J. Appl. Math. Comput. (2016) 50:227–241
DOI 10.1007/s12190-015-0867-9

ORIGINAL RESEARCH

A new linearized implicit iteration method for nonsymmetric algebraic Riccati equations

Huaize Lu · Changfeng Ma

Received: 6 August 2014 / Published online: 3 February 2015

Abstract For the nonsymmetric algebraic Riccati equation, we establish a new linearized implicit iteration method (LI) for computing its minimal nonnegative solution. And a modified linearized implicit iteration method (MLI) is obtained through Shamanskii technique. Under suitable conditions, we prove the monotone convergence of the LI and MLI iteration methods. Numerical experiments show that the LI and MLI iteration methods are feasible and effective. Moreover, the MLI iteration method outperforms the alternately linearized implicit iteration method (in: Bai et al., Numer. Linear Algebr. Appl. 13:655–674, 2006).

Keywords Nonsymmetric algebraic Riccati equation · Minimal nonnegative solution · Linearized implicit method · Modified linearized implicit method · Shamanskii technique

Mathematics Subject Classification 15A24 · 65F10 · 65H10

1 Introduction

We study numerical solution of the nonsymmetric algebraic Riccati equation (NARE)

$$R(X) = XCX - XD - AX + B = 0, \tag{1}$$

H. Lu · C. Ma (✉)
School of Mathematics and Computer Science, Fujian Normal University,
Fuzhou 350007, People's Republic of China
e-mail: macf@fjnu.edu.cn

H. Lu
e-mail: luhz2013@163.com

where A, B, C and D are real matrices of sizes $m \times m, m \times n, n \times m$ and $n \times n$, respectively.

We define the matrix

$$K = \begin{pmatrix} D & -C \\ -B & A \end{pmatrix}. \tag{2}$$

When K is a nonsingular M-matrix or an irreducible M-matrix, the NARE (1) has a minimal nonnegative solution [10,15]. The NARE (1) that arises in applied probability and transportation theory has been studied for years, see [1,3,4,6,7,17,22,23, 25,26,29,31] and the references therein. Numerical methods for finding the minimal nonnegative solution of the NARE (1) include the alternately linearized implicit iteration method[2], the basic fixed-point iteration method and the Newton iteration method [14], the Schur method [11], the structure-preserving doubling algorithm [16] and the alternating-directional doubling algorithm [30]. For more other methods see [5,9,12,13,18–21,24,28] and the references therein.

Recently, Bai [2] has shown that the alternately linearized implicit iteration method (ALI) was a feasible and effective solver for the NARE (1). Besides, ALI has comparable computing cost and fast convergence rate. Based on the ALI, we establish a linearized implicit iteration method (LI) for computing the minimal nonnegative solution of the NARE (1). Taking advantage of the simple structure of LI iteration method and the idea of Shamanskii technic [27], we get a modified linear implicit iteration method (MLI), which has less computing cost and faster convergence speed than ALI iteration method. Under suitable conditions, we prove the monotone convergence of the LI and MLI iteration methods. Numerical experiments show that LI is a feasible and effective iteration method, and can perform as well as the ALI iteration method. Besides, the MLI iteration method can outperform the ALI iteration method.

This paper is organized as follows. In Sect. 2, we introduce some necessary notations and lemmas. We establish the LI and MLI iteration methods in Sect. 3. In Sect. 4, we prove the monotone convergence of the LI and MLI iteration methods. Numerical results are given in Sect. 5. Finally, we draw a brief conclusion in Sect. 6.

2 Notations and lemmas

Throughout this paper, we use the following definitions and notations. I_n denotes the identity matrix with dimension n. For two matrices $A = (a_{ij}), B = (b_{ij}) \in \Re^{m\times n}$, we write $A \geq B (A > B)$ if $a_{ij} \geq b_{ij} (a_{ij} > b_{ij})$ holds for all $1 \leq i \leq m$ and $1 \leq j \leq n$. A matrix $A \in \Re^{m\times n}$is said to be nonnegative (positive) if its entries satisfy $a_{ij} \geq 0 (a_{ij} > 0)$ for all $1 \leq i \leq m$ and $1 \leq j \leq n$. A matrix $A \in \Re^{m\times n}$ is said to be a Z-matrix if all of its off-diagonal elements are non-positive. It follows that any Z-matrix A can be written as the form $A = sI - B$, with s a positive real number and B a non-negative matrix. A Z-matrix $A \in \Re^{n\times n}$ is called an M-matrix if $s \geq \rho(B)$, where $\rho(B)$ denotes the spectral radius of B. It is called a nonsingular M-matrix if $s > \rho(B)$ and a singular M-matrix if $s = \rho(B)$. $\|\cdot\|_\infty$ denotes the ∞-norm of a matrix.

The following lemmas describe several important properties about a nonsingular M-matrix.

Lemma 1 (see [2]) *Let $A \in \Re^{n\times n}$ be a Z-matrix. Then the following statements are equivalent:*

(1) *A is a nonsingular M-matrix;*
(2) *$A^{-1} \geq 0$;*
(3) *$Av > 0$ for some positive vector $v \in \Re^n$;*
(4) *All eigenvalues of A have positive real parts.*

Lemma 2 (see [2]) *Let $A \in \Re^{n\times n}$ be a nonsingular M-matrix. If the matrix $B = (b_{ij}) \in \Re^{n\times n}$ satisfies*

$$b_{ii} \geq a_{ii}, \quad a_{ij} \leq b_{ij} \leq 0, \quad i \neq j, \quad 1 \leq i, j \leq n$$

then B also is a nonsingular M-matrix. In particular, for any positive real $\theta, B = \theta I + A$ is a nonsingular M-matrix.

Lemma 3 (see [2]) *Let $A, B \in \Re^{n\times n}$ be nonsingular M-matrices satisfying $A \leq B$, Then $A^{-1} \geq B^{-1}$*

Lemma 4 (see [2]) *If K in (2) is a nonsingular M-matrix, then the NARE (1) has a minimal nonnegative solution S such that both matrices $D - CS$ and $A - SC$ are nonsingular M-matrices. Moreover, any solution S^* of the NARE (1) such that both matrices $D - CS^*$ and $A - S^*C$ are nonsingular M-matrices must satisfy $S^* = S$.*

3 The LI iteration method and MLI iteration method

The ALI iteration method [2] has comparable computing cost and fast convergence rate for solving the NARE (1), whose advantage over the fixed-point iteration method is that the ALI may has faster convergence rate and better numerical behaviour as it is more implicit and has exploited more information from the nonlinear term XCX.

We have known the following alternately linearized implicit iteration method for solving the NARE (1): Set $X_0 = 0 \in \Re^{m\times n}$, for $k = 0, 1, 2, \ldots$ until $\{X_k\}$ convergence, compute X_{k+1} from X_k by solving the following two systems of linear matrix equations:

$$X_{k+\frac{1}{2}}(\alpha I + (D - CX_k)) = (\alpha I - A)X_k + B, \tag{3}$$

$$\left(\alpha I + (A - X_{k+\frac{1}{2}}C)\right) X_{k+1} = X_{k+\frac{1}{2}}(\alpha I - D) + B, \tag{4}$$

where $\alpha \geq \max\{\max_{1\leq i\leq m}\{a_{ii}\}, \max_{1\leq j\leq n}\{d_{jj}\}\}$.

Motivated by the ALI iteration method, we present a linearized implicit iteration method by reformulating the NARE (1) as the following equation:

$$(\alpha I + (A - XC))\, X = X(\alpha I - D) + B, \tag{5}$$

where α is a given positive parameter.

Then the following iteration equation is built:

$$(\alpha I + (A - X_k C))\, X_{k+1} = X_k(\alpha I - D) + B, \tag{6}$$

where α is a given positive iteration parameter.

The linearized implicit iteration method only needs to solve a linear matrix equation at each iteration step. Hence, it has comparable computing cost and fast convergence rate. For $m = n$, the computing cost at each iteration step of the LI iteration method is $\frac{20}{3}n^3 + \mathcal{O}(n^2)$ [8]. The linearized implicit iteration is similar to the ALI iteration, but it has a more simple structure. So the LI iteration method may be a good numerical algorithm for solving the NARE (1) in practical.

We find that the Eq. (6) is similar to the Eq. (4) which is the subproblem of the ALI iteration method. Obviously, we can get another LI iteration method: for given X_k, by solving X_{k+1} from

$$X_{k+1}\,(\alpha I + (D - CX_k)) = (\alpha I - A)X_k + B \tag{7}$$

where α is a given positive iteration parameter.

By observing the structure of (6), the most of computing cost at each iteration step is spent on computing the inverse of $(\alpha I + (A - X_k C))$. To further reduce the computing cost, we derived a modified LI iteration method (MLI) by using Shamanskii technique.

Hence, we have some following iteration equalities: for given $s \geq 1$ and $k = 0, 1, 2, \ldots,$

$$\begin{aligned} X_{k,1} &= (\alpha I + (A - X_{k,0}C))^{-1}(X_{k,0}(\alpha I - D) + B), \\ X_{k,q+1} &= (\alpha I + (A - X_{k,0}C))^{-1}(X_{k,q}(\alpha I - D) + B), \quad 1 \leq q \leq s-1, \\ X_{k+1} &= X_{k,s}, \end{aligned} \tag{8}$$

where $X_{k,0} = X_k$.

The algorithm of the modified linearized implicit iteration method:

Algorithm 1 *(The modified linearied implicit iteration)*

Step1 Input matrices A, B, C and D. Choose parameters α, ε, and s such that

$$\alpha \geq \max\left\{\max_{1\leq i\leq m}\{a_{ii}\},\ \max_{1\leq j\leq n}\{d_{jj}\}\right\},\ \varepsilon = 10^{-12},\ s \geq 1. \text{ Set } k := 0;$$

Step2 Compute $R(X_k)$, if $\|R(X_k)\|_F \leq \varepsilon$, stop. Otherwise, go to step3.
Step3 Get X_{k+1} by solving the iteration the equalities (8).
Step4 Set k:=k+1 and go to step2.

Remark 1 In general, the optimal parameter s is difficult to be determined. So we set $s = 4$ or $s = 6$ in the MLI iteration method to reduce computing cost.

4 Convergence analysis

Lemma 5 *Suppose that S is the minimal nonnegative solution of the NARE (1). Matrix sequence $\{X_k\}$ is generated by LI iteration method. Then the following equalities hold true:*

(a) $(\alpha I + (A - X_k C))(X_{k+1} - S) = (X_k - S)(\alpha I - (D - CS))$
(b) $(\alpha I + (A - X_k C))(X_{k+1} - X_k) = R(X_k)$
(c) $R(X_{k+1}) = (X_{k+1} - X_k)(\alpha I - (D - CX_{k+1}))$

Proof (a) Making use of (6), we get

$$\begin{aligned}(\alpha I + (A - X_k C))(X_{k+1} - S) &= (\alpha I + (A - X_k C))X_{k+1} - (\alpha I + (A - X_k C))S \\ &= X_k(\alpha I - D) + B - (\alpha I + (A - X_k C))S \\ &= \alpha(X_k - S) - X_k D + X_k CS + B - AS. \end{aligned} \tag{9}$$

Since S is the minimal nonnegative solution of the NARE (1), we obtain

$$R(S) = SCS - SD - AS + B = 0,$$

i.e.,

$$B - AS = SD - SCS.$$

By substituting this identity into the Eq. (9), we immediately obtain

$$\begin{aligned}(\alpha I + (A - X_k C))(X_{k+1} - S) &= \alpha(X_k - S) - X_k D + X_k CS + SD - SCS \\ &= (X_k - S)(\alpha I - (D - CS)).\end{aligned}$$

(b) Equality (b) follows from straightforward computations. In fact, it holds that

$$\begin{aligned}(\alpha I + (A - X_k C))(X_{k+1} - X_k) &= (\alpha I + (A - X_k C))X_{k+1} - (\alpha I + (A - X_k C))X_k \\ &= X_k(\alpha I - D) + B - \alpha X_k - AX_k + X_k C X_k \\ &= -X_k D + B - AX_k + X_k C X_k \\ &= R(X_k).\end{aligned}$$

(c) Making use of (6), we have

$$\begin{aligned}\alpha X_{k+1} + AX_{k+1} - X_{k+1}CX_{k+1} &= (\alpha I + (A - X_k C))X_{k+1} - (X_{k+1} - X_k)CX_{k+1} \\ &= X_k(\alpha I - D) + B - (X_{k+1} - X_k)CX_{k+1} \\ &= \alpha X_k - X_k D + B - (X_{k+1} - X_k)CX_{k+1}.\end{aligned}$$

It then follows that

$$\begin{aligned} R(X_{k+1}) &= X_{k+1}CX_{k+1} - AX_{k+1} - X_{k+1}D + B \\ &= \alpha X_{k+1} - \alpha X_k + (X_{k+1} - X_k)CX_{k+1} - X_{k+1}D \\ &= (X_{k+1} - X_k)(\alpha I - (D - CX_{k+1})). \end{aligned}$$

So far, we have completed the whole proof. □

Lemma 6 *Suppose that S is a solution of the NARE (1), the matrix subsequence $\{X_{k,0}\}$ and $\{X_{k,1}\}$ are generated by MLI iteration method. Then the following equalities hold true:*

(a) $(\alpha I + (A - X_{k,0}C))(X_{k,1} - S) = (X_{k,0} - S)(\alpha I - (D - CS))$;
(b) $(\alpha I + (A - X_{k,0}C))(X_{k,1} - X_{k,0}) = R(X_{k,0})$;
(c) $R(X_{k,1}) = (X_{k,1} - X_{k,0})(\alpha I - (D - CX_{k,1}))$.

Proof The proof of this remark is analogous to that of Lemma (5). □

Theorem 1 *Suppose that the matrix K defined in (2) is a nonsingular M-matrix, and S is the minimal nonnegative solution of the NARE (1). The initial matrix $X_0 = 0$ and α is a prescribed iteration parameter such that*

$$\alpha \geq \max\left\{ \max_{1\leq i\leq m}\{a_{ii}\}, \max_{1\leq j\leq n}\{d_{jj}\}\right\}$$

where a_{ii} and d_{ii} are the ith diagonal elements of the matrices A and D, respectively. Then the matrix sequence $\{X_k\}$ generated by LI iteration method is well defined, and it holds that

(a) *$\{X_k\}$ is bounded and monotone increasing, i.e.,*

$$0 \leq X_0 \leq X_1 \leq \cdots \leq X_k \leq X_{k+1} \leq \cdots \leq S$$

(b) *$\{X_k\}$ is convergent to S, i.e., $\lim_{k\to\infty} X_k = S$.*

Proof Notice that K is a nonsingular M-matrix, its diagonal matrices A and D are nonsingular M-matrix and $B, C \geq 0$. Hence, when $\alpha \geq \max\left\{ \max_{1\leq i\leq m}\{a_{ii}\}, \max_{1\leq j\leq n}\{d_{jj}\}\right\}$, the matrix $\alpha I - D$ is a nonnegative matrix.

The result (a) is equivalent to the following conclusion:

$$0 \leq X_k \leq X_{k+1} \leq S, \quad R(X_k) \geq 0, \quad k = 0, 1, 2 \cdots .$$

We can prove above conclusion by induction. When $k = 0$, we have $R(X_0) = B \geq 0$. And by substituting $X_0 = 0$ into (6) and lemma 5 (a), we get

$$(\alpha I + A)X_1 = B \text{ and } (\alpha I + A)(X_1 - S) = (-S)(\alpha I - (D - CS)).$$

As the matrix A is a nonsingular M-matrix, the matrix $\alpha I + A$ also is a nonsingular M-matrix from Lemma 2. So

$$X_1 = (\alpha I + A)^{-1} B \geq 0 = X_0$$

and

$$X_1 - S = -(\alpha I + A)^{-1} S(\alpha I - (D - CS)) \leq 0.$$

These show that $0 \leq X_0 \leq X_1 \leq S$ and $R(X_0) \geq 0$.

Assume that the conclusion holds for $k = l - 1$, i.e., $0 \leq X_{l-1} \leq X_l \leq S$ and $R(X_{l-1}) \geq 0$. Since $C \geq 0$, it then follows that

$$A - SC \leq A - X_l C \leq A.$$

By making use of Lemma 2, we know that $A - X_l C$ and $\alpha I + (A - X_l C)$ are both nonsingular M-matrices. Then from lemma 5, we have

$$\begin{aligned}
(X_{l+1} - S) &= (\alpha I + (A - X_l C))^{-1} (X_l - S)(\alpha I - (D - CS)),\\
X_{l+1} - X_l &= (\alpha I + (A - X_l C))^{-1} R(X_l),\\
R(X_l) &= (\alpha I + (A - X_{l-1} C))(X_l - X_{l-1}).
\end{aligned}$$

Notice that $(\alpha I + (A - X_l C))^{-1} \geq 0$ and $\alpha I - (D - CS) \geq 0$, it's easily to get that

$$0 \leq X_l \leq X_{l+1} \leq S \quad \text{and} \quad R(X_l) \geq 0.$$

Hence, we have proved the result (a) by induction.

Now we come to prove (b). Because $\{X_k\}$ is nonnegative, monotonically increasing, and bounded from above, there exists a nonnegative matrix S^* such that $\lim_{k \to \infty} X_k = S^*$. Obviously, the above result implies $S^* \leq S$. On the other hand, by taking limits in (6), we see that S^* also is a nonnegative solution of NARE (1). Hence, it must hold that $S \leq S^*$ due to the minimal property of S. It then follows that $S^* = S$. □

We have proved that the matrix sequence $\{X_k\}$ generated by LI iteration method converges monotonically to the minimal nonnegative of the NARE (1). In the sequel, the convergence of MLI iteration method will be analyzed.

Theorem 2 *Suppose that K defined in (2) is a nonsingular M-matrix, S is the minimal nonnegative solution of the NARE (1). Set the initial matrix $X_0 = 0$ and α is a prescribed iteration parameter such that*

$$\alpha \geq \max \left\{ \max_{1 \leq i \leq m} \{a_{ii}\}, \max_{1 \leq j \leq n} \{d_{jj}\} \right\}$$

where a_{ii} and d_{ii} are the ith diagonal elements of the matrices A and D, respectively. For $k = 0, 1, 2, \ldots, s \geq 1$, matrix sequence $\{X_{k,q}\}(0 \leq q \leq s)$ generated by the MLI *iteration method, if* $0 \le X_{k,0} \le S$, $R(X_{k,0}) \ge 0$, *then*

$$0 \le X_{k,q-1} \le X_{k,q} \le S,\ \ R(X_{k,q}) \ge 0\ holds\ \ for\ 1 \le q \le s.$$

Proof Notice that K is a nonsingular M-matrix, whose main diagonal blocks A and D are nonsingular M-matrix and $B, C \ge 0$. Therefore, for $\alpha \ge \max\left\{ \max_{1 \le i \le m} \{a_{ii}\}, \max_{1 \le j \le n} \{d_{jj}\} \right\}$, the matrix $\alpha I - D$ also is a nonnegative matrix.

Since $0 \le X_{k,0} \le S$ and $C \ge 0$, we have

$$A - SC \le A - X_{k,0}C \le A.$$

Thus, $A - X_{k,0}C$ and $\alpha I + (A - X_{k,0}C)$ are both nonsingular M-matrices from Lemma 2 and Lemma 4. By the MLI iteration method and Lemma 6, we obtain that

$$\begin{aligned} X_{k,1} - X_{k,0} &= (\alpha I + (A - X_{k,0}C))^{-1} R(X_{k,0}), \\ X_{k,1} - S &= (\alpha I + (A - X_{k,0}C))^{-1} (X_{k,0} - S)(\alpha I - (D - CS)), \\ R(X_{k,1}) &= (X_{k,1} - X_{k,0})(\alpha I - (D - CX_{k,1})), \end{aligned}$$

for given $k > 0$.

Since $0 \le X_{k,0} \le S$, $R(X_{k,0}) \ge 0$, we get

$$\begin{aligned} &X_{k,1} - X_{k,0} \ge 0,\ i.e.,\ X_{k,1} \ge X_{k,0}, \\ &X_{k,1} - S \le 0,\ i.e.,\ X_{k,1} \le S. \end{aligned}$$

and

$$R(X_{k,1}) \ge 0.$$

Hence, the conclusion holds true for $q = 1$.

Assume that the conclusion holds true for all $q \le l - 1$ $(2 \le l \le s + 1)$, i.e.,

$$0 \le X_{k,q-1} \le X_{k,q} \le S,\ \ R(X_{k,q}) \ge 0\ holds\ \ for\ 1 \le q \le l - 1.$$

By MLI iteration method, we get

$$\left(\alpha I + (A - X_{k,0}C)\right) X_{k,l} = X_{k,l-1}(\alpha I - D) + B,$$

i.e.,

$$X_{k,l} = \left(\alpha I + (A - X_{k,0}C)\right)^{-1} (X_{k,l-1}(\alpha I - D) + B).$$

As $\alpha I - D \geq 0$, $B \geq 0$ and $X_{k,l-1} \geq X_{k,l-2}$, we have

$$X_{k,l} \geq \left(\alpha I + (A - X_{k,0}C)\right)^{-1} (X_{k,l-2}(\alpha I - D) + B) = X_{k,l-1}.$$

In addition,

$$\begin{aligned}
(\alpha I + \left(A - X_{k,0}C\right)) (X_{k,l} - S) &= \left(\alpha I + (A - X_{k,0}C)\right) X_{k,l} - \left(\alpha I + (A - X_{k,0}C)\right) S \\
&= X_{k,l-1}(\alpha I - D) - \alpha S + X_{k,0}CS + B - AS \\
&= \alpha(X_{k,l-1} - S) - X_{k,l-1}D + X_{k,0}CS + SD - SCS \\
&\leq \alpha(X_{k,l-1} - S) - X_{k,l-1}D + X_{k,l-1}CS + SD - SCS \\
&= (X_{k,l-1} - S)(\alpha I - (D - CS)) \leq 0.
\end{aligned}$$

Hence, $0 \leq X_{k,l-1} \leq X_{k,l} \leq S$.

According to the MLI iteration method, we get

$$\begin{aligned}
\alpha X_{k,l} + AX_{k,l} - X_{k,l}CX_{k,l} &= \left(\alpha I + (A - X_{k,0}C)\right) X_{k,l} - (X_{k,l} - X_{k,0})CX_{k,l} \\
&= X_{k,l-1}(\alpha I - D) + B - (X_{k,l} - X_{k,0})CX_{k,l} \\
&= \alpha X_{k,l-1} - X_{k,l-1}D + B - (X_{k,l} - X_{k,0})CX_{k,l}
\end{aligned}$$

As $X_{k,l-1} \geq X_{k,0}$, it follows that

$$\begin{aligned}
R(X_{k,l}) &= X_{k,l}CX_{k,l} - AX_{k,l} - X_{k,l}D + B \\
&= \alpha X_{k,l} - \alpha X_{k,l-1} + (X_{k,l} - X_{k,0})CX_{k,l} - X_{k,l}D + X_{k,l-1}D \\
&\geq (X_{k,l} - X_{k,l-1}) \left(\alpha I - (D - CX_{k,l})\right) \\
&\geq 0
\end{aligned}$$

By the principle of induction, $0 \leq X_{k,q-1} \leq X_{k,q} \leq S$ and $R(X_{k,q}) \geq 0$ holds for $1 \leq q \leq s$. □

Theorem 3 *Suppose that the matrix K defined in (2) is a nonsingular M-matrix and S is the minimal nonnegative solution of the NARE (1). Let the initial matrix $X_0 = 0$ and α is a prescribed iteration parameter such that*

$$\alpha \geq \max\left\{ \max_{1 \leq i \leq m} \{a_{ii}\}, \max_{1 \leq j \leq n} \{d_{jj}\} \right\}$$

where a_{ii} and d_{ii} are the ith diagonal elements of the matrices A and D, respectively. Then the matrix sequence $\{X_k\}$ generated by MLI iteration method is well defined, and it holds that

(a) $0 \leq X_0 \leq X_1 \leq \cdots \leq X_k \leq S$ *and* $R(X_k) \geq 0$

(b) *X_k is convergent to S, i.e.,* $\lim_{k\to\infty} X_k = S$.

Proof According to the MLI iteration method, we know that X_k is equivalent to $X_{k-1,s}$ or $X_{k,0}$. Since $X_0 = 0 \leq S$ and $R(X_0) = B \geq 0$, we get $S \geq X_1 = X_{0,s} \geq X_{0,0} = X_0$ and $R(X_1) = R(X_{0,s}) \geq 0$ from Theorem 2. Hence, the conclusion (a) is true for $k = 1$.

Assume that the conclusion (a) is true for $k = l - 1$, i.e.,

$$0 \leq X_0 \leq X_1 \leq \cdots \leq X_{l-1} \leq S \ \ and \ \ R(X_{l-1}) \geq 0.$$

Taking advantage of the Theorem 2, it's easily to get that

$$S \geq X_l = X_{l-1,s} \geq X_{l-1,0} = X_{l-1} \ \ and \ \ R(X_l) = R(X_{l-1,s}) \geq 0,$$

which shows that $0 \leq X_0 \leq X_1 \leq \cdots \leq X_l \leq S$ and $R(X_l) \geq 0$.

By the principe of induction, the result (a) is true.

We now turn to prove the conclusion (b). Because matrix sequence $\{X_k\}$ is nonnegative, monotonically increasing, and bounded from above, there exists a nonnegative matrix S^* such that $\lim_{k\to\infty} X_k = S^*$. Obviously, the above result implies $S^* \leq S$. On the other hand, by taking limits in (6), we see that S^* is also a nonnegative solution of NARE (1). Hence, it must hold that $S \leq S^*$ due to the minimal property of S. It then follows that $S^* = S$. □

We have completed the convergence analysis of the LI and MLI iteration methods. In addition, if matrix sequences $\{X_k\}$ is generated by MLI iteration method and $\{\widetilde{X}_k\}$ is generated by LI iteration method, we can see that $X_k \leq \widetilde{X}_k$ from the proof of Theorems 1, 2 and 3. Therefore, the convergence rate of the MLI iteration method is faster than the LI iteration method.

5 Numerical results

In this section, we use three examples to show the numerical feasibility and effectiveness of the LI and MLI iteration methods. The numerical behaviours of the LI and MLI iteration methods will be compared with the ALI iteration method with respect to the number of iteration steps (IT), the computing times (CPU) and the relative residual error (denoted by ERR, where ERR$= \|\Re(X_k)\|_\infty / \|\Re(X_0)\|_\infty$).

All implementations are run in MATLAB R2010b (7.11) on a personal computer CORE i5. In actual computations, each iteration is terminated when the current iterate satisfies ERR$< 10^{-12}$.

Example 1 (see [2]) We consider the NARE (1), for which

$$A = D = \mathrm{Tridiag}(-I, T, -I) \in \Re^{n\times n}$$

Table 1 Numerical results of Example 1 when $m = 16,\ n = 256$

Method		ALI	LI	MLI ($s = 4$)	MLI ($s = 6$)
$\xi = 0.2$	IT	44	87	22	15
	CPU	1.4535	1.8776	0.8211	0.7503
	RES	5.25e−13	7.28e−13	5.32e−13	2.81e−13
$\xi = 0.5$	IT	44	87	22	15
	CPU	1.5224	1.9529	0.838655	0.7840
	RES	5.52e−13	7.66e−13	5.72e−13	3.12e−13
$\xi = 1.0$	IT	44	87	22	15
	CPU	1.4636	1.8421	0.8307	0.7851
	RES	5.98e−13	8.28e−13	6.41e−13	3.65e−13

are block tridiagonal matrices,

$$B = \frac{1}{50}\mathrm{tridiag}(1, 2, 1) \in \Re^{n\times n}$$

is a tridiagonal matrix and $C = \xi B$, where ξ is a positive constant such that K defined in (2) is a nonsingular M-matrix. Here,

$$T = \mathrm{tridiag}\left(-1, 4 + \frac{200}{(m+1)^2}, -1\right) \in \Re^{m\times m}$$

and $n = m^2$.

We take $m = 16$, $X_0 = 0$ and set $\alpha = 4 + \frac{200}{(m+1)^2}$. The numerical results for this example are listed in table 1.

From Table 1, we see that all iterations can converge rapidly to the exact minimal non-negative solution of NARE (1) with high accuracy. According to the iteration step, the MLI ($s = 6$) iteration method is the least. Besides, the iteration steps of the ALI iteration method are almost half of the LI iteration method and twice of the MLI ($s = 4$) iteration method. According to the computing times, the MLI ($s = 6$) iteration method also is the least and the LI iteration method is the longest for the increasing of iteration steps. According to the iteration equalities (3), (4), (6) and (8), the computing cost of ALI is twice of the LI iteration method and more than the MLI iteration method. Therefore, with respect to the computing efficiency, the LI iteration method is as well as the ALI iteration method and the MLI iteration method outperforms the ALI and LI iteration methods in large matrices computation. Moreover, when ξ or C becomes large, the iteration numbers and computing time of the LI and MLI iteration methods are almost fixed. This shows that the LI and MLI iteration methods could successfully solve the NARE (1). Finally, the MLI ($s = 6$) iteration method is the best in this example.

Table 2 Numerical comparisons about the mentioned algorithms for Example 2

Method		ALI	LI	MLI ($s = 4$)	MLI ($s = 6$)
$\xi = 0.2$	IT	9	18	7	7
	CPU	0.4143	0.4815	0.3546	0.428982
	RES	7.47e−13	7.47e−13	1.29e−13	5.77e−14
$\xi = 0.5$	IT	10	19	9	9
	CPU	0.4753	0.4879	0.4100	0.5001
	RES	1.88e−13	8.10e−13	7.52e−14	4.90e−14
$\xi = 1.0$	IT	11	21	11	11
	CPU	0.5108	0.5337	0.4686	0.6479
	RES	2.25e−13	8.40e−13	2.97e−13	2.27e−13

Example 2 (see [2]) Consider the NARE (1), for which

$$A = D = \begin{pmatrix} 3 & -1 & & \\ & 3 & \ddots & \\ & & \ddots & -1 \\ & & & 3 \end{pmatrix} \in \Re^{n \times n}, \quad B = I_n \ \ and \ \ C = \xi I_n$$

with $\xi > 0$ a given constant. Here, I_n is the $n \times n$ identity matrix.

We take $n = 256$ and $X_0 = 0$, and change the problem parameter ξ in our implementations. The numerical results for this example are listed in Table 2.

From Table 2, we observe that all iterations can converge rapidly to the exact minimal nonnegative solution of the NARE (1) with high accuracy. According to the iteration step, the MLI ($s = 4, 6$) iteration method is the least. Besides, the iteration steps of the LI iteration method are twice of the ALI iteration method and the iteration steps of the MLI ($s = 4, 6$) iteration method are nearly to the ALI iteration method. According to the computing times, the MLI ($s = 4$) iteration method is almost the least and the MLI ($s = 6$) iteration method is almost longest. With respect to the computing efficiency, the LI iteration method is as well as the ALI iteration method. Therefore, the MLI ($s = 4$) iteration method outperforms the ALI and LI iteration methods in large matrices computation. Moreover, when ξ or C becomes large, the iteration numbers and computing time of the LI and MLI ($s = 4, 6$) iteration methods are almost fixed. This shows that the LI and MLI iteration methods could successfully solve the NARE (1).

Example 3 (see [2]) Consider the NARE (1), for which A, B, C and D are generated according to the following rule: first, generate and save a random 1000×1000 nonzero matrix R by using rand (1000, 1000); then set $W = \text{diag}(Re) - R$, with $e = (1, 1, \ldots, 1)^T \in \Re^{1000}$; and finally, for a given positive constant κ, define

$$D = W(1:500, 1:500) + \kappa I, \quad A = W(501:1000, 501:1000) + \kappa I,$$
$$B = -W(501:1000, 501:1000), \quad C = -\xi W(1:500, 501:1000)$$

Table 3 Numerical comparisons about the mentioned algorithms for Example 3 with κ=10

Method		ALI	LI	MLI($s=4$)	MLI($s=6$)
$\xi = 0.2$	IT	15	30	11	11
	CPU	3.6381	4.1863	2.8072	3.4487
	RES	1.04e−13	8.20e−13	1.23e−13	1.03e−13
$\xi = 0.5$	IT	19	37	15	16
	CPU	4.0860	5.1156	3.8298	4.1852
	RES	4.18e−13	9.45e−13	5.88e−13	4.50e−13
$\xi = 1.0$	IT	50	97	49	46
	CPU	11.6078	14.4417	11.7070	11.5908
	RES	7.81e−13	9.86e−13	9.57e−13	7.08e−13

where ξ is a positive constant. Note that the so-generated matrix W is a nonsingular M-matrix, and A, B, C and D are 500×500 matrices with A, D being non singular M-matrices and B, C being nonnegative matrices,respectively.

We take $X_0 = 0$, $\kappa = 10$. The numerical results for this example are listed in Table 3.

From Table 3, we see that all iterations can converge rapidly to the exact minimal nonnegative solution of the NARE (1) with high accuracy. According to the iteration step, the MLI ($s = 4, 6$) iteration method is the least. Besides, the iteration steps of the LI iteration method are almost twice of the ALI iteration method again. According to the computing times, the MLI ($s = 4$) iteration method or the MLI ($s = 6$) iteration method is the least. With respect to the computing efficiency, the LI iteration method is as well as the ALI iteration method. The MLI ($s = 4, 6$) and ALI iteration methods outperform the LI iteration methods in large matrices computation. Moreover, when ξ or C becomes large, the iteration numbers and computing time of the LI and MLI ($s = 4, 6$) iteration methods are almost fixed. This shows that the LI and MLI iteration methods could successfully solve the NARE (1) again.

6 Conclusions

Theoretical analysis and numerical implementations have shown that the LI and MLI iteration methods are feasible and effective solvers for the NARE (1). The LI iteration method is as well as the ALI iteration method. The iteration steps of the LI iteration method may be twice of the ALI iteration method according to iteration equalities (3), (4) and (6). In addition, the MLI iteration method has further reduced the computing cost and improved the convergence rate of the LI iteration method. According to the above three examples, the MLI ($s = 6$) and MLI ($s = 4$) iteration methods can do better in medium-sized matrices. Even though setting $\alpha \geq \max\{\max_{1\leq i\leq m}\{a_{ii}\}, \max_{1\leq j\leq n}\{d_{jj}\}\}$ and $s = 4, 6$ can get a good MLI iteration methods, the choice of a practically optimal parameter α and s are considerably difficult in the viewpoints of both theory and application. Finally, we know that $\alpha \geq \max_{1\leq i\leq n}\{d_{ii}\}$ also can keep the convergence property of the LI and MLI iteration methods according to the Sect. 4.

Acknowledgments The project is supported by National Natural Science Foundation of China (Grant No. 11071041) and Fujian Natural Science Foundation (Grant No.2013J01006).

References

1. De Moor, B., David, J.: Total linear least squares and the algebraic Riccati equation. Syst. Control Lett. **18**, 329–337 (1992)
2. Bai, Z.Z., Guo, X.X., Xu, S.F.: Alternately linearized implicit iteration methods for the minimal nonnegative solutions of the nonsymmetric algebraic Riccati equations. Numer. Linear Algebr. Appl. **13**(8), 655–674 (2006)
3. Bernstein, D., Haddad, W.: LQG control with an H performance bound: a Riccati equation approach. IEEE Trans. Autom. Control **34**, 293–305 (1989)
4. Clancey, K., Gohberg, I.: Factorization of matrix functions and singular integral operators. In operator theory: advances and applications, vol. 3. Birkhauser Verlag, Basel (1981)
5. Gao, Y.H., Bai, Z.Z.: On inexact Newton methods based on doubling iteration scheme for nonsymmetric algebraic Riccati equations. Numer. Linear Algebr. Appl. **18**(3), 325–341 (2011)
6. Gohberg, I., Kaashoek, M.: An inverse spectral problem for rational matrix functions and minimal divisibility. Integral Equ. Oper. Theory **10**, 437–465 (1987)
7. Gohberg, I., Rubinstein, S.: Proper contractions and their unitary minimal completions. In operator theory: advances and applications, vol. 33. Birkhauser, Basel (1988)
8. Golub, G., Van Loan, C.: Matrix Comput., 3rd edn. The Johns Hopkins University Press, Baltimore/London (1996)
9. Guo, C.H.: A new class of nonsymmetric algebraic Riccati equations. Linear Algebr. Appl. **426**(2), 636–649 (2007)
10. Guo, C.H.: Nonsymmetric algebraic Riccati equations and Wiener–Hopf factorization for M-matrices. SIAM J. Matrix Anal. Appl. **23**, 225–242 (2001)
11. Guo, C.H.: Efficient methods for solving a nonsymmetric algebraic Riccati equation arising in stochastic fluid models. J. Comput. Appl. Math. **192**(2), 353–373 (2006)
12. Guo, C.H., Higham, N.: Iterative solution of a nonsymmetric algebraic Riccati equation. SIAM J. Matrix Anal. Appl. **29**, 396–412 (2007)
13. Guo, C.H., Iannazzo, B., Meini, B.: On the doubling algorithm for a (shifted) nonsymmetric algebraic Riccati equation. SIAM J. Matrix Anal. Appl. **29**, 1083–1100 (2007)
14. Guo, C.H., Laub, A.J.: On the iterative solution of a class of nonsymmetric algebraic Riccati equations. SIAM J. Matrix Anal. Appl. **22**(2), 376–391 (2000)
15. Guo, X.X., Bai, Z.Z.: On the minimal nonnegative solution of nonsymmetric algebraic Riccati equation. J. Comput. Math. **23**, 305–320 (2005)
16. Guo, X.X., Lin, W.W., Xu, S.F.: A structure-preserving doubling algorithm for nonsymmetric algebraic Riccati equation. Numer. Math. **103**(3), 393–412 (2006)
17. Hinrichsen, D., Kelb, B., Linnemann, A.: An algorithm for the computation of the structured complex stability radius. Automatica **25**, 771–775 (1989)
18. Huang, N., Ma, C.F.: The inversion-free iterative methods for solving the nonlinear matrix equation. Abstract and applied analysis, p. 7. Hindawi Publishing Corporation, New York (2013). Article ID 843785
19. Huang, N., Ma, C.F.: Some predictor-corrector-type iterative schemes for solving nonsymmetric algebraic Riccati equations arising in transport theory. Numer. Linear Algebr. Appl. (2014). Published online in Wiley Online Library (wileyonlinelibrary.com). doi:10.1002/nla.1932
20. Huang, N., Ma, Cf: The modified conjugate gradient methods for solving a class of the generalized coupled Sylvester-transpose matrix equations. Comput. Math. Appl. **67**, 1545–1558 (2014)
21. Juang, J.: Existence of algebraic matrix Riccati equations arising in transport theory. Linear Algebr. Appl. **230**, 89–100 (1995)
22. Juang, J., Lin, W.W.: Nonsymmetric algebraic Riccati equations and Hamiltonian-like matrices. SIAM J. Matrix Anal. Appl. **20**, 228–243 (1999)
23. Lancaster, P., Rodman, L.: Solutions of the continuous and discrete-time algebraic Riccati equations: a review. In: Bittanti, S., Laub, A.J., Willems, J.C. (eds.) The Riccati equation. Springer, Berlin (1991)
24. Lin, W.W., Xu, S.F.: Convergence analysis of structure-preserving doubling algorithms for Riccati-type matrix equations. SIAM J. Matrix Anal. Appl. **28**, 26–39 (2006)

25. Petersen, I.: Disturbance attenuation and H-optimization: a design method based on the algebraic Riccati equation. IEEE Trans. Autom. Control **32**, 427–429 (1987)
26. Van der Schaft, A.: L2-gain and passivity techniques in nonlinear control, 2nd edn. Springer, London (2000)
27. Shamanskii, V.E.: A modification of Newton's method. Ukr. Math. J. **19**, 133–138 (1967)
28. Smith, R.A.: Matrix equation XA + BX = C. SIAM J. Appl. Math. **16**, 198–201 (1968)
29. Tadmor, G.: Worst-case design in the time domain: the maximum principle and the standard H problem. Math. Control Signals Syst. **3**, 301–324 (1990)
30. Wang, W., Wang, W., Li, R.C.: Alternating-directional doubling algorithm for M-matrix algebraic Riccati equations. SIAM J. Matrix Anal. Appl. **33**(1), 170–194 (2012)
31. Zhou, K.M., Khargonekar, P.: An algebraic Riccati equation approach to H optimization. Syst. Control Lett. **11**, 85–91 (1988)