mathematics

Article

On the Iterative Methods for the Solution of Three Types of Nonlinear Matrix Equations

Ivan G. Ivanov [1,*,†] and Hongli Yang [2,†]

[1] Faculty of Economics and Business Administration, Sofia University "St.Kl.Ohridski", 1000 Sofia, Bulgaria
[2] College of Mathematics and Systems Science, Shandong University of Science and Technology, Qingdao 266590, China; yanghongli@sdust.edu.cn
* Correspondence: i_ivanov@feb.uni-sofia.bg
† These authors contributed equally to this work.

Abstract: In this paper, we investigate the iterative methods for the solution of different types of nonlinear matrix equations. More specifically, we consider iterative methods for the minimal nonnegative solution of a set of Riccati equations, a nonnegative solution of a quadratic matrix equation, and the maximal positive definite solution of the equation $X + A^*X^{-1}A = Q$. We study the recent iterative methods for computing the solution to the above specific type of equations and propose more effective modifications of these iterative methods. In addition, we make comments and comparisons of the existing methods and show the effectiveness of our methods by illustration examples.

Keywords: Riccati equation; nonlinear matrix equation; M-matrix; minimal nonnegative solution

MSC: 15A24; 15A45; 65F10; 65F35

check for updates

Citation: Ivanov, I.G.; Yang, H. On the Iterative Methods for the Solution of Three Types of Nonlinear Matrix Equations. *Mathematics* **2023**, *11*, 4436. https://doi.org/10.3390/math11214436

Academic Editors: Carlo Bianca and Ioannis K. Argyros

Received: 12 September 2023
Revised: 8 October 2023
Accepted: 23 October 2023
Published: 26 October 2023

1. Introduction

Nonlinear matrix equations are commonly used in many fields of scientific and engineering computing. Research on the existence and properties of the solution to the matrix equations, as well as the corresponding numerical methods, has important theoretical significance and practical value. In this paper, we focus on the iterative methods for the solution of different types of nonlinear matrix equations. More specifically, we consider the iterative methods for computing the minimal nonnegative solution of a set of Riccati equations, a nonnegative solution of the quadratic matrix equation, and the maximal positive definite solution of the equation $X + A^*X^{-1}A = Q$. Iterative methods for solving the nonlinear matrix equation, which avoid the calculation of the inverse matrix at each iteration step, have gained wide popularity [1,2]. Without commenting on the reliability of this approach, this is an efficient approach that speeds up convergence. Users of similar methods should be aware of the possibility that some of these methods may lose accuracy during the calculations and may not reach the result. Moreover, we study the recent iterative methods for computing the above specific type of equations and propose more effective modifications of these iterative methods.

The investigated equations can be encountered in various applied tasks, for example, in the solution of problems for stability analysis [3,4]. There have been many published papers on the field of matrix iterative schemes and their applications. We cite some of them related to our investigation [5–8].

We will exploit a class of nonnegative matrices for the first two equations. Some notations are made throughout this paper. A matrix is nonnegative if all entries are either greater than zero or equal to zero. The set of real $r \times n$ matrices is denoted as $\mathbf{R}^{r\times n}$. The notations I or I_r are used for an unit $r \times r$ matrix. We need an elementwise order relation. The inequality $A \geq B (A > B)$ for $A = (a_{ij}), B = (b_{ij})$ means that $a_{ij} \geq b_{ij} (a_{ij} > b_{ij})$ for all indexes i and j. A matrix $A = (a_{ij}) \in \mathbf{R}^{p\times p}$ is said to be a Z-matrix if it has non-positive off-diagonal elements. A Z-matrix Q has the presentation $Q = \gamma I - P$, with P being a nonnegative matrix. Each M-matrix is a Z-matrix if $\gamma \geq \rho(P)$, where $\rho(P)$ is the spectral radius of P. A Z-matrix Q is called a non-singular M-matrix if $\gamma > \rho(P)$; otherwise, it is a singular M-matrix. For the third equations, we will exploit a class of Hermitian matrices. If the matrix Q is positive definite, we write $Q \succ 0$ or $Q \succeq 0$ for positive semidefinite. Therefore, $P \succeq Q$ means the matrix $P - Q$ is positive semidefinite.

2. Numerical Methods for the Solution of a Set of Riccati Equations

In this section, we investigate different iterative methods to compute the minimal nonnegative solution of a set of matrix Riccati equations, where the matrix coefficients of each equation are associated with an M-matrix. Our investigation follows the ideas of Bai and coauthors in [9], Ma and Lu [10], Guan and Lu [11], Guan [12], and Ivanov and Yang [13]. In fact, we propose a new modification of the alternate linear implicit method introduced in [14] and modified in [15]. We propose a different iterative method to compute the minimal nonnegative solution and derive convergence properties of the new iteration. We apply some properties of M-matrices in the proof and show that the new iteration method is faster than Newton's method investigated in [16] by Liu, Zhang, and Luo.

Consider a set of nonsymmetric coupled Riccati equations (SNCRE) associated with M-matrices:

$$\mathcal{M}_i(X_1, \ldots, X_q) := X_i C_i X_i - X_i D_i - A_i X_i + B_i + \sum_{j \neq i} e_{ij} X_j = 0, \tag{1}$$

$i = 1, \ldots, q,$, which is introduced in [14]. The coefficients of matrix X_i are $A_i = (a^i_{kp}) \in R^{m \times m}, B_i \in R^{m \times n}, C_i \in R^{n \times m}, D_i = (d^i_{kp}) \in R^{n \times n}$. Let $(X_1, \ldots, X_s)$ be a solution of the set of Equation (1) with $X_i \in R^{m \times n}, i = 1, \ldots, q$. Entries of $E = (e_{ij})$ are nonnegative constants.

The couple of matrices $(\tilde{X}_1, \ldots, \tilde{X}_q)$ is the minimal nonnegative solution to (1) if $\tilde{X}_i \leq X_i, i = 1, \ldots, q$ (elementwise order) for any nonnegative solution $(X_1, \ldots, X_q)$ to (1).

Zhang and Tan [14] have investigated the inexact Newton method and the alternate linear implicit method (ALI) to compute the minimal nonnegative solution of the SNCRE (1). They have proved the convergence properties of these iterations. We define the ALI iterative method with initial matrices $X_i^{(0)} = 0 \in R^{n \times n} (m = n)$. The method uses positive constants $\gamma_i, i = 1, \ldots, q$, which are computed via ((31), [14]):

$$\gamma_i = \max\{max_j a^i_{jj},\ max_j d^i_{jj}\}. \tag{2}$$

$$\begin{aligned} &\text{for } k = 0, 1, 2, \ldots : \\ &Y_i^{(k)}(\gamma_i I_n + D_i - C_i X_i^{(k)}) = (\gamma_i I_n - A_i) X_i^{(k)} + B_i + \textstyle\sum_{j \neq i} e_{ij} X_j^{(k)}, \\ &(\gamma_i I_n + A_i - Y_i^{(k)} C_i) X_i^{k+1} = Y_i^{(k)}(\gamma_i I_n - D_i) + B_i + \textstyle\sum_{j \neq i} e_{ij} Y_j^{(k)}. \end{aligned} \tag{3}$$

Iteration (3) computes two inverse matrices at each iteration step. In order to avoid the computation of inverse matrices at each step, we have proposed a modification, as in [15]:

$$\begin{aligned} &X_i^{(0)} = 0, i = 1, 2, \ldots q, \\ &k = 0, 1, 2, \ldots : \\ &Y_i^{(k)}(\gamma_i I_n + D_i) = (\gamma_i I_n - A_i + X_i^{(k)} C_i) X_i^{(k)} + B_i + \textstyle\sum_{j \neq i} e_{ij} X_j^{(k)}, \\ &(\gamma_i I_n + A_i) X_i^{k+1} = Y_i^{(k)}(\gamma_i I_n - D_i + C_i Y_i^{(k)}) + B_i + \textstyle\sum_{j \neq i} e_{ij} Y_j^{(k)}. \end{aligned} \tag{4}$$

Iteration (4) and its convergence properties are derived in [15]. Here, the computations of the inverse matrices of $(\gamma_i I_n + D_i)$ and $(\gamma_i I_n + A_i)$ are executed in the beginning of the iterative process, i.e., it operates only one time. This fact significantly reduces the computational cost throughout the iteration process, which is confirmed by the numerical experiments executed in [15].

2.1. Newton Method and Its Modifications

Authors Liu, Zhang, and Luo [16] investigated the Newton method to compute the positive minimal solution to the set of Riccati Equation (1).

$$\begin{aligned}
&X_i^{(0)} = 0, i = 1,2,\dots q, \\
&k = 0,1,2,\dots : \\
&(A_i - X_i^{(k)}C_i)X_i^{(k+1)} + X_i^{(k+1)}(D_i - C_iX_i^{(k)}) = B_i + \\
&+ \textstyle\sum_{j\neq i} e_{ij}X_j^{(k)} - C_iX_i^{(k)}C_i.
\end{aligned} \tag{5}$$

Together with (5), the following modifications are studied by the same authors:

$$\begin{aligned}
&X_i^{(0)} = 0, i = 1,2,\dots q, \\
&k = 0,1,2,\dots : \\
&(A_i - X_i^{(k)}C_i)X_i^{(k+1)} + X_i^{(k+1)}(D_i - C_iX_i^{(k)}) = B_i + \\
&+ \textstyle\sum_{j<i} e_{ij}X_j^{(k+1)} + \sum_{j>i} e_{ij}X_j^{(k)} - C_iX_i^{(k)}C_i.
\end{aligned} \tag{6}$$

and

$$\begin{aligned}
&X_i^{(0)} = 0, i = 1,2,\dots q, \\
&k = 0,1,2,\dots : \\
&(A_i - X_i^{(k)}C_i)X_i^{(k+1)} + X_i^{(k+1)}(D_i - C_iX_i^{(k)}) = B_i + \\
&+ \textstyle\sum_{j<i} e_{ij}(\omega X_j^{(k+1)} + (1-\omega)X_j^{(k)}) + \sum_{j>i} e_{ij}X_j^{(k)} - C_iX_i^{(k)}C_i.
\end{aligned} \tag{7}$$

The proof of the convergence for (5) is derived by Liu, Zhang, and Luo in [16], whereas the iterations (6) and (7) are used by them as an empirical experiment. The idea of applying approximation $X_i^{(k+1)}$ for computing $X_j^{(k+1)}, j > i$, as in (6), is an effective one.

2.2. Our New Iteration Scheme and Convergence Proof

Here, we propose the following iteration strategy to compute the minimal nonnegative solution to (1):

$$\begin{aligned}
&X_i^{(0)} = 0, \quad i = 1,2,\dots q \\
&\gamma_i \text{ as in } (2), \quad i = 1,2,\dots q \\
&k = 0,1,2,\dots :, \quad 0 \le \omega \\
&Y_i^{(k)}(\gamma_i I + D_i) = (\gamma_i I - A_i + X_i^{(k)}C_i)X_i^{(k)} + B_i \\
&\qquad + \textstyle\sum_{j<i} e_{ij}[\omega Y_j^{(k)} + (1-\omega)X_j^{(k)}] + \sum_{j>i} e_{ij}X_j^{(k)}, \\
&(\gamma_i I + A_i)X_i^{(k+1)} = Y_i^{(k)}(\gamma_i I - D_i + C_iY_i^{(k)}) + B_i \\
&\qquad + \textstyle\sum_{j<i} e_{ij}[\omega X_j^{(k+1)} + (1-\omega)Y_j^{(k)}] + \sum_{j>i} e_{ij}Y_j^{(k)}.
\end{aligned} \tag{8}$$

We derive several matrix identities for matrices obtained by iteration (8) in the lemma.

Lemma 1. *The matrix sequences* $\{X_i^{(k)}, Y_i^{(k)}\}_{k=0}^{\infty}$ *are constructed by iteration (8) with initial values* $X_i^{(0)} = 0, i = 1, 2, ..q$. *The following matrix identities are satisfied for* $k = 0, 1, \ldots, \infty$*:*

$$
\begin{aligned}
(i)\ \ (Y_i^{(k)} - X_i^{(k)})(\gamma_i I + D_i) &= (X_i^{(k)} - Y_i^{(k-1)})(\gamma_i I - D_i) \\
&+ X_i^{(k)} C_i (X_i^{(k)} - Y_i^{(k-1)}) + (X_i^{(k)} - Y_i^{(k-1)}) C_i Y_i^{(k-1)} \\
&+ \textstyle\sum_{j<i} e_{ij}[\omega(X_j^{(k+1)} - X_j^{(k)}) + (1-\omega)(Y_j^{(k)} - X_j^{(k)})] \\
&+ \textstyle\sum_{j>i} e_{ij}(X_j^{(k)} - Y_j^{(k)}), \\
(ii)\ (\gamma_i I + A_i)(X_i^{(k+1)} - Y_i^{(k)}) &= (\gamma_i I - A_i)(Y_i^{(k)} - X_i^{(k)}) \\
&+ Y_i^{(k)} C_i (Y_i^{(k)} - X_i^{(k)}) + (Y_i^{(k)} - X_i^{(k)}) C_i X_i^{(k)} \\
&+ \textstyle\sum_{j<i} e_{ij}[\omega(X_j^{(k+1)} - Y_j^{(k)}) + (1-\omega)(Y_j^{(k)} - X_j^{(k)})] \\
&+ \textstyle\sum_{j>i} e_{ij}(Y_j^{(k)} - X_j^{(k)}),
\end{aligned}
$$

where I is an identity $n \times n$ *matrix.*

Moreover, if $(\tilde{X}_1, \ldots, \tilde{X}_q)$ *is an exact nonnegative solution of* $\mathcal{M}_i(X_1, \ldots, X_q) = 0$*, the subsequent identities can be verified:*

$$
\begin{aligned}
(iii)\ \ (\tilde{X}_i - Y_i^{(k)})(\gamma_i I + D_i) &= (\gamma_i I - A_i)(\tilde{X}_i - X_i^{(k)}) \\
&+ X_i^{(k)} C_i (\tilde{X}_i - X_i^{(k)}) + (\tilde{X}_i - X_i^{(k)}) C_i \tilde{X}_i \\
&+ \textstyle\sum_{j<i} e_{ij}[\omega(\tilde{X}_j - Y_j^{(k)}) + (1-\omega)(\tilde{X}_j - X_j^{(k)})] \\
&+ \textstyle\sum_{j>i} e_{ij}(\tilde{X}_j - X_j^{(k)}) \\
(iv)\ (\gamma_i I + A_i)(\tilde{X}_i - X_i^{(k+1)}) &= (\tilde{X}_i - Y_i^{(k)})(\gamma_i I - D_i) \\
&+ (\tilde{X}_i - Y_i^{(k)}) C_i \tilde{X}_i + Y_i^{(k)} C_i (\tilde{X}_i - Y_i^{(k)}) \\
&+ \textstyle\sum_{j<i} e_{ij}[\omega(\tilde{X}_i - X_j^{(k+1)}) + (1-\omega)(\tilde{X}_i - Y_j^{(k)})] \\
&+ \textstyle\sum_{j>i} e_{ij}(\tilde{X}_i - Y_j^{(k)})
\end{aligned}
$$

Proof. The proof is completed by direct calculations and matrices manipulations. We rewrite Equation (8) for $X_i^{(k)}$ and consider the difference $Y_i^{(k)}(\gamma I + A_i) - (\gamma I_{2n} + D_i) X_i^{(k)}$. After some matrix calculations, we obtain the matrix identity (i). Subtracting matrix equations in (8), we derive (ii). □

We prove the convergence of the matrix sequence generated by (8).

Theorem 1. *Suppose the matrix coefficients* A_i, D_i *of (1) are Z-matrices and* $B_i, C_i, (i = 1, \ldots, q)$ *are nonnegative. There must exist positive scalars* γ_i *such that* $(\gamma_i\, I + A_i)$ *and* $(\gamma_i\, I + D_i)$ *are nonsingular M-matrices.*

If there exits a nonnegative solution to set of matrix Equation (1), then the matrix sequences $\{X_i^{(k)}, Y_i^{(k)}\}_{k=0}^{\infty}$ $i = 1, \ldots, q$ *generated by (8) satisfy the following monotonicity property:*

(i) $\hat{X}_i \geq X_i^{(k+1)} \geq Y_i^{(k)} \geq X_i^{(k)}$ *for* $i = 1, \ldots, q$, $k = 0, 1, \ldots$ *for an exact nonnegative solution* $(\hat{X}_1, \ldots, \hat{X}_q)$ *of (1). Moreover, the same matrix sequences converge to the nonnegative minimal solution of (1).*

(ii) Moreover, if $A_i - \hat{X}_i C_i$ *and* $D_i - C_i \hat{X}_i, i = 1, \ldots, q$ *are nonsingular M-matrices, then* $A_i - \tilde{X}_i C_i$ *and* $D_i - C_i \tilde{X}_i, i = 1, \ldots, q$ *are nonsingular M-matrices, i.e., matrices* $-A_i + \tilde{X}_i C_i$ *and* $-D_i + C_i \tilde{X}_i, i = 1, \ldots, q$ *are c-stable.*

Proof. Under assumptions that we have $(\gamma_i I + A_i)^{-1} \geq 0$ and $(\gamma_i I + D_i)^{-1} \geq 0, i = 1, \ldots, q$. Apply recurrence Equation (8) with $X_1^{(0)} = \ldots = X_q^{(0)} = 0$ and γ_i computed by (2).

For $Y_1^{(0)}$, we have $Y_1^{(0)}(\gamma_1 I + D_1) = B_1 \geq 0$ and $Y_1^{(0)} = B_1(\gamma_i I + D_1)^{-1} \geq 0$. For $Y_2^{(0)}$, we have $Y_2^{(0)}(\gamma_2 I + D_2) = B_2 + e_{21}\omega Y_1^{(0)} \geq 0$. Thus, $Y_2^{(0)} \geq 0$. Therefore $Y_i^{(0)} \geq 0$, and $Y_i^{(0)} \geq X_i^{(0)} = 0, i = 1, \ldots, q$.

Construct matrix sequences $\{X_i^{(k)}, Y_i^{(k)}\}_{k=0}^{\infty}$ $i = 1, \ldots, q$ by (8) and exploit the facts $\gamma_i I - D_i \geq 0$ and $\gamma_i I - A_i \geq 0, i = 1, \ldots, q$.

We assume that the inequalities are true $X_i^{(p)} \geq Y_i^{(p-1)} \geq X_i^{(p-1)} \geq 0$ for some integer p.

Next, we prove that $X_i^{(p+1)} \geq Y_i^{(p)} \geq X_i^{(p)} \geq 0, i = 1, \ldots, q$.

Taking into account of Lemma 1(i), we get:

$$(Y_i^{(p)} - X_i^{(p)}) = F_i^{(p)}\,(\gamma_i I + D_i)^{-1} \geq 0\,,$$

because

$$\begin{aligned} F_i^{(p)} := {} & (X_i^{(p)} - Y_i^{(p-1)})(\gamma_i I - D_i) \\ & + X_i^{(p)} C_i (X_i^{(p)} - Y_i^{(p-1)}) + (X_i^{(p)} - Y_i^{(p-1)}) C_i Y_i{}^{(p-1)} \\ & + \textstyle\sum_{j<i} e_{ij} [\omega(X_j^{(p+1)} - X_j^{(p)}) + (1-\omega)(Y_j^{(p)} - X_j^{(p)})] \\ & + \textstyle\sum_{j>i} e_{ij} (X_j^{(p)} - Y_j^{(p)}) \geq 0 \ . \end{aligned}$$

Note that:

$$\omega(X_j^{(p+1)} - X_j^{(p)}) + (1-\omega)(Y_j^{(p)} - X_j^{(p)}) = (Y_j^{(p)} - X_j^{(p)}) + \omega(X_j^{(p+1)} - Y_j^{(p)}) \geq 0.$$

Thus, $\omega(X_j^{(p+1)} - X_j^{(p)}) + (1-\omega)(Y_j^{(p)} - X_j^{(p)}) \geq 0$ for positive ω and all j.

Therefore, $Y_i^{(p)} - X_i^{(p)} \geq 0,\ i = 1, \ldots, q$.

Taking account of Lemma 1(ii), we have:

$$(X_i^{(p+1)} - Y_i^{(p)}) = (\gamma_i I + A_i)^{-1}\, G_i^{(p)},$$

where

$$\begin{aligned} G_i^{(p)} = {} & (\gamma_i I - A_i)(Y_i^{(p)} - X_i^{(p)}) \\ & + Y_i^{(p)} C_i (Y_i^{(p)} - X_i^{(p)}) + (Y_i^{(p)} - X_i^{(p)}) C_i X_i^{(p)} \\ & + \textstyle\sum_{j<i} e_{ij} [\omega(X_j^{(p+1)} - Y_j^{(p)}) + (1-\omega)(Y_j^{(p)} - X_j^{(p)})] \\ & + \textstyle\sum_{j>i} e_{ij} (Y_j^{(p)} - X_j^{(p)}) \geq 0. \end{aligned}$$

Thus, $X_i^{(p+1)} - Y_i^{(p)} \geq 0,\ i = 1, \ldots, q$.

We conclude that the matrix sequences $\{X_i^{(k)}, Y_i^{(k)}\}_{k=0}^{\infty}$ are monotone increasing. We have to prove that they are bonded above. Consider any exact nonnegative solution $(\hat{X}_1, \ldots, \hat{X}_q)$ of (1). We shall prove that the solution is an upper bound of the matrix sequences.

For $k = 0$, we have $\hat{X}_i \geq X_i^{(0)} = 0$. We compute $Y_i^{(0)}, i = 1 \ldots, q$, and by (Lemma 1(iii)):

$$\hat{X}_i - Y_i^{(0)} = (Q_i^{(0)} + S_i^{(0)})\,(\gamma_i I + D_i)^{-1},$$

where

$$Q_i^{(0)} = (\gamma_i I - A_i)(\hat{X}_i - X_i^{(0)}) + X_i^{(0)} C_i (\hat{X}_i - X_i^{(0)}) + (\hat{X}_i - X_i^{(0)}) C_i \hat{X}_i,$$

and

$$\begin{aligned} S_i^{(0)} = {} & \textstyle\sum_{j<i} e_{ij} [\omega(\tilde{X}_j - Y_j^{(0)}) + (1-\omega)(\tilde{X}_j - X_j^{(0)})] \\ & + \textstyle\sum_{j>i} e_{ij} (\tilde{X}_j - X_j^{(0)}). \end{aligned}$$

Note that $Q_i^{(0)} \geq 0,\ i = 1, \ldots, q$.
Moreover, for i = 1 we have $\hat{X}_1 - Y_1^{(0)} \geq 0$, because:

$$S_1^{(0)} = \sum_{j>1} e_{1j}(\hat{X}_j - X_j^{(0)}) \geq 0.$$

For $i = 2$, we obtain:

$$S_2^{(0)} = e_{21}[\omega(\hat{X}_1 - Y_1^{(0)}) + (1-\omega)(\hat{X}_1 - X_1^{(0)})] \\ + \sum_{j>2} e_{2j}(\hat{X}_j - X_j^{(0)}) \geq 0\,.$$

Thus, $\hat{X}_2 - Y_2^{(0)} \geq 0$. We conclude that $\hat{X}_j - Y_j^{(0)} \geq 0\ j = 1, \ldots, q$.
Thus:

$$\hat{X}_j \geq Y_j^{(0)} \geq X_j^{(0)} \geq 0,\ j = 1, \ldots, q.$$

We will prove:

$$\hat{X}_j \geq X_j^{(1)},\ j = 1, \ldots, q.$$

From Lemma 1(iv), for $k = 0$, we obtain:

$$\hat{X}_i - X_i^{(k+1)} = (\gamma_i I + A_i)^{-1}\,(G_{Yi}^{(0)} + L_i^{(0)}),$$

where

$$G_{Yi}^{(0)} = (\tilde{X}_i - Y_i^{(0)})(\gamma_i I - D_i) + (\hat{X}_i - Y_i^{(0)})C_i\hat{X}_i + Y_i^{(0)}C_i(\hat{X}_i - Y_i^{(0)}),$$

which is a nonnegative matrix. For the matrix $L_i^{(0)}$, write:

$$L_i^{(0)} = \sum_{j<i} e_{ij}[\omega(\hat{X}_j - X_j^{(1)}) + (1-\omega)(\hat{X}_j - Y_j^{(0)})] + \sum_{j>i} e_{ij}(\hat{X}_j - Y_j^{(0)}).$$

For $i = 1$, we obtain:

$$L_1^{(0)} = \sum_{j>1} e_{1j}(\hat{X}_j - Y_j^{(0)}) \geq 0,$$

and thus $\hat{X}_1 - X_1^{(1)} \geq 0$. For $i = 2$, write:

$$L_2^{(0)} = e_{21}[\omega(\tilde{X}_1 - X_1^{(1)}) + (1-\omega)(\hat{X}_1 - Y_1^{(0)})] + \sum_{j>2} e_{2j}(\hat{X}_j - Y_j^{(0)}) \geq 0,$$

which leads to $\hat{X}_2 - X_2^{(1)} \geq 0$.
Consequently, we infer $\hat{X}_j - X_j^{(1)} \geq 0,\ j = 1, \ldots, q$.
Assume:

$$\hat{X}_j \geq X_j^{(k)} \geq Y_j^{(k-1)} \geq 0,\ j = 1, \ldots, q.$$

With similar reasoning, we derive the inequalities:

$$\hat{X}_j \geq X_j^{(k+1)} \geq Y_j^{(k)} \geq 0,\ j = 1, \ldots, q.$$

Both matrix sequences are monotone increasing in the elemenwise order and bounded by the above. They converge to same limit $(\tilde{P}_1, \ldots, \tilde{P}_q)$. Going to the limits in Equation (8), one concludes that $(\tilde{P}_1, \ldots, \tilde{P}_q)$ is a nonnegative solution of (1).

Suppose there is another solution $(\tilde{S}_1, \ldots, \tilde{S}_q)$ with $\tilde{S}_j \leq \tilde{P}_j$. The last inequalities lead us to a contradiction with the inequalities $\tilde{S}_j \leq \tilde{P}_j$. Therefore, the solution $(\tilde{S}_1, \ldots, \tilde{S}_q)$ is the minimal one.

Furthermore, we shall prove point (ii) of the theorem. Matrices $A_i - \hat{X}_iC_i$ and $D_i - C_i\hat{X}_i$, $i = 1, \ldots, q$ are nonsingular M-matrices for an upper nonnegative limit $(\hat{X}_1, \ldots, \hat{X}_q)$ for nonnegative solutions of (1). According to properties of M-matrices, we conclude that

$A_i - \tilde{X}_i C_i$ is a nonsingular M-matrix for $i = 1, \ldots, q$ and, moreover, $-A_i + \tilde{X}_i C_i, i = 1, \ldots, s$ is c-stable for the minimal nonnegative solution $(\tilde{X}_1, \ldots, \tilde{X}_q)$ of (1). □

Remark 1. *The existence of a nonnegative solution for the set of matrix Equation (1) is commented by [16]. Two assumptions are necessary in [14], which involve the existence of a nonnegative matrices sequence $Z_1, \ldots, Z_q$ such that $\mathcal{M}_i(Z_1, \ldots, Z_q) \leq 0$. However, we drop this condition in our investigation. We derive a direct convergence proof for iteration (8) based on Lemma 1.*

Remark 2. *We use the parameter ω, the values of which are bigger than 2 in (8) in order to speed up the rate of convergence for (8) comparing with the case $\omega = 1$. We denote $W_{j,\omega} = \omega Y_j^{(k)} + (1 - \omega) X_j^{(k)}$ and $V_{j,\omega} = \omega X_j^{(k+1)} + (1 - \omega) Y_j^{(k)}$. For $\omega > 2$, we have $W_{j,\omega} - W_{j,\omega=1} = \omega Y_j^{(k)} + (1 - \omega) X_j^{(k)} - Y_j^{(k)} = (\omega - 1) Y_j^{(k)} + (1 - \omega) X_j^{(k)} \geq 0$. That means $W_{j,\omega>2} - W_{j,\omega=1} \geq 0$. Analogously, the inequality $V_{j,\omega>2} - V_{j,\omega=1} \geq 0$ is true for all values of j. We expect that iteration (8) for $\omega \geq 2$ makes a smaller number of iteration steps than the case $\omega = 1$. We shall track this fact in numerical experiments.*

The above remarks allow choosing $\omega > 1$, and confirm that the choice preserves the monotony of the matrix sequences $\{X_i^{(k)}, Y_i^{(k)}\}_{k=0}^{\infty}\, i = 1, \ldots, q$.

2.3. Numerical Experiments

We provide numerical experiments to compute the minimal nonnegative solution to (1). We compare the results of iterations (5)–(7) with the results of the proposed new iterations (8). All experiments are performed in MATLAB (version R2018b) on a personal computer. The iterations stop when the current iterative step satisfies $RES_i \leq 10 \times 10^{-12}$, where RES_i is defined as [14]:

$$RES_i := \frac{\|\mathcal{M}_i({X_1}^{(k)}, \ldots, {X_q}^{(k)})\|}{\|\mathcal{M}_i({X_1}^{(0)}, \ldots, {X_q}^{(0)})\|},$$

$i = 1, \ldots, q$.

In the experiments, we choose the parameters γ_i, as defined in (2). We take $X_1^{(0)} = \ldots = X_q^{(0)} = 0$ for all examples and all iterative methods. Thus, $\mathcal{M}_i({X_1}^{(0)}, \ldots, {X_q}^{(0)}) = B_i$.

Example 1. *A set of $n \times n$ matrix coefficients for different values of n are tested. The matrices $A_i, D_i, i = 1, 2, 3$ are introduced following the Matlab terminology:*
A_1 = A_2=A_3 = zeros(n,n);
For $i = 1$:n, $A_1(i,i) = 4$; $A_2(i,i) = 3$; $A_3(i,i) = 2$; end
For $i = 1$:$n - 1$, $A_1(i, i+1) = -0.5$; $A_1(i+1, i) = -0.03$; end
For $i = 1$:$n - 2$, $A_1(i, i+2) = -0.25$; $A_1(i+2, i) = -0.9$; end
$A_1(1, n) = -0.05$; $A_1(n, 1) = -0.4$;
$A_2 = A_1$; $A_2(1, n) = -0.8$; $A_2(n, 1) = -0.06$;
$A_3 = A_1$; $A_3(1, n) = -0.7$; $A_3(n, 1) = -0.09$;
$B_1 = B_2 = B_3 = 0.75\, I_n$, $B_2 = B_1$, $B_3 = B_1$, $C_1 = 0.92\, I_n$, $C_2 = C_1$, $C_3 = C_1$, where I_n is an identity matrix order n.

$$E = (e_{ij}) = \begin{pmatrix} 0.0661 & 0.4512 & 0.8887 \\ 0.4965 & 0.3156 & 0.8780 \\ 0.6542 & 0.8914 & 0.1947 \end{pmatrix},$$

The results from the experiments are presented at Table 1. A hundred runs are executed for each example for $n = 12$ and $n = 24$. Ten runs are executed for $n = 48$. In this case, iterations (5) and (6) are very slowly (in the used computer), whereas iteration (8) is fastest.

Table 1. Example 1 with (5)–(8).

	(5)		**(6)**		**(7), $\omega = 1.2$**		**(8), $\omega = 2.5$**	
n	It	**CPU**	It	**CPU**	It	**CPU**	It	**CPU**
12	34	4.3 s	19	2.57 s	18	2.44 s	25	0.10 s
24	38	128.7 s	21	81.9 s	19	70.7 s	28	0.33 s
				10 runs				
48	22	323.0 s	22	281 s	20	220.8 s	33	0.23 s

Example 2. *A set of $n \times n$ matrix examples with the matrix coefficients for different values of n are tested.*

The matrices $A_i, D_i, i = 1,2,4$ are introduced following the Matlab terminology:
$A1 = gallery('tridiag', n, 0, 1, -1); A1 = full(A1);$
$A2 = gallery('tridiag', n, 0, 2, -1); A2 = full(A2);$
$A3 = gallery('tridiag', n, 0, 3, -1); A3 = full(A3);$
$A4 = gallery('tridiag', n, 0, 4, -1); A4 = full(A4);$
$D1 = gallery('tridiag', n, 0, 2, -1); D1 = full(D1);$
$D2 = gallery('tridiag', n, 0, 4, -1); D2 = full(D2);$
$D3 = gallery('tridiag', n, 0, 6, -1); D3 = full(D3);$
$D4 = gallery('tridiag', n, 0, 8, -1); D4 = full(D4);$
$B1 = 0.5 * eye(n, n); B2 = B1; B3 = B1; B4 = B1;$
$C1 = 0.2 * eye(n, n); C2 = C1; C3 = C1; C4 = C1;$
$E = rand(4);$
The results from the experiments are presented in Table 2.

Table 2. Example 2 for 10 runs with (5)–(8).

	(5)		**(6)**		**(7), $\omega = 1.2$**		**(8), $\omega = 2.5$**	
n	It	**CPU**	It	**CPU**	It	**CPU**	It	**CPU**
12	31	0.63 s	17	0.32 s	14	0.3 s	17	0.03 s
24	32	10.7 s	16	4.62 s	16	4.6 s	19	0.03 s
48	28	406.2 s	25	386.6 s	19	291.9 s	29	0.24 s
96			slow	convergence			35	0.98 s

The experiments with the above examples show the effectiveness of the proposed iteration Formula (8). Moreover, the high value of ω speeds up the convergence.

3. Numerical Method for the Maximal Solutions of Specifical Nonlinear Matrix Equations

Consider the iterative solution to the following nonlinear matrix equations:

$$X + A^* X^{-1} A = Q,$$

$$MY^2 + NY + P = 0,$$

investigated in [1,2,17]. Numerical methods on the specific solutions of the above matrix equations (maximal positive definite and minimal nonnegative) are investigated and some families of iterative formulas are proposed in [1,2,17]. However, comments and improvements of the proposed iteration schemes are provided to improve and accelerate the convergence. In this section, we will focus on the problem of how to accelerate the numerical solution of the above nonlinear matrix equations. The main tricks in the iterative methods proposed in these publications are to avoid the computation of an inverse matrix at each iteration step.

In general cases, the matrix A may be a real or complex square matrix. The notation A^* denotes a complex conjugate operation.

3.1. Iterative Solution of $X + A^*X^{-1}A = Q$

We firstly list several known algorithms for computing the maximal solution of $X + A^*X^{-1}A = Q$ and compare their computational behavior.

Algorithm 1 follows iterative Formula (2.2) and the corresponding algorithm from [1].

Algorithm 1 For matrix equation $X + A^*X^{-1}A = Q$

1: Introduce matrix coefficients $A, Q = I$ and a small positive number tol.
Take $X_0 = Y_0 = I$ (the identity matrix).
2: $Y_{k+1} = -I + Y_k(3I + X_k - 2X_kY_k)$,
$X_{k+1} = I - A^*Y_{k+1}A$,
3: Stop if $\|X_{k+1} + A^*X_{k+1}^{-1}A - I\| \le tol$. Otherwise, $k := k+1$ go to 2.
end

Algorithm 2 follows iterative Formula (3.3) and the corresponding algorithm from [2].

Algorithm 2 For matrix equation $X + A^*X^{-1}A = Q$

1: Introduce matrix coefficients $A, Q = I$ and a small positive number tol. Choose $p = 1, m = 1, q_1 = -1$ for Equation (1.9) [2].
Take $X_0 = Y_0 = I$ (the identity matrix).
2: $E_k = X_kY_k$,
$Y_{k+1} = -\frac{2}{5}I + \frac{12}{5}Y_k + \frac{1}{5}(E_k + E_k^*) - \frac{7}{5}Y_kE_k)$,
$X_{k+1} = I - A^*Y_{k+1}A$,
$Res_k = \|X_{k+1} + A^*X_{k+1}^{-1}A - I\| \le tol$.
3: If $Res_k \le tol$ then stop. Otherwise, $k := k+1$ go to 2.
end

In addition, we apply iterative Formula (3) from [18] to compute the same solution. The iteration (3) is:

$$X_{k+1} = I - A^*X_k^{-1}A, \quad X_0 = \alpha I, \ 0.5 \le \alpha \le 1, \ k = 0, 1, \ldots. \tag{9}$$

Here, we apply Algorithms 1 and 2 and iteration (9) to compute the maximal positive definite solution to $X + A^*X^{-1}A = I$. We use $tol = 10^{-16}$ in examples. The computations are performed on a computer Intel(R) Core(TM) i7-1065G7 CPU @ 1.30 GHz via Matlab R2018b.

Example 3. *Consider the Example 3.1 introduced in [1]. The matrix is:*

$$A = \frac{1}{40}\begin{pmatrix} 2 & -1 & 3 & 4 \\ 7 & 6 & -5 & 9 \\ 4 & 8 & 10 & 6 \\ -3 & 5 & 2 & 8 \end{pmatrix}.$$

We have executed 100 runs with all algorithms. Algorithm 1 makes 26 iteration steps for 0.0276 s. Algorithm 2 makes 21 iteration steps for 0.0238 s. The computer realization of iteration (9) performs 21 iteration steps for 0.0186 s. The performance results of the three algorithms are comparable and show their applicability.

Example 4. *The example is considered in [2] as Example 4.1.*

$$A = \begin{pmatrix} 0.37 & 0.13 & 0.12 \\ -0.30 & 0.34 & 0.12 \\ 0.11 & -0.17 & 0.29 \end{pmatrix}.$$

We have executed 100 runs with all algorithms. Algorithm 1 needs 81 iteration steps for 0.0754 s. Algorithm 2 needs 111 iteration steps for 0.1005 s. Iteration (9) performs 124 iteration steps for 0.0942 s. All three algorithms are working effectively for this example. The computational time is almost the same.

Example 5. *The example is introduced by Guo and Lancaster in [19] with:*

$$A = \begin{pmatrix} 0.2 & 0.2 & 0.1 \\ 0.2 & 0.15 & 0.15 \\ 0.1 & 0.15 & 0.25 \end{pmatrix},$$

We have executed 100 runs with all algorithms using two different values of tol. We take $tol = 10^{-4}$. *Algorithm 1 needs 48 iteration steps to compute the solution for 0.05 s. Algorithm 2 needs 59 iteration steps for 0.0586 s. Iteration (9) applies only three iteration steps for 0.0064 s with* $\alpha = 0.5$. *Further on, we take* $tol = 10^{-8}$. *Algorithm 1 needs 4714 iteration steps to compute the solution for 4.1306 s (for 100 runs). Algorithm 2 needs 5893 iteration steps for 5.7824 s (for 100 runs). However, iteration (9) has done only five iteration steps for 0.0101 s (for 100 runs) with* $\alpha = 0.5$. *Thus, iteration (9) is superior than Algorithms 1 and 2 when the maximal solution is computed in this example.*

Example 6. *The example is firstly considered in [20] and, next, is investigated in [18]. The matrix A is defined:*

$$A = \frac{\tilde{A}}{2\|\tilde{A}\|}, \quad \tilde{A} = \begin{pmatrix} 0.1 & -0.15 & -0.2598076 \\ 0.15 & 0.2125 & -0.0649519 \\ 0.2598076 & -0.0649519 & 0.1375 \end{pmatrix}.$$

Algorithms 1 and 2 do not converge for this example. Iteration (9) with $\alpha = 0.5$ *converges to the maximal solution after 11 iteration steps for* $tol = 10^{-7}$. *The maximal solution* $\tilde{X}$ *is:*

$$\tilde{X} = \begin{pmatrix} 0.500000082310064 & -0.000000016964994 & 0.000000002309095 \\ -0.000000016964994 & 0.729639588876686 & -0.132582448109853 \\ 0.000000002309095 & -0.132582448109853 & 0.576546597071862 \end{pmatrix}.$$

The results of the experiments in this section show that the introduced iterative method (9) in [18] is effective and comparable to the iterative methods introduced in [1,2], and even better. Iterative method (9) uses the choice of an initial approximation depending on the value of α. How to make the choice of α can be read in [18]. Algorithms 1 and 2 avoid the computation of the inverse matrix, but this is not always reliable, as can be seen from the examples discussed in this section. Thus, we have to be careful where the inverse free algorithm is applied.

3.2. Numerical Method for the Solution of $MY^2 + NY + P = 0$

In this section, we study square matrix equation $MY^2 + NY + P = 0$, where M, N, P are real matrix coefficients. Different iterative methods are analyzed in [17] . The authors of [17] have investigated a family of iterative methods for finding the minimal nonnegative solution to $MY^2 + NY + P = 0$. Their conclusion shows that Algorithms 1 and 6 defined in [17] are able to find the corresponding solution with the given accuracy. We will present these two algorithms and propose their modifications to improve their computational behavior, i.e., we will propose new modifications of both algorithms to make them more effective in the computational aspects.

We describe Algorithm 1 proposed in [17] as Algorithm 3 here.

Algorithm 3 Algorithm 1 [17]

1: Input $n \times n$ matrices M, N, P.
2: We take $Y0$ and $\alpha > 0$.
3: Compute $V_M = \alpha M$, and $W_M = (1-\alpha)M$.
Note that $M = V_M + W_M$.
4: Compute Y_{r+1} from
$(V_M Y_r + N + R)Y_{r+1} = (R - W_M Y_r)Y_r - P$.
5: If $\|MY_r^2 + NY_r + P\| < tol$ then stop.

Now, we introduce our modifications to the above algorithms. The aim of the modifications is to use a diagonal matrix $W_M = \xi * I_n$. Then, the matrix multiplication $W_M Y_r$ can be realized as $\xi * Y_r$ in Matlab. Taking W_M as a diagonal matrix, we preserve the properties of Theorem 2.4 proved by Erfanifar and Hajarian [17]. Thus, the matrix $V_M Y_{r+1} + N$ is an M-matrix and the matrix sequence $\{Y_r\}$ is monotone increasing and then converges to the minimal nonnegative solution. Moreover, applying a diagonal form for the matrix W_M, we avoid a matrix multiplication and replace it with a matrix multiplication by a number.

Compare the results from Algorithms 3 and 4 by Example 7.

Algorithm 4 Our modification to Algorithm 3

1: Input $n \times n$ matrices M, N, P.
2: Take $Y0 = 0$ and $\alpha > 0, R = \alpha * I_n$.
3: Compute $V_M = M + R$, and $W_M = -R$, and $NN = N + R$.
Note that $M = V_M + W_M$.
4: Compute Y_{r+1} using the equation
$(V_M Y_r + N + R)Y_{r+1} = (R - W_M Y_r)Y_r - P$.
4.1: Compute $im = inv(VM * Y0 + NN)$. (Remark $Y0 = Y_r$.)
4.2: Compute $tQ = (R + \alpha * Y0) * Y0 - P$.
4.3: Compute $Y0 = im * tQ$;. (Remark $Y0 = Y_{r+1}$ here).
4.4: If $norm((M * Y0 + N) * Y0 + P) \leq tol$ then stop. Orthерwise, $r = r + 1$ and go to Step 4.2.
5: The computed solution is $Y0$.

Example 7 (Example 4.1, [17])**.** *For $s \times s$ matrix coefficients $M = (m_{ij}), P = (p_{ij}), N = (n_{ij})$, we have:*

$$\begin{cases} m_{ii} = -1.5, \quad i = 1, \dots s; \\ m_{i,i+1} = -8, \;\; m_{i+1,i} = -5, \;\; i = 1, \dots s-1; \\ p_{ii} = -0.5, \quad i = 1, \dots s; \\ p_{i,i+1} = -0.8, \quad p_{i+1,i} = -1.5, \;\; i = 1, \dots s-1; \\ n_{ii} = 45, \quad i = 1, \dots s, \\ n_{i,i+1} = -6, \;\; n_{i+1,i} = -4, \quad i = 1, \dots s-1, \\ n_{11} = n_{ss} = 18. \end{cases}$$

Introducing a vector-row of size s of units, i.e., $e = (1, \dots, 1)$, we compute $emat = 0.1 * e' * e$, and $M = M\text{-}emat$.

Based on the matrices M, N, P, we compute a nonnegative solution of matrix Equation (1) with Algorithms 3 and 4 with the stop criterion with $tol = 10^{-14}$ and compare numbers of iteration steps (It) and CPU time for 1000 runs for each value of s. The results are listed in Table 3.

Table 3. Example 7 for 1000 runs with Algorithms 3 and 4.

	Algorithm 3		Algorithm 4	
$s(\alpha)$	It	CPU Time Seconds	It	CPU Time Seconds
10 (0.6)	14	0.20	13	0.19
20 (0.6)	14	0.48	13	0.42
30 (0.6)	14	0.80	13	0.74
40 (0.6)	14	1.45	13	1.41
50 (0.6)	14	4.35	13	3.87
60 (0.6)	14	5.56	13	5.25
70 (0.6)	15	8.60	14	8.22
80 (0.7)	no convergence		14	7.52
80 (0.9)	15	8.37	15	8.12
		$tol = 10^{-13}$		
90 (0.6)	14	10.51	13	9.43
100 (0.6)	14	14.1	13	13.71

Further on, we describe Algorithm 6 introduced in [17].
Applying the same approach, we obtain a modification of Algorithm 5.

Algorithm 5 Algorithm 6 [17]

1: Input $n \times n$ matrices M, N, P.
2: We take $Y0$ and $\alpha > 0, \beta > 0$.
3: Compute $V_M = \alpha M$, and $W_M = (1-\alpha)M$.
$V_N = \alpha N$, and $W_N = (1-\beta)N$.
Note that $M = V_M + W_M$ and $N = V_N + W_N$.
4: Compute Z_r, Y_{r+1} from
$(V_M Y_r + V_N + R)Z_r = (R - W_M Y_r - W_N)Y_r - P$,
$(W_M Z_r + V_N + S)Y_{r+1} = (S - V_M Z_r - W_N)Z_r - P$.
5: If $\|MY_r^2 + NY_r + P\| < tol$ then stop . Ortherwise, $r = r+1$ and go to Step 4.

We have performed experiments with Algorithms 4 and 5 for Example 7. The tol value is $tol = 10^{-14}$ and 1000 runs for each value of s are played. The results can be found in Table 4.

Table 4. Example 7 for 1000 runs with Algorithms 5 and 6.

	Algorithm 5		Algorithm 6		Algorithm 6	
	$\alpha = 0.8, \beta = 0.9$		$\alpha = \beta = 0.94$		$\alpha = 0.8; \beta = 0.95$	
$n(\alpha, \beta)$	It	CPU Time Seconds	It	CPU Time Seconds	It	CPU Time Seconds
10	7	0.15	6	0.14	6	0.12
20	7	0.34	6	0.31	6	0.28
30	7	0.59	6	0.51	6	0.48
40	7	1.05	6	0.87	6	0.85
50	7	3.32	6	1.92	6	1.82
60	7	3.11	6	2.38	6	2.46
70	7	4.25	6	3.56	6	3.62
			$tol = 10^{-13}$			
80	7	5.32	6	4.70	6	4.44
90	7	5.62	6	5.60	6	5.87
100	7	10.26	6	8.30	6	8.24

Algorithm 6 Our modification of Algorithm 5

1: Input $n \times n$ matrices M, N, P.
2: We take $Y0$ and $\alpha > 0, \beta > 0, R = \alpha * I_n, S = \beta * I_n$.
3: Compute $V_M = M + R$, and $W_M = -R$, and $V_N = \beta * N$, $W_N = (1 - \beta) * N$ and $NN = V_N + R, NM = V_N + S$.
Note that $M = V_M + W_M$ and $N = V_N + W_N$.
4: Compute Z_r, Y_{r+1} from matrix equations:
$(V_M Y_r + V_N + R)Z_r = (R - W_M Y_r - W_N)Y_r - P$,
$(W_M Z_r + V_N + S)Y_{r+1} = (S - V_M Z_r - W_N)Z_r - P$.
4.1: Compute $im = inv(V_M * Y0 + NN)$. (Remark $Y0 = Y_r$.)
4.2: Compute $tQ = (R + \alpha * Y0 - W_N) * Y0 - P$.
4.3: Compute $Z0 = im * tQ$;. (Remark $Z0 = Z_r$ here).
4.4: Compute $im = inv(NM - \alpha * Z0)$.
4.5: Compute $tQ = (S - V_M * X0 - W_N) * Y0 - P$.
4.6: Compute $Y0 = im * tQ$;. (Remark $Y0 = Y_{r+1}$ here).
4.7: If $norm((M * Y0 + N) * Y0 + P) \leq tol$ then stop. Orthерwise, $r = r + 1$ and go to Step 4.1.
5: The computed solution is $Y0$.

Comparing Tables 3 and 4, we conclude that Algorithm 5 is faster than Algorithm 3, and Algorithm 6 is faster than Algorithm 4. Algorithm 6 is faster than the remaining algorithms. The approach to divide the given iteration from Algorithm 3 in two parts, as it is shown in Algorithm 5, is more effective than the original one.

4. Conclusions

In this paper, we have studied numerical methods for three computational tasks: (a) to compute the minimal nonnegative solution of a set of Riccati equations, (b) to compute the maximal positive definite solution of the equation $X + A^* X^{-1} A = Q$, and (c) to compute the minimal nonnegative solution to the quadratic matrix equation $MY^2 + NY + P = 0$. We have considered the existing iterative methods and have proposed their improvements to accelerate the convergence process. We have performed several numerical experiments for each task where to show the effectiveness of the proposed modifications.

Moreover, as a weakness of the iterative methods for solving task (b), we note that the application of the inverse free approach, where the computation of an inverse matrix is avoided, will save the cost of computation. However, the use of this approach is limited. This fact is confirmed by experiments in Section 3.1. In recent years, this approach has been widely used in the analysis of iterative solutions of matrix equations. The effectiveness of this approach will be investigated in our future work more deeply.

Author Contributions: Conceptualization, I.G.I. and H.Y.; methodology, I.G.I. and H.Y.; software, I.G.I. and H.Y.; validation, I.G.I. and H.Y.; formal analysis, I.G.I. and H.Y.; investigation, I.G.I. and H.Y.; supervision, I.G.I. and H.Y.; writing—original draft and improved manuscript, I.G.I. and H.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: Thanks to the reviewers for their useful comments, remarks, and constructive recommendations, which have increased the value of this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Erfanifar, R.; Sayevand, K.; Esmaeili, H. A novel iterative method for the solution of a nonlinear matrix equation. *Appl. Numer. Math.* **2020**, *153*, 503–518. [CrossRef]
2. Erfanifar, R.; Sayevand, K.; Hajarian, M. An efficient inversion-free method for solving the nonlinear matrix equation $X^P + \sum_{j=1}^{m} A_j^* X^{-q_j} A_j = Q$. *J. Frankl. Inst.* **2022**, *359*, 3071–3089.
3. Yang, Q.; Wang, X.; Cheng, X.; Du, B.; Zhao, Y. Positive periodic solution for neutral-type integral differential equation arising in epidemic model. *Mathematics* **2023**, *11*, 2701. [CrossRef]
4. Fan, L.; Zhu, Q.; Zheng, W.X. Stability analysis of switched stochastic nonlinear systems with state-dependent delay. *IEEE Trans. Autom. Control* **2023**, 1–8. [CrossRef]
5. Erfanifar, R.; Sayevand, K.; Hajarian, M. Convergence analysis of Newton method without inversion for solving discrete algebraic Riccati equations. *J. Frankl. Inst.* **2022**, *359*, 7540–7561. [CrossRef]
6. Erfanifar, R.; Sayevand, K.; Hajarian, M. Solving system of nonlinear matrix equations over Hermitian positive definite matrices. *Linear Multilinear Algebra* **2023**, *71*, 597–630. [CrossRef]
7. Hasanov, V.I.; Ali, A.A. On convergence of three iterative methods for solving of the matrix equation $X + A^* X^{-1} A + B^* X^{-1} B = Q$. *Comput. Appl. Math.* **2017**, *36*, 79–87. [CrossRef]
8. El-Sayed, S.M.; Ivanov, I.G.; Petkov, M.G. A new modification of the Rojo method for solving symmetric circulant five-diagonal systems of linear equations. *Comput. Math. Appl.* **1998**, *35*, 35–44. [CrossRef]
9. 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 Algebra Appl.* **2006**, *13*, 655–674.
10. Ma, C.; Lu, H. Numerical Study on Nonsymmetric Algebraic Riccati Equations. *Mediterr. J. Math.* **2016**, *13*, 4961–4973. [CrossRef]
11. Guan, J.; Lu, L. New alternately linearized implicit iteration for M-matrix algebraic Riccati equations. *J. Math. Study* **2017**, *50*, 54–64.
12. Guan, J. Modified alternately linearized implicit iteration method for M-matrix algebraic Riccati equations. *Appl. Math. Comput.* **2019**, *347*, 442–448.
13. Ivanov, I.; Yang, H. An effective approach to solve a nonsymmetric algebraic Riccati equation. *Innov. Model. Anal. J. Res.* **2021**, *6*, 7–14.
14. Zhang, J.; Tan, F. Numerical methods for the minimal non-negative solution of the non-symmetric coupled algebraic Riccati equation. *Asian J. Control* **2021**, *23*, 374–386. [CrossRef]
15. Ivanov, I. Iterative computing the minimal solution of the coupled nonlinear matrix equations in terms of nonnegative matrices. *Ann. Acad. Rom. Sci. Ser. Math. Appl.* **2020**, *12*, 226–237. [CrossRef]
16. Liu, J.; Zhang, J.; Luo, F. Newton's method for the positive solution of the coupled algebraic Riccati equation applied to automatic control. *Comput. Appl. Math.* **2020**, *39*, 113. I. [CrossRef]
17. Erfanifar, R.; Hajarian, M. Weight splitting iteration methods to solve quadratic nonlinear matrix equation $MY^2 + NY + P = 0$. *J. Frankl. Inst.* **2023**, *360*, 1904–1928.
18. Ivanov, I.G.; Hasanov, V.I.; Uhlig, F. Improved methods and starting values to solve the matrix equations $X \pm A * X^{-1} A = I$ iteratively. *Math. Comput.* **2005**, *74*, 263–278.
19. Guo, C.-H.; Lancaster, P. Iterative Solution of Two Matrix Equations. *Math. Comput.* **1999**, *68*, 1589–1603. [CrossRef]
20. Zhan, X. Computing the extremal positive definite solutions of a matrix equation. *SIAMJ Sci. Comput.* **1996**, *17*, 1167–1174. [CrossRef]