Tanzania Journal of Science 47(4): 1392-1401, 2021
ISSN 0856-1761, e-ISSN 2507-7961

Inversion–free Iterative Method for Finding Symmetric Solution of the Nonlinear Matrix Equation $X - A^{*}X^{q}A = I\ (q \geq 2)$

Chacha Stephen Chacha
Department of Mathematics, Physics and Informatics, Mkwawa University College of Education, University of Dar es Salaam, P. O. Box 2513, Iringa, Tanzania
E-mail: chchstephen@yahoo.com, chchstephen@muce.ac.tz
Received 21 Jun 2021, Revised 22 Sep 2021, Accepted 25 Sep 2021, Published Oct 2021
DOI: https://dx.doi.org/10.4314/tjs.v47i4.5

Abstract

In this paper, we propose the inversion free iterative method to find symmetric solution of the nonlinear matrix equation $\boldsymbol{X - A^{*}X^{q}A = I\ (q \geq 2)}$, where X is an unknown symmetric solution, A is a given Hermitian matrix and q is a positive integer. The convergence of the proposed method is derived. Numerical examples demonstrate that the proposed iterative method is quite efficient and converges well when the initial guess is sufficiently close to the approximate solution.

Keywords: Symmetric solution, nonlinear matrix equation, inversion free, iterative method.

Introduction

The nonlinear matrix equation

$$X - A^{*}X^{q}A = I(q \geq 2) \qquad (1)$$

is considered, where A is the given square matrix, I is an identity matrix and X is an unknown Symmetric Solution (SS) to be determined. Different approaches and solutions of the generic nonlinear matrix equation of the form

$$X + A^{*}\mathfrak{F}(X)A = Q(Q > 0) \qquad (2)$$

has been widely explored for different $\mathfrak{F}(X)$ (see, Zhang et al. 2011, Huang and Ma 2015, Gao 2016, Chacha and Naqvi 2018 and Chacha and Kim 2019a). Equation (1) arises in modelling of physical processes in statistics, control theory, stochastic filtering, Kalman filtering, quasi-birth-death processes among others.

Symmetric solution has piqued the interest of numerous authors because of its practical importance. It has been extensively studied for different matrix equations (Peng et al. 2006, Sheng and Chen 2010, Huan 2011, Dehghan and Hajarian 2011, Chacha and Kim 2019b and the references therein).

There are numerous renowned iterative methods for solving Equation (1) such as variants of fixed point method and Newton's method. Newton's method is the best when exploring Elementwise Minimal Nonnegative Solution (EMNS), while fixed point method fits best when exploring positive definite solution. Recently, Chacha and Kim (2019a) explored the EMNS of the nonlinear matrix equation $X - A^{*}X^{q}A = I\ (q \geq 2)$, which is Equation (1) for $q = 2$ by employing pure Newton's method. However, Pure Newton's method does not guarantee the existence of symmetric solution.

To the best of our knowledge, the inversion free iterative method has not been exploited in finding SS of Equation (1). Many authors have applied fixed point method and its variants to find Hermitian positive definite solution of Equation (1) for $q = -1$ (Erfanifar et al. 2020 and the references therein).

In this paper, we are interested in investigating SS of Equation (1). An inversion free iterative method which guarantees the existence of SS for Equation (1) is proposed. The only big advantage regarding this method is that there is no computation of inverse of Kronecker Fréchet derivative and ensures the existence of SS.

The following notations and definitions will be used throughout this paper: $\rho(\blacksquare)$ stands for spectral radius; $vec\,(A) = [a_1{}^T, a_2{}^T. \cdots, a_n{}^T]^T$ is the column-wise vector representation of matrix A and $vec(AXB)$=$(B^T \otimes A)vec(X)$; $C \otimes D = [c_{ij}]B$ is the tensor or Kronecker product of the matrices C and D; $\overline{B_\epsilon(X_0)}$ stands for a closed ball with a radius ϵ and centreX_0; A^T stands for transpose of matrix A; 0 represents square null matrix; $\oplus$ stands for Kronecker sum; $\|\blacksquare\| \coloneqq \|\blacksquare\|_2$ is the spectral norm; $\|\blacksquare\|_F$ stands for Frobenius norm and for any matrices $C, D \in \mathbb{R}^{m\times n}, C \geq D (C > D)$ if $[c_{ij}] \geq [d_{ij}]([c_{ij}] > [d_{ij}])$ for all i, j.

Definition 1
For a general function $F: C^{n\times n} \to C^{n\times n}$, Newton's method for the solution of $\mathcal{F}(X) = 0$ is specified by an initial approximation X_0 and the recurrence
$X_{k+1} = X_k - F'(X_k)^{-1}F(X_k)$, for all $k = 0, 1, 2, \cdots$, where F' denotes the Fr é chet derivative.

Definition 2
Let matrix A be $m \times m$ square matrix, A is a Z – matrix if all its off-diagonal elements are non-positive.

Definition 3
A matrix $A \in \mathbb{R}^{m\times m}$ is an M-matrix if $A = sI - B$ for some nonnegative B and s with s> $\rho(B)$.

Lemma 1 (Chacha and Kim 2019a)
For a Z-matrix, the following are equivalent:

i. A is non-singular M-matrix.
ii. A^{-1} is nonnegative.
iii. $Av > 0\ (\geq 0)$ for some vector $v > 0\ (\geq 0)$.
iv. All eigenvalues of A have positive real parts.

Materials and Methods

Newton's method

Let Equation (1) be represented by the map
$$\mathcal{F}(X) = X - A^*X^qA - I = 0 \qquad (3)$$

From Equation (3), we see the that Fréchet derivative $\mathcal{F}'_X$ is a linear operator $\mathcal{F}'_X[E]: \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$, defined by

$$\mathcal{F}'_X[E] = E - \sum_{\mu=1}^{q} A^*\, X^{q-\mu} E X^{\mu-1} A. \qquad (4)$$

From Equation (4) we have
$\text{vec}(\mathcal{F}'_X[E]) = \mathcal{D}_X \text{vec}(E)$, where
$$\mathcal{D}_X = I_{n^2} - \sum_{\mu=1}^{q} (X^{\mu-1}A)^T \otimes (A^*X^{q-\mu}).$$

Lemma 2
Suppose that
$0 \leq \sum_{\mu=1}^{q} (X^{\mu-1}A)^T \otimes (A^*X^{q-\mu}) < I_{n^2}$, then $\mathcal{D}_X$ is a nonsingular M-matrix.

Proof: The proof is straight forward from Definitions 2, 3 and Lemma 1. Thus, the proof is omitted here.
Since $\mathcal{D}_X$ is invertible. It implies that $\mathcal{F}'_X$ regular. Thus, Newton's step E is calculated in
$$E - \sum_{\mu=1}^{q} A^*\, X^{q-\mu} E X^{\mu-1} A = -\mathcal{F}(X) \qquad (5)$$

Algorithm 1 (Newton's method for Equation (1))
Step 1: Given symmetric matrix A and initial guess X_0
Step 2: Solve Newton's step in
$$E - \sum_{\mu=1}^{q} A^*\, X^{q-\mu} E X^{\mu-1} A = -\mathcal{F}(X).$$
Step 3:$X_{i+1} = X_i + E_i$, for all $i = 0, 1, 2, \cdots$
Step 4: Check if $\|\mathcal{F}(X_k)\|_F \leq n.$eps, where n is the size of matrix A and eps = 2.22×10^{-16}, otherwise go to **Step 2**.
Step 5: Display the approximate solution X.

Remark 1
We see that Newton's method for Equation (1) is applicable if $\mathcal{D}_X$ is nonsingular. Now suppose that $\sum_{\mu=1}^{q}(X^{\mu-1}A)^T \otimes (A^* X^{q-\mu}) = I_{n^2}$. In this case, Newton's method will not work. Moreover, Algorithm 1 does not guarantee the existence of SS. Now we provide an inversion free iterative method to calculate the Newton's step. The proposed iterative method ensures the existence of SS. We also derive necessary condition for existence of SS.

Algorithm 2: Inversion free iterative method for solving Newton's step

1. Let $A \in \mathbb{R}^{n\times n}$ and symmetric $X_p \in \mathbb{R}^{n\times n}$, choose initial symmetric Newton's step $E_{pk} \in \mathbb{R}^{n\times n}$
2. For $k = 0$, evaluate
   i. $\mathcal{R}_0 = -\mathcal{F}(X_p) - [E_{p0} - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} E_{p0} {X_p}^{\mu-1} A]$
   ii. $Y_0 = \mathcal{R}_0 - \sum_{\mu=1}^{q} \left(A^* {X_p}^{q-\mu}\right)^T \mathcal{R}_0 \left({X_p}^{\mu-1} A\right)^T$
   iii. $Q_0 = \frac{1}{2}\left(Y_0 + {Y_0}^T\right)$
3. While $\mathcal{R}_k \neq 0$, evaluate
   i. $\alpha_k = \frac{\|\mathcal{R}_k\|^2}{\|Q_k\|^2}$
   ii. $E_{pk+1} = E_{pk} + \alpha_k Q_k$
   iii. $\mathcal{R}_{k+1} = -\mathcal{F}(X_p) - [E_{pk+1} - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} E_{pk+1} {X_p}^{\mu-1} A]$
   iv. $Y_{k+1} = \mathcal{R}_{k+1} - \sum_{\mu=1}^{q} \left(A^* {X_p}^{q-\mu}\right)^T \mathcal{R}_{k+1} \left({X_p}^{\mu-1} A\right)^T$
   v. $\beta_k = \frac{\|\mathcal{R}_{k+1}\|^2}{\|\mathcal{R}_k\|^2}$
   vi. $Q_{k+1} = \frac{1}{2}\left(Y_{k+1} + {Y_{k+1}}^T\right) + \beta_k Q_k$
4. end

Lemma 3 (Chacha and Kim 2019b**)**
For any symmetric solution X, it holds that
$\mathrm{tr}\left[{}^1\!/_2\, (M + M^T)^T X\right] = \mathrm{tr}(M^T X)$, where M is any arbitrary $n \times n$ real matrix.

Remark 2
In **Algorithm 2**, the sequence of matrices Q_k and E_{pk} are symmetric for all $k = 0,1,\cdots$.
Based on Algorithm 2, we derive the following results.

Lemma 4
Let E_p be a symmetric solution of the p^{th} Newton's iteration (5), and the sequences $\{Y_k\}$, $\{\mathcal{R}_k\}$, and $\{E_{pk}\}$ be generated by Algorithm 2. Then,
$\mathrm{tr}\left[{Y_k}^T\left(E_p - E_{pk}\right)\right] = \|\mathcal{R}_k\|^2$, for all $k = 0, 1, \cdots$.

Proof: From Algorithm 2, it follows that

$$\mathrm{tr}\big[{Y_k}^T\big(E_p - E_{pk}\big)\big] = \mathrm{tr}\Big\{\Big[\mathcal{R}_k - \sum_{\mu=1}^{q}\big(A^* {X_p}^{q-\mu}\big)^T \mathcal{R}_k\big({X_p}^{\mu-1}A\big)^T\Big]^T \Big(\big(E_p - E_{pk}\big)\Big)\Big\}$$

$$= \mathrm{tr}\big\{{\mathcal{R}_k}^T\big[E_p - E_{pk} - \sum_{\mu=1}^{q}\big(A^* {X_p}^{q-\mu}\big)\big(E_p - E_{pk}\big)\,\big({X_p}^{\mu-1}A\big)\big]\big\}$$

$$= \mathrm{tr}\big\{{\mathcal{R}_k}^T\big[-\mathcal{F}\big(X_p\big) - [E_{pk} - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} E_{pk} {X_p}^{\mu-1} A]\big]\big\}$$

$$= \mathrm{tr}\big\{{\mathcal{R}_k}^T \mathcal{R}_k\big\} = \|\mathcal{R}_k\|^2.$$

This marks the end of the proof.

Lemma 5

Let E_p be a symmetric solution of the p^{th} Newton's iteration (5). Then,
$\mathrm{tr}\big[{Q_k}^T\big(E_p - E_{pk}\big)\big] = \|\mathcal{R}_k\|^2,$ for all $k = 0, 1, \cdots.$

Proof: We prove via mathematical induction.
When $k = 0$, from Algorithm 2, Lemmas 3 and 4, we have

$$\mathrm{tr}\big[{Q_k}^T\big(E_p - E_{pk}\big)\big] = \mathrm{tr}\Big[\tfrac{1}{2}(Y_0 + Y_0)^T\big(E_p - E_{pk}\big)\Big]$$

$$= \mathrm{tr}\big[(Y_0)^T\big(E_p - E_{pk}\big)\big] = \|\mathcal{R}_k\|^2. \qquad (6)$$

Now, suppose statement (6) holds for $h = k$, it follows that

$$\mathrm{tr}\big[{Q_{h+1}}^T\big(E_p - E_{ph+1}\big)\big] = \mathrm{tr}\Big[\Big[\tfrac{1}{2}(Y_{h+1} + Y_{h+1})^T + \beta_h Q_h\Big]^T \big(E_p - E_{ph+1}\big)\Big]$$

$$= \mathrm{tr}\big[{Y_{h+1}}^T\big(E_p - E_{ph+1}\big)\big] + \beta_h \mathrm{tr}\big[Q_h\big(E_p - E_{ph+1}\big)\big]$$

$$= \|\mathcal{R}_{h+1}\|^2 + \beta_h \mathrm{tr}\big[{Q_h}^T\big(E_p - E_{ph} - \alpha_h Q_h\big)\big]$$

$$= \|\mathcal{R}_{h+1}\|^2 + \beta_h \mathrm{tr}\big[{Q_h}^T\big(E_p - E_{ph}\big)\big] + \beta_h \alpha_h \|Q_h\|^2$$

$$= \|\mathcal{R}_{h+1}\|^2 + \beta_h\|\mathcal{R}_h\|^2 - \beta_h\|\mathcal{R}_h\|^2 = \|\mathcal{R}_{h+1}\|^2 \blacksquare$$

This marks the end of Lemma 5.

Remark 3

From Lemma 5, for Newton iteration (5) to have symmetric solution, the sequences $\{\mathcal{R}_k\}$, and $\{Q_k\}$ generated by Algorithm 2 should non-zero.

Lemma 6

For the sequences $\{\mathcal{R}_k\}$ and $\{Q_k\}$ generated by Algorithm 2, we have $\mathrm{tr}\big({\mathcal{R}_k}^T\mathcal{R}_k\big) = 0$ for all $k > j = 0, 1, \cdots l,\quad l \geq 1.$ (7)

Proof: We prove via mathematical induction. Case I: When $l = 1$, it follows that

$$\mathrm{tr}\big({\mathcal{R}_1}^T\mathcal{R}_0\big) = \Big\{\big[-\mathcal{F}\big(X_p\big) - [E_{p1} - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} E_{p1} {X_p}^{\mu-1} A]\big]^T \mathcal{R}_0\Big\}$$

$$= \quad \mathrm{tr}\left\{\left[-\mathcal{F}(X_p) - [E_0 - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} E_{p1} {X_p}^{\mu-1} A\right.\right.$$

$$\left.\left. + \quad \alpha_0 \left(Q_0 - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} Q_0 {X_p}^{\mu-1} A \right)]\right]^T \mathcal{R}_0 \right\}$$

$$= \mathrm{tr}\left\{\left[\mathcal{R}_0 - \alpha_0 \left(Q_0 - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} Q_0 {X_p}^{\mu-1} A \right)\right]^T \mathcal{R}_0 \right\}$$

$$= \|\mathcal{R}_0\|^2 - \mathrm{tr}\left\{ \alpha_0 \left({Q_0}^T \left[\mathcal{R}_0 - \sum_{\mu=1}^{q} \left(A^* {X_p}^{q-\mu}\right)^T \mathcal{R}_0 \left({X_p}^{\mu-1} A\right)^T \right]\right)\right\}$$

$$= \|\mathcal{R}_0\|^2 - \alpha_0 \mathrm{tr}\left[{Q_0}^T Y_0\right] = \|\mathcal{R}_0\|^2 - \alpha_0 \mathrm{tr}\left[{Q_0}^T \, {}^1\!/_2 \left(Y_0 + {Y_0}^T\right)\right]$$

$$= \|\mathcal{R}_0\|^2 - \alpha_0 \mathrm{tr}\left[{Q_0}^T Q_0\right] = 0, \qquad \text{and}$$

$$\mathrm{tr}\left({Q_1}^T Q_0\right) \quad = \mathrm{tr}\left[\left[{}^1\!/_2 \left(Y_0 + {Y_0}^T\right) + \beta_0 Q_0\right]^T Q_0\right]$$

$$= \mathrm{tr}\left[{Y_1}^T Q_0\right] + \beta_0 \mathrm{tr}\left[{Q_0}^T Q_0\right]$$

$$= \mathrm{tr}\left[\left[\mathcal{R}_1 - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} \mathcal{R}_1 {X_p}^{\mu-1} A\right]^T Q_0\right] + \beta_0 \|Q_0\|^2$$

$$= \mathrm{tr}\left[{\mathcal{R}_1}^T \left[Q_0 - \sum_{\mu=1}^{q} \left(A^* {X_p}^{q-\mu}\right) Q_0 \left({X_p}^{\mu-1} A\right)\right]\right] + \frac{\|\mathcal{R}_1\|^2}{\|\mathcal{R}_0\|^2} \|Q_0\|^2$$

$$= \mathrm{tr}\left[{\mathcal{R}_1}^T \left[{}^1\!/_{\alpha_0} \left(E_{p1} - E_{p0}\right) - {}^1\!/_{\alpha_0} \sum_{\mu=1}^{q} \left(A^* {X_p}^{q-\mu}\right) \left(E_{p1} - E_{p0}\right)\left({X_p}^{\mu-1} A\right)\right]\right]$$

$$+ \frac{\|\mathcal{R}_1\|^2}{\|\mathcal{R}_0\|^2} \|Q_0\|^2$$

$$= {}^1\!/_{\alpha_0} \mathrm{tr}\left[{\mathcal{R}_1}^T \left[\left(E_{p1} - E_{p0}\right) - \sum_{\mu=1}^{q} \left(A^* {X_p}^{q-\mu}\right) \left(E_{p1} - E_{p0}\right)\left({X_p}^{\mu-1} A\right)\right]\right] + \frac{\|\mathcal{R}_1\|^2}{\|\mathcal{R}_0\|^2} \|Q_0\|^2$$

$$= {}^1\!/_{\alpha_0} \mathrm{tr}\left[{\mathcal{R}_1}^T [(\mathcal{R}_0 - \mathcal{R}_1)]\right] + \frac{\|\mathcal{R}_1\|^2}{\|\mathcal{R}_0\|^2} \|Q_0\|^2$$

$$= {}^1\!/_{\alpha_0} \left(\mathrm{tr}\left[{\mathcal{R}_1}^T \mathcal{R}_0\right] - \mathrm{tr}\left[{\mathcal{R}_1}^T \mathcal{R}_1\right]\right) + \frac{\|\mathcal{R}_1\|^2}{\|\mathcal{R}_0\|^2} \|Q_0\|^2$$

$$= -{}^1\!/_{\alpha_0} \left(\mathrm{tr}\left[{\mathcal{R}_1}^T \mathcal{R}_1\right]\right) + \frac{\|\mathcal{R}_1\|^2}{\|\mathcal{R}_0\|^2} \|Q_0\|^2 = -\frac{\|\mathcal{R}_1\|^2}{\|\mathcal{R}_0\|^2} \|Q_0\|^2 + \frac{\|\mathcal{R}_1\|^2}{\|\mathcal{R}_0\|^2} \|Q_0\|^2 = 0.$$

Now assume that Equation (7) holds for $l = s$. It follows that

$$\mathrm{tr}\left({\mathcal{R}_{s+1}}^T \mathcal{R}_s\right) = \left\{\left[[\mathcal{R}_s - \alpha_s\left(Q_s - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} Q_s {X_p}^{\mu-1} A\right)]\right]^T \mathcal{R}_s\right\}$$

$$= \mathrm{tr}\left({\mathcal{R}_s}^T \mathcal{R}_s\right) - \alpha_s \mathrm{tr}\left[\left[\left(Q_s - \sum_{\mu=1}^{q} A^* {X_p}^{q-\mu} Q_s {X_p}^{\mu-1} A\right)\right]^T \mathcal{R}_s\right]$$

$$= \|\mathcal{R}_s\|^2 - \alpha_s \mathrm{tr}\left[Q_s{}^T \left(\sum_{\mu=1}^{q} \left(A^* X_p{}^{q-\mu}\right)^T \mathcal{R}_s \left(X_p{}^{\mu-1} A\right)^T\right)\right]$$

$$= \|\mathcal{R}_s\|^2 - \alpha_s \mathrm{tr}\left[Q_s{}^T Y_s\right]$$

$$= \|\mathcal{R}_s\|^2 - \alpha_s \mathrm{tr}\left[Q_s{}^T\, {}^1\!/_2 \left(Y_s + Y_s{}^T\right)\right]$$

$$= \|\mathcal{R}_s\|^2 - \alpha_s \|Q_s\|^2 + \alpha_s \beta_{s-1} \mathrm{tr}\left[Q_s{}^T Q_{s-1}\right] = 0.$$

Moreover,

$$\mathrm{Tr}\left(Q_{s+1}{}^T Q_s\right) = \mathrm{tr}\left[\left[{}^1\!/_2 \left(Y_{s+1} + Y_{s+1}{}^T\right) + \beta_s Q_s\right]^T Q_s\right] = \mathrm{tr}\left[Y_{s+1}{}^T Q_s\right] + \beta_s \mathrm{tr}\left[Q_s{}^T Q_s\right]$$

$$= \mathrm{tr}\left[\left[\mathcal{R}_{s+1} - \sum_{\mu=1}^{q} A^* X_p{}^{q-\mu} \mathcal{R}_{s+1} X_p{}^{\mu-1} A\right]^T Q_s\right] + \beta_s \|Q_s\|^2$$

$$= \mathrm{tr}\left[\mathcal{R}_{s+1}{}^T \left[Q_s - \sum_{\mu=1}^{q} \left(A^* X_p{}^{q-\mu}\right) Q_s \left(X_p{}^{\mu-1} A\right)\right]\right] + \beta_s \|Q_s\|^2$$

$$= \mathrm{tr}\left[\mathcal{R}_{s+1}{}^T \left[{}^1\!/_{\alpha_0} \left(\mathcal{R}_s - \mathcal{R}_{s+1}\right)\right]\right] + \beta_s \|Q_s\|^2$$

$$= {}^1\!/_{\alpha_0} \mathrm{tr}\left[\mathcal{R}_{s+1}{}^T \left[\left(\mathcal{R}_s - \mathcal{R}_{s+1}\right)\right]\right] + \frac{\|\mathcal{R}_{s+1}\|^2}{\|\mathcal{R}_s\|^2} \|Q_s\|^2$$

$$= {}^1\!/_{\alpha_0} \left(\mathrm{tr}\left[\mathcal{R}_{s+1}{}^T \mathcal{R}_s\right] - \mathrm{tr}\left[\mathcal{R}_{s+1}{}^T \mathcal{R}_{s+1}\right]\right) + \frac{\|\mathcal{R}_{s+1}\|^2}{\|\mathcal{R}_s\|^2} \|Q_s\|^2$$

$$= {}^{-1}\!/_{\alpha_0} \left(\mathrm{tr}\left[\mathcal{R}_{s+1}{}^T \mathcal{R}_{s+1}\right]\right) + \frac{\|\mathcal{R}_{s+1}\|^2}{\|\mathcal{R}_s\|^2} \|Q_s\|^2$$

$$= -\frac{\|\mathcal{R}_{s+1}\|^2}{\|\mathcal{R}_s\|^2} \|Q_s\|^2 + \frac{\|\mathcal{R}_{s+1}\|^2}{\|\mathcal{R}_s\|^2} \|Q_s\|^2 = 0.$$

Thus, we have $\mathrm{tr}\left[\mathcal{R}_k{}^T \mathcal{R}_{k-1}\right] = 0$ and $\mathrm{tr}\left[Q_k{}^T Q_{k-1}\right] = 0$, for all $k = 0, 1, \cdots, l$.
We now assume that $\mathrm{tr}\left[\mathcal{R}_s{}^T \mathcal{R}_j\right] = 0$ and $\mathrm{tr}\left[Q_s{}^T Q_j\right] = 0$, for all $j = 0, 1, \cdots, l-1$.
By Algorithm 2 and Lemma 3, in line with assumptions made herein, it follows that

$$\mathrm{tr}\left(\mathcal{R}_{s+1}{}^T \mathcal{R}_j\right) = \left\{\left[\left[\mathcal{R}_s - \alpha_s \left(Q_s - \sum_{\mu=1}^{q} A^* X_p{}^{q-\mu} Q_s X_p{}^{\mu-1} A\right)\right]\right]^T \mathcal{R}_j\right\}$$

$$= \mathrm{tr}\left(\mathcal{R}_s{}^T \mathcal{R}_j\right) - \alpha_s \mathrm{tr}\left[Q_s{}^T \left(\mathcal{R}_j - \sum_{\mu=1}^{q} \left(A^* X_p{}^{q-\mu}\right)^T \mathcal{R}_j \left(X_p{}^{\mu-1} A\right)^T\right)\right]$$

$$= \mathrm{tr}\left(\mathcal{R}_s{}^T \mathcal{R}_j\right) - \alpha_s \mathrm{tr}\left[Q_s{}^T Y_j\right]$$

$$= 0 - \alpha_s \mathrm{tr}\left[Q_s{}^T\, {}^1\!/_2 \left(Y_j + Y_j{}^T\right)\right]$$

$$= 0 - \alpha_s \mathrm{tr}\left[Q_s{}^T \left(Q_s - \beta_{j-1} Q_{j-1}\right)\right] = 0.$$

Finally, we prove that $\mathrm{tr}\left(Q_{s+1}{}^T Q_j\right) = 0$.

$$\mathrm{tr}\left(Q_{s+1}{}^T Q_j\right) = \mathrm{tr}\left[\left[{}^1\!/_2 \left(Y_{s+1} + Y_{s+1}{}^T\right) + \beta_s Q_s\right]^T Q_j\right]$$

$$= \mathrm{tr}\left[Y_{s+1}{}^T Q_j\right]$$

$$= \mathrm{tr}\left[\left[\mathcal{R}_{s+1} - \sum_{\mu=1}^{q} A^* X_p{}^{q-\mu} \mathcal{R}_{s+1} X_p{}^{\mu-1} A\right]^T Q_j\right]$$

$$= \mathrm{tr}\left[\mathcal{R}_{s+1}{}^T \left[Q_j - \sum_{\mu=1}^{q} \left(A^* X_p{}^{q-\mu}\right) Q_j \left(X_p{}^{\mu-1} A\right)\right]\right]$$

$$= \mathrm{tr}\left[\mathcal{R}_{s+1}{}^{T}\left[{}^{1}/_{\alpha_s}\left(\mathcal{R}_j - \mathcal{R}_{j+1}\right)\right]\right] = 0 \text{ for all } j = 0,1,\cdots,s-1.$$

From Lemma 6, it is easy to see that if $k > 0$, and $\mathcal{R}_i \neq 0$ for all $i = 0,1,\cdots,k$. Then, sequences $\mathcal{R}_i$ and $\mathcal{R}_j$ generated by Algorithm 2 are orthogonal for $i \neq j$.

Remark 4

If there exists a positive number $k > 0$ such that $\mathcal{R}_i \neq 0$ for all $i = 0,1,\cdots,k$ in Algorithm 2, then, the matrices $\mathcal{R}_i$ and $\mathcal{R}_j$ are perpendicular for $i \neq j$ from Lemma 6.

Theorem 1

Presume that the pth Newton's iteration (5) has symmetric solution. Then, for any initial matrix E_{p0} its symmetric solution can be obtained within at most n^2 iterative steps.

Proof: From Lemma 6, suppose that $\mathcal{R}_k \neq 0$ for $k = 0,1,\cdots,n^2-1$. We see that the set $\{\mathcal{R}_0, \mathcal{R}_1,\cdots,\mathcal{R}_{n^2-1}\}$ forms an orthogonal basis of the finite dimension matrix space $\mathbb{R}^{n\times n}$. Since the p^{th} Newton's iteration (5) has SS, then from Remark 4, it is certain that there exist a positive integer k such that $Q_k \neq 0$. E_{Pn^2} and $\mathcal{R}_{n^2}$ can be evaluated in Algorithm 2, and from Lemma 6 we know that $\mathrm{tr}\left[\mathcal{R}_{n^2}{}^{T}\mathcal{R}_k\right] = 0$. However, we know that $\left[\mathcal{R}_{n^2}{}^{T}\mathcal{R}_k\right] = 0$ is valid if $\mathcal{R}_{n^2} = 0$, this implies that E_{Pn^2} is the solution of iteration (5).

It is now high time to prove the convergence of Algorithm 2 to SS based on the established results.

Theorem 2

Assume that Equation (1) has SS and each Newton's iteration is consistent for symmetric initial guess X_0. The sequence $\{X_k\}$ is generated by Algorithm 1 with X_0 such that $\lim_{k\to\infty} X_k = X_p$, and X_p satisfies $\mathcal{F}(X_p) = 0$, then, X_p is the SS of Equation (1).

Proof: Let E_0 be the initial SS of Algorithm 2, then, it follows that $E_1 = E_0 + \alpha_0 Q_0$. We know that matrix sequence generated by Q_k is symmetric for $k = 0,1,\cdots$ and α_k is a positive number. Thus, From Algorithm 1 and Theorem 1, we have $X_{k+1} = X_k + E_k$.

Since $E_k{}'$s and $X_{k+1}{}'$ s are symmetric matrices, then the sequence $\{X_k\}$ converges to symmetric matrix X_p satisfying $\mathcal{F}(X_p) = 0$. Then, X_p is the SS of Equation (1).

Results and Discussion

In this section, we use numerical tests to illustrate the effectiveness of the algorithm developed to find SS of Equation (1). Our experiments were done in MATLAB R2015a and the loops were terminated whenever the error, $\|F(X_k)\|_F \leq 10^{-06}$. Summaries of results are presented in Table 1, Table 2 and Table 3.

Example 1

We now consider a matrix used in a model for the population of the bilby by Bean, Bright, Latouche, Pearce, Pollett, and Taylor (1987) for the quasi-stationary behaviour of quasi-birth-death processes. The bilby is an endangered Australian marsupial. Define the 5×5 matrix $B = \beta A_2{}^{T}$, where $\beta = 0.5$ and $d = [0,\ 0.5,\ 0.55,\ 0.8,\ 1]$ is the vector of probability that the population moves down a level given phase j and $g = 0.2$. We now have Equation (1) with a symmetric matrix given by $A = 0.5(B^T + B)\delta$, where $\delta = \{0.1,\ 0.001, 0.0001\}$.

$$A_2 = Q(g,d) = \begin{pmatrix} gd_1 & (1-g)d_1 & 0 & 0 & 0 \\ gd_2 & 0 & (1-g)d_2 & 0 & 0 \\ gd_3 & 0 & 0 & (1-g)d_3 & 0 \\ gd_4 & 0 & 0 & 0 & (1-g)d_4 \\ gd_5 & 0 & 0 & 0 & (1-g)d_5 \end{pmatrix}$$

Table 1 summarizes the findings obtained by incorporating Algorithm 2 into Algorithm 1 and solving Equation (1).

Table 1: Results summary for **Example 1**

δ	Iteration allowed	Iteration executed	Error	$\rho(A)$
0.1	100	Over 100	$1.0 \times 10^{+01}$	0.0460
0.01	100	Over 100	$9.90 \times 10^{+00}$	0.0046
0.001	100	1	1.03×10^{-07}	0.0004

The approximate SS for Equation (1) is

$$X_1 = \begin{pmatrix} 1.000000104440213 & 0.000000001336501 & 0.000000003353403 & 0.000000001858951 & 0.000000004082401 \\ 0.000000001336501 & 1.000000108142929 & 0.000000000334125 & 0.000000005832005 & 0.000000000060749 \\ 0.000000003353403 & 0.000000000334125 & 1.000000114087321 & 0.000000000534598 & 0.000000008620427 \\ 0.000000001858951 & 0.000000005832005 & 0.000000000534598 & 1.000000122078983 & 0.000000031201206 \\ 0.000000004082401 & 0.000000000060749 & 0.000000008620427 & 0.000000031201206 & 1.000000193192948 \end{pmatrix}.$$

Example 2

We consider matrix A in **Example 1**, with $\delta = 0.01, X_0 = \{0.6I, 0.7I, 0.8I, 0.9I, I\}$, and $q = 3$. Numerical results are recorded in Table 2.

Example 3

We consider real symmetric matrix $A = 0.5(B^T + B)$, where $B = 0.01 * H/N$; $N = \text{sum}(H(1,:)); H = \text{magic}(n)$ and $n = \{10,20,30,40,50,60,70,80,90,100,150\}$. with $X_0 = I$, $q = 3$ Then, a summary of results is given in Table 3.

Table 2: Results summary for **Example 2**

X_0	Iterations allowed	Iteration(s) executed	Error
$0.6I$	100	Over 100	$4.95\times\ 10^{+01}$
$0.6I$	100	Over 100	$3.96\times\ 10^{+01}$
$0.7I$	100	Over 100	$2.97\times\ 10^{+01}$
$0.8I$	100	Over 100	$1.98\ \times\ 10^{+01}$
0.9I	100	Over 100	$9.90\ \times\ 10^{00}$
I	100	1	$3.02\ \times\ 10^{-09}$

Table 3: Results summary for **Example 3**

Matrix size (n)	Iterations allowed	Iteration(s) executed	Error
10	100	1	$5.56\ \times\ 10^{-09}$
20	100	1	$4.97\ \times\ 10^{-09}$
30	100	1	$5.07\ \times\ 10^{-09}$
40	100	1	$4.79\ \times\ 10^{-09}$
50	100	1	$4.89\ \times\ 10^{-09}$
60	100	1	$4.72\ \times\ 10^{-09}$
70	100	1	$4.80\ \times\ 10^{-09}$
80	100	1	$4.69\ \times\ 10^{-09}$
90	100	1	$4.75\ \times\ 10^{-09}$
100	100	1	$4.67\ \times\ 10^{-09}$
150	100	1	$4.67\ \times\ 10^{-09}$

Remark 5

In Table 1, the error decreases as the spectral radius drops from 0.046 to 0.0004. This implies that the convergence of the developed algorithm is highly dependent on the spectral radius of the matrix considered. When the spectral radius reaches 0.0004, the algorithm starts to yield better results within the iterations permitted.

In Table 2, the error decreases as initial solution is decreased. The convergence improves significantly as the initial solution gets closer to I. In fact, the solution is very close to I. As the initial solution is quite away from I, Algorithm 2 tends to diverge. Algorithm 2 being Quasi-Newton, it depicts properties of pure Newton's method.

In Table 3, Algorithm 2 seems to converge to the solution by only a single iteration for all matrices considered. This is because we have considered a relatively good initial solution and a matrix with a relatively smaller spectral radius.

Conclusion

This work introduced inversion free method for obtaining symmetric solution of Equation (1). Basic conditions for the convergence of Algorithm 2 have been presented. Numerical results show that Algorithm 2 performs well when the coefficient matrix A has a relatively smaller spectral radius and for initial guess closer to identity matrix I.

Acknowledgements

The author thanks the reviewers for their constructive comments which led to the improvement of this paper.

Declaration: The author has no conflict of interest.

References

Chacha CS and Kim HM 2019a Elementwise minimal nonnegative solutions for a class of nonlinear matrix equations. *East Asian J. Appl. Math.* 9(4): 665-682.

Chacha CS and Kim HM 2019b An efficient iterative algorithm for finding a nontrivial symmetric solution of the Yang–Baxter-like matrix equation *J. Nonlinear Sci. Appl.* 12: 21-29.

Chacha CS and Naqvi SMRS 2018 Condition numbers of the nonlinear matrix equation $X^p - A^*e^XA = I$. *J. Funct. Spaces* 2018: 1-8.

Dehghan M and Hajarian M 2011 The (R, S)-Symmetric and (R, S)-Skew Symmetric Solutions of the pair of matrix equations $A_1XB_1= C_1$ and $A_2XB_2= C_2$. *Bull. Iranian Math. Soc.* 37(3): 269-279.

Erfanifar R, Sayevand K and Esmaeili H 2020 A novel iterative method for the solution of a nonlinear matrix equation. *Appl. Numer. Math.* 153: 503-518.

Gao D 2016 On Hermitian positive definite solution of the nonlinear matrix equation $X - A^*e^XA = I$. *J. Appl. Math. Comput.* 50: 109-116.

Huang N and Ma CF 2015 Two structure-preserving-doubling like algorithms for obtaining the positive definite solution to a class of nonlinear matrix equation. *J. Comput. Math. Appl.* 69: 494-502.

Huan HY 2011 *Finding special solvents to some nonlinear matrix equations*. PhD thesis, Pusan National University.

Peng ZH, Hu XY and Zhang L 2006 An iterative method for symmetric solutions and the optimal approximation solution of the system of matrix equations $A_1X_1B_1 = C_1$, $A_2 X_2 B_2 = C_2$, *Appl. Math. Comput.* 183: 1127-1137.

Sheng X and Chen G 2010 An iterative method for the symmetric and skew symmetric solutions of a linear matrix equation $AXB + CYD = E$. *J. Comput. Appl. Math.* 223: 3030-3040.

Zhang GF, Xie WW and Zhao JY 2011 Positive definite solution of the nonlinear matrix positive definite solutions of the nonlinear matrix equation$X + A^*X^qA = Q$ $(q > 0)$. *Appl. Math. Comput*. 217(22): 9182-9188.