Scholars Journal of Physics, Mathematics and Statistics OPEN ACCESS
Abbreviated Key Title: Sch J Phys Math Stat
ISSN 2393-8056 (Print) | ISSN 2393-8064 (Online)
Journal homepage: https://saspublishers.com

A New Type of Shifted Method for the Stein-Like Matrix Equation X+M (X) = C

Shuai Liu[1*]

[1]College of Applied Mathematics, Chengdu University of Information Technology, Chengdu 610025, Sichuan, P. R. China

DOI: 10.36347/sjpms.2023.v10i09.001 | **Received:** 24.09.2023 | **Accepted:** 30.10.2023 | **Published:** 08.11.2023

***Corresponding author:** Shuai Liu
College of Applied Mathematics, Chengdu University of Information Technology, Chengdu 610025, Sichuan, P. R. China

Abstract **Review Article**

In this paper, we presented the shifted global full orthogonalization method (SGl-FOM(m)) and shifted global generalized minimum residual method (SGl-GMRES(m)) to solve the Stein-like matrix equation $X+\mathrm{M}(X)=C$ with $\mathrm{M}(X)=AXB$ $\mathrm{M}(X)=AX^{\cdot}B$, $\mathrm{M}(X)=A\bar{X}B$ or $\mathrm{M}(X)=AX^{H}B$, respectively, where X is an unknown matrix to be solved. By utilizing the collinearity of the residuals of the seed system $\mathrm{M}(X)=C$ and the shifted system $X+\mathrm{M}(X)=C$, the solution of the Stein-like matrix equation can be derived. Some results are also given. Numerical experiments show its efficiency.

Keywords: FOM method; GMRES method; Stein-like matrix equation; collinearity.

1. INTRODUCTION

In this paper, we consider the following Stein-like matrix equation:

$$X+\mathrm{M}(X)=C \quad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

With $\mathrm{M}(X)=AXB$, $\mathrm{M}(X)=AX^{\cdot}B$, $\mathrm{M}(X)=A\bar{X}B$ or $\mathrm{M}(X)=AX^{H}B$ and A, B, C being known matrices, If $\mathrm{M}(X)=AXB$, Eq. (1) becomes the standard Stein matrix equation which plays an important role in the fields of discrete-time system [1], statistics [2, 3], probability [4] and spectral analysis [5]. For further details on this topic, we refer to see [6] and the references therein. Under the condition that $\lambda_i(A)\lambda_j(B)\neq -1$ for all i, j, the solution X exists and is unique for any C (e.g., [7]). Necessary and sufficient conditions for the existence and uniqueness of the solution X were obtained in [8]. We also refer the reader to [17] for a solution expressed in terms of the companion form of the given matrices, and to [18] for related computational considerations. Recently, the solvability conditions of T-Stein equation, given by $X=AX^{\cdot}B+C$, have been analyzed in [11-13]. In [14, 15], some explicit solutions of the matrix equation $X-A\bar{X}B=C$ have been established. More generally, Zhou *et al*., [16] discussed the solvability, existence of unique solution, closed-form solution and numerical solution of matrix equation $X=Af(X)B+C$ with $f(X)=X^{\cdot}$, $f(X)=\bar{X}$ or $f(X)=X^{H}$.

Recently, many iterative methods for solving Eq. (1) have received great attention. Some Arnoldi-type matrix Krylov subspace methods were presented to solve the Stein matrix equation $X=AXB+C$ in [17-23]. By Smith-type iterative methods, Zhou *et al*., [24, 25] gained the numerical solution under some restrictions on coefficient matrices. While in [26, 27], Wu *et al*., obtained the iterative solution of Eq. (1) by extending the CGNE method [28] and the gradient-based iterative (GI) method [29]. The CGNE method denotes the conjugate gradient (CG) method applied to the normal equation of the linear system $Ax=b$. The basic idea of the Gl method is to regard the unknown matrix as the parameter of a system

to be identified. However, the extended CGNE method and the extended GI method have shown slow convergence rate in numerical experiments, especially when dealing with the complex matrix equations.

Li and Huang [30] make full use of the shifted structure of $X + AXB = C$ to solve the matrix equations, inspired by this, we consider using a seed system to solve the shifted system. In [31], Frommer and Glassner solved the shifted linear systems by forcing the residual of the shifted system to be collinear to the residual of the seed system. To my knowledge, no one has yet solve the Stein-like matrix equations by forcing the residual of the shifted system to be collinear to the residual of the seed system. In the present work, we are interested in using shifted global full orthogonalization method (SGl-FOM(m)) and shifted global generalized minimum residual method (SGl-GMRES(m)) to solve Eq. (1). To derive the new method, we define the seed system as follows:

$$\mathrm{M}(X) = C \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2)$$

With $\mathrm{M}(X) = AXB, \mathrm{M}(X) = AX^{\bullet}B, \mathrm{M}(X) = A\bar{X}B$ or $\mathrm{M}(X) = AX^{H}B$. Then, the system $X + \mathrm{M}(X) = C$ may be called the shifted system generated by the seed system (2).

We give the following notations to be used later. The notations tr(A) and $A^{\intercal}$ denote the trace and transpose of a matrix A, respectively. For two matrices A, B $\in R^{n\times n}$, A$\otimes$B is their Kronecker product and $\langle A,B\rangle_F = tr(A^{\intercal}B)$ is the inner product which generates the Frobenius norm as $\|A\|_F = \sqrt{\langle A, A\rangle_F} = \sqrt{tr(A^{\bullet}A)}$. The matrices A, B are called orthogonal if $\langle A, A\rangle_F = 0$.

The rest of this paper is organized as follows. In Section 2, we review the generalized global arnoldi process. In Section 3, the shifted Gl-FOM(m) method and shifted Gl-GMRES(m) method are given. Some numerical examples are given to illustrate the efficiency of the proposed method in Section 4. Finally, the paper is ended with a brief conclusion in Section 5.

2. Generalized global Arnoldi process

In this Section, we review the generalized global arnoldi process in [30].

Definition 2.1 ([32]) *Let $A = [A_1, A_2, ..., A_p]$ and $B = [B_1, B_2, ..., B_l]$ be matrices of dimension $n \times ps$ and $n \times ls$, respectively, where $A_i, B_j \in R^{n\times s}$ $(i = 1, ..., p; j = 1, ..., l)$. Then the $p \times l$ matrix $A^{\intercal} \diamond_F B$ is defined by*

$$A^{\bullet} \diamond_F B = \begin{pmatrix} \langle A_1, B_1\rangle_F & \langle A_1, B_2\rangle_F & \cdots & \langle A_1, B_l\rangle_F \\ \langle A_2, B_1\rangle_F & \langle A_2, B_2\rangle_F & \cdots & \langle A_2, B_l\rangle_F \\ \vdots & \vdots & \vdots & \vdots \\ \langle A_p, B_1\rangle_F & \langle A_p, B_2\rangle_F & \cdots & \langle A_p, B_l\rangle_F \end{pmatrix}.$$

For a given matrix V $\in R^{n\times p}$, we define the matrix Krylov subspace as follows:

$$\mathrm{K}_m(\mathrm{M}, V) = span\{V, \mathrm{M}(V), \ldots, \mathrm{M}^{m-1}(V)\} \quad \ldots\ldots\ldots\ldots\ldots \quad (3)$$

Here $\mathrm{M}^{i}(V)$ is defined recursively as $\mathrm{M}(\mathrm{M}^{i-1}(V))$ and $\mathrm{M}^{0}(V) = V$. We describe the generalized global Arnoldi process global Arnoldi process as follows.

Algorithm 2.1 Generalized global Arnoldi process
1. Set $\beta = \|V\|F$, $V_1 = V/\beta$.
2. For j = 1, 2, ..., m, do
3. $W = \mathrm{M}(V_j)$;
4. For i = 1, 2, ..., j, do
5. $h_{ij} = \langle W, V_i\rangle F$;
6. $W = W - h_{ij} V_j$;
7. Enddo;
8. $h_{j+1,j} = \|W\|F$, If $h_{j+1,j} = 0$, Stop; 9. $Vj+1 = W/hj+1,j$; 10. Enddo.

Let $V_m = [V_1, V_2, ..., V_m]$ and $\overline{H}_m$ be the (m + 1) × m upper Hessenberg matrix whose nonzero entries h_{ij} are computed by Algorithm 2.1. Suppose that H_m is the m×m matrix obtained from $\overline{H}_m$ by deleting its last row. It is not difficult to verify that the global Arnoldi process produces an F-orthonormal basis $V_1, V_2, ..., V_m$ for the matrix Krylov subspace $K_m(M,V)$, i.e., $V_m^{\bullet} \diamond_F V_m = I_m$.

Theorem 2.1 ([30]) *Let* V_m *be defined as above. Then we have*

$$\left\| V_m\left(\alpha \otimes I_p\right)\right\|_F = \|\alpha\|_2 \quad \text{..... (4)}$$

Where $\alpha = (\alpha_1, \alpha_2, ..., \alpha_m)^{\bullet}$ *is a real vector of* R^m.

Theorem 2.2 ([30]) *Let* V_m, H_m *and* $\overline{H}_m$, *be defined as above. Then the following relations hold*

$$\left[\mathrm{M}\left(V_1\right), \mathrm{M}\left(V_2\right), \ldots \mathrm{M}\left(V_m\right)\right] = V_{m+1}\left(\hat{H}_m \otimes I_p\right) \text{..... (5)}$$

$$\left[\mathrm{M}\left(V_1\right), \mathrm{M}\left(V_2\right), \ldots \mathrm{M}\left(V_m\right)\right] = V_m\left(H_m \otimes I_p\right) + h_{m+1,m} V_{m+1}\left(e_m^{\bullet} \otimes I_p\right) \text{..... (6)}$$

Where $e_m^{\bullet} = [0, \ldots, 0, 1]_{1\times m}$.

Then let $\overline{H}_m = H_m + \begin{bmatrix} I_m \\ 0 \end{bmatrix}$, the following relation holds,

$$(\mathrm{M} + I)V_m = V_{m+1}\left(\bar{H}_m \otimes I_p\right) \text{..... (7)}$$

3. Shifted Gl-FOM(m) and Shifted Gl-GMRES(m) algorithms

In this section, we propose the Shifted Gl-FOM(m) and Shifted Gl-GMRES(m) algorithms for solving Eq. (1).

3.1 Shifted Gl-FOM(m) algorithm

Let $X_0 \in R^{n\times p}$ be an initial guess and $R_0 = C - \mathrm{M}(X_0)$ be the corresponding residual. At the *m*th iterate, the Gl-FOM algorithm computes the new approximate solution X_m such that

$$X_m \in X_0 + \mathrm{K}(\mathrm{M}, R_0), \text{..... (8)}$$

and

$$R_m = C - \mathrm{M}(X_m) \perp_F \mathrm{K}_m(\mathrm{M}, R_0). \text{..... (9)}$$

The symbol $\perp_F$ denotes the orthogonality with respect to the inner product $\langle\cdot,\cdot\rangle_F$. Considering the *F*-orthonormal basis $V_m = [V_1, V_2, \ldots, V_m]$ of $\mathrm{K}_m(\mathrm{M}, R_0)$ constructed by the generalized global Arnoldi process, we have:

$$X_m = X_0 + \sum_{i=1}^{m} y_m^{(i)} V_i = X_0 + V_m(y_m \otimes I_p) \text{..... (10)}$$

where the vector $y_m = (y_m^{(1)}, y_m^{(2)}, \ldots, y_m^{(m)})^{\bullet} \in R^m$ is obtained by the orthogonality condition (9). Substituting (10) to (9), by (6), we get

$$\begin{aligned} R_m &= C - \mathrm{M}\,(X_0 + \mathrm{V}_m(y_m \otimes I_p)) \\ &= R_0 - \mathrm{M}\,(\mathrm{V}_m(y_m \otimes I_p)) \\ &= \beta V_1 - \left[\mathrm{M}\,(V_1), \mathrm{M}\,(V_2), \ldots \mathrm{M}\,(V_m)\right](y_m \otimes I_p) \\ &= \mathrm{V}_m(\beta e_1 \otimes I_p) - \mathrm{V}_m(H_m \otimes I_p)(y_m \otimes I_p) - h_{m+1,m} V_{m+1}(e_m^{\cdot} \otimes I_p)(y_m \otimes I_p) \\ &= \mathrm{V}_m((\beta e_1 - H_m y_m) \otimes I_p) - h_{m+1,m} V_{m+1}(e_m^{\cdot} y_m \otimes I_p). \end{aligned} \quad \ldots\ldots\ldots\ldots (11)$$

According to the orthogonality condition (9), we have

$$\begin{aligned} 0 &= \mathrm{V}_m^{\cdot} \Diamond_F R_m \\ &= \mathrm{V}_m^{\cdot} \Diamond_F \left(\mathrm{V}_m((\beta e_1 - H_m y_m) \otimes I_p) - h_{m+1,m} V_{m+1}(e_m^{\cdot} y_m \otimes I_p)\right) \\ &= \mathrm{V}_m^{\cdot} \Diamond_F \mathrm{V}_m(\beta e_1 - H_m y_m) - h_{m+1,m}(\mathrm{V}_m^{\cdot} \Diamond_F V_{m+1})(e_m^{\cdot} y_m). \end{aligned}$$

Owing to $\mathrm{V}_m^{\cdot} \Diamond_F \mathrm{V}_m = I_m$ and $\mathrm{V}_m^{\cdot} \Diamond_F V_{m+1} = 0$ we conclude that the vector y_m is the solution of the following linear system:

$$H_m y_m = \beta e_1. \quad \ldots\ldots\ldots\ldots\ldots\ldots (12)$$

Meanwhile, by (5), R_m can also be represented as:

$$\begin{aligned} R_m &= C - \mathrm{M}\,(X_0 + \mathrm{V}_m(y_m \otimes I_p)) \\ &= R_0 - \mathrm{M}\,(\mathrm{V}_m(y_m \otimes I_p)) \\ &= \beta V_1 - \left[\mathrm{M}\,(V_1), \mathrm{M}\,(V_2), \ldots \mathrm{M}\,(V_m)\right](y_m \otimes I_p) \\ &= \mathrm{V}_{m+1}(\beta \hat{e}_1 \otimes I_p) - \mathrm{V}_{m+1}((\hat{H}_m y_m) \otimes I_p) \\ &= \mathrm{V}_{m+1}((\beta \hat{e}_1 - \hat{H}_m y_m) \otimes I_p) \\ &= \mathrm{V}_{m+1}(\omega_{m+1} \otimes I_p) \end{aligned} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (13)$$

Where $\hat{e}_1 = [0, \ldots, 0, 1]_{1\times(m+1)}$, and $\omega_{m+1} = \beta \hat{e}_1 - \hat{H}_m y_m$.

For the shifted system (1), we have

$$\bar{X}_m = \bar{X}_0 + \sum_{i=1}^{m} \bar{y}_m^{(i)} V_i = \bar{X}_0 + \mathrm{V}_m(\bar{y}_m \otimes I_p) \quad \ldots\ldots\ldots\ldots\ldots\ldots (14)$$

$$\bar{R}_m = C - \bar{X}_m - \mathrm{M}\,(\bar{X}_0 + \mathrm{V}_m(\bar{y}_m \otimes I_p)) \quad \ldots\ldots\ldots\ldots\ldots\ldots (15)$$

Now assume the initial residuals are colinear,

$$\bar{R}_0 = \beta_0 R_0, \beta_0 \in R \quad \ldots\ldots\ldots\ldots\ldots\ldots (16)$$

Then we require that

$$\bar{R}_m = \beta_m R_m, \beta_m \in R \quad \ldots\ldots\ldots\ldots\ldots\ldots (17)$$

According to equations (7), (13), (15) and (16), the colinearity condition (17) can be written as:

$$\begin{aligned} & C - \bar{X}_m - \mathrm{M}\,(\bar{X}_0 + \mathrm{V}_m(\bar{y}_m \otimes I_p)) = \beta_m R_m \\ & \bar{R}_0 - (\mathrm{M} + I)(\mathrm{V}_m(\bar{y}_m \otimes I_p)) = \beta_m R_m \\ & \beta_0 R_0 - \mathrm{V}_{m+1}\left(\bar{H}_m \otimes I_p\right)(\bar{y}_m \otimes I_p) = \beta_m \mathrm{V}_{m+1}(\omega_{m+1} \otimes I_p) \\ & \beta_0 \mathrm{V}_{m+1}(\beta \hat{e}_1 \otimes I_p) - \mathrm{V}_{m+1}\left(\bar{H}_m \otimes I_p\right)(\bar{y}_m \otimes I_p) = \beta_m \mathrm{V}_{m+1}(\omega_{m+1} \otimes I_p). \end{aligned} \quad \ldots\ldots\ldots\ldots\ldots\ldots (18)$$

Multiply equation (18) from left by $\mathrm{V}_{m+1}^{\bullet}$, we have

$$\beta_0\beta\hat{e}_1 - \bar{H}_m\bar{y}_m = \beta_m\omega_{m+1}$$

Which leads to a linear system for the unknown $\bar{y}_m$ and β_m,

$$\begin{pmatrix}\bar{H}_m & \omega_{m+1}\end{pmatrix}\begin{pmatrix}\bar{y}_m \\ \beta_m\end{pmatrix} = \beta_0\beta\hat{e}_1. \quad \text{..... (19)}$$

The shifted Gl-FOM algorithm requires the storage of V_m. As m increases it becomes expensive. To overcome the storage problem of the shifted Gl-FOM algorithm, one can restart the algorithm every *m* iterations. Then the restarted and shifted Gl-FOM algorithm denoted by SGl-FOM(m) is given as follows.

Algorithm 3.1 SGl-FOM(m)
1. Choosing X_0, and a tolerance ε, set $\bar{X}_0 = X_0$, set $\beta_0 = 1$;
2. Compute $R_0 = C - \mathrm{M}(X_0)$ and $\bar{R}_0 = C - \bar{X}_0 - \mathrm{M}(\bar{X}_0)$, set $\mathrm{V}=\mathrm{R}_0$;
3. Construct the *F*-orthonormal basis $V_1, V_2, ..., V_m$ by Algorithm 2.1.
4. Determine y_m as the solution of $H_m y_m = \beta e_1$;
5. Determine $\bar{y}_m$ and β_m as the solution of $\begin{pmatrix}\bar{H}_m & \omega_{m+1}\end{pmatrix}\begin{pmatrix}\bar{y}_m \\ \beta_m\end{pmatrix} = \beta_0\beta\hat{e}_1$;
6. Compute $X_m = X_0 + \mathrm{V}_m(y_m \otimes I_p)$ and $R_m = C - \mathrm{M}(X_m)$;
7. Compute $\bar{X}_m = \bar{X}_0 + \mathrm{V}_m(\bar{y}_m \otimes I_p)$ and $\bar{R}_m = C - \bar{X}_m - \mathrm{M}(\bar{X}_m)$;
8. If $\|\bar{R}_m\|_F < \varepsilon$, then stop; else, set $X_0 = X_m, R_0 = R_m, V = R_0, \bar{X}_0 = \bar{X}_m$ $\bar{R}_0 = \bar{R}_m, \beta_0 = \beta_m$, go to 3.

3.2 Shifted Gl-GMRES(m) algorithm

Like the SGl-FOM(m) algorithm, the *m*th iterate $\mathrm{X_m}$ of the Gl-GMRES algo-rithm belongs to the affine matrix Krylov subspace $X_0 + \mathrm{K}(\mathrm{M}, R_0)$. On the other hand, in the GL-GMRES algorithm, the vec $\mathrm{y_m}$ in (10) is obtained by imposing the following orthogonality condition:

$$R_m = C - \mathrm{M}(X_m) \perp_F \mathrm{K}_m(\mathrm{M}, \mathrm{M}(R_0)). \quad \text{..... (20)}$$

From (13), we have

$$R_m = \mathrm{V}_{m+1}((\beta\hat{e}_1 - \hat{H}_m y_m) \otimes I_p) \quad \text{..... (21)}$$

As $\mathrm{V}_{m+1}^H \Diamond_F \mathrm{V}_{m+1} = I_{m+1}$, we have

$$\|R_m\|_F = \left\|\beta\hat{e}_1 - \hat{H}_m y_m\right\|_2.$$

The orthogonality condition (20) shows that $\mathrm{y_m}$ is the solution of the following minimization problem

$$\min_{y\in R^m}\left\|\beta\hat{e}_1 - \hat{H}_m y_m\right\|_2.$$

Consider the QR factorization of the (m+1) × m matrix $\hat{H}_m$, i.e., $\hat{Q}_m\hat{H}_m = \hat{R}_m$, where $\hat{Q}_m$ is a product of Givens rotations and $\hat{R}_m$ is a upper triangular. Assume that

$$\hat{g}_m = \beta\hat{Q}_m\hat{e}_1,$$

And R_m denotes the m × m matrix obtained from $\hat{R}_m$ by deleting its last row and g_m is the m-dimensional vector obtained from $\hat{g}_m$ by deleting its last component. Straightforward computations show that $y_m = R_m^{-1} g_m$.

For the shifted system (1), equations (15) to (18) hold for Gl-GMRES algorithm, so we can obtain $\bar{y}_m$ and β_m by solving the linear system:

$$\left(\bar{H}_m \quad \omega_{m+1}\right)\begin{pmatrix}\bar{y}_m \\ \beta_m\end{pmatrix} = \beta_0 \beta \hat{e}_1.$$

Like the SGl-FOM algorithm, to remedy the storage problem, the restarted and shifted Gl-GMRES algorithm denoted by SGl-GMRES(m) is given as follows.

Algorithm 3.2 SGl-GMRES(m)
1. Choosing X_0, and a tolerance ε, set $\bar{X}_0 = X_0$, set $\beta_0 = 1$;
2. Compute $R_0 = C - \mathrm{M}(X_0)$ and $\bar{R}_0 = C - \bar{X}_0 - \mathrm{M}(\bar{X}_0)$, set $\mathrm{V}=\mathrm{R}_0$;
3. Construct the F-orthonormal basis $V_1, V_2, ..., V_m$ by Algorithm 2.1.
4. Determine y_m as the solution of following least square problem $\min\limits_{y \in R^m} \left\| \beta \hat{e}_1 - \hat{H}_m y_m \right\|_2$;
5. Determine $\bar{y}_m$ and β_m as the solution of $\left(\bar{H}_m \quad \omega_{m+1}\right)\begin{pmatrix}\bar{y}_m \\ \beta_m\end{pmatrix} = \beta_0 \beta \hat{e}_1$;
6. Compute $X_m = X_0 + \mathrm{V}_m(y_m \otimes I_p)$ and $R_m = C - \mathrm{M}(X_m)$;
7. Compute $\bar{X}_m = \bar{X}_0 + \mathrm{V}_m(\bar{y}_m \otimes I_p)$ and $\bar{R}_m = C - \bar{X}_m - \mathrm{M}(\bar{X}_m)$;
8. If $\left\|\bar{R}_m\right\|_F < \varepsilon$, then stop; else, set $X_0 = X_m, R_0 = R_m, V = R_0, \bar{X}_0 = \bar{X}_m$ $\bar{R}_0 = \bar{R}_m, \beta_0 = \beta_m$, go to 3.

4. Numerical examples

In this section, we give some numerical examples to illustrate the effectiveness of the SGl-FOM(m) algorithm and SGl-GMRES(m) algorithm for solving Eq. (1). Also, we compare our new methods with the extended CGNE and Gl methods from [25, 26]. The experiments are performed in MATLAB. All tests start from the zero matrix with suitable size and stop as soon as $\| R_m \|_F \leq 10^{-9}$.

Example 4.1 We consider the Stein matrix equation

$$X + AXB = C$$

Where $A = [diag([1:100]) + diag(1 * ones(299,1),1)]$ and

$B = [diag([1:100]) + diag(-1 * ones(299,1),-1)]$.

Here C is chosen such that $X = tridiag(-1,0,1)$ is the exact solution of the system.

The numerical comparison results are given in Fig. 1. As it is seen, the SGl-FOM (10) and SGl-GMRES (10) algorithms have almost the same convergence behavior and require fewer iteration steps than the the extended CGNE and Gl methods for achieving the prescribed stopping criterion.

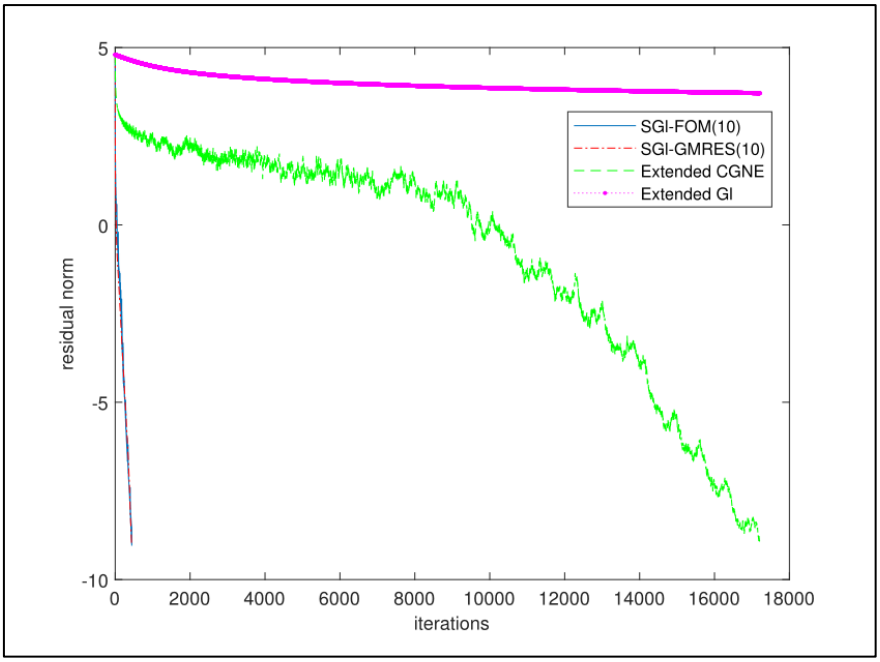

Figure 1: Comparison results of Example 4.1

Example 4.2 We consider the $\bullet$ -Stein matrix equation

$$X + AX^{\bullet} B = C$$

where $A = B = tridiag(-1, 4, -1) \in R^{n\times n}$. Here C is chosen such that the exact solution of the system is $X = tridiag(-1, 0, 1)$.

When n = 200, the numerical comparison results are depicted in Fig. 2. We see from Table 2 that the SGl-FOM(m) and SGl-GMRES(10) algorithms have significant advantages in terms of convergence speed compared to extended CGNE and Gl methods.

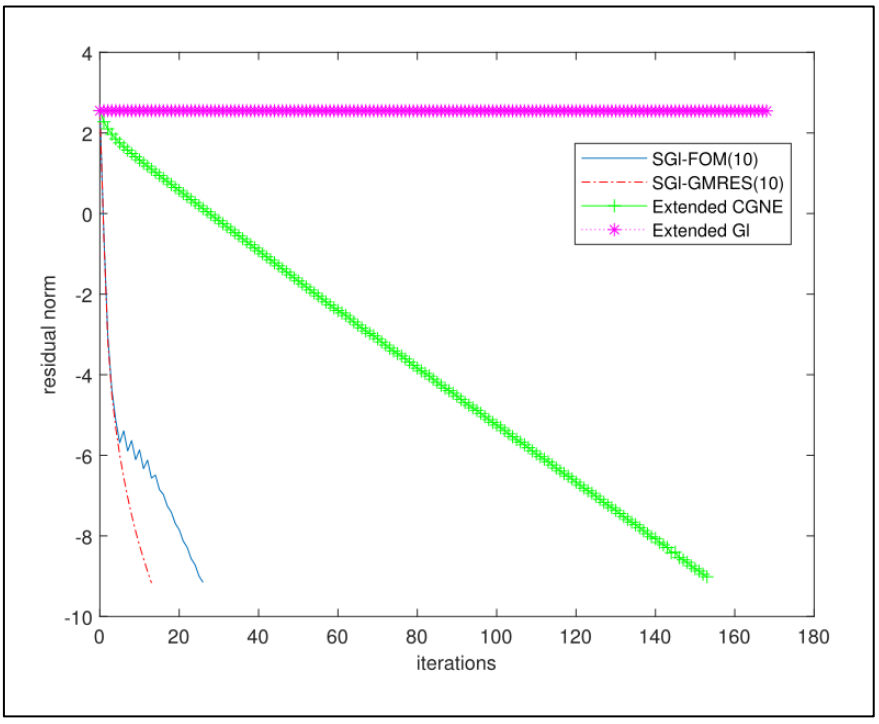

Figure 2: Comparison results of Example 4.2

Example 4.3 We consider the Stein-like matrix equation

$$X + A\bar{X}B = C$$

Where $A = tridiag(-i, 3, i) \in C^{n\times n}$ and $B = tridiag(-i, 0, i) \in C^{s\times s}$. Here C is chosen such that the exact solution of the system is $X = (x_{ij})_{n\times s}$ with $x_{ij} = 1 + i$.

When n = 200 and s = 100, the numerical comparison results are given in Fig.3. As it is seen, the SGl-FOM(10) and SGl-GMRES(10) algorithms require fewer iteration steps than the the extended CGNE and Gl methods for achieving the prescribed stopping criterion.

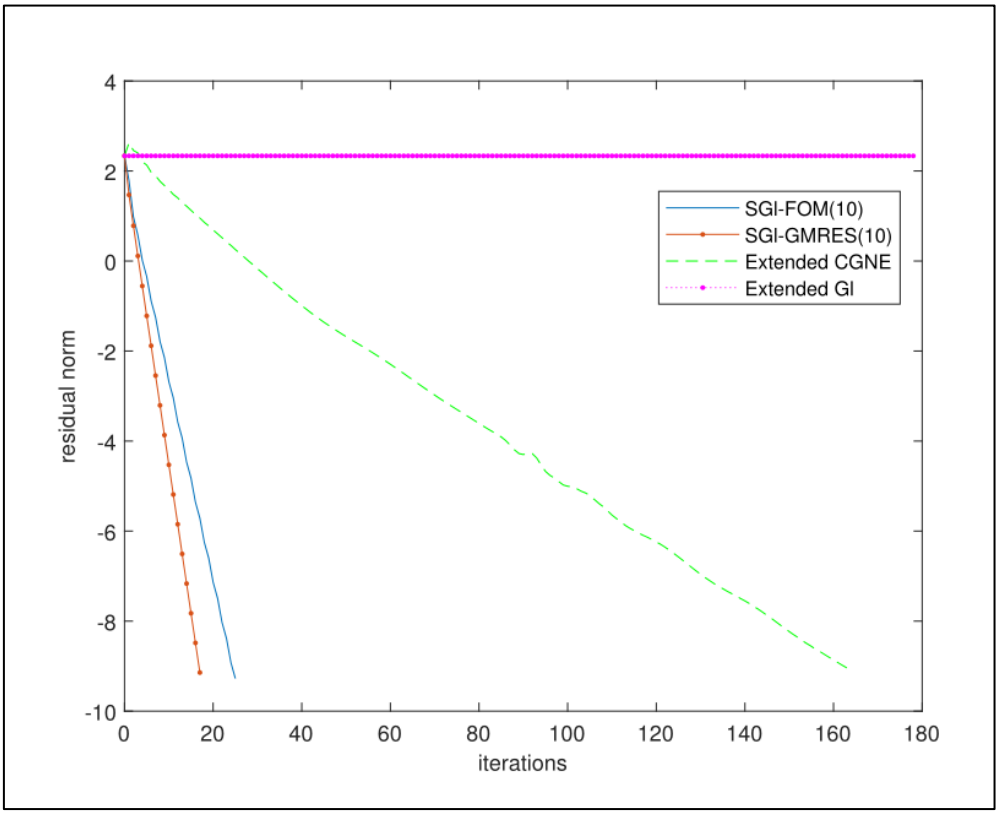

Figure 3: Comparison results of Example 4.3

Example 4.4 We consider the Stein-like matrix equation

$$X + AX^{H}B = C$$

where the matrices A, B, and C are generated by the following MATLAB codes.

```
rand('state',0)

A=diag(10+diag(rand(n)))+triu(rand(n,n),1)*i
B=diag(10+diag(rand(n)))+tril(rand(n,n),1)*i
C=rand(n,n)+rand(n,n)*i
```

When n = 100, the numerical comparison results are given in Fig. 4. As it is seen, the SGl-FOM(10) and SGl-GMRES(10) algorithms require fewer iteration steps than the the extended CGNE and Gl methods for achieving the prescribed stopping criterion.

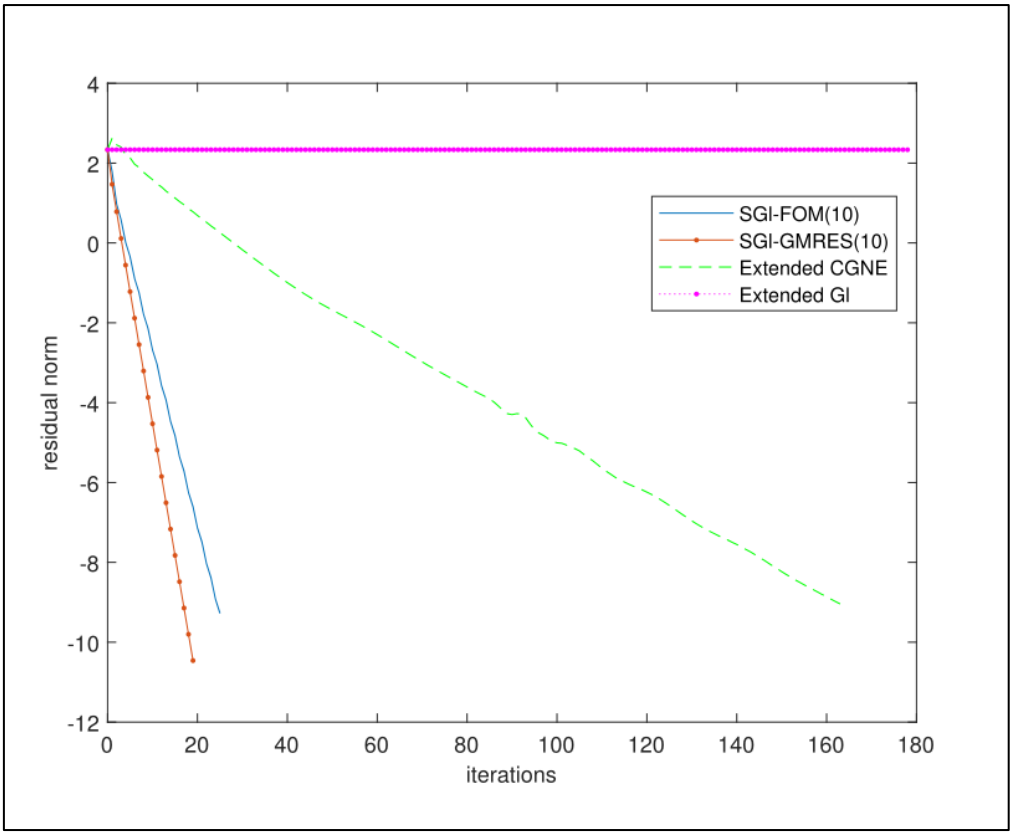

Figure 4: comparison results of Example 4.4

5. CONCLUSION

In this paper, we propose the SGl-FOM(m) method and SGl-GMRES(m) method to solve the Stein-like matrix equation. From numerical examples we conclude that the SGl-FOM(m) method and SGl-GMRES(m) method may be a useful tool for solving the Stein-like matrix equation. In the future, we will study the possibility of combining the augment technology with SGl-FOM(m) method and SGl-GMRES(m) method.

ACKNOWLEDGEMENTS

This research is supported by the Science Research Fund of Chengdu University of Information Technology (KYTD202243).

REFERENCES

1. Antoulas, A. C. (2005). Approximation of large-scale dynamical systems, Advances in Design and Control, SIAM., Philadelphia.
2. Klein, A., & Spreij, P. J. C. (2001). On Steins equation, Vandermonde matrices and Fishers information matrix of time series processes. Part i: the autoregressive moving average process, *Linear Algebra Appl*, 329, 9-47.
3. Klein, A., & Spreij, P. (2005). On the solution of Steins equation and Fishers information matrix of an ARMAX process, *Linear Algebra Appl*, 396, 1-34.
4. Barbour, A. D., & Utev, S. (1998). Solving the stein equation in compound Poisson approximation, *Adv Appl Probab*, 30, 449-475.
5. Hu, G. D., & Zhu, Q. (2013). Bounds of modulus of eigenvalues based on Stein equation, Kybernetika 46 (2010) 655-664. *Remote Sens*, 51, 4045-4058.
6. Simoncini, V. (2016). Computational methods for linear matrix equations, *SIAM Rev*, 58, 377-441.
7. Lancaster, P., & Rodman, L. (1995). Algebraic Riccati equations, Oxford University Press, Oxford, UK.
8. Wimmer, H. K. (1988). The matrix equation X + AXB = C and an analogue of Roths theorem, *Linear Algebra Appl*, 109, 145-147.
9. Lancaster, P., Lerer, L., & Tismenetsky, M. (1984). Factored forms for solutions of AX−XB = C and X−AXB = C in companion matrices, *Linear Algebra Appl*, 62, 19-49.
10. Betser, A., Cohen, N., Zeheb, E. (1995). On solving the Lyapunov and stein equations for a companion matrix, Syst. *Control Lett*, 25, 211-218.
11. Lin, M. M., & Chiang, C. Y. (2012). On the T-Stein Matrix Equation X = AX^TB + C, Tech. Report 2012-10-002,

National Center for Theoretical Sciences, National Tsing Hua University, Taiwan.
12. Chiang, C. Y. (2013). A note on the T-stein matrix equation, *Abstr Appl Anal*. Article ID 824641, 8.
13. Ikramov, K. D., & Vorontsov, Y. O. (2012). The matrix equation X = AX^{T}B +C: conditions for unique solvability and a numerical algorithm for its solution, *Dokl Math*, 85, 265-267.
14. Jiang, T., & Wei, M. (2003). On solutions of the matrix equations X−AXB = C and $X - A\bar{X}B = C$, *Linear Algebra Appl*, 367, 225-233.
15. Wu, A. G., Wang, H. Q., Duan, G. R., & Comput, J. (2009). On matrix equations X−AXF = C and $X - A\bar{X}F = C$, *Appl. Math*, 230, 690-698.
16. Zhou, B., Lam, J., & Duan, G. R. (2011). Toward solution of matrix equationX = Af(X)B +C, *Linear Algebra Appl*, 435, 1370-1398.
17. Bao, L., Lin, Y. Q., & Wei, Y. M. (2007). A new projection method for solving large Sylvester equations, *Appl Numer Math*, 57, 521-532.
18. Bao, L., Lin, Y. Q., & Wei, Y. M. (2006). Krylov subspace methods for the generalized Sylvester equation, *Appl Math Comput*, 175, 557-573.
19. Jbilou, K. (2006). Low rank approximate solutions to large Sylvester matrix equations, *Appl Math Comput*, 177, 365-376.
20. Bouhamidi, A., & Jbilou, K. (2008). A note on the numerical approximate solutions for generalized Sylvester matrix equations with applications, *Appl Math Comput*, 206, 687-694.
21. Beik, F. P. A., & Salkuyeh, D. K. (2011). On the global Krylov subspace methods for solving general coupled matrix equations, *Comput Math Appl*, 62, 4605-4613.
22. Beik, F. P. A. (2014). Theoretical results on the global GMRES method for solving generalized Sylvester matrix equations, *Bull Iran Math Soc*, 40, 1097-1117.
23. Li, S. K., & Huang, T. Z. (2012). LSQR iterative method for generalized coupled Sylvester matrix equations, *Appl Math Model*, 36, 3545-3554.
24. Zhou, B., Lam, J., & Duan, G. R. (2011). Toward solution of matrix equation X = Af(X)B+C, *Linear Algebra Appl*, 435, 1370-1398.
25. Zhou, B., Lam, J., & Duan, G. R. (2009). On Smith-type iterative algorithms for the Stein matrix equation, *Appl Math Lett*, 22, 1038-1044.
26. Wu, A. G., Feng, G., Duan, G. R., & Wu, W. J. (2010). Finite iterative solutions to a class of complex matrix equations with conjugate and transpose of the unknowns, *Math Comput Model*, 52, 1463-1478.
27. Wu, A. G., Lv, L. L., & Duan, G. R. (2011). Iterative algorithms for solving a class of complex conjugate and transpose matrix equations, *Appl Math Comput*, 217, 83438353.
28. Saad, Y. (2003). Iterative Methods for Sparse Linear Systems, (2nd edition), SIAM., Philadelphia.
29. Ding, F., & Chen, T. (2005). Gradient based iterative algorithms for solving a class of matrix equations, IEEE Trans. *Autom Control*, 50, 1216-1221.
30. Li, S. K., & Huang, T. Z. (2019). Restarted global FOM and GMRES algorithms for the Stein-like matrix equation X + M(X) = C, *Appl Math Comput*, 348, 206-214.
31. Frommer, A., & Glassner, U. (1998). Restarted GMRES for shifted linear systems, *SIAM J Sci Comput, 19*(1), 15-26.
32. Bouyouli, R., Jbilou, K., Sadaka, R., & Sadok, H. (2006). Convergence properties of some block Krylov subspace methods for multiple linear systems, *J Comput Appl Math*, 196, 498–511.