LOW RANK APPROXIMATION SOLUTION OF A CLASS OF GENERALIZED LYAPUNOV EQUATION
*

Xuefeng Duan
College of Mathematics and Computational Science, Guilin University of
Electronic Technology, Guilin 541004, P.R. China

Guangxi Colleges and Universities Key Laboratory of Data Analysis and Computation,
Guilin University of Electronic Technology, Guilin 541004, P.R. China
Email: guidian520@126.com

Zhuling Jiang
College of Mathematics and Computational Science, Guilin University of
Electronic Technology, Guilin 541004, P.R. China
Email: 543274306@qq.com

Anping Liao
College of Mathematics and Econometrics, Hunan University, Changsha 410082, P.R. China
Email: liaoap@hnu.edu.cn

Abstract
In this paper, we consider the low rank approximation solution of a generalized Lyapunov equation which arises in the bilinear model reduction. By using the variation principle, the low rank approximation solution problem is transformed into an unconstrained optimization problem, and then we use the nonlinear conjugate gradient method with exact line search to solve the equivalent unconstrained optimization problem. Finally, some numerical examples are presented to illustrate the effectiveness of the proposed methods.


Key words: Generalized Lyapunov equation, Bilinear model reduction, Low rank approximation solution, Numerical method.

1. Introduction

Denoted by $R^{n \times n}$ be the set of $n \times n$ real matrices, $SR^{n \times n}$ be the set of $n \times n$ real symmetric matrices, $SR^{+ \times n}$ be the set of $n \times n$ real symmetric positive definite matrices. We write $B > 0$ ($B \geq 0$) if the matrix $B$ is positive definite (semidefinite). The symbol $B^T$ stands for the transpose of the matrix $B$, and the symbol $\otimes$ stands for the Kronecker product. For the $n \times n$ matrix $B = (b_1, b_2, \cdots, b_n) = (b_{ij})$, $[B]_{ij}$ stands for the element of the $i$th row and $j$th column, that is, $[B]_{ij} = b_{ij}$, and vec($B$) stands for a vector defined by vec($B$) = $(b_1^T, b_2^T, \cdots, b_n^T)^T$. The symbols rank($B$) and tr($B$) stand for the rank and trace of the matrix $B$, respectively. We use $\lambda_1(B)$ and $\lambda_n(B)$ to denote the maximal and minimal eigenvalues of an $n \times n$ symmetric matrix $B$, respectively. We use $\|B\|_F$ to denote the Frobenius norm of a matrix $B$.

In this paper, we consider the low rank approximation solution of the generalized Lyapunov equation

\[ AX + XA^T + \sum_{j=1}^{m} N_j X N_j^T + Q = 0, \]  

(1.1)
where \( A, N_1, N_2, \cdots, N_m \in \mathbb{R}^{n \times n} \), \( Q \) is an \( n \times n \) symmetric semidefinite matrix, and

\[
I_n \otimes A + A \otimes I_n + \sum_{j=1}^{m} N_j \otimes N_j \in S\mathbb{R}^{n^2 \times n^2}
\]

The low rank approximation solution of (1.1) arises in bilinear model reduction, which can be stated as follows (see [4,11,38] for more details). Consider the following bilinear control systems

\[
\begin{align*}
\dot{x}(t) &= Ax(t) + \sum_{j=1}^{m} N_j x(t)u_j(t) + Bu(t), \\
y(t) &= Cx(t), & x(0) = x_0,
\end{align*}
\]

where \( A, N_j \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m} \) and \( C \in \mathbb{R}^{k \times n} \). Let

\[
\begin{align*}
P_1(t_1) &= e^{At_1}B, \\
P_i(t_1, t_2, \cdots, t_i) &= e^{At_i}[N_1 P_{i-1}, \cdots, N_m P_{i-1}], & i = 2, 3, \cdots
\end{align*}
\]

Then the reachability Gramian

\[
P = \sum_{i=1}^{\infty} \int_{0}^{\infty} \cdots \int_{0}^{\infty} P_i P_i^T dt_1 \cdots dt_i
\]

of (1.2) satisfies (1.1) with \( Q = BB^T \).

In the last few years there has been a constantly increasing interest in developing effective numerical methods for the standard Lyapunov equation (i.e. Eq. (1.1) with \( m = 0 \)). The numerical methods can be generally separated into two classes. The first class consists of direct method, such as the Bartels-Stewart method [3] and the Hammarling method [20]. The second class is the iterative method, such as Krylov subspace method [22], ADI method [26], matrix sign function method [6], Smith’s method [34], block successive overrelaxation method [35] and the matrix splitting methods [14]. In particular, Bai [2] presented a HSS iterative method for solving large sparse continuous Sylvester equations with non-Hermitian and positive definite (semidefinite) matrices. Motivated by the classical conjugate direction method for Hermitian positive definite linear systems, Deng, Bai and Gao [7] constructed orthogonal direction methods for solving two classes of linear matrix equations. Some other matrix equations were also studied in [8,9,15,29]. However, when \( m > 0 \), the theory and numerical methods for the generalized Lyapunov equation (1.1) are fewer than the case \( m = 0 \), due to the complicated structure. By means of the linear operator theory and spectral analysis, Damm [10] and Zhang-Chen [37] gave some sufficient conditions for the existence of a positive (semi)definite solution of Eq. (1.1), but how to verify these conditions is difficult. By using the vectorizing operator and Kronecker product, Huang [18,19] transformed Eq. (1.1) into a system of linear equations and derived some sufficient conditions for the existence of a symmetric solution. A parameter iterative method was constructed to compute the symmetric solution, but how to choose the optimal parameter is unknown.

Recent interests on the Lyapunov equation are directed more towards large and sparse coefficients matrices \( A, N_j \) and \( Q = BB^T \) with very low rank, where \( B \) has only a few columns. In this case, the standard methods are often too expensive to be practical, and some low rank iterative methods become more viable choices. Common ones are based on the ADI or Smith method (see [12,25,29]), on Krylov subspace techniques (see [15,25,26]), and on low rank
In this paper, a new numerical method for computing the low rank approximation solution of Eq. (1.1) is developed by minimizing a trace function on the set of $n \times n$ symmetric positive semidefinite solution with rank no more than $k$. By characterizing this set by $X = YY^T, Y \in \mathbb{R}^{n \times k}$, the low rank approximation solution problem is transformed into an equivalent unconstrained optimization problem. Then we use the nonlinear conjugate gradient method with exact line search to solve the equivalent unconstrained optimization problem. Consequently, the low rank approximation solution of Eq. (1.1) is derived.

This paper is organized as follows. In the section 2, a new low rank iterative method is developed to compute the low rank approximation solution of Eq. (1.1). In the section 3, some numerical examples are used to illustrate that the new method is feasible and effective.

2. Low Rank Approximation Solution of Eq. (1.1)

Based on the variation principle, the low rank approximation solution of Eq. (1.1) is transformed into an optimization problem $\min_{X \in \Omega} f(X)$. We characterize the feasible set $\Omega$ by $X = YY^T, Y \in \mathbb{R}^{n \times k}$, and then transform the optimization problem into an unconstrained optimization problem. Finally, we use the nonlinear conjugate gradient method with exact line search to solve it.

By taking vec() to the both sides of Eq. (1.1) we obtain that

$$L \text{vec}(X) = - \text{vec}(Q), \quad L = I_n \otimes A + A \otimes I_n + \sum_{j=1}^{m} N_j \otimes N_j.$$ (2.1)

Since $L \in \mathbb{S}^{n^2 \times n^2}_{+}$, then it is easy to verify that Eq. (1.1) has a unique solution $X_*$, and $X_*$ is symmetric. Here, we assume that the solution $X_*$ has the so-called low rank property, i.e., the eigenvalues of the solution $X_*$ have an exponential decay.

Since solving system (2.1) is equivalent to find the minimizer $X_*$ of the quadratic form

$$\varphi(X) = \frac{1}{2} \text{vec}(X)^T L \text{vec}(X) + \text{vec}(X)^T \text{vec}(Q).$$

Noting that

$$\text{tr}(X^T (AX + XA^T + \sum_{j=1}^{m} N_j X N_j^T)) = \text{vec}(X)^T L \text{vec}(X),$$

$$\text{tr}(X^T Q) = \text{vec}(X)^T \text{vec}(Q),$$

then computing the low rank approximation solution of (1.1) is equivalent to solving the nonlinear optimization problem

$$\min_{X \in \Omega} f(X),$$ (2.2)
where
\[
f(X) = \text{tr}[X^T(AX + XA^T + \sum_{j=1}^{m} N_j XN_j^T)] + 2\text{tr}(X^TQ),
\]
\[
\Omega = \left\{ X \in \mathbb{S}^{n \times n} \mid X \succeq 0, \text{rank}(X) \leq k \right\}.
\]

Since the feasible set \( \Omega \) of (2.2) can be characterized by \( X = ZZ^T, Z \in \mathbb{R}^{n \times k} \), then the nonlinear optimization problem (2.2) can be rewritten as
\[
\min_{Z \in \mathbb{R}^{n \times k}} f(ZZ^T).
\]
Set
\[
g(Z) = f(ZZ^T),
\]
and noting that (2.3) we have
\[
g(Z) = f(ZZ^T)
\]
\[
= \text{tr}\left( (ZZ^T)^T (AZZ^T + ZZ^TA^T + \sum_{j=1}^{m} N_j ZZ^T N_j^T) \right) + 2\text{tr}(ZZ^TQ)
\]
\[
= \text{tr}\left( (ZZ^T)^T (AZZ^T + ZZ^TA^T + \sum_{j=1}^{m} N_j ZZ^T N_j^T + 2Q) \right)
\]
\[
= \text{vec}(ZZ^T)^T \text{vec}(AZZ^T + ZZ^TA^T + \sum_{j=1}^{m} N_j ZZ^T N_j^T + 2Q).
\]
Hence the nonlinear optimization problem (2.2) can be rewritten as
\[
\min_{Z \in \mathbb{R}^{n \times k}} g(Z).
\]
That is to say, if the nonlinear optimization problem (2.6) has a solution \( \hat{Z} \), then \( X = \hat{Z} \hat{Z}^T \) is the solution of problem (2.2), which is also the low rank approximation solution of Eq. (1.1).

Compared with the optimization problem (2.2), the optimization problem (2.6) has the following advantages: (1) The rank constrain of the feasible set \( \Omega \) has been eliminated; (2) The new variable \( Y \in \mathbb{R}^{n \times k} \) amounts to searching a space of dimension \( nk \), which can be much lower than the dimension of \( X \in \mathbb{R}^{n \times n} \) in (2.2). Now we will use the nonlinear conjugate gradient method to solve (2.6). The difficulty for using nonlinear conjugate gradient method to solve (2.6) is how to compute the gradient of the objective function \( g(Z) \). We will overcome this difficulty by making use of the properties of the matrix trace and Taylor’s formula, and the expression for the gradient of the objective function \( g(Z) \) is given as follows.

**Theorem 2.1.** The gradient of the objective function \( g(Z) \) is
\[
\nabla g(Z) = 4 \left( AZZ^T + ZZ^TA^T + \sum_{j=1}^{m} N_j ZZ^T N_j^T + Q \right) Z.
\]

**Proof.** For convenience, we firstly set
\[
T(Z) = AZZ^T + ZZ^TA^T + \sum_{j=1}^{m} N_j ZZ^T N_j^T.
\]
It is easy to verify that $T(Z)$ is symmetric, i.e. $T(Z) = [T(Z)]^T$. For the $n \times k$ incremental matrix $H$, by (2.5), (2.7), and noting that $A$, $N_j$ and $Q$ are symmetric, then we have

$$g(Z + H) = \text{vec}[(Z + H)(Z + H)^T] \text{vec}[A(Z + H)(Z + H)^T]$$

$$+ (Z + H)(Z + H)^T A^T + \sum_{j=1}^{m} N_j(Z + H)(Z + H)^T N_j^T + 2Q]$$

$$= W_4 + W_5 + O(\|H\|^2), \quad (2.8)$$

where

$$W_4 = g(Z) = \text{vec}(ZZ^T) \text{vec} \left( AZZ^T + ZZ^T A^T + \sum_{j=1}^{m} N_j ZZ^T N_j^T + 2Q \right),$$

$$W_5 = \text{vec}(ZZ^T) \left[ \text{vec}(AZH^T) + \text{vec}(AHZ^T) + \text{vec}(ZH^T A^T) \right.$$

$$\left. + \text{vec}(HZ^T A^T) + \text{vec} \left( \sum_{j=1}^{m} N_j ZH^T N_j^T \right) + \text{vec} \left( \sum_{j=1}^{m} N_j HZ^T N_j^T \right) \right]$$

$$+ \text{vec}(ZH^T)^T \text{vec} \left( T(Z) + 2Q \right) + \text{vec}(HZ^T)^T \text{vec}[T(Z) + 2Q]$$

$$= \text{tr}(ZZ^T AZH^T) + \text{tr}(ZZ^T AHZ^T) + \text{tr}(ZZ^T ZH^T A^T)$$

$$+ \text{tr}(ZZ^T HZ^T A^T) + \text{tr} \left( ZZ^T \sum_{j=1}^{m} N_j ZH^T N_j^T \right) + \text{tr} \left( ZZ^T \sum_{j=1}^{m} N_j HZ^T N_j^T \right)$$

$$+ \text{tr}[ZH^T(T(Z) + 2Q)] + \text{tr}[HZ^T(T(Z) + 2Q)]$$

$$= \text{tr} \left\{ [2T(Z) + (T(Z) + 2Q)^T + (T(Z) + 2Q)]ZH^T \right\}$$

$$= \text{tr} \left\{ 4[T(Z) + Q]ZH^T \right\}. \quad (2.10)$$

On the other hand, according to Taylor’s Theorem (P.15, [27]) we have

$$g(Z + H) = g(Z) + [\text{vec}(\nabla g(Z))^T \text{vec}(H) + O(\|H\|^2)]$$

$$= g(Z) + \text{tr}[\nabla g(Z)^T H] + O(\|H\|^2). \quad (2.9)$$

Comparing (2.8) and (2.9), we obtain that

$$\text{tr}[\nabla g(Z)^T H] = W_5 = \text{tr} \left\{ 4[T(Z) + Q]ZH^T \right\}. \quad (2.10)$$

Set $H = e_ie_j^T$, where $e_i$ is the $i$th column of the $n \times n$ identity matrix and $e_j$ is the $j$th column of the $k \times k$ identity matrix, then by (2.10) we have

$$\text{tr}[\nabla g(Z)^T e_ie_j^T] = W_5 = \text{tr} \left\{ 4[T(Z) + Q]Z(e_i e_j^T)^T \right\}. \quad (2.11)$$
Applying $\operatorname{tr}(e_i^T B e_j) = b_{ij} = [B]_{ij}$ to the both sides of (2.11) we obtain

$$
\operatorname{tr}[\nabla g(Z)^T e_i e_j^T] = \operatorname{tr}[e_j^T \nabla g(Z)] = \operatorname{tr}[e_i^T \nabla g(Z)e_j] = [\nabla g(Z)]_{ij}.
$$

(2.12)

$$
\operatorname{tr}\{4[T(Z) + Q]Z(e_j^T e_i^T)\} = \operatorname{tr}[4(T(Z) + Q)Ze_j e_i^T]
= 4\operatorname{tr}\{e_i^T [T(Z) + Q]Ze_j\} = 4[(T(Z) + Q)Z]_{ij}.
$$

(2.13)

Combining (2.11)-(2.13) we have

$$
[\nabla g(Z)]_{ij} = 4[(T(Z) + Q)Z]_{ij},
$$

or equivalently

$$
\nabla g(Z) = 4[T(Z) + Q]Z
= 4\left( A Z Z^T + Z Z^T A^T + \sum_{j=1}^{m} N_j Z Z^T N_j^T + Q \right) Z.
$$

This completes the proof of the lemma. 

The nonlinear conjugate gradient method (see [24] for more details) with exact line search for solving the minimization problem (2.6) can be stated as follows.

\begin{algorithm}
1. Given initial matrix $Z_0 \in \mathbb{R}^{n \times k}$ and error tolerant $\varepsilon > 0$;
2. Evaluate $g_0 = g(Z_0)$, $\nabla g_0 = \nabla g(Z_0)$, $D_0 = -\nabla g(Z_0)$, $k = 0$;
3. While $\|\nabla g_k\|_F > \varepsilon$
    Find $t_k$ such that $g(Z_k + t_k D_k) = \min_{t > 0} g(Z_k + t D_k)$;
    $Z_{k+1} = Z_k + t_k D_k$;
    $\nabla g_{k+1} = \nabla g(Z_{k+1})$;
    $\beta_{k+1} = \frac{\|\nabla g_{k+1}\|_F^2}{\|\nabla g_k\|_F^2}$;
    $D_{k+1} = -\nabla g_{k+1} + \beta_{k+1} D_k$;
end
\end{algorithm}

In the following, several comments can be made on Algorithm 2.1.

(1) Algorithm 2.1 is implemented with exact line search for the step length $t_k$. We can use the exact line search method in Higham-Kim [17] to compute the step length $t_k$, because the univariate function $\phi(\cdot)$ defined by

$$
\phi(t) = g(Z_k + t D_k) = a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0,
$$

where

\[
a_4 = \text{tr}[D_k D_k^T D(D_k)]; \\
a_3 = \text{tr}[D_k D_k^T (A Z_k D_k^T + A D_k Z_k^T + Z_k D_k^T A^T + D_k Z_k^T A^T) \\
+ \sum_{j=1}^m N_j Z_k D_k^T N_j^T + \sum_{j=1}^m N_j D_k Z_k^T N_j^T)]; \\
a_2 = 2\text{tr}[D_k D_k^T T(Z_k)] + 2\text{tr}\left[D_k Z_k^T (A Z_k D_k^T + A D_k Z_k^T + Z_k D_k^T A^T \\
+ D_k Z_k^T A^T + \sum_{j=1}^m N_j Z_k D_k^T N_j^T + \sum_{j=1}^m N_j D_k Z_k^T N_j^T)\right]; \\
a_1 = 4\text{tr}\left[(T(Z_k) + Q) Z_k\right]; \\
a_0 = \text{tr}\left[(Z_k Z_k^T)^T (A Z_k Z_k^T + Z_k Z_k^T A^T + \sum_{j=1}^m N_j Z_k Z_k^T N_j^T + 2Q)\right]
\]

is quadratic, which is similar as the merit function for Newton’s method with exact line search for solving a quadratic matrix equation in Higham-Kim [17].

(2) Note that the exact line search always satisfy the following equality

\[
\text{tr}[\nabla g_{k+1}^T D_k] = \text{vec}(\nabla g_{k+1})^T \text{vec}(D_k) = 0. \quad (2.14)
\]

By applying vec(·) to the equality \(D_{k+1} = -\nabla g_{k+1} + \beta_{k+1} D_k\) in Algorithm 4.1 and premultiplying by [vec(\nabla g_{k+1})]^T we have

\[
[\text{vec}(\nabla g_{k+1})]^T \text{vec}(D_{k+1}) = -||\nabla g_{k+1}||^2_F + \beta_{k+1} \text{vec}(\nabla g_{k+1})^T \text{vec}(D_k). \quad (2.15)
\]

By (2.14) and (2.15) we have

\[
[\text{vec}(\nabla g_{k+1})]^T \text{vec}(D_{k+1}) < 0,
\]

which implies that \(D_{k+1}\) is a descent direction.

(3) By Li-Tong-Wan [24, P.81], we can establish the global convergence theorem for Algorithm 4.1.

**Theorem 2.2.** Suppose that \(g(Z)\) is continuously differentiable and bounded below. If the gradient \(\nabla g(Z)\) is Lipschitz continuous, that is, there exists a constant \(L > 0\) such that

\[
||\nabla g(Z_1) - \nabla g(Z_2)||_F \leq L||Z_1 - Z_2||_F, \quad \forall \ Z_1, Z_2 \in \mathbb{R}^{n \times k}, \quad (2.16)
\]

then the matrix sequence \(\{Z_k\}\) generated by Algorithm 2.1 satisfies

\[
\lim_{k \to \infty} \inf \|\nabla g(Z_k)\|_F = 0. \quad (2.17)
\]

Further analysis on the convergence theorem 2.2 are as follows. Let \(X\) be the solution of Eq. (1.1), that is,

\[
-Q = AX + X A^T + \sum_{j=1}^m N_j X N_j^T, \quad (2.18)
\]
then we have
\[
\| - Q \| = \| AX + XA^T + \sum_{j=1}^{m} N_j XN_j^T \| \leq \left( 2\| A \| + \sum_{j=1}^{m} \| N_j \|^2 \right) \| X \|, \tag{2.19}
\]
which implies,
\[
\| X \| \geq \frac{\| Q \|}{2\| A \| + \sum_{j=1}^{m} \| N_j \|^2}. \tag{2.20}
\]
Therefore we can restrict the feasible set \( \Omega \) of the nonlinear optimization problem (2.2) to
\[
\begin{cases}
X \in S R^{n \times n} \mid X \geq 0, \text{ rank}(X) \leq k, \| X \| \geq \frac{\| Q \|}{2\| A \| + \sum_{j=1}^{m} \| N_j \|^2}
\end{cases} \tag{2.21}
\]
Note that
\[
g(Z) = f(ZZ^T) = f(X) = tr \left[ X^T (AX + XA^T + \sum_{j=1}^{m} N_j XN_j^T) \right] + 2tr(X^T Q)
\]
\[
= tr(X^T AX) + Tr(XAX^T) + \sum_{j=1}^{m} tr(N_j XN_j^T) + 2tr(X^T Q)
\]
\[
= tr(X^T AX) + tr(XAX^T) + \sum_{j=1}^{m} tr \left[ (X^{1/2} N_j X^{1/2}) (X^{1/2} N_j X^{1/2})^T \right] + 2tr(X^{1/2} Q X^{1/2})
\]
\[
\geq 2\lambda_n(A) tr(X^T X) = 2\lambda_n(A) \| X \|^2 \geq \frac{2\lambda_n(A) \| Q \|}{2\| A \| + \sum_{j=1}^{m} \| N_j \|^2}, \tag{2.22}
\]
which shows that \( g(Z) \) is bounded below. Now we begin to discuss the Lipschitz continuity of the gradient \( \nabla g(Z) \). Set
\[
F(Z) = AZZ^T + ZZ^T A^T + \sum_{j=1}^{m} N_j ZZ^T N_j^T + Q. \tag{2.23}
\]
Then for arbitrary \( Z_1, Z_2 \) we have
\[
\| \nabla g(Z_1) - \nabla g(Z_2) \| = \| 4F(Z_1) Z_1 - 4F(Z_2) Z_2 \|
\]
\[
= 4\| F(Z_1) Z_1 - F(Z_1) Z_2 + F(Z_1) Z_2 - F(Z_2) Z_2 \|
\]
\[
\leq 4\| F(Z_1) \| \| Z_1 - Z_2 \| + 4\| F(Z_1) - F(Z_2) \| \| Z_2 \|. \tag{2.24}
\]
We further note that
\[
\| F(Z_1) \| = \| AZ_1 Z_1^T + Z_1 Z_1^T A^T + \sum_{j=1}^{m} N_j Z_1 Z_1^T N_j^T + Q \|
\]
\[
\leq 2\| A \| \| Z_1 \| ^2 + \sum_{j=1}^{m} \| N_j \|^2 \| Z_1 \|^2 + \| Q \|, \tag{2.25}
\]
and

\[ \| F(Z_1) - F(Z_2) \| = \| A Z_1 Z_1^T + Z_1 Z_1^T A^T + \sum_{j=1}^{m} N_j Z_1 Z_1^T N_j^T + Q - A Z_2 Z_2^T - Z_2 Z_2^T A^T - \sum_{j=1}^{m} N_j Z_2 Z_2^T N_j^T - Q \| \]

\[ \leq 2\| A \| \| Z_1 Z_1^T - Z_2 Z_2^T \| + \sum_{j=1}^{m} \| N_j \| \| Z_1 Z_1^T - Z_2 Z_2^T \| \]

\[ = \left( 2\| A \| + \sum_{j=1}^{m} \| N_j \| ^2 \right) \| Z_1 Z_1^T - Z_2 Z_2^T \| \]

\[ = \left( 2\| A \| + \sum_{j=1}^{m} \| N_j \| ^2 \right) \| Z_1 (Z_1^T - Z_2^T) + (Z_1 - Z_2) Z_2^T \| \]

\[ \leq \left( 2\| A \| + \sum_{j=1}^{m} \| N_j \| ^2 \right) \| Z_1 \| \| Z_1^T - Z_2^T \| + \| Z_1 - Z_2 \| \| Z_2^T \| \]

\[ \leq \left( 2\| A \| + \sum_{j=1}^{m} \| N_j \| ^2 \right) (\| Z_1 \| + \| Z_2 \| ) \| Z_1 - Z_2 \|. \] (2.26)

Assume that the Frobenius norm of the solution of Eq. (1.1) has an upper bound \( \mu \), which implies \( \| X \| = \| ZZ^T \| \leq \| Z \|^2 \leq \mu \). Then we can restrict the feasible set \( \Omega \) to

\[ \left\{ X \in SR^{n \times n} \mid X \geq 0, \ \text{rank}(X) \leq k, \ \frac{\| Q \|}{2\| A \| + \sum_{j=1}^{m} \| N_j \| ^2} \leq \| X \| \leq \mu, \right\}. \] (2.27)

Therefore, combining (2.24)-(2.27) we have

\[ \| \nabla g(Z_1) - \nabla g(Z_2) \| \leq L \| Z_1 - Z_2 \|, \]

where

\[ L = 4 \left( 2\| A \| + \sum_{j=1}^{m} \| N_j \| ^2 \right) (\mu + 2\sqrt{\mu}) + \| Q \|. \]

In a word, if the Frobenius norm of the solution of Eq. (1.1) has an upper bound \( \mu \), i.e., the feasible set \( \Omega \) of (2.2) can be restrict to (2.27), then \( g(Z) \) is bounded below and the gradient \( \nabla g(Z) \) is Lipschitz continuous. Consequently, the matrix sequence \( \{ Z_k \} \) generated by Algorithm 2.1 satisfies (2.17),

\[ \lim_{k \to \infty} \inf \| \nabla g(Z_k) \|_F = 0. \]

3. Numerical Experiments

In this section we illustrate the performance of Algorithms 2.1 with some numerical examples. All experiments are performed in MATLAB R2010a on a PC with an Intel Core i3 processor at 2.13GHz with machine precision \( \varepsilon = 2.22 \times 10^{-16} \). For Algorithm 2.1, we denote the relative residual error

\[ R(Z_k) = \left( \| AZ_k Z_k^T + Z_k Z_k^T A^T + \sum_{j=1}^{m} N_j Z_k Z_k^T N_j^T + Q \|_F \right) / \| Q \|_F, \]
and use the practical stopping criterion $R(Z_k) \leq 1.0 \times 10^{-6}$.

**Example 3.1.** Consider the following generalized Lyapunov equation

$$AX + XA^T + \sum_{j=1}^{4} N_j X N_j^T + Q = 0,$$

(3.1)

where

$$A = -0.1 \times [I_m \otimes T_m + T_m \otimes I_m + E_1 \otimes I_m + I_m \otimes E_1 + E_m \otimes I_m + I_m \otimes E_m],$$

$$N_1 = E_1 \otimes I_m, \quad N_2 = I_m \otimes E_1, \quad N_3 = E_m \otimes I_m, \quad N_4 = I_m \otimes E_m,$$

$$Q = BB^T, \quad B = [E_1 \otimes e \quad e \otimes E_1 \quad E_m \otimes e \quad e \otimes E_m],$$

with

$$T_m = \begin{pmatrix}
-2 & 1 & \cdots & 0 & 0 \\
1 & -2 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & -2 & 1 \\
0 & 0 & \cdots & 1 & -2
\end{pmatrix} \in \mathbb{R}^{m \times m}, \quad E_1 = \begin{pmatrix}
1 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0
\end{pmatrix} \in \mathbb{R}^{m \times m}$$

$$E_m = \begin{pmatrix}
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 1
\end{pmatrix} \in \mathbb{R}^{m \times m}, \quad e = \begin{pmatrix}
1 \\
1 \\
\vdots \\
1
\end{pmatrix} \in \mathbb{R}^m.$$

It is easy to verify that $A, N_1, N_2, N_3, N_4, Q \in \mathbb{S}^{m^2 \times m^2}$ and $I_n \otimes A + A \otimes I_n + \sum_{j=1}^{4} N_j \otimes N_j$ is positive definite. Now we use Algorithm 2.1 to compute the low rank approximation $\hat{X} = \hat{Z} \hat{Z}^T$ to the true solution $X_*$ with different set $\Omega$, that is, with different $k$ and $n$ in $\Omega$. Here, $n = m^2$.

The convergence curves of the relative residual error $R(Z_k)$ are presented in Fig. 3.1.

Fig. 3.1 shows that Algorithm 2.1 is feasible to compute the low rank approximation solution $\hat{X} = \hat{Z} \hat{Z}^T$ of Eq. (1.1), whose rank is no more than $k$. 

![Fig. 3.1. Convergence curves of the relative residual error $R(Z_k)$](image-url)
In the following example, we will compare Algorithm 2.1 with LR-ADI method ([25,29]) and Krylov subspace method ([21,10]) which are the popular methods to compute the low rank approximation solution of generalized Lyapunov equations.

Example 3.2. In this example, we will compute the reachability Gramian of the bilinear control system
\[
\dot{x} = Ax + Nux + Bu, \tag{3.2}
\]
which arises from FEM discretization of a heat conduction problem (see [28] for more details). The reachability Gramian corresponding to (4.2) satisfies the generalized Lyapunov equation
\[
AX + XA^T + NN^T + BB^T = 0, \tag{3.3}
\]
where
\[
A = (a_{ij})_{n \times n} = \begin{cases} 
1.6, & i = j; \\
0.3, & |i - j| = 1; \\
0, & \text{else}
\end{cases} \quad N = (n_{ij})_{n \times n} = \begin{cases} 
0.05, & i = j; \\
-0.01, & |i - j| = 1; \\
0, & \text{else}
\end{cases}
\]
\[
B = -A^{-1} \left( \begin{array}{cc}
O_{n_1 \times n_1} & O_{n_1 \times n_2} \\
O_{n_2 \times n_1} & I_{n_2}
\end{array} \right) A^{-1}, \quad n_2 = \frac{n}{10000}, \quad n_1 = n - n_2.
\]

We use Algorithm 2.1 (denoted by “NCGM”), LR-ADI method (denoted by “LRADI”) and Krylov subspace method (denoted by “KSM”) to compute the low rank approximation solution of Eq. (3.3), respectively. Under the same stopping criterion \( R(Z_k) \leq 1.0 \times 10^{-6} \), we list the number of iteration (denoted by “IT”), CPU time (denoted by “CPU”) and the relative error (denoted by “ERR”) in Table 1. Here it is worth to point out that, in principle, the optimization formulated in Algorithm 2.1 suffices to find a low rank approximation if the rank is known in advance. In practice, this rank is unknown, so we constantly increase \( k \) until the stopping criterion is satisfied.

<table>
<thead>
<tr>
<th>( n )</th>
<th>NCGM</th>
<th>LRADI</th>
<th>KSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000</td>
<td>IT</td>
<td>58</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>15.6</td>
<td>20.3</td>
</tr>
<tr>
<td></td>
<td>ERR</td>
<td>( 6.36 \times 10^{-7} )</td>
<td>( 2.10 \times 10^{-7} )</td>
</tr>
<tr>
<td>30000</td>
<td>IT</td>
<td>137</td>
<td>149</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>40.7</td>
<td>51.9</td>
</tr>
<tr>
<td></td>
<td>ERR</td>
<td>( 9.45 \times 10^{-7} )</td>
<td>( 3.81 \times 10^{-7} )</td>
</tr>
<tr>
<td>70000</td>
<td>IT</td>
<td>220</td>
<td>287</td>
</tr>
<tr>
<td></td>
<td>CPU</td>
<td>194.2</td>
<td>298.4</td>
</tr>
<tr>
<td></td>
<td>ERR</td>
<td>( 3.11 \times 10^{-7} )</td>
<td>( 2.89 \times 10^{-7} )</td>
</tr>
</tbody>
</table>

From Table 3.1 we can see that: (1) Algorithm 2.1 outperforms LR-ADI method in iteration steps and CPU time, so our algorithm has faster convergence rate than LR-ADI method, but LR-ADI method has higher approximating accuracy than our algorithm; (2) When the size \( n \) of the coefficient matrices is very large, Algorithm 2.1 slightly outperforms Krylov subspace
method in iteration steps and CPU time, so our algorithm is more effective than Krylov subspace method. The reason is that the search space of our Algorithm is smaller than Krylov subspace method.

4. Conclusion

In this paper, we study the low rank approximation solution of the generalized Lyapunov equation which arises in the model order reduction of bilinear control systems. We first transform this problem into an unconstrained optimization problem, and then develop a new numerical method to compute the low rank approximation solution of this equation. Numerical examples show that new methods are feasible and effective. It is worth to point out that the numerical method of this paper is different from the Iterative Orthogonal Direction (IOD) method in Deng, et al. [7]. In this paper, the low rank approximation solution of Eq. (1.1) is transformed into the minimization problem of a trace function on the set of $n \times n$ symmetric positive semidefinite solution with rank no more than $k$, and the nonlinear conjugate gradient method is used to solve the equivalent problem. However, Deng, Bai and Gao [7] used the conjugate gradient method to the normal equation of $AXB = C$, which is the IOD method. The two numerical methods are different in essence. The IOD method can’t use to compute the low rank approximation solution of Eq. (1.1).

Acknowledgements. Part of this work was done when the first author was visiting the Department of Mathematics, College of William and Mary. He is grateful for providing good working conditions. We also thank to Prof. Chi-Kwong Li for discussing the low rank iterative methods of matrix equations. The authors wish to thank the Editor and anonymous referees for providing very useful suggestions to improve this paper.

The work was supported by the National Natural Science Foundation of China (No. 11561015; 11271117; 11261014; 11301107; 11361018), the Natural Science Foundation of Guangxi Province (No.2012GXNSFBA053006; 2014GXNSFFA118001; 2014GXNSFAA118004), and the Innovation Project of GUET Graduate Education (No. 2016YJCX45).

References


