NEWTON'S METHOD FOR SYMMETRIC AND
BISYMMETRIC SOLVENTS OF THE NONLINEAR MATRIX
EQUATIONS

YIN-HUAN HAN AND HYUN-MIN KIM

ABSTRACT. One of the interesting nonlinear matrix equations is the quadratic matrix equation defined by

\[ Q(X) = AX^2 + BX + C = 0, \]

where \( X \) is a \( n \times n \) unknown real matrix, and \( A, B \) and \( C \) are \( n \times n \) given matrices with real elements. Another one is the matrix polynomial

\[ P(X) = A_0X^m + A_1X^{m-1} + \cdots + A_m = 0, \quad X, A_i \in \mathbb{R}^{n \times n}. \]

Newton’s method is used to find the symmetric and bisymmetric solvents of the nonlinear matrix equations \( Q(X) \) and \( P(X) \). The method does not depend on the singularity of the Fréchet derivative. Finally, we give some numerical examples.

1. Introduction

We consider two kinds of nonlinear matrix equations, namely, the quadratic matrix equation

\[ Q(X) = AX^2 + BX + C = 0, \quad A, B, C \text{ and } X \in \mathbb{R}^{n \times n}, \]

and the matrix polynomial

\[ P(X) = A_0X^m + A_1X^{m-1} + \cdots + A_m = 0, \quad A_i, X \in \mathbb{R}^{n \times n}. \]

Solving nonlinear matrix equations addresses many problems which occur in many applications and in modeling of scientific problems.

Newton’s method is a natural approach in solving nonlinear matrix equations. For the quadratic case (1), Davis [2, 3] considered Newton’s method and Higham and Kim [5, 6] incorporated the exact line searches into Newton’s method, which reduced the number of iterations required for the most part. For solving the matrix polynomial (2), Newton’s method was considered by Kratz and Stickel [8].
However, we need to overcome the following challenges when solving matrix equations by Newton’s method.

(i) The method only works well when the Fréchet derivative is nonsingular.

(ii) How to guarantee the convergence of a particular starting matrix.

Guo and Laub [4] considered the nonsymmetric algebraic Riccati equation

\[ R(X) = XEX + XG + HX + F = 0, \]

which arises from the transport theory. They proposed an algorithm for Newton’s method with a special starting matrix to find the elementwise minimal positive solvent. Kim [7] presented that the elementwise minimal positive definite solvent for some different types of quadratic matrix equations can be found by Newton’s method with a zero starting matrix.

In this paper, we introduce two iterative algorithms for solving the Newton step with the symmetric and bisymmetric solutions. Then we apply Newton’s method with iterative algorithms to solve the quadratic matrix equation (1) and the matrix polynomial (2). We show that for the symmetric (bisymmetric) starting matrix \( X_0 \), our Newton’s method converges to a solvent which has the same properties as \( X_0 \). Finally, we give some numerical experiments that confirm our Newton’s method is efficient for solving the case when the Fréchet derivative is singular.

Definition 1.1 ([12]). A matrix \( A \in \mathbb{R}^{n \times n} \) is called a bisymmetric (BS) matrix if its elements \( a_{ij} \) satisfy the properties

\[ a_{ij} = a_{ji} \text{ and } a_{ij} = a_{n-j+1,n-i+1} \quad \text{for } 1 \leq i, j \leq n. \]

In order to construct an iterative method for finding a bisymmetric solution of the Newton step, the following basic properties of bisymmetric matrices are needed.

Lemma 1.2 ([10]). A matrix \( B \) is bisymmetric if and only if \( B = B^T = S_nBS_n \), where \( S_n = [e_n, e_{n-1}, \ldots, e_1] \) and \( e_i \) denotes the elementary standard vector of \( \mathbb{R}^n \).

Lemma 1.3 ([11]). If the matrix \( X \in \mathbb{R}^{n \times n} \) is a symmetric matrix, then \( X + S_nXS_n \) is a bisymmetric matrix.

2. Newton’s method

In this section, let us review Newton’s method for the general nonlinear matrix equation \( G : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n \times n} \) such that

\[ G(X) = 0. \]

Let the matrix \( S \) be a solvent of equation (3) and write

\[ X_k = S + H_k. \]
Then by Taylor’s Theorem, we have

\[ G(X_k) = G(S + H_k) \]

\[ = G(S) + G'(S)H_k + O(H_k^2) \]

\[ = G'(S)H_k + O(H_k^2), \]

where \( G' : \mathbb{R}^{n \times n} \to \mathbb{R}^{n \times n} \) is the Fréchet derivative as \( G(X) \) at \( X \). If \( H_k \) is the value that we know, then the matrix equation (3) can be automatically solved from (4). Moreover, if we evaluate the Fréchet derivative \( G' \) at \( X_k \), replace \( H_k \) in (5) by \( X_k - X_{k+1} \) rather than \( X_k - S \) and ignore the second order terms, then we get

\[ G(X_k) = G'(X_k)(X_k - X_{k+1}). \]

So if \( G'(X_k) \) is nonsingular, then from (6), we obtain the next approximation \( X_{k+1} \) as

\[ X_{k+1} = X_k - [G'(X_k)]^{-1}G(X_k), \]

which is called Newton’s method [1, 9].

In the nonsingular Fréchet derivative case, the Kantorovich theorem gives information on the convergence of Newton’s method for solving the nonlinear matrix equation (3) [2].

**Theorem 2.1 (Kantorovich).** If there exists \( K \) such that

\[ \|G'(X) - G'(Y)\| \leq K\|X - Y\| \quad \text{for all} \quad X, Y \in \mathbb{R}^{n \times n} \]

in some closed ball \( \bar{U}(X_0, r) \) and \( h_0 = B_0\eta_0K \leq \frac{1}{2} \) with \( \|G'(X_0)\|^{-1} \leq B_0 \) and \( \|X_1 - X_0\| \leq \eta_0 \), then the Newton sequence starting from \( X_0 \) will converge to a solvent \( S \) of \( G(X) \) which exists in \( \bar{U}(X_0, r) \), provided that

\[ r \geq r_0 = \frac{1 - \sqrt{1 - 2h_0}}{h_0}\eta_0. \]

However, Theorem 2.1 cannot affect the settlement for the weak points of Newton’s method.

If we define \( E_k \) as the solution of the linear equation \( G(X_k) + G'(X_k)E_k = 0 \), then Newton’s method for the nonlinear matrix equations (3) with the given starting matrix \( X_0 \) can be written in the iteration form

\[
\begin{cases}
G'(X_k)E_k = -G(X_k), \\
X_{k+1} = X_k + E_k,
\end{cases}
\]

where \( k = 0, 1, \ldots \).

Thus each step of Newton’s method requires finding of the solution \( E \) of the linear equation

\[ G'(X)E = -G(X). \]

Now, we should derive the Fréchet derivatives of the quadratic matrix equations (1) and the matrix polynomial (2) to solve them using Newton’s method.
From the definition of the quadratic matrix equation (1), we easily obtain
\[ Q'(X)[E] = (AX + B)E + AEX, \]
which is the Fréchet derivative of equation (1) at \( X \) in the direction \( E \). The
Fréchet derivative of the matrix polynomial (2) is
\[
P'(X)[H] = \left( \sum_{\nu=0}^{m-1} A_{\nu} X^{(m-1)-\nu} \right) H + \left( \sum_{\nu=0}^{m-2} A_{\nu} X^{(m-2)-\nu} \right) HX + \cdots + A_0 H X^{m-1}.
\]
Therefore, each step of Newton’s method for equations (1) and (2) involves
finding the solution \( E \) and the solution \( H \) of
\[
(9) \quad (AX + B)E + AEX = -Q(X)
\]
and
\[
(10) \quad \left( \sum_{\nu=0}^{m-1} A_{\nu} X^{(m-1)-\nu} \right) H + \left( \sum_{\nu=0}^{m-2} A_{\nu} X^{(m-2)-\nu} \right) HX + \cdots + A_0 H X^{m-1} = -P(X),
\]
respectively.

3. The symmetric solvents of \( Q(X) \) and \( P(X) \)

In Section 2, we have already seen that for solving \( Q(X) \) and \( P(X) \) using
Newton’s method, we need to solve the linear equations (9) and (10), respectively. So we first give an iterative method to find a symmetric solution of (9), then extend it to solve equation (10). Then we consider the convergence of our
Newton’s method. From here, \( \| \cdot \| \) denotes the Euclidean norm of matrices.

3.1. An iterative method for solving equation (9)

The following algorithm is to find a symmetric solution of the \( q \)th Newton
step (9).

\begin{algorithm}
\textbf{Algorithm 3.1}. Let \( A, B, C \in \mathbb{R}^{n \times n} \) and a symmetric matrix \( X_q \in \mathbb{R}^{n \times n} \) be
given. Choose a symmetric starting matrix \( E_{q_0} \in \mathbb{R}^{n \times n} \).

\begin{align*}
& k = 0; \quad R_0 = -Q(X_q) - [(AX_q + B)E_{q_0} + AE_{q_0}X_q] \\
& Z_0 = (AX_q + B)T R_0 + A^T R_0 (X_q)^T \\
& P_0 = \frac{1}{2} (Z_0 + Z_0^T) \\
& \alpha_0 = \frac{\| R_0 \|^2}{\| P_0 \|^2} \\
\text{while } & R_k \neq 0 \text{ or } P_k \neq 0 \\
& \alpha_k = \frac{\| R_k \|^2}{\| P_k \|^2} \\
& E_{q_{k+1}} = E_{q_k} + \alpha_k P_k \\
& R_{k+1} = -Q(X_q) - [(AX_q + B)E_{q_{k+1}} + AE_{q_{k+1}}X_q] \\
& Z_{k+1} = (AX_q + B)^T R_{k+1} + A^T R_{k+1} (X_q)^T \\
& \beta_k = \frac{\| R_{k+1} \|^2}{\| R_k \|^2}
\end{align*}
\end{algorithm}
Lemma 3.6. For the sequences $P$ and $(12)$ have that

\[ P_{k+1} = \frac{1}{2} (Z_{k+1} + Z_{k+1}^T) + \beta_k P_k \]

Remark 3.2. In Algorithm 3.1, the matrices $P$ and $E_{q_k}$ are symmetric matrices for all $k = 0, 1, 2, \ldots$

From Algorithm 3.1, we have some basic properties.

**Lemma 3.3.** Let $E_q$ be a symmetric solution of the $q$th Newton step (9), and the sequences $\{Z_k\}, \{R_k\}, \{E_{q_k}\}$ be generated by Algorithm 3.1. Then the following statement holds.

\[ \text{tr} [Z_k^T (E_q - E_{q_k})] = \|R_k\|^2 \quad \text{for all } k = 0, 1, \ldots \]

**Proof.** From Algorithm 3.1, for any $k$, we have that

\[ \text{tr} [Z_k^T (E_q - E_{q_k})] = \text{tr} \left\{ [(AX_q + B)^T R_k + A^T R_k (X_q)^T] (E_q - E_{q_k}) \right\} \]

\[ = \text{tr} \left\{ R_k^T [(AX_q + B)(E_q - E_{q_k}) + A(E_q - E_{q_k})X_q] \right\} \]

\[ = \text{tr} \left\{ R_k^T \left[ -Q(X_q) - (AX_q + B)E_{q_k} - AE_{q_k}X_q \right] \right\} \]

\[ = \|R_k\|^2. \quad \Box \]

**Lemma 3.4.** Suppose $E_q$ is a symmetric solution of equation (9). Then

\[ \text{tr} [P_k^T (E_q - E_{q_k})] = \|R_k\|^2 \quad \text{for all } k = 0, 1, \ldots \]

**Proof.** We prove the conclusion (11) by induction.

When $k = 0$, from Algorithm 3.1 and Lemma 3.3, we have

\[ \text{tr} [P_0^T (E_q - E_{q_0})] = \text{tr} [Z_0^T (E_q - E_{q_0})] = \|R_0\|^2. \]

Assume that the conclusion (11) holds for $k = l$. Then when $k = l + 1$,

\[ \text{tr} [P_{l+1}^T (E_q - E_{q_{l+1}})] = \text{tr} [Z_{l+1}^T (E_q - E_{q_{l+1}})] + \beta_l \text{tr} [P_l^T (E_q - E_{q_{l+1}})] \]

\[ = \|R_{l+1}\|^2 \]

by Lemma 3.3 since

\[ \text{tr} [P_l^T (E_q - E_{q_{l+1}})] = \text{tr} [P_l^T (E_q - E_{q_l} - \alpha_l P_l)] \]

\[ = \text{tr} [P_l^T (E_q - E_{q_l})] - \alpha_l \text{tr} (P_l^T P_l) = 0. \quad \Box \]

Remark 3.5. Lemma 3.4 implies that if the $q$th Newton step (9) has a symmetric solution and $R_k \neq 0$ for some integer $k$, then $P_k \neq 0$ must hold for $k$.

**Lemma 3.6.** For the sequences $\{R_i\}$ and $\{P_i\}$ generated by Algorithm 3.1, we have that

\[ \text{tr} (R_i^T R_j) = 0 \quad \text{and} \quad \text{tr} (P_i^T P_j) = 0 \quad \text{for } i > j = 0, 1, \ldots, k, k \geq 1. \]
Proof. We prove (12) by induction.

Step 1. When \( k = 1 \),
\[
\text{tr} (R_1^T R_0) = \text{tr} \left\{ (R_0 - \alpha_0 (AX_q + B) P_0 - \alpha_0 A^T R_0) R_0 \right\} \\
= \|R_0\|^2 - \alpha_0 \text{tr} \left\{ P_0^T [(AX_q + B)^T R_0 + A^T R_0 (X_q)^T] \right\} \\
= \|R_0\|^2 - \alpha_0 \text{tr} (P_0^T P_0) \\
= 0,
\]
and
\[
\text{tr} (P_1^T P_0) = \text{tr} (Z_0^T P_0) + \beta_0 \text{tr} (P_0^T P_0) \\
= \text{tr} \left\{ R_0^T [(AX_q + B) P_0 + AR_0 X_q] \right\} + \frac{\|R_0\|^2 \|P_0\|^2}{\|R_0\|^2} \\
= -\frac{1}{\alpha_0} \text{tr} (R_1^T R_1) + \frac{\|R_1\|^2 \|P_0\|^2}{\|R_0\|^2} \\
= 0.
\]
Assume the statement (12) holds for \( k = l \), i.e., \( \text{tr} (R_l^T R_{l-1}) = 0 \) and \( \text{tr} (P_l^T P_{l-1}) = 0 \). Then
\[
\text{tr} (R_{l+1}^T R_l) = \text{tr} (R_l^T R_l) - \alpha_l \text{tr} \left\{ [(AX_q + B) P_l + AP_l X_q]^T R_l \right\} \\
= \|R_l\|^2 - \alpha_l \text{tr} (P_l^T Z_l) \\
= \|R_l\|^2 - \alpha_l \text{tr} (P_l^T P_l) - \alpha_l \beta_{l-1} \text{tr} (P_l^T P_{l-1}) \\
= 0,
\]
and
\[
\text{tr} (P_{l+1}^T P_l) = \text{tr} (Z_l^T P_l) + \beta_l \text{tr} (P_l^T P_l) \\
= \text{tr} \left\{ R_{l+1}^T [(AX_q + B) P_l + AP_l X_q] \right\} + \frac{\|R_{l+1}\|^2 \|P_l\|^2}{\|R_l\|^2} \\
= -\frac{1}{\alpha_l} \text{tr} (R_{l+1}^T R_{l+1}) + \frac{\|R_{l+1}\|^2 \|P_l\|^2}{\|R_l\|^2} \\
= 0.
\]
Step 2. Suppose that \( \text{tr} (R_j^T R_j) = 0 \) and \( \text{tr} (P_j^T P_j) = 0 \) for all \( j = 0, 1, \ldots, l-1 \), i.e., \( \text{tr} (P_j^T P_{j-1}) = 0 \). Now we show that \( \text{tr} (R_{l+1}^T R_j) = 0 \) and \( \text{tr} (P_{l+1}^T P_j) = 0 \) for \( j = 0, 1, \ldots, l-1 \).

By Algorithm 3.1 and the accompanying assumptions, we have
\[
\text{tr} (R_{l+1}^T R_j) = \text{tr} (R_j^T R_j) - \alpha_j \text{tr} \left\{ [(AX_q + B) P_j + AP_j X_q]^T R_j \right\} \\
= -\alpha_j \text{tr} (P_j^T Z_j) \\
= -\alpha_j \text{tr} [P_j^T (P_j - \beta_{j-1} P_{j-1})] \\
= 0,
\]
and
\[
\text{tr} \left( P_{l+1}^T P_j \right) = \text{tr} \left( Z_{l+1}^T P_j \right) + \beta \text{tr} \left( P_{l}^T P_j \right) \\
= \text{tr} \left\{ R_{l+1}^T \left[ (AX_q + B)P_j + AP_j X_q \right] \right\} \\
= \frac{1}{\alpha_j} \text{tr} \left[ R_{l+1}^T (R_j - R_{j+1}) \right] \\
= 0.
\]
Hence the statement (12) holds for \( k = l + 1 \). Therefore, from Steps 1 and 2, we complete the proof. □

**Theorem 3.7.** Assume the \( q \)th Newton step (9) has a symmetric solution. Then for any symmetric starting matrix \( E_{q_0} \), its symmetric solution can be obtained, at most, in \( n^2 \) steps.

**Proof.** Suppose that \( R_k \neq 0 \) for \( k = 0, 1, \ldots, n^2 - 1 \). Then from Lemma 3.6, the set \( \{ R_0, R_1, \ldots, R_{n^2 - 1} \} \) is an orthogonal basis of the matrix space \( \mathbb{R}^{n \times n} \). Since the \( q \)th Newton step (9) has a symmetric solution, \( P_k \neq 0 \) for \( k = 0, 1, \ldots, n^2 - 1 \) by Lemma 3.6. Therefore, we can evaluate \( E_{q,w} \) and \( R_{n^2} \) in Algorithm 3.1, and \( \text{tr} \left( R_{n^2}^T R_k \right) = 0 \) for \( k = 0, 1, \ldots, n^2 - 1 \) by Lemma 3.6. But \( \text{tr} \left( R_{n^2}^T R_k \right) = 0 \) holds only when \( R_{n^2} = 0 \), which implies that \( E_{q,w} \) is a solution of equation (9). □

From Newton’s method and Theorem 3.7, we have the following main theorem.

**Theorem 3.8.** Suppose that the quadratic matrix equation (1) has a symmetric solvent and each Newton step is consistent for a symmetric starting matrix \( X_0 \). The sequence \( \{ X_k \} \) is generated by Newton’s method with \( X_0 \) such that

\[
\lim_{k \to \infty} X_k = S,
\]

and if the matrix \( S \) satisfies \( Q(S) = 0 \), then \( S \) is a symmetric solvent.

**Proof.** Let \( E_0 \) be a symmetric solution of the first Newton step

\[
(AX_0 + B)E_0 + AE_0X_0 = -Q(X_0) = -AX_0^2 - BX_0 - C
\]

with the symmetric starting matrix \( X_0 \). Then according to Newton’s method and Theorem 3.7, we obtain the symmetric matrix

\[
X_{k+1} = X_k + E_k \\
= X_0 + E_0 + \cdots + E_k
\]

for all \( k = 0, 1, \ldots \) with starting matrix \( X_0 \). Since the matrix \( X_0 \) guarantees

\[
\lim_{k \to \infty} X_{k+1} = \lim_{k \to \infty} (X_0 + E_0 + \cdots + E_k) = S,
\]

the matrix \( S \) is a symmetric matrix. □
3.2. An iterative method for solving (10)

We now propose an iterative method for solving the qth Newton step (10) of a matrix polynomial.

**Algorithm 3.9.** Input $n \times n$ real matrices $A_0, A_1, \ldots, A_m$ and a symmetric matrix $X_q \in \mathbb{R}^{n \times n}$. Choose a symmetric starting matrix $H_{q_0} \in \mathbb{R}^{n \times n}$.

$k = 0; \quad R_0 = -P(X_q) - \left(\sum_{\nu=0}^{m-1} A_\nu X^{(m-1)-\nu}\right) H_{q_0} - \cdots - A_0 H_{q_0} X_q^{m-1} \quad Y_0 = \left(\sum_{\nu=0}^{m-1} A_\nu X^{(m-1)-\nu}\right)^T R_0 + \cdots + A_0^T R_0 (X_q^{m-1})^T \quad Q_0 = \frac{1}{2}(Y_0 + Y_0^T)$

while $R_k \neq 0$ or $Q_k \neq 0$

$H_{q_k+1} = H_{q_k} + \frac{1}{\|R_k\|} Q_k$

$R_{k+1} = -P(X_q) - \left(\sum_{\nu=0}^{m-1} A_\nu X^{(m-1)-\nu}\right) H_{q_k+1} - \cdots - A_0 H_{q_k+1} X_q^{m-1} \quad Y_{k+1} = \left(\sum_{\nu=0}^{m-1} A_\nu X^{(m-1)-\nu}\right)^T R_{k+1} + \cdots + A_0^T R_{k+1} (X_q^{m-1})^T \quad Q_{k+1} = \frac{1}{2}(Y_{k+1} + Y_{k+1}^T) + \frac{1}{\|R_{k+1}\|} Q_k$

end

Regarding Algorithm 3.9, we have the following basic properties.

**Lemma 3.10.** Suppose $H_q$ is a symmetric solution of the qth Newton step (10), and the sequences $\{R_k\}$, $\{Y_k\}$ and $\{H_{q_k}\}$ are generated by Algorithm 3.9. Then we have

$\text{tr} [Y_k^T (H_q - H_{q_k})] = \|R_k\|^2 \quad \text{for all} \quad k = 0, 1, \ldots$  

**Lemma 3.11.** Let $H_q$ be a symmetric solution of equation (10). Then for any starting symmetric matrix $H_{q_0}$, we have

$\text{tr} [Q_k^T (H_q - H_{q_k})] = \|R_k\|^2 \quad \text{for} \quad k = 0, 1, \ldots$

**Remark 3.12.** Lemma 3.11 implies that if there exists an integer $k$ such that $Q_k = 0$ but $R_k \neq 0$, then the matrix equation (10) is inconsistent over symmetric matrices.

**Lemma 3.13.** Suppose that the sequences $\{R_k\}$ and $\{Q_k\}$ are generated by Algorithm 3.9. Then we have

$\text{tr} (R_i^T R_j) = 0 \quad \text{and} \quad \text{tr} (Q_i^T Q_j) = 0 \quad \text{for} \quad i > j = 0, 1, \ldots, k, \quad k \geq 1.$

Similar to Theorem 3.7, we can prove the following theorem by using Lemmas 3.11, 3.13, and Remark 3.12.

**Theorem 3.14.** Suppose that the qth Newton step (10) is consistent. Then for any symmetric starting matrix $H_{q_0}$, its symmetric solution can be obtained by Algorithm 3.9.

From Newton’s method and the above theorem, we can easily prove the following result.
Theorem 3.15. Suppose that the matrix polynomial (2) has a symmetric solvent and each Newton step is consistent for a symmetric starting matrix \(X_0\). The sequence \(\{X_k\}\) is generated by Newton’s method with \(X_0\) such that
\[
\lim_{k \to \infty} X_k = S,
\]
and if the matrix \(S\) satisfies \(P(S) = 0\), then \(S\) is a symmetric solvent.

4. The BS solvents of \(Q(X)\) and \(P(X)\)

In this section, we consider Newton’s method for finding the BS solvents of the quadratic matrix equation (1) and matrix polynomial (2).

4.1. An iterative method for solving (9) over BS matrices

Before proposing the iterative method for finding the BS solution of (9), we give the following well-known results.

Lemma 4.1 ([11]). Assume that \(X\) is BS. Then for any \(n \times n\) real matrix \(Y\),
\[
\text{tr} \left\{ \frac{1}{4} ((Y + Y^T) + S_n (Y + Y^T) S_n)^T X \right\} = \text{tr} (Y^T X).
\]

Algorithm 4.2. The matrices \(A, B, C, X_q \in \mathbb{R}^{n \times n}\) are given, where \(X_q \in \mathbb{R}^{n \times n}\) is BS. Choose a BS starting matrix \(E_q \in \mathbb{R}^{n \times n}\).

\[
k = 0; \quad R_0 = -Q(X_q) - (AX_q + B)E_q - AE_q X_q
\]
\[
Z_0 = (AX_q + B)^T R_0 + A^T R_0 (X_q)^T
\]
\[
P_0 = \frac{1}{4} \left[ (Z_0 + Z_0^T) + S_n (Z_0 + Z_0^T) S_n \right]
\]
\[
\alpha_0 = \frac{\|R_0\|^2}{\|P_0\|^2}
\]

while \(R_k \neq 0\) or \(P_k \neq 0\)

\[
\alpha_k = \frac{\|R_k\|^2}{\|P_k\|^2}
\]
\[
E_{q+k} = E_{q+k-1} + \alpha_k P_k
\]
\[
R_{k+1} = -Q(X_q) - (AX_q + B)E_{q+k} - AE_{q+k} X_q
\]
\[
Z_{k+1} = (AX_q + B)^T R_{k+1} + A^T R_{k+1} (X_q)^T
\]
\[
\beta_k = \frac{\text{tr} (Z_{k+1}^T P_k)}{\|P_k\|^2}
\]
\[
P_{k+1} = \frac{1}{4} \left[ (Z_{k+1} + Z_{k+1}^T) + S_n (Z_{k+1} + Z_{k+1}^T) S_n \right] - \beta_k P_k
\]

end

Note that the matrices \(P_k\) and \(E_{q+k}\) are BS matrices in Algorithm 4.2.

Lemma 4.3. If the matrix \(E_q\) is a BS solution of the \(q\)th Newton step (9) and the sequences \(\{Z_k\}\), \(\{R_k\}\) and \(\{E_{q+k}\}\) are generated by Algorithm 4.2, then
\[
\text{tr} \left[ Z_k^T (E_q - E_{q+k}) \right] = \|R_k\|^2 \text{ for all } k = 0, 1, \ldots.
\]

Similarly as in Lemma 3.3, from Algorithm 4.2, we can prove the conclusion (14). By Lemma 4.3, we easily prove the following property.
Lemma 4.4. Assume that $E_q$ is a BS solution of (9). Then
\[
(15) \quad \text{tr} \left[ P_i^T (E_q - E_{q_0}) \right] = \| R_0 \|^2 \quad \text{for all } k = 0, 1, \ldots.
\]

Proof. When $k = 0$, from Algorithm 4.2, Lemmas 4.1 and 4.3, we have
\[
\text{tr} \left[ P_0^T (E_q - E_{q_0}) \right] = \text{tr} \left[ Z_0^T (E_q - E_{q_0}) \right] = \| R_0 \|^2.
\]
Assume that the conclusion (15) holds for $k = l$. Since \( \text{tr} \left[ P_l^T (E_q - E_{q_{l+1}}) \right] = \text{tr} \left[ P_l^T (E_q - E_q - \alpha_l P_l) \right] = \| R_l \|^2 - \alpha_l \| P_l \|^2 \) and by Lemma 4.1 and 4.3, we have
\[
\text{tr} \left[ P_{l+1}^T (E_q - E_{q_{l+1}}) \right] = \text{tr} \left[ Z_{l+1}^T (E_q - E_{q_{l+1}}) \right] - \beta_l \text{tr} \left[ P_l^T (E_q - E_{q_{l+1}}) \right] = \| R_{l+1} \|^2.
\]
Hence the conclusion (15) holds for $k = 0, 1, \ldots$ by the principle of induction. \( \square \)

Lemma 4.5. For the sequences \( \{ R_i \} \) and \( \{ P_i \} \) generated by Algorithm 3.1, we have
\[
(16) \quad \text{tr} \left( R_i^T R_j \right) = 0 \quad \text{and} \quad \text{tr} \left( P_i^T P_j \right) = 0 \quad \text{for } i > j = 0, 1, \ldots, k, \ k \geq 1.
\]

Proof. We prove (16) by induction.

Step 1. When $k = 1$, we have
\[
\text{tr} \left( R_1^T R_0 \right) = \text{tr} \left\{ [R_0 - \alpha_0 (AX_q + B) P_0 - \alpha_0 AP_0 X_q]^T R_0 \right\} \\
= \text{tr} \left( R_0^T R_0 \right) - \alpha_0 \left\{ P_0^T (AX_q + B)^T R_0 + A^T R_0 (X_q)^T \right\} \\
= \| R_0 \|^2 - \alpha_0 \text{tr} \left( P_0^T Z_0 \right) \\
= \| R_0 \|^2 - \alpha_0 \text{tr} \left( P_0^T P_0 \right) = 0,
\]
and
\[
\text{tr} \left( P_1^T P_0 \right) = \text{tr} \left( Z_1^T P_0 \right) - \beta_0 \text{tr} \left( P_0^T P_0 \right) = 0.
\]
Suppose that (16) holds for $k = l$. Then we have
\[
\text{tr} \left( R_{l+1}^T R_l \right) = \text{tr} \left\{ [R_l - \alpha_l (AX_q + B) P_l - \alpha_l AP_0 X_q]^T R_l \right\} \\
= \text{tr} \left( R_l^T R_l \right) - \alpha_l \text{tr} \left\{ P_l^T (AX_q + B)^T R_l + A^T R_l (X_q)^T \right\} \\
= \text{tr} \left( R_l^T R_l \right) - \alpha_l \text{tr} \left( P_l^T Z_l \right) \\
= \text{tr} \left( R_l^T R_l \right) - \alpha_l \text{tr} \left[ P_l^T (P_l + \beta_l P_{l-1}) \right] \\
= \| R_l \|^2 - \alpha_l \text{tr} \left( P_l^T P_l \right) \\
= 0,
\]
and
\[
\text{tr} \left( P_{l+1}^T P_l \right) = \text{tr} \left( Z_{l+1}^T P_l \right) - \beta_l \text{tr} \left( P_l^T R_l \right) = 0.
\]
Step 2. Assume that \( \text{tr} \left( R_l^T R_j \right) = 0 \) and \( \text{tr} \left( P_l^T P_j \right) = 0 \) for all \( j = 0, 1, \ldots, l-1 \). Then from Algorithm 4.2 and the above assumptions, we have

\[
\text{tr} \left( R_{l+1}^T R_j \right) = \text{tr} \left\{ [R_l - \alpha_l (AX_q + B) P_l - \alpha_l A P_l X_q]^T R_j \right\} \\
= \text{tr} \left( R_l^T R_j \right) - \alpha_l \text{tr} \left\{ P_l^T \left[ (AX_q + B)^T R_j + A^T R_j (X_q)^T \right] \right\} \\
= -\alpha_l \text{tr} \left( P_l^T Z_j \right) \\
= \alpha_l \text{tr} \left[ P_l^T (P_j + \beta_j P_{j-1}) \right] \\
= 0,
\]

and

\[
\text{tr} \left( P_{l+1}^T P_j \right) = \text{tr} \left( Z_{l+1}^T P_j \right) - \beta_j \text{tr} \left( P_l^T P_j \right) \\
= \text{tr} \left[ R_{l+1}^T [(AX_q + B) P_j + A P_j X_q] \right] \\
= \frac{1}{\alpha_j} \text{tr} \left[ R_{l+1}^T (R_j - R_{j+1}) \right] \\
= 0.
\]

Thus we complete the proof by Steps 1 and 2. \( \square \)

**Theorem 4.6.** Suppose that the \( q \)th Newton step (9) has a bisymmetric solution. Then for any bisymmetric starting matrix \( E_q \), its symmetric solution can be obtained, at most, in \( n^2 \) steps.

**Proof.** This proof is similar to that of Theorem 3.7. \( \square \)

**Theorem 4.7.** Suppose that the quadratic matrix equation (1) has a bisymmetric solvent and each Newton step is consistent for a bisymmetric starting matrix \( X_0 \). The sequence \( \{X_k\} \) is generated by Newton’s method with \( X_0 \) such that

\[
\lim_{k \to \infty} X_k = S,
\]

and if the matrix \( S \) satisfies \( Q(S) = 0 \), then \( S \) is a bisymmetric solvent.

**Proof.** Similar to Theorem 3.8, we can complete the proof by Newton’s method and Theorem 4.7. \( \square \)

### 4.2. An iterative method for finding the BS solution of equation (10)

**Algorithm 4.8.** Input \( n \times n \) real matrices \( A_0, A_1, \ldots, A_m \) and BS matrix \( X_q \in \mathbb{R}^{n \times n} \). Choose a BS starting matrix \( H_{q0} \in \mathbb{R}^{n \times n} \).

\[
k = 0; \quad R_0 = -P(X_q) - \left( \sum_{\nu=0}^{m-1} A_\nu X_q^{(m-1)-\nu} \right) H_{q0} - \cdots - A_0 H_{q0} X_q^{m-1}
\]

\[
Y_0 = \left( \sum_{\nu=0}^{m-1} A_\nu X_q^{(m-1)-\nu} \right)^T R_0 + \cdots + A_0^T R_0 (X_q^{m-1})^T \\
Q_0 = \frac{1}{4} \left[ (Y_0 + Y_0^T) + S_n (Y_0 + Y_0^T) S_n \right]
\]

while \( R_k \neq 0 \) or \( Q_k \neq 0 \).
Lemma 4.11. Suppose that the sequences \( \{R_k\} \) and \( \{Q_k\} \) are generated by Algorithm 4.8. Then we have

\[
H_{qk+1} = H_{qk} + \frac{\|R_k\|^2}{\|Q_k\|^2} Q_k
\]

\[
R_{k+1} = -P(X_q) - \left( \sum_{\nu = 0}^{m-1} A_{\nu} X^{(m-1) - \nu} \right) H_{qk+1} \cdots - A_0 H_{qk+1} X_q^{m-1}
\]

\[
Y_{k+1} = \left( \sum_{\nu = 0}^{m-1} A_{\nu} X^{(m-1) - \nu} \right)^T R_{k+1} \cdots + A_0^T R_{k+1} (X_q^{m-1})^T
\]

\[
Q_{k+1} = \frac{1}{2} \left[ (Y_{k+1} + Y_{k+1}^T) + S_n (Y_{k+1} + Y_{k+1}^T) S_n \right] - \frac{\text{tr}(Y_{k+1}^T Q_k)}{\|Q_k\|^2} Q_k
\]

end

Regarding Algorithm 4.8, we have the following basic properties.

Lemma 4.9. Suppose \( H_q \) is a BS solution of the \( q \)th Newton step (10), and the sequences \( \{R_k\} \), \( \{Y_k\} \) and \( \{H_{qk}\} \) are generated by Algorithm 4.8. Then we have

\[
\text{tr} \left[ Y_q^T (H_q - H_{qk}) \right] = \|R_k\|^2 \text{ for all } k = 0, 1, \ldots.
\]

Lemma 4.10. Let \( H_q \) be a BS solution of equation (10). Then for any starting BS matrix \( H_{q0} \), we have

\[
\text{tr} \left[ Q_k^T (H_q - H_{qk}) \right] = \|R_k\|^2 \text{ for } k = 0, 1, \ldots.
\]

Lemma 4.11. Suppose that the sequences \( \{R_i\} \) and \( \{Q_i\} \) are generated by Algorithm 4.8. Then we have

\[
\text{tr} (R_i^T R_j) = 0 \text{ and } \text{tr} (Q_i^T Q_j) = 0 \text{ for } i > j = 0, 1, \ldots, k, k \geq 1.
\]

Similar to Theorem 4.6, we can prove the following theorem by using Lemmas 4.10 and 4.11.

Theorem 4.12. Suppose that the \( q \)th Newton step (10) is consistent. Then for any BS starting matrix \( H_{q0} \), its BS solution can be obtained by Algorithm 4.8.

From Newton’s method and the above theorem, we easily prove the following main theorem.

Theorem 4.13. Suppose that the matrix polynomial (2) has a BS solvent and each Newton step is consistent for a BS starting matrix \( X_0 \). The sequence \( \{X_k\} \) is generated by Newton’s method with \( X_0 \) such that

\[
\lim_{k \to \infty} X_k = S,
\]

and if the matrix \( S \) satisfies \( P(S) = 0 \), then \( S \) is a bisymmetric solvent.

5. Numerical examples

In this section, we give some numerical experiments to present the convergence of our Newton’s method. Computations were done in MATLAB 7.1 and we regard the relative residuals \( \rho_Q(X_k) \) and \( \rho_P(X_k) \) as zeros if

\[
\rho_Q(X_k) = \frac{\|P(Q_i(X_k))\|}{\|A_0\| \|X_k\|^{m+1} + \|A_1\| \|X_k\|^{m+1} \cdots + \|A_m\| \|X_k\|^0} \leq \eta \mu,
\]

\[
\rho_P(X_k) = \frac{\|P(P_i(X_k))\|}{\|A_0\| \|X_k\|^{m+1} + \|A_1\| \|X_k\|^{m+1} \cdots + \|A_m\| \|X_k\|^0} \leq \eta \mu,
\]
where $n$ is the maximum size of $A$ and $A_0$, and $\mu = 2^{-53} \simeq 1.1102e-016$ is the unit round off. In Algorithms 3.1, 3.9, 4.2, and 4.8, the iteration will be terminated whenever $\|R_k\| < \epsilon = 1.0e-016$.

Now, we give four numerical examples with Fréchet derivatives that are all singular according to their respective starting matrices.

**Example 5.1.** Let the coefficients of the quadratic matrix equation $Q_1(X)$ be

\begin{equation}
A = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} -2 & 0 \\ -2 & 0 \end{bmatrix} \quad \text{and} \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}.
\end{equation}

Starting Newton’s method with Algorithm 3.1 with a symmetric matrix $X_0 = 1_2$, where $1_2$ denote the $2 \times 2$ identity matrix, we obtain the symmetric solvent of problem (17), that is, $X_{13} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. In this case, $\rho_{Q_1}(X_{13}) = 5.55e-017 < 2\mu$. The convergence results are presented in Figure 1, which confirms the conclusion of Theorem 3.8.

**Example 5.2.** For convenience, we consider a simple matrix polynomial of degree 3 given as

\begin{equation}
P_1(X) = Q_1(X)X = 0.
\end{equation}

Similarly as in the first example, the symmetric starting matrix is chosen to be $X_0 = 1_2$. Then, for this starting matrix $X_0$, the Fréchet derivative of problem (18)

\begin{align*}
P'_1(X_0) &= 1_2 \otimes (AX^2 + BX + C) + X^T \otimes (AX + B) + (X^2)^T \otimes A \\
&= 0_4,
\end{align*}

where $0_4$ is the $4 \times 4$ zero matrix, is singular. So Kratz and Stickel’s method cannot solve problem (18). But by using Newton’s method with Algorithm 3.9
and 13 iterative steps, we can obtain the symmetric solvent of equation (18) as follows:

\[ X_{13} = \begin{bmatrix} 1 & 0 \\ 0 & 4096 \end{bmatrix} \]

with

\[ \rho_{P_1}(X_{13}) = 1.16e - 017 < 2\mu. \]

This is a very simple example that verifies Theorem 3.15.

**Example 5.3.** We consider the quadratic matrix equation

\[ Q_2(X) = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} X^2 + \begin{bmatrix} -4 & 0 & -4 \\ -4 & 0 & -4 \\ -4 & 0 & -4 \end{bmatrix} X + \begin{bmatrix} 12 & 0 & 3 \\ 12 & 0 & 3 \\ 12 & 0 & 3 \end{bmatrix} = 0. \]

Choose the BS starting matrix \( X_0 = \begin{bmatrix} 2 & 0 & 4 \\ 0 & 6 & 0 \\ 4 & 0 & 2 \end{bmatrix} \). Applying Newton’s method with Algorithm 4.2, we obtain the BS solvent \( X_5 = \begin{bmatrix} 1 & 0 & 4 \\ 0 & 36 & 0 \\ 4 & 0 & 1 \end{bmatrix} \) with the corresponding relative residual \( \rho_{Q_2}(X_5) = 4.01e - 018 < 3\mu \). The obtained convergence results from our Newton’s method are shown in the first figure of Figure 2. By using Algorithm 4.2, we can get BS solutions of five Newton steps within 9 iterations. This is illustrated in the second figure of Figure 2. In this figure, we can see that using iterations 3, 5, 3, 5, and 4 yields the BS solutions of the 1st, 2nd, 3rd, 4th, and 5th Newton steps, respectively. This demonstrates the conclusion of Theorem 4.6 for problem (19).

**Example 5.4.** We consider a matrix polynomial of degree 3 given as follows:

\[ P_2(X) = \begin{bmatrix} -2 & 1 \\ -2 & 1 \end{bmatrix} X^3 + \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} X^2 + \begin{bmatrix} 6 & -1 \\ 6 & -1 \end{bmatrix} X + \begin{bmatrix} 14 & 14 \\ 14 & 14 \end{bmatrix} = 0. \]

By applying Newton’s method with Algorithm 4.8 for the BS starting matrix \( X_0 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \), the BS solvent \( X_{10} = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} \) of \( P_2(X) \) can be obtained. The results
Table 1. Comparison of the relative residuals from the Kratz and Stickel’s method with those from our Newton’s method for problem (20).

<table>
<thead>
<tr>
<th>No.ite</th>
<th>Kratz and Stickel’s method</th>
<th>Our Newton’s method</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.00e + 000</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2.66e − 001</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2.30e − 001</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1.69e − 001</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>fail</td>
<td>8.71e − 002</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>2.13e − 002</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>1.20e − 003</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>3.70e − 016</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>3.53e − 011</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>6.18e − 017</td>
</tr>
</tbody>
</table>

are provided in Table 1. Here, Kratz and Stickel’s method also fails to find the BS solvent since the Fréchet derivative for the starting matrix $X_0$ is singular.

Example 5.5. Our final example is

$$Q_3(X) = AX^2 + BX + C,$$

where

$$A = I_5, \quad B = \begin{bmatrix} 20 & -10 & 0 & 0 & 0 \\ -10 & 30 & -10 & 0 & 0 \\ 0 & -10 & 30 & -10 & 0 \\ 0 & 0 & -10 & 30 & -10 \\ 0 & 0 & 0 & -10 & 20 \end{bmatrix},$$


The equation $Q_3(X)$ has a symmetric solvent $S$, where $S_{ij} = \min\{i, j\}$. Our iteration converges to $S$ with the starting symmetric matrices $I_5$ as well as with a matrix whose entries are all 1.

6. Conclusion

In this work, four iterative methods are introduced for solving Newton steps (9) and (10) over symmetric and BS matrices, respectively. Then we incorporated the iterative methods into Newton’s method to find the symmetric and
BS solvents of the quadratic matrix equation and the matrix polynomial. The contributions of this paper are as follows:

1. Our Newton’s method can solve $Q(X)$ and $P(X)$ even when the Fréchet derivative is singular;
2. Our Newton’s method can find the symmetric and BS solvents for any given symmetric and BS starting matrices, respectively.

References