_{++}} \left[ X^k - \frac{(A^T A X^k + A^T X^k A^T + A X^k + X^k A^T A)}{\tau} + \frac{(AC + CA^T)}{\tau} \right], \]

where the second equality follows from the two properties in Section 1, i.e.,
\[ \| \Delta^T \text{vec}(X^k) \| = \left\| \left( I_m \otimes A^T A + A \otimes A^T A + A^T A \otimes I_m \right) \text{vec}(X^k) \right\| = \left\| A^T A^T A X^k + A^T X^k A^T + A X^k + X^k A^T A \right\|, \]
\[ \| \Delta^T \text{vec}(C) \| = \left\| \left( I_m \otimes A + A \otimes I_m \right) \text{vec}(C) \right\| = \| A^T C + CA \|. \]

Then, \( \bar{X}^k \) can be computed by the formula (7).

From [16], we have the following theorem, which means that the iterative sequence \( \{X^k\} \) generated by RPPA is Fejér monotone with respective to the solution set. The proof of its global convergence follows immediately from this property and the results in [17].

**Theorem 1.** Let the sequences \( \{X^k\} \) be generated by the RPPA. Then, for any \( k \geq 0 \), one has
\[ \|X^{k+1} - X^*\| \leq \|X^k - X^*\| - \gamma (2 - \gamma) \|X^k - \bar{X}^k\|. \]  

where \( X^* \) is a solution of CMO.

### 3. The Second Iterative Algorithm and Its Convergence

In this section, based on the Peaceman–Rachford splitting methods in [18, 19], we shall present another iterative algorithm for the constrained Lyapunov matrix equations and discuss its convergence.

By introducing an auxiliary matrix variable \( Y \in \mathbb{R}^m \), CMO (8) can be further written as the following separable convex optimization (termed as SCO):

\[ \min \frac{1}{2} \| A X + X A^T - C \|^2 \]
\[ \text{s.t.} \quad X - Y = 0, \]
\[ X \in \mathbb{R}^{m \times m}, Y \in S^m_{++}. \]

Since SCO (16) can be viewed as the matrix form of the traditionary convex optimization with vector variables, we apply the Peaceman–Rachford splitting method in [18, 19] to solve SCO (15), and get the following Peaceman–Rachford splitting method (termed as PRSM).

**Algorithm 2.** The PRSM for problem (16)

Step 1. Input two parameters \( \alpha \in (0, 1), \beta > 0 \), a tolerance \( \varepsilon > 0 \). Initialize \( X^0 \in \mathbb{R}^{m \times m}, Y^0 \in S^m_{++}, A^0 \in \mathbb{R}^{m \times m}. \) Set \( k = 0 \).
Step 2. Generate the new iterate $W^{k+1} = (X^{k+1}, \ Y^{k+1}, \ \Lambda^{k+1})$ by

$$
\begin{align*}
X^{k+1} &= \arg \min_{X \in \mathbb{R}^{m \times m}} \left\{ \frac{1}{2} \left\| AX + XA^\top - C \right\|^2 - (\Lambda^k)^\top X + \frac{\beta}{2} \left\| X - Y^k \right\|^2 \right\}, \\
\Lambda^{k+(1/2)} &= \Lambda^k - a\beta(X^{k+1} - Y^k), \\
Y^{k+1} &= \arg \min_{Y \in \mathbb{R}^m} \left\{ (\Lambda^{k+(1/2)})^\top Y + \frac{\beta}{2} \left\| X^{k+1} - Y \right\|^2 \right\}, \\
\Lambda^{k+1} &= \Lambda^{k+(1/2)} - a\beta(X^{k+1} - Y^{k+1}).
\end{align*}
$$

(17)

where

$$
R = AX + XA^\top - C.
$$

(20)

Thus, we have

$$
\vec{X}^{k+1} = (I_m \otimes A^\top A + A \otimes A^\top + A^\top \otimes A + A A^\top \otimes I_m + \beta I_m)^{-1} \cdot \vec{X}^k + \vec{C}.
$$

(21)

Step 3. If

$$
\max\left\{ \left\| X^k - X^{k+1} \right\|, \left\| Y^k - Y^{k+1} \right\|, \left\| \Lambda^k - \Lambda^{k+1} \right\| \right\} < \varepsilon,
$$

(18)

then stop and return an approximate solution $(X^{k+1}, \ Y^{k+1})$ of (16); else set $k := k + 1$, and goto Step 2.

Now, let us elaborate on the two optimization subproblems in (17).

For the given $Y^k$, $\Lambda^k$, the $X$ subproblem in (17) is amount to solving the following normal equation:

$$
A^\top R + RA + \beta \left( \frac{X - Y^k - \Lambda^k}{\beta} \right) = 0,
$$

(19)

which can be computed by the formula given in (7).

Remark 2. Though the computation of $X^{k+1}$ involves an inverse of a matrix. However, since this term is invariant in each iteration, we only need to compute it once before all iterations.

From [18, 19], we have the following theorem, which means that the iterative sequence $\{W^k\}$ generated by PRSM is Fejér monotone with respective to the solution set. Obviously, SCO is a convex optimization problem. Thus, its global convergence follows immediately from this property and the results in [17].

\[\text{Theorem 2. Let } \{W^k\} \text{ be the sequence generated by the PRSM. Then, for any } k \geq 0, \text{ one has} \]

\[
\left\| W^{k+1} - W^* \right\|_H \leq \left\| W^k - W^* \right\|_H - \left\| W^k - W^k \right\|_N^2,
\]

(23)

where $H$ and $N$ are two positive definite matrices, $W^* = (X^*, \ Y^*, \ \Lambda^*)$ with $(X^*, \ Y^*)$ being a solution of SCO, $\Lambda^*$ being the corresponding Lagrangian multiplier, and $\{\bar{W}^k\}$ is an auxiliary sequence defined as

\[
\bar{W}^k = \begin{pmatrix} \bar{X}^k \\ \bar{Y}^k \\ \bar{\Lambda}^k \end{pmatrix} = \begin{pmatrix} X^{k+1} \\ Y^{k+1} \\ \Lambda^k - \beta(X^{k+1} - Y^k) \end{pmatrix}.
\]

(24)
Remark 3. Following the above procedure, some famous iterative algorithms for convex programming, such as the proximal point algorithm (PPA), the relaxed PPA [16], can also be used to solve SCO.

4. The Third Iterative Algorithm and Its Convergence

In this section, motivated by the alternating direction methods of multiplier (ADMM) in [12, 13], we shall present the third iterative algorithm for the constrained Lyapunov equations and discuss its convergence.

Firstly, CMO (8) can be written as the following equivalent form:

\[
\min \frac{1}{2} \| AX + Y A^\top - C \|^2 \\
\text{s.t.} \quad X - Y = 0, Y - Z = 0, \\
X \in \mathbb{R}^{m \times m}, Y \in \mathbb{R}^{m \times m}, Z \in \mathbb{S}_+^m.
\]  

(25)

The augmented Lagrangian function of (25) is

\[
\mathcal{L}(X, Y, Z, \Lambda, \Pi) = \frac{1}{2} \| AX + YA^\top - C \|^2 - \langle A, X - Y \rangle - \langle \Pi, Y - Z \rangle + \frac{\alpha}{2} \| X - Y \|^2 + \frac{\beta}{2} \| Y - Z \|^2,
\]

(26)

where the matrices \( \Lambda, \Pi \in \mathbb{R}^{m \times m} \) are Lagrangian multipliers and the constants \( \alpha, \beta > 0 \) are penalty parameters. Then, we can derive the second iterative algorithm (termed as ADMM) for CMO by alternating minimizing one variable in \( \mathcal{L}(X, Y, Z, \Lambda, \Pi) \) when other variables is fixed.

Algorithm 3. The ADMM for problem (25)

**Step 1.** Input three parameters \( \alpha, \beta > 0, \gamma \in (0, (1 + \sqrt{5})/2) \), a tolerance \( \varepsilon > 0 \). Initialize \( X^0, Y^0, \Lambda^0, \Pi^0 \in \mathbb{R}^{m \times m} \), and \( Z^0 \in \mathbb{S}_+^m \). Set \( k = 0 \).

**Step 2.** Compute the new iterate \( W^{k+1} = (X^{k+1}, Y^{k+1}, Z^{k+1}, \Lambda^{k+1}, \Pi^{k+1}) \) by

\[
X^{k+1} = \arg \min_{X \in \mathbb{R}^{m \times m}} \mathcal{L}(X, Y^k, Z^k, \Lambda^k, \Pi^k), \\
Y^{k+1} = \arg \min_{Y \in \mathbb{R}^{m \times m}} \mathcal{L}(X^{k+1}, Y, Z^k, \Lambda^k, \Pi^k), \\
Z^{k+1} = \arg \min_{Z \in \mathbb{S}_+^m} \mathcal{L}(X^{k+1}, Y^{k+1}, Z, \Lambda^k, \Pi^k), \\
\Lambda^{k+1} = \Lambda^k - \gamma \alpha (X^{k+1} - Y^{k+1}), \\
\Pi^{k+1} = \Pi^k - \gamma \beta (Y^{k+1} - Z^{k+1}).
\]

(27)

**Step 3.** If

\[
\max \left\{ \| X^k - X^{k+1} \|, \| Y^k - Y^{k+1} \|, \| Z^k - Z^{k+1} \|, \| X^k - X^{k+1} \|, \| Y^k - Y^{k+1} \|, \| \Pi^k - \Pi^{k+1} \| \right\} < \varepsilon,
\]

(28)

then stop and return an approximate solution \( (X^{k+1}, Y^{k+1}, Z^{k+1}) \) of (25); else, set \( k = k + 1 \), and goto Step 2.

Remark 4. It is obvious that the two matrices \( A^\top A \) and \( (\alpha + \beta) I_m + A^\top A \) are symmetric positive definite for any real constants \( \alpha, \beta > 0 \), which are invariant during the iteration, and we only need to compute their inverses once before all iterations.
The KKT conditions for problem (25) are:

\[
\begin{align*}
A^T(AX + YA^T - C) - \Lambda &= 0, \\
(AX + YA^T - C)A + \Lambda - \Pi &= 0, \\
\Pi - \Gamma &= 0, \\
X - Y &= 0, \\
Y - Z &= 0, \\
\langle \Gamma, Z \rangle &= 0, Z \in S^n_+, \Gamma \in S^n_+,
\end{align*}
\]  

(34)

where \(\Lambda, \Pi, \) and \(\Gamma\) are three Lagrangian multipliers corresponding to the constraints \(X - Y = 0, Y - Z = 0\) and \(Z \in S^n_+\) in (25), respectively. Eliminating the variable \(\Gamma\), the above KKT system can be written as:

\[
\begin{align*}
A^T(AX + YA^T - C) - \Lambda &= 0, \\
(AX + YA^T - C)A + \Lambda - \Pi &= 0, \\
X - Y &= 0, \\
Y - Z &= 0, \\
\langle \Pi, Z \rangle &= 0, Z \in S^n_+ , \Pi \in S^n_+.
\end{align*}
\]

(35) – (38)

If the point \((X, Y, Z, \Lambda, \Pi)\) satisfies the above conditions, then \((X, Y, Z)\) is called KKT point, and \((\Lambda, \Pi)\) is the corresponding multiplier.

To simplify notation, we set \(W = (X, Y, Z, \Lambda, \Pi)\). Now, let us analyze the convergence of ADMM. Due to the objective function of (25) is not separable, we only get the following weak convergence result.

**Theorem 3.** Let \([W^k]\) be the sequence generated by ADMM which satisfies the condition

\[
\lim_{k \to \infty} (W^{k+1} - W^k) = 0.
\]

(40)

Then, any accumulation point of \([W^k]\) satisfies the KKT conditions for problem (25).

**Proof.** From (11)–(13), it holds that

\[
\begin{align*}
(aI_m + A^T A)(X^{k+1} - X^k) &= A^T(C - AX^k - Y^k A^T) + \Lambda^k + \alpha(Y^k - X^k), \\
(Y^{k+1} - Y^k)((a + \beta)I_m + A^T A) &= (C - AX^{k+1} - Y^{k+1} A^T)A - \Lambda^k + \Pi^k + \alpha(X^{k+1} - Y^k) + \beta(Z^k - Y^k), \\
&\quad + \alpha(X^{k+1} - Y^k) + \beta(Z^k - Y^k), \\
Z^{k+1} - Z^k &= P_{S^n_+} \left[ \begin{pmatrix} (Y^{k+1} - \Pi^k/\beta) + (Y^{k+1} - \Pi^k/\beta) \end{pmatrix}^T \right] - Z^k, \\
\Lambda^{k+1} - \Lambda^k &= -\gamma\alpha(X^{k+1} - Y^{k+1}), \\
\Pi^{k+1} - \Pi^k &= -\gamma\beta(Y^{k+1} - Z^{k+1}).
\end{align*}
\]

(41)

From \(\lim_{k \to \infty} (W^{k+1} - W^k) = 0\), we can find that both the sides in the above five equations tend entirely to zero, i.e.,

\[
\begin{align*}
A^T(C - AX^k - Y^k A^T) + \Lambda^k &\to 0, \\
(C - AX^{k+1} - Y^{k+1} A^T)A - \Lambda^k + \Pi^k &\to 0, \\
P_{S^n_+} \left[ \begin{pmatrix} (Y^{k+1} - \Pi^k/\beta) + (Y^{k+1} - \Pi^k/\beta) \end{pmatrix}^T \right] - Z^k &\to 0, \\
X^{k+1} - Y^{k+1} &\to 0, \\
Y^{k+1} - Z^{k+1} &\to 0.
\end{align*}
\]

(42) – (46)

For any limit \(\tilde{W} = (\tilde{X}, \tilde{Y}, \tilde{Z}, \tilde{\Lambda}, \tilde{\Pi})\) of the sequence \([W^k]\), from (42), (43), (45) and (46), it is obvious that the first four equations (14)–(17) in the KKT conditions are satisfied.

In the following, we only need to prove that (39) also hold for \(\tilde{W}\). From (35) and (36), it holds that

\[
A^T(A\tilde{X} + \tilde{Y}A^T - C) - \tilde{\Lambda} = 0, (\tilde{A}X + \tilde{Y}A^T - C)A + \tilde{\Lambda} - \tilde{\Pi} = 0.
\]

(47)

This together with \(\tilde{X} = \tilde{Y}\) implies that

\[
\tilde{\Pi} = A\tilde{X}A + \tilde{X}A^T A - CA + A^T A\tilde{X} + A^T \tilde{X}A^T - A^T C.
\]

(48)

This and the symmetry of \(\tilde{X}\) and \(C\) imply that the matrix \(\tilde{\Pi}\) is also symmetric. From (44) and (46), it holds that
\[ Z = P_{\frac{\lambda}{\beta}} \left[ \overline{Z} - \overline{\Pi} \right]. \]  

(49)

From this and (6), we have

\[ \langle \overline{\Pi}, B - \overline{Z} \rangle \geq 0, \quad \forall B \in S^m_+. \]

(50)

Setting \( B = 0 \) and \( B = 2\overline{Z} \) in the above inequality, we get

\[ \langle \overline{\Pi}, \overline{Z} \rangle \leq 0, \quad \langle \overline{\Pi}, \overline{Z} \rangle \geq 0. \]

(51)

Thus, \( \langle \overline{\Pi}, \overline{Z} \rangle = 0 \). Let \( \lambda_i \quad (i = 1, 2, \ldots, m) \) with \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m \) be the \( m \) eigenvalues of the matrix \( \overline{\Pi} \). Suppose the matrix \( \overline{\Pi} \) is not positive semidefinite, then \( \lambda_m \leq 0 \). Let \( P \in \mathbb{R}^{m\times m} \) be an orthogonal matrix, such that \( \overline{\Pi} = P \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m) P^T \). Setting \( B = \overline{Z} + P \text{diag}(0, \ldots, t) P^T \in S^m_+ \) with \( t > 0 \) in (50), we have

\[ \langle \overline{\Pi}, P \text{diag}(0, 0, \ldots, t) P^T \rangle \geq 0. \]

(52)

Thus,

\[ \langle P^T \overline{\Pi} P, \text{diag}(0, 0, \ldots, t) \rangle \geq 0, \]

(53)

That is,

\[ \langle \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m), \text{diag}(0, 0, \ldots, t) \rangle \geq 0. \]

(54)

Then, \( t \lambda_m \geq 0 \), which contradicts with \( t > 0 \) and \( \lambda_m < 0 \). Thus, the matrix \( \overline{\Pi} \in S^m_+ \). This completes the proof. \( \square \)

Remark 5. Based on the proximal ADMM [20], we can design an inverse-free ADMM for the studied problem by attaching some proximal terms to the subproblems with respective to \( X \) and \( Y \) in ADMM. Furthermore, based on the parallel or partially parallel splitting method or their proximal versions [21, 22], we can design some inverse-free (partially) parallel splitting methods for the studied problem [23, 24].

5. Numerical Results

In this section, we evaluate the performance of the three iterative algorithms designed in this paper for solving the Lyapunov matrix equations, and report some preliminary numerical results. The codes of our algorithms were written entirely in Matlab R2014a, and were executed on a ThinkPad notebook with Pentium(R) Dual-Core CPU T4400@2.2 GHz, 2 GB of memory.

The parameters in the three tested algorithms are set as follows:

For RPPA: \( \tau = \| \Delta^T \Delta \| \), \( \gamma = 1.8 \)

For PRSM: \( \alpha = 0.9 \), \( \beta = 2 \)

For ADMM: \( \alpha = 2 \), \( \beta = 2 \), \( \gamma = 1.618 \)

Problem 1. Consider the following constrained Lyapunov matrix equations:

\[ AX^* + X^* A^T = -C, \quad X^* \in S^m_+, \]

(55)

where \( A \) and \( C \) are set as the following three cases:

\[ A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}, \]

\[ X^* = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, \]

\[ C = -AX^* - X^* A^T = \begin{bmatrix} -8 & -8 \\ -8 & -6 \end{bmatrix}, \]

(56)

\[ A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}, \]

\[ X^* = \begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix}, \]

\[ C = -AX^* - X^* A^T = \begin{bmatrix} -6 & -4 \\ -10 & 8 \end{bmatrix}, \]

(57)

\[ A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}, \]

\[ X^* = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}, \]

\[ C = -AX^* - X^* A^T = \begin{bmatrix} -2 & -2 \\ -2 & -6 \end{bmatrix}. \]

All initial matrices are set zero matrix. Due to the iterate \( Y^k \) generated by PRSM satisfies the constraint \( Y \in S^m_+ \) and

\[ \lim_{k \to \infty} \| X^k - Y^k \| = 0, \]

(57)

and RPPA, ADMM also have similar property, then we adopt the following stopping criterion:

\[ \text{Error} (k) = \| AX^k + X^k A^T + C \| < 10^{-6}, \]

(58)

or the number of iterations \( k > 1000 \).

The numerical results of the three tested algorithms for the first case of Problem 1 are plotted in Figure 1, from which we find that they successfully solve this problem within 66, 35, and 37 iterations, respectively, and the final iterates of the three tested algorithms are

\[ X_{\text{RPPA}}^{66} = \begin{bmatrix} 1.99999588105564 & 1.000000128578955 \\ 1.000000128578955 & 1.000000128578955 \end{bmatrix}, \]

(59)

\[ X_{\text{PRSM}}^{35} = \begin{bmatrix} 1.99999561481541 & 0.99999561481541 \\ 0.99999561481540 & 0.99999561481540 \end{bmatrix}, \]

\[ X_{\text{ADMM}}^{37} = \begin{bmatrix} 2.000046066342317 & 1.000046066349865 \\ 1.000046066349865 & 1.000046066349865 \end{bmatrix}. \]

The corresponding absolute errors are

\[ \| X_{\text{RPPA}}^{66} - X^* \| = 9.3941 \times 10^{-7}, \| X_{\text{PRSM}}^{35} - X^* \| = 8.7704 \times 10^{-7}, \| X_{\text{ADMM}}^{37} - X^* \| = 6.0869 \times 10^{-7}. \]

(60)
Though the matrix

\[ X^* = \begin{bmatrix} 2 & 0 \\ 1 & 1 \end{bmatrix} \]  

(61)

Satisfies the equations \( AX + XA^T = -C \), it is not symmetric. Then, it is not a solution of CMO. The numerical results of the three tested algorithms for the second case of Problem 1 are plotted in Figure 2, from which we find that the Error\((k)\)s generated by the three algorithms do not converge to zero, and the final iterate is about

\[ X^{1000} = \begin{bmatrix} 2 & 0.5 \\ 0.5 & 1 \end{bmatrix} \]  

(62)

Similarly, the matrix

\[ X^* = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \]  

(63)

satisfies the equations \( AX + XA^T = -C \), it is not positive definite. Then, it is not a solution of CMO. The numerical results of the three tested algorithms for the third case of Problem 1 are plotted in Figure 3, from which we find that the Error\((k)\)s generated by the three algorithms do not converge to zero, and the final iterate is about

\[ X^{1000} = \begin{bmatrix} 0.15653 & 0.41045 \\ 0.41045 & 1.0763 \end{bmatrix} \]  

(64)

Problem 2. Consider the following constrained Lyapunov matrix equations:

\[ AX^* + X^* A^T = -C, \quad X^* \in S^+_n, \]  

(65)

where \( A \) and \( C \) are set as

\[ A = \begin{bmatrix} 1 & 3 \\ 2 & 1 \end{bmatrix}, \]  

\[ X^* = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, \]  

\[ C = -AX^* - X^* A^T = \begin{bmatrix} -10 & -9 \\ -10 & -6 \end{bmatrix}. \]  

(66)

Three curves in Figure 4 indicates that for CLMEs with nonsymmetric matrix \( A \), RPPA and PRSM fall into the premature convergence, while ADMM successfully solve this problem.
Problem 3. Consider the following constrained Lyapunov matrix equations:

\[ AX^* + X^* A^\top = -C, \quad X^* \in S_m^+, \]  

(67)

where \( A \) and \( X^* \) are set by

\[ A = \text{rand}(n), \]
\[ A = A + A^\top, \]
\[ D = \text{diag}(\text{rand}(n,1)), \]
\[ U = \text{orth}(\text{rand}(n,n)), \]
\[ X^* = U^\top \cdot D \cdot U. \]  

(68)

Then, \( C \) is set as \( C = -(AX^* + X^* A^\top) \). Furthermore, we set \( n = 20 \).

The numerical results of the three tested algorithms are plotted in Figure 5, from which we find that the performance of ADMM is a litter better than that of PRSM, which is obviously better than that of RPPA.

6. Conclusions

In this paper, three iterative algorithms have been proposed to solve constrained Lyapunov matrix equations, and their convergence is discussed in detail. Furthermore, preliminary numerical results have been included, which verify the efficiency of the proposed algorithms.

Among various challenges for large-scale constrained Lyapunov matrix equations, high performance is a must, not an option. Therefore, an object of our future investigations is to design more efficient iterative algorithms and propose some valid criteria, from which we can directly determine whether the Lyapunov matrix equations has a symmetric positive semidefinite solution or not.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This research was partially supported by the National Natural Science Foundation of Shandong Province (no. ZR2016AL05) and the Doctoral Foundation of Zaozhuang University.

References

[1] A. G. Wu, Y. M. Fu, and G. R. Duan, “On solutions of matrix equations \( V - AVF = BW \) and \( V - aVF = BW \),” *Mathematical and Computer Modelling*, vol. 47, no. 11-12, pp. 1181–1197, 2008.

[2] A. G. Wu, H. Q. Wang, and G. R. Duan, “On matrix equations \( X - AXF = C \) and \( JX - AXF = X \),” *Journal of Computational and Applied Mathematics*, vol. 230, no. 2, pp. 690–698, 2009.

[3] H. Zhang and F. Ding, “Iterative algorithms for \( X + A^\top X A = I \) by using the hierarchical identification principle,” *Journal of the Franklin Institute*, vol. 353, no. 5, pp. 1132–1146, 2016.

[4] A.-G. Wu, G. Feng, G.-R. Duan, and W. Liu, “Iterative solutions to the Kalman-Yakubovich-conjugate matrix equation,” *Applied Mathematics and Computation*, vol. 217, no. 9, pp. 4427–4438, 2011.

[5] M. Hajarian, “Developing biCOR and CORS methods for coupled Sylvester-transpose and periodic Sylvester matrix equations,” *Applied Mathematical Modelling*, vol. 39, no. 9, pp. 6073–6084, 2015.
[6] C. Q. Song and J. E. Feng, "On solutions to the matrix equations $XB - AX = CY$ and $XB - AX^T = CY$,” *Journal of the Franklin Institute*, vol. 353, no. 5, pp. 1075–1088, 2016.

[7] B. N. Datta, *Numerical Methods for Linear Control Systems*, Elsevier Science & Technology Books, New York, NY, USA, 2003.

[8] F. Ding, P. X. Liu, and J. Ding, "Iterative solutions of the generalized Sylvester matrix equations by using the hierarchical identification principle," *Applied Mathematics and Computation*, vol. 197, pp. 41–50, 2008.

[9] Q. Niu, X. Wang, and L.-Z. Lu, "A relaxed gradient based algorithm for solving Sylvester equations," *Asian Journal of Control*, vol. 13, no. 3, pp. 461–464, 2011.

[10] Y.-J. Xie and C.-F. Ma, "The accelerated gradient based iterative algorithm for solving a class of generalized Sylvester-transpose matrix equation," *Applied Mathematics and Computation*, vol. 273, pp. 1257–1269, 2016.

[11] N. Huang and C. Ma, "The modified conjugate gradient methods for solving a class of generalized coupled Sylvester-transpose matrix equations," *Computers & Mathematics with Applications*, vol. 67, no. 8, pp. 1545–1558, 2014.

[12] Y. Ke and C. Ma, "An alternating direction method for nonnegative solutions of the matrix equation $AX + YB = C$," *Computational and Applied Mathematics*, vol. 36, no. 1, pp. 359–365, 2017.

[13] Y. Xu, W. Yin, Z. Wen, and Y. Zhang, "An alternating direction algorithm for matrix completion with nonnegative factors," *Frontiers of Mathematics in China*, vol. 7, no. 2, pp. 365–384, 2012.

[14] H. M. Zhang and F. Ding, "On the Kronecker products and their applications," *Journal of Applied Mathematics*, vol. 2013, Article ID 296185, 8 pages, 2013.

[15] Y. J. Wang and N. H. Xiu, *Nonlinear Programming Theory and Algorithm*, Shaanxi Science and Technology Press, Xian, China, 2008.

[16] M. Li and Y. F. You, "Convergence analysis of the relaxed proximal point algorithm," *Abstract and Applied Analysis*, vol. 2013, Article ID 912846, 6 pages, 2013.

[17] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, Berlin, Germany, 2011.

[18] B. He, H. Liu, Z. Wang, and X. Yuan, "A strictly contractive peaceman–rachford splitting method for convex programming," *SIAM Journal on Optimization*, vol. 24, no. 3, pp. 1011–1040, 2014.

[19] M. Sun and J. Liu, "A proximal Peaceman-Rachford splitting method for compressive sensing," *Journal of Applied Mathematics and Computing*, vol. 50, no. 1-2, pp. 349–363, 2016.

[20] B. He, L.-Z. Liao, D. Han, and H. Yang, "A new inexact alternating directions method for monotone variational inequalities," *Mathematical Programming*, vol. 92, no. 1, pp. 103–118, 2002.

[21] L. Hou, H. He, and J. Yang, "A partially parallel splitting method for multiple-block separable convex programming with applications to robust PCA," *Computational Optimization and Applications*, vol. 63, no. 1, pp. 273–303, 2016.

[22] M. Sun, Y. Wang, and J. Liu, "Generalized Peaceman-Rachford splitting method for multiple-block separable convex programming with applications to robust PCA," *Calcolo*, vol. 54, no. 1, pp. 77–94, 2017.

[23] M. Sun and Y. Wang, "The conjugate gradient methods for solving the generalized periodic Sylvester matrix equations," *Journal of Applied Mathematics and Computing*, vol. 60, no. 1-2, pp. 413–434, 2019.

[24] M. Sun and J. Liu, "An accelerated proximal augmented Lagrangian method and its application in compressive sensing," *Journal of Inequalities and Applications*, vol. 2017, p. 263, 2017.