Filomat 38:27 (2024), 9399–9434
https://doi.org/10.2298/FIL2427399S

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

On solving coupled Sylvester-conjugate transpose matrix equations over generalized reflexive matrices and anti-reflexive matrices

Akbar Shirilord[a], Mehdi Dehghan[a]

[a]*Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences, Amirkabir University of Technology (Tehran Polytechnic), No. 424, Hafez Ave.,15914, Tehran, Iran*

Abstract. A square matrix P is considered a generalized reflection matrix if being Hermitian and having its square equal to the identity matrix. Given two generalized reflection matrices P and Q, a matrix A is said to be reflexive (anti-reflexive) with respect to pair (P, Q) if $A = PAQ$ $(A = -PAQ)$. This manuscript introduces some iterative algorithms that utilizes the gradient method to solve coupled Sylvester-conjugate transpose matrix equations over generalized reflexive matrices and anti-reflexive matrices. Furthermore, we will conduct an analysis of the convergence properties of these methods. Then, we provide numerical techniques to determine these solutions. To summarize, the numerical examples utilized in this study have effectively demonstrated the efficacy of the iterative methods presented.

1. Introduction

Matrix equations have become a significant area of research in computational mathematics and control. They are used in diverse fields of engineering and mathematics. Control theory heavily relies on understanding the solutions of matrix equations, especially in analyzing the stability of systems. For example, Sylvester matrix equations are crucial in equilibrium realization, optimal control, and robust pole assignment of discrete periodic systems [2, 28, 46].

Lyapunov or Riccati matrix equations are also essential in converting system stability problems into existence problems, as the existence of positive definite solutions to these equations is crucial. Therefore, investigating matrix equations is critical in computational mathematics and control, and its significance cannot be underestimated [6, 16, 17, 22, 23, 30, 31, 43].

The computation of the least squares solution for the Sylvester-type matrix equation $AXB + CX^TD = E$ was carried out using an approach known as the alternating direction method, as described in reference [25].

Zhou et al. proposed an iteration algorithm in [32] for solving matrix equations $(A_1XB_1, A_2XB_2) = (C_1, C_2)$ with an unknown reflexive matrix X relative to generalized reflection matrices. This method guarantees convergence within a finite number of iterations, assuming no round-off errors. Another study in linear matrix equation can be found in [1, 4, 11–14, 21, 24, 27, 33, 34, 37]. In [33], an algorithm was introduced

2020 *Mathematics Subject Classification*. Primary 15A24; Secondary 39B42, 65F10, 65F30.
Keywords. Generalized reflexive matrix, Generalized anti-reflexive matrix, Hierarchical identification principle, Sylvester-conjugate transpose matrix equations, Gradient method, Control theory.
Received: 02 August 2023; Revised: 10 November 2023; Accepted: 16 July 2024
Communicated by Dragan S. Djordjević
Email addresses: akbar.shirilord@aut.ac.ir (Akbar Shirilord), mdehghan@aut.ac.ir, mdehghan.aut@gmail.com (Mehdi Dehghan)

that can determine the solvability of the matrix equation automatically, and it converges to the solution if the system is consistent. In [27], the least squares solution of matrix equation $AXB + CYD = F$ was presented, and the existence and uniqueness of the solution were deeply discussed. To solve generalized matrix equations, the gradient method was extended to the matrix equation

$$(AXB + CYD, EXF + GYH) = (M, N), \tag{1}$$

in [9], and the corresponding generalized bi-symmetric solution was obtained.

By introducing the modular operator, a cyclic gradient based iterative algorithm is provided for solving a class of generalized coupled Sylvester-conjugate matrix equations [36]

$$\sum_{j=1}^{p} \left(A_{ij} X_j B_{ij} + C_{ij} \overline{X_j} D_{ij} \right) = F_i, \quad i = 1, .., N,$$

where $A_{ij}, C_{ij} \in \mathbb{C}^{m_i \times s_j}, B_{ij}, D_{ij} \in \mathbb{C}^{t_j \times n_i}, F_i \in \mathbb{C}^{m_i \times n_i}$ are the known coefficient matrices, and $X_j \in \mathbb{C}^{s_j \times t_j}$ $(j = 1, .., p)$ are the matrices that need to be determined. Author of [19] introduced CGS and Bi-CGSTAB methods for solving the Sylvester-transpose matrix equation

$$\sum_{i=1}^{k} \left(A_i X B_i + C_i X^T D_i \right) = E$$

where $A_i, B_i, C_i, D_i, E \in \mathbb{R}^{m \times m}$ are known matrices for $i = 1, 2, \ldots, k$ and $X \in \mathbb{R}^{m \times m}$ is the matrix to be determined. Also these methods are suggestion for obtaining the solution of periodic Sylvester matrix equation

$$\widehat{A}_j \widehat{X}_j \widehat{B}_j + \widehat{C}_j \widehat{X}_{j+1} \widehat{D}_j = \widehat{E}_j,$$

for $j = 1, 2, \ldots$, where coefficient matrices and solutions are periodic with period λ, i.e., $\widehat{A}_{j+\lambda} = \widehat{A}_j, \widehat{B}_{j+\lambda} = \widehat{B}_j, \widehat{C}_{j+\lambda} = \widehat{C}_j, \widehat{D}_{j+\lambda} = \widehat{D}_j, \widehat{E}_{j+\lambda} = \widehat{E}_j$ and $\widehat{X}_{j+\lambda} = \widehat{X}_j$.

In [3], Bai introduced an iterative algorithm based on the Hermitian and skew-Hermitian splitting (HSS) method to tackle the Sylvester matrix equation $AX+XB = F$, where the involved matrices are non-Hermitian and positive definite or semi-definite. Ding et al. [15] utilized the Jacobi and Gauss-Seidel iterations to extend their iterative solutions beyond the standard $Ax = b$ matrix equation, allowing them to solve more complex matrix equations $AXB = C$ and $AXB + CXD = F$ in their study.

In a separate paper, Xie and Ma proposed a modified conjugate gradient method that is specifically designed to solve either the reflexive or anti-reflexive solutions for a given problem [45]. Their method is particularly applicable to solving the following problem:

$$\begin{cases} AXB + CY^T D = S_1, \\ EX^T F + GYH = S_2, \end{cases} \tag{2}$$

where $A, E \in \mathbb{R}^{p \times n}, C, G \in \mathbb{R}^{p \times m}, B, F \in \mathbb{R}^{n \times q}, D, H \in \mathbb{R}^{m \times q}, S_1, S_2 \in \mathbb{R}^{p \times q}$ are given constant matrices, and $X \in \mathbb{R}^{n \times n}, Y \in \mathbb{R}^{m \times m}$ are unknown matrices to be determined.

The computation of symmetric solutions for the generalized Sylvester matrix equation, represented as $\sum_{i=1}^{t} (A_i X B_i + C_i Y D_i + E_i Z F_i) = G_i$, was achieved through the utilization of a variant of the biconjugate residual algorithm called Lanczos, as documented in [20].

Several iterative algorithms have been developed for solving linear matrix equations, both coupled and uncoupled, utilizing the conjugate gradient (CG) approach. This method has been explored extensively in the literature [10, 38].

In their work, Wu et al. [39–41] tackled the matrix equation $A\bar{X} + BY = XF, X - A\bar{X}F = BY + R$ as well as $AV + BW = EVF$, and presented analytical solutions to these equations.

The solution for a set of linear equations involving matrices of known constants and an unknown matrix X was presented in [29] and [44]. The equations are given by:

$$\sum_{i=1}^{r} A_i X B_i + \sum_{j=1}^{s} C_j X^T D_j = E,$$

where A_i, B_i, C_j, D_j ($i = 1, \ldots, r$, $j = 1, \ldots, s$) and E are matrices of appropriate dimensions.

The research presented in [7] explores the use of a novel approach to matrix splitting and applies it in combination with the hierarchical identification principle to develop iterative techniques for solving linear matrix equations and generalized coupled Sylvester matrix equations.

Our proposed approach for solving the Sylvester-conjugate transpose matrix equations:

$$\begin{cases} A_1 X + X^H B_1 = F_1, \\ A_2 X + X^H B_2 = F_2, \\ \quad \vdots \\ A_r X + X^H B_r = F_r, \end{cases} \tag{3}$$

where A_i, B_i, F_i ($i = 1, \ldots, r$) are matrices of appropriate dimensions, utilizes a novel generalized matrix splitting method and is based on the research by [7]. We employ an effective gradient method for the implementation of our approach. We determine the solution to the Sylvester-conjugate transpose matrix equations on matrices that are both generalized reflexive

$$\mathbb{C}_r^{n\times n}(P, Q) = \{X \in \mathbb{C}^{n\times n} : X = PXQ\},$$

and generalized anti-reflexive:

$$\mathbb{C}_a^{n\times n}(P, Q) = \{X \in \mathbb{C}^{n\times n} : X = -PXQ\}.$$

The paper is structured as follows: Section 2 provides useful definitions and lemmas. Also, in this section, the necessary and sufficient conditions for the solvability of equations (3) are determined with the help of Kronecker product. In addition, a closed form is determined for the solution of these equations. In Section 3, we introduce a novel iterative approach to solve the coupled Sylvester-conjugate transpose matrix equations (3) and provide a convergence analysis. We then extend the iterative method to derive generalized reflexive and generalized anti-reflexive solutions for equations (3) in Section 4 and conduct a convergence analysis of these methods. The numerical results are presented in Section 5, then an application for coupled Sylvester-conjugate transpose matrix equations to the palindromic eigenvalue problem is studied. Finally, in Section 6, we conclude the manuscript with some remarks.

2. Preliminaries

In this paper, the notations A^T, $\overline{A}$, A^H , and $\|.\|$ are utilized to represent the transpose, conjugate, conjugate transpose, and norm of a matrix A, respectively. The spectral norm of A is denoted as $\|.\|_2$. Additionally, the Kronecker product of matrices A and B is represented by $A \otimes B$. To facilitate ease of understanding, we present the following definitions:

Definition 2.1. *[5] A square matrix of size $n \times n$ is considered a generalized reflection matrix if it satisfies two conditions: being Hermitian and having its square equal to the identity matrix I. Given two generalized reflection matrices P and Q, a matrix A of size $n \times n$ is said to be reflexive with respect to pair (P, Q) if $A = PAQ$, and anti-reflexive with respect to pair (P, Q) if $A = -PAQ$. We use the notation $\mathbb{C}_r^{n\times n}(P, Q)$ to denote the set of matrices that are reflexive with respect to pair (P, Q), and $\mathbb{C}_a^{n\times n}(P, Q)$ to denote the set of matrices that are anti-reflexive with respect to pair (P, Q), where P and Q are two generalized reflection matrices.*

Definition 2.2. *[44] Let* $X = [x_1, x_2, \ldots, x_n] \in \mathbb{C}^{n\times n}$ *with* $x_i \in \mathbb{C}^n$ *being the i-th column of X. Then* Col $[X]$ *is an* n^2*-dimensional vector formed by columns of X, i.e.,*

$$\mathrm{Col}\,[X] = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \in \mathbb{C}^{n^2}.$$

Definition 2.3. *[44] Consider a square matrix* $P_n \in \mathbb{R}^{n^2\times n^2}$ *partitioned into* $n \times n$ *submatrices where each submatrix is an elementary matrix of order* $n \times n$ *denoted by* $E_{ij} = e_i e_j^{\mathrm{T}}$, *with* e_i *being a column vector of order* $n \times 1$ *with a unity in the* i^{th} *position and zeros elsewhere. Thus, we can express* P_n *as a sum of such submatrices:*

$$P_n = \sum_{i=1}^{n}\sum_{j=1}^{n} E_{ij} \otimes E_{ij}^{\mathrm{T}}.$$

For example for $n = 2$ we have

$$P_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad X = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{12} \end{bmatrix},$$

and

$$P_2\,\mathrm{Col}\,[X] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_{11} \\ x_{21} \\ x_{12} \\ x_{12} \end{bmatrix} = \begin{bmatrix} x_{11} \\ x_{12} \\ x_{21} \\ x_{12} \end{bmatrix} = \mathrm{Col}\left[X^T\right].$$

In general by using this definition, we can show that P_n satisfies the following properties [44]:

- $\mathrm{Col}\left[X^T\right] = P_n\,\mathrm{Col}\,[X]$,
- $P_n^2 = I_{n^2}$,
- $P_n^T = P_n^{-1} = P_n$.

Lemma 2.4. *[25] If the equation* $AXB = F$ *has a unique solution* X^*, *then the gradient-based iterative (GI) algorithm,*

$$X(k+1) = X(k) + \mu A^H(F - AX(k)B)B^H, \tag{4}$$

where

$$0 < \mu < \frac{2}{\lambda_{\max}(AA^H)\,\lambda_{\max}(B^HB)} \quad or \quad \mu \le \frac{2}{\|A\|^2\|B\|^2}, \tag{5}$$

is such that $X(k) \to X^*$.

3. Main Results

From the property $\mathrm{Col}\left[X^T\right] = P_n\,\mathrm{Col}\,[X]$ it is easy to see that

$$\mathrm{Col}\left[X^H\right] = \overline{P_n\,\mathrm{Col}\,[X]} = \overline{P_n}\;\overline{\mathrm{Col}\,[X]} = P_n\,\mathrm{Col}\left[\overline{X}\right].$$

Thus the solution of Sylvester-conjugate transpose matrix equations (3) can be found by the following lemma.

Lemma 3.1. *The matrix equations* (3) *have a unique solution X if and only if the matrix*

$$\Theta_1 = \begin{bmatrix} I \otimes A_1 & \left(B_1^T \otimes I\right) P_n \\ \vdots & \vdots \\ I \otimes A_r & \left(B_r^T \otimes I\right) P_n \\ (B_1^H \otimes I)P_n & I \otimes \overline{A_1} \\ \vdots & \vdots \\ (B_r^H \otimes I)P_n & I \otimes \overline{A_r} \end{bmatrix},$$

has full column rank and the rank of $[\Theta_1, f_1]$ *is equal to the rank of* Θ_1, *where*

$$f_1 = \begin{bmatrix} \mathrm{Col}[F_1] \\ \vdots \\ \mathrm{Col}[F_r] \\ \mathrm{Col}[\overline{F_1}] \\ \vdots \\ \mathrm{Col}[\overline{F_r}] \end{bmatrix}.$$

In such cases, the solution to (3) *can be obtained by solving the following linear system:*

$$\begin{bmatrix} I \otimes A_1 & \left(B_1^T \otimes I\right) P_n \\ \vdots & \vdots \\ I \otimes A_r & \left(B_r^T \otimes I\right) P_n \\ (B_1^H \otimes I)P_n & I \otimes \overline{A_1} \\ \vdots & \vdots \\ (B_r^H \otimes I)P_n & I \otimes \overline{A_r} \end{bmatrix} \begin{bmatrix} \mathrm{Col}[X] \\ \mathrm{Col}[\overline{X}] \end{bmatrix} = \begin{bmatrix} \mathrm{Col}[F_1] \\ \vdots \\ \mathrm{Col}[F_r] \\ \mathrm{Col}[\overline{F_1}] \\ \vdots \\ \mathrm{Col}[\overline{F_r}] \end{bmatrix}.$$

Furthermore, it is observed that the corresponding homogeneous matrix equations (3) *possesses a unique zeros solution,* $X = \boldsymbol{O}_n$.

Proof. Taking conjugate from (3) yields:

$$\begin{cases} A_1 X + X^H B_1 = F_1, \\ \vdots \\ A_r X + X^H B_r = F_r, \\ \overline{A_1}\,\overline{X} + X^T \overline{B_1} = \overline{F_1}, \\ \vdots \\ \overline{A_r}\,\overline{X} + X^T \overline{B_1} = \overline{F_r}. \end{cases} \tag{6}$$

Now by using Kronecker product and property $\mathrm{Col}\left[X^H\right] = P_n\,\mathrm{Col}\left[\overline{X}\right]$ we obtain:

$$\begin{cases} (I \otimes A_1)\,\mathrm{Col}\,[X] + (B_1^T \otimes I)\,\mathrm{Col}\left[X^H\right] = \mathrm{Col}\,[F_1]\,, \\ \vdots \\ (I \otimes A_r)\,\mathrm{Col}\,[X] + (B_r^T \otimes I)\,\mathrm{Col}\left[X^H\right] = \mathrm{Col}\,[F_r]\,, \\ (I \otimes \overline{A_1})\,\mathrm{Col}\left[\overline{X}\right] + (\overline{B_1}^T \otimes I)\,\mathrm{Col}\left[X^T\right] = \mathrm{Col}\left[\overline{F_1}\right], \\ \vdots \\ (I \otimes \overline{A_r})\,\mathrm{Col}\left[\overline{X}\right] + (\overline{B_r}^T \otimes I)\,\mathrm{Col}\left[X^T\right] = \mathrm{Col}\left[\overline{F_r}\right], \end{cases} \tag{7}$$

or

$$\begin{cases} (I \otimes A_1)\,\mathrm{Col}\,[X] + (B_1^T \otimes I)P_n\,\mathrm{Col}\left[\overline{X}\right] = \mathrm{Col}\,[F_1]\,, \\ \vdots \\ (I \otimes A_r)\,\mathrm{Col}\,[X] + (B_r^T \otimes I)P_n\,\mathrm{Col}\left[\overline{X}\right] = \mathrm{Col}\,[F_r]\,, \\ (I \otimes \overline{A_1})\,\mathrm{Col}\left[\overline{X}\right] + ({B_1}^H \otimes I)P_n\,\mathrm{Col}\,[X] = \mathrm{Col}\left[\overline{F_1}\right], \\ \vdots \\ (I \otimes \overline{A_r})\,\mathrm{Col}\left[\overline{X}\right] + ({B_r}^H \otimes I)P_n\,\mathrm{Col}\,[X] = \mathrm{Col}\left[\overline{F_r}\right], \end{cases} \tag{8}$$

or

$$\begin{cases} (I \otimes A_1)\,\mathrm{Col}\,[X] + (B_1^T \otimes I)P_n\,\mathrm{Col}\left[\overline{X}\right] = \mathrm{Col}\,[F_1]\,, \\ \vdots \\ (I \otimes A_r)\,\mathrm{Col}\,[X] + (B_r^T \otimes I)P_n\,\mathrm{Col}\left[\overline{X}\right] = \mathrm{Col}\,[F_r]\,, \\ ({B_1}^H \otimes I)P_n\,\mathrm{Col}\,[X] + (I \otimes \overline{A_1})\,\mathrm{Col}\left[\overline{X}\right] = \mathrm{Col}\left[\overline{F_1}\right], \\ \vdots \\ ({B_r}^H \otimes I)P_n\,\mathrm{Col}\,[X] + (I \otimes \overline{A_r})\,\mathrm{Col}\left[\overline{X}\right] = \mathrm{Col}\left[\overline{F_r}\right]. \end{cases} \tag{9}$$

Now the above equations can be written as

$$\Theta_1 \begin{bmatrix} \mathrm{Col}(X) \\ \mathrm{Col}(\overline{X}) \end{bmatrix} = f.$$

Hence matrix equations (3) have a unique solution X if and only if the matrix Θ_1 has full column rank, and $\mathrm{Rank}[\Theta_1, f_1] = \mathrm{Rank}[\Theta_1] = 2rn^2$. In this case we have

$$\begin{bmatrix} \mathrm{Col}(X) \\ \mathrm{Col}(\overline{X}) \end{bmatrix} = (\Theta_1^H \Theta_1)^{-1} \Theta_1^H f_1. \tag{10}$$

Consequently, the exact solution of matrix equations (3) can be determined by (10). Moreover for $f = 0$, we have $\begin{bmatrix} \mathrm{Col}(X) \\ \mathrm{Col}(\overline{X}) \end{bmatrix} = 0$ that yields the solution $X = \mathbf{O}_n$. □

Lemma 3.2. *The matrix equations* (3) *have a unique solution X if and only if the matrix*

$$\Theta_2 = \begin{bmatrix} I \otimes A_1 & B_1^T \otimes I \\ \vdots & \vdots \\ I \otimes A_r & B_r^T \otimes I \\ I \otimes B_1^H & \overline{A_1} \otimes I \\ \vdots & \vdots \\ I \otimes B_r^H & \overline{A_r} \otimes I \end{bmatrix},$$

has full column rank and the rank of $[\Theta_2, f_2]$ *is equal to the rank of* Θ_2*, where*

$$f_2 = \begin{bmatrix} \mathrm{Col}[F_1] \\ \vdots \\ \mathrm{Col}[F_r] \\ \mathrm{Col}[F_1^H] \\ \vdots \\ \mathrm{Col}[F_r^H] \end{bmatrix}.$$

In such cases, the solution to (3) *can be obtained by solving the following linear system:*

$$\begin{bmatrix} I\otimes A_1 & B_1^T\otimes I \\ \vdots & \vdots \\ I\otimes A_r & B_r^T\otimes I \\ I\otimes B_1^H & \overline{A_1}\otimes I \\ \vdots & \vdots \\ I\otimes B_r^H & \overline{A_r}\otimes I \end{bmatrix} \begin{bmatrix} \mathrm{Col}[X] \\ \mathrm{Col}[X^H] \end{bmatrix} = \begin{bmatrix} \mathrm{Col}[F_1] \\ \vdots \\ \mathrm{Col}[F_r] \\ \mathrm{Col}[F_1^H] \\ \vdots \\ \mathrm{Col}[F_r^H] \end{bmatrix}. \tag{11}$$

Proof. From (3) we have:

$$\begin{cases} A_1X + X^HB_1 = F_1, \\ \vdots \\ A_rX + X^HB_r = F_r, \\ \left(A_1X + X^HB_1\right)^H = F_1^H, \\ \vdots \\ \left(A_rX + X^HB_r\right)^H = F_r^H, \end{cases} \tag{12}$$

that yields:

$$\begin{cases} A_1X + X^HB_1 = F_1, \\ \vdots \\ A_rX + X^HB_r = F_r, \\ B_1^HX + X^HA_1^H = F_1^H, \\ \vdots \\ B_r^HX + X^HA_r^H = F_r^H. \end{cases} \tag{13}$$

Now by using Kronecker product we obtain:

$$\begin{cases} (I\otimes A_1)\,\mathrm{Col}\,[X] + (B_1^T\otimes I)\,\mathrm{Col}\left[X^H\right] = \mathrm{Col}\,[F_1]\,, \\ \vdots \\ (I\otimes A_r)\,\mathrm{Col}\,[X] + (B_r^T\otimes I)\,\mathrm{Col}\left[X^H\right] = \mathrm{Col}\,[F_r]\,, \\ (I\otimes {B_1}^H)\,\mathrm{Col}\,[X] + (\overline{A_1}\otimes I)\,\mathrm{Col}\left[X^H\right] = \mathrm{Col}\left[{F_1}^H\right], \\ \vdots \\ (I\otimes {B_r}^H)\,\mathrm{Col}\,[X] + (\overline{A_r}\otimes I)\,\mathrm{Col}\left[X^H\right] = \mathrm{Col}\left[{F_r}^H\right], \end{cases} \tag{14}$$

or

$$\begin{bmatrix} I\otimes A_1 & B_1^T\otimes I \\ \vdots & \vdots \\ I\otimes A_r & B_r^T\otimes I \\ I\otimes B_1^H & \overline{A_1}\otimes I \\ \vdots & \vdots \\ I\otimes B_r^H & \overline{A_r}\otimes I \end{bmatrix}\begin{bmatrix} \mathrm{Col}[X] \\ \mathrm{Col}[X^H] \end{bmatrix} = \begin{bmatrix} \mathrm{Col}[F_1] \\ \vdots \\ \mathrm{Col}[F_r] \\ \mathrm{Col}[F_1^H] \\ \vdots \\ \mathrm{Col}[F_r^H] \end{bmatrix}.$$

Hence matrix equations (3) have a unique solution X if and only if $\mathrm{Rank}[\Theta_2, f_2] = \mathrm{Rank}[\Theta_2] = 2rn^2$. Moreover, the exact solution of matrix equations (3) can be determined by (11). □

In the continuation of this work, the lemmas that we require will pertain to the particular solutions of system of matrix equations (3).

Lemma 3.3. *System of equations* (3) *has a generalized reflexive solution* $X \in \mathbb{C}_r^{n\times n}(P,Q)$ *if and only if the following system of linear matrix equations is consistent:*

$$\begin{cases} A_1X + X^HB_1 = F_1, \\ A_1PXQ + (PXQ)^HB_1 = F_1, \\ \vdots \\ A_rX + X^HB_r = F_r, \\ A_rPXQ + (PXQ)^HB_r = F_r. \end{cases} \tag{15}$$

Proof. Suppose that the system (15) is consistent, then there exists a matrix $\tilde{X}$ such that (similar to the approach in [8]):

$$\begin{cases} A_1\tilde{X} + \tilde{X}^HB_1 = F_1, \\ A_1P\tilde{X}Q + (P\tilde{X}Q)^HB_1 = F_1, \\ \vdots \\ A_r\tilde{X} + \tilde{X}^HB_r = F_r, \\ A_rP\tilde{X}Q + (P\tilde{X}Q)^HB_r = F_r. \end{cases} \tag{16}$$

Define

$$\hat{X} = \frac{\tilde{X} + P\tilde{X}Q}{2}.$$

Then;

$$P\hat{X}Q = \frac{P\tilde{X}Q + P^2\tilde{X}Q^2}{2} = \frac{P\tilde{X}Q + \tilde{X}}{2} = \hat{X} \Rightarrow \hat{X} \in \mathbb{C}_r^{n\times n}(P,Q).$$

Additionally, it can be written:

$$A_j\hat{X} + \hat{X}^HB_j = A_j\left(\frac{\tilde{X} + P\tilde{X}Q}{2}\right) + \left(\frac{\tilde{X} + P\tilde{X}Q}{2}\right)^H B_j$$

$$= \frac{1}{2}\left(A_j\tilde{X} + \tilde{X}^HB_j\right) + \frac{1}{2}\left(A_jP\tilde{X}Q + (P\tilde{X}Q)^HB_j\right) \overset{(16)}{=} \frac{F_j}{2} + \frac{F_j}{2} = F_j, \quad j = 1,2,\dots,r.$$

On the other hand, if system (3) possesses the generalized reflexive solution $Z \in \mathbb{C}_r^{n\times n}(P,Q)$, then we can derive:

$$A_jPZQ + (PZQ)^H B_j = A_jZ + Z^H B_j = F_j, \quad j = 1,2,...,r. \tag{17}$$

From (17), it follows that the generalized reflexive solution Z is a solution to the system of matrix equations (15), meaning that system (15) is consistent. This completes the proof. □

Lemma 3.4. *System of equations* (3) *has a generalized ant-reflexive solution* $X \in \mathbb{C}_a^{n\times n}(P,Q)$ *if and only if the following system of linear matrix equations is consistent:*

$$\begin{cases} A_1X + X^H B_1 = F_1, \\ A_1PXQ + (PXQ)^H B_1 = -F_1, \\ \vdots \\ A_rX + X^H B_r = -F_r, \\ A_rPXQ + (PXQ)^H B_r = F_r, \end{cases} \tag{18}$$

Proof. Suppose that system (18) is consistent, then there exists a matrix $\tilde{X}$ such that (similar to the approach in [8]):

$$\begin{cases} A_1\tilde{X} + \tilde{X}^H B_1 = F_1, \\ A_1P\tilde{X}Q + (P\tilde{X}Q)^H B_1 = -F_1, \\ \vdots \\ A_r\tilde{X} + \tilde{X}^H B_r = -F_r, \\ A_rP\tilde{X}Q + (P\tilde{X}Q)^H B_r = F_r. \end{cases} \tag{19}$$

Define

$$\hat{X} = \frac{\tilde{X} - P\tilde{X}Q}{2}.$$

Then;

$$P\hat{X}Q = \frac{P\tilde{X}Q - P^2\tilde{X}Q^2}{2} = \frac{-\tilde{X} + P\tilde{X}Q}{2} = -\hat{X} \Rightarrow \hat{X} \in \mathbb{C}_a^{n\times n}(P,Q).$$

Additionally, it can be written:

$$A_j\hat{X} + \hat{X}^H B_j = A_j\left(\frac{\tilde{X} - P\tilde{X}Q}{2}\right) + \left(\frac{\tilde{X} - P\tilde{X}Q}{2}\right)^H B_j$$

$$= \frac{1}{2}\left(A_j\tilde{X} + \tilde{X}^H B_j\right) - \frac{1}{2}\left(A_jP\tilde{X}Q + (P\tilde{X}Q)^H B_j\right) = \frac{1}{2}F_j - \frac{1}{2}\left(-F_j\right) = F_j, \quad j = 1,2,...,r.$$

On the other hand, if system (3) possesses the generalized anti-reflexive solution $Z \in \mathbb{C}_a^{n\times n}(P,Q)$, then $PZQ = -Z$, $A_jZ + Z^H B_j = F_j$, $j = 1,2,...,r$ and:

$$A_jPZQ + (PZQ)^H B_j = -A_jZ - Z^H B_j = -F_j, \quad j = 1,2,...,r. \tag{20}$$

From (20), it follows that the generalized ant-reflexive solution Z is a solution to the system of matrix equations (18), meaning that system (18) is consistent. This completes the proof. □

3.1. Derive iterative algorithm

Consider a matrix A that can be expressed as the sum of three matrices: $A = M + N + G$, where M, N, and G are arbitrary matrices. For example a choice for these matrices is

$$M = \frac{1}{2}(L + L^H + U + U^H), \quad N = \frac{1}{2}(D + D^H), \quad G = \frac{1}{2}(L - L^H + U - U^H + D - D^H), \tag{21}$$

where U, L and D are the upper, lower, and diagonal parts of matrix A, respectively. Then it is easy to see that in (21), M and N are Hermitian and G is skew-Hermitian matrix.

Let's examine the following decompositions.

$$A_i = M_{a,i} + N_{a,i} + G_{a,i}, \quad B_i = M_{b,i} + N_{b,i} + G_{b,i}, \ i = 1, 2, \dots, r, \tag{22}$$

$$A_i = (M_{a,i} + \tau\Delta_a) - (\tau\Delta_a - G_{a,i} - N_{a,i}) \equiv (G_{a,i} + N_{a,i} + \gamma\Gamma_a) - (\gamma\Gamma_a - M_{a,i}), \tag{23}$$

$$B_i = (M_{b,i} + \tau\Delta_b) - (\tau\Delta_b - G_{b,i} - N_{b,i}) \equiv (G_{b,i} + N_{b,i} + \gamma\Gamma_b) - (\gamma\Gamma_b - M_{b,i}), \tag{24}$$

where τ and γ are real numbers, $M_{a,i}$ and $N_{a,i}$ are Hermitian matrices, $M_{b,i}$ and $N_{b,i}$ are also Hermitian matrices, and $G_{a,i}$ and $G_{b,i}$ are skew-Hermitian matrices for $i = 1, 2$. Furthermore, Δ_a, Δ_b, Γ_a, and Γ_b are arbitrary known matrices.

Drawing inspiration from the approach taken in [7], we can employ a hierarchical identification principle to solve system of equations (3). Utilizing the decompositions given in (23), we can express the system as:

$$(M_{a,i} + \tau\Delta_a)X = (\tau\Delta_a - G_{a,i} - N_{a,i})X - X^H B_i + F_i, \tag{25}$$

or

$$Z_{1,1}X = J_{1,1}, \ \dots, \ Z_{1,r}X = J_{1,r}, \tag{26}$$

such that

$$Z_{1,1} = M_{a,1} + \tau\Delta_a, \ \dots, \ Z_{1,r} = M_{a,r} + \tau\Delta_a, \tag{27}$$

and

$$\begin{aligned} J_{1,1} &= (\tau\Delta_a - G_{a,1} - N_{a,1})X - X^H B_1 + F_1, \\ &\vdots \\ J_{1,r} &= (\tau\Delta_a - G_{a,r} - N_{a,r})X - X^H B_r + F_r. \end{aligned} \tag{28}$$

The formula stated in the above equations concludes that:

$$S_1: \ Z_1 X = J_1, \tag{29}$$

where

$$Z_1 = \begin{bmatrix} Z_{1,1} \\ \vdots \\ Z_{1,r} \end{bmatrix}, \quad J_1 = \begin{bmatrix} J_{1,1} \\ \vdots \\ J_{1,r} \end{bmatrix}. \tag{30}$$

Applying the decompositions given in equation (23) to equations (3), gives:

$$(G_{a,i} + N_{a,i} + \gamma\Gamma_a)X = (\gamma\Gamma_a - M_{a,i})X - X^H B_i + F_i, \tag{31}$$

or

$$Z_{2,1}X = J_{2,1}, \ \dots, \ Z_{2,r}X = J_{2,r}, \tag{32}$$

where

$$Z_{2,1} = G_{a,1} + N_{a,1} + \gamma\Gamma_a, \ \dots, \ Z_{2,r} = G_{a,r} + N_{a,r} + \gamma\Gamma_a, \tag{33}$$

and

$$\begin{aligned} J_{2,1} &= (\gamma\Gamma_a - M_{a,1})\, X - X^H B_1 + F_1, \\ &\vdots \\ J_{2,r} &= (\gamma\Gamma_a - M_{a,r})\, X - X^H B_r + F_r. \end{aligned} \tag{34}$$

In a comparable manner, we can derive:

$$Z_{2,1}X = J_{2,1}, \ \dots, \ Z_{2,r}X = J_{2,r}, \tag{35}$$

or

$$S_2: \ Z_2X = J_2, \tag{36}$$

such that

$$Z_2 = \begin{bmatrix} Z_{2,1} \\ \vdots \\ Z_{2,r} \end{bmatrix}, \quad J_2 = \begin{bmatrix} J_{2,1} \\ \vdots \\ J_{2,r} \end{bmatrix}. \tag{37}$$

By substituting the splittings described in equation (24) into equations (3), the resulting expression is:

$$X^H(M_{b,i} + \tau\Delta_b) = X^H(\tau\Delta_b - G_{b,i} - N_{b,i}) - A_iX + F_i. \tag{38}$$

The matrices are defined as follows:

$$Z_{3,1} = M_{b,1} + \tau\Delta_b, \ \dots, \ Z_{3,r} = M_{b,r} + \tau\Delta_b, \tag{39}$$

and

$$\begin{aligned} J_{3,1} &= X^H(\tau\Delta_b - G_{b,1} - N_{b,1}) - A_1X + F_1, \\ &\vdots \\ J_{3,r} &= X^H(\tau\Delta_b - G_{b,r} - N_{b,r}) - A_rX + F_r. \end{aligned} \tag{40}$$

Thus, we can represent equation (38) in the following form:

$$S_3: X^HZ_3 = J_3, \tag{41}$$

where

$$Z_3 = \begin{bmatrix} Z_{3,1}, & \dots, & Z_{3,r} \end{bmatrix}, \quad J_3 = \begin{bmatrix} J_{3,1}, & \dots, & J_{3,r} \end{bmatrix}. \tag{42}$$

Applying comparable computations leads to the following relationships:

$$X^H(G_{b,i} + N_{b,i} + \gamma\Gamma_b) = X^H(\gamma\Gamma_b - M_{b,i}) - A_iX + F_i. \tag{43}$$

By defining

$$Z_{4,1} = G_{b,1} + N_{b,1} + \gamma\Gamma_b, \ \dots, \ Z_{4,r} = G_{b,r} + N_{b,r} + \gamma\Gamma_b, \tag{44}$$

and

$$\begin{aligned} J_{4,1} &= X^H(\gamma\Gamma_b - M_{b,1}) - A_1X + F_1, \\ &\vdots \\ J_{4,r} &= X^H(\gamma\Gamma_b - M_{b,r}) - A_rX + F_r, \end{aligned} \tag{45}$$

we get

$$S_4 : X^H Z_4 = J_4, \tag{46}$$

such that

$$Z_4 = \left[\begin{array}{ccc} Z_{4,1}, & \dots, & Z_{4,r} \end{array}\right], \quad J_4 = \left[\begin{array}{ccc} J_{4,1}, & \dots, & J_{4,r} \end{array}\right]. \tag{47}$$

Iterative methods for system S_1 can be obtained by utilizing gradient method (4) in the following manner:

$$X_1(k+1) = X_1(k) + \mu_1 \begin{bmatrix} M_{a,1} + \tau\Delta_a \\ \vdots \\ M_{a,r} + \tau\Delta_a \end{bmatrix}^H \begin{bmatrix} F_1 - A_1X_1(k) - X_1(k)^H B_1 \\ \vdots \\ F_r - A_rX_1(k) - X_1(k)^H B_r \end{bmatrix}.$$

The same procedure applied to system S_2 in (36) results in:

$$X_2(k+1) = X_2(k) + \mu_2 \begin{bmatrix} G_{a,1} + N_{a,1} + \gamma\Gamma_a \\ \vdots \\ G_{a,r} + N_{a,r} + \gamma\Gamma_a \end{bmatrix}^H \begin{bmatrix} F_1 - A_1X_2(k) - X_2(k)^H B_1 \\ \vdots \\ F_r - A_rX_2(k) - X_2(k)^H B_r \end{bmatrix}.$$

Applying gradient method (4) to system S_3 in (41), confirms:

$$\begin{aligned} X_3(k+1) &= X_3(k) + \mu_1 \left[M_{b,1} + \tau\Delta_b, \dots, M_{b,r} + \tau\Delta_b\right] \Big[F_1 - A_1X_3(k) - X_3(k)^H B_1, \\ &\dots, F_r - A_rX_3(k) - X_3(k)^H B_r\Big]^H. \end{aligned}$$

Consequently, we arrive at the following equation:

$$\begin{aligned} X_4(k+1) &= X_4(k) + \mu_2 \left[G_{b,1} + N_{b,1} + \gamma\Gamma_b, \dots, G_{b,r} + N_{b,r} + \gamma\Gamma_b\right] \\ &\times \Big[F_1 - A_1X_4(k) - X_4(k)^H B_1, \dots, F_r - A_rX_4(k) - X_4(k)^H B_r\Big]^H. \end{aligned}$$

A gradient iterative algorithm can be derived by computing the mean of $Y_i(k)$ for $i = 1, 2, 3, 4$ (refer to [7]):

$$X(k+1) = \frac{\sum_{i=1}^{4} X_i(k+1)}{4}, \tag{48}$$

where

$$X_1(k+1) = X(k) + \mu_1 \begin{bmatrix} M_{a,1} + \tau\Delta_a \\ \vdots \\ M_{a,r} + \tau\Delta_a \end{bmatrix}^H \begin{bmatrix} F_1 - A_1X(k) - X(k)^H B_1 \\ \vdots \\ F_r - A_rX(k) - X(k)^H B_r \end{bmatrix}, \tag{49}$$

$$X_2(k+1) = X(k) + \mu_2 \begin{bmatrix} G_{a,1} + N_{a,1} + \gamma\Gamma_a \\ \vdots \\ G_{a,r} + N_{a,r} + \gamma\Gamma_a \end{bmatrix}^H \begin{bmatrix} F_1 - A_1X(k) - X(k)^H B_1 \\ \vdots \\ F_r - A_rX(k) - X(k)^H B_r \end{bmatrix}, \tag{50}$$

$$\begin{aligned} X_3(k+1) &= X(k) + \mu_1 \left[M_{b,1} + \tau\Delta_b,\ \ldots, M_{b,r} + \tau\Delta_b\right] \Big[F_1 - A_1 X(k) - X(k)^H B_1, \\ &\ldots,\ F_r - A_r X(k) - X(k)^H B_r\Big]^H, \end{aligned} \tag{51}$$

and

$$\begin{aligned} X_4(k+1) &= X(k) + \mu_2 \left[G_{b,1} + N_{b,1} + \gamma\Gamma_b,\ \ldots\ , G_{b,r} + N_{b,r} + \gamma\Gamma_b\right] \\ &\times \Big[F_1 - A_1 X(k) - X(k)^H B_1, \ldots,\ F_r - A_r X(k) - X(k)^H B_r\Big]^H. \end{aligned} \tag{52}$$

For the sake of brevity, we will adopt the following notation:

$$\Psi_i(k) = F_i - A_i X(k) - X(k)^H B_i, \quad i = 1, 2, \ldots, r.$$

The new algorithm can be expressed in the following manner:

$$X_1(k+1) = X(k) + \mu_1 \begin{bmatrix} M_{a,1} + \tau\Delta_a \\ \vdots \\ M_{a,r} + \tau\Delta_a \end{bmatrix}^H \begin{bmatrix} \Psi_1(k) \\ \vdots \\ \Psi_s(k) \end{bmatrix},$$

$$X_2(k+1) = X(k) + \mu_2 \begin{bmatrix} G_{a,1} + N_{a,1} + \gamma\Gamma_a \\ \vdots \\ G_{a,r} + N_{a,r} + \gamma\Gamma_a \end{bmatrix}^H \begin{bmatrix} \Psi_1(k) \\ \vdots \\ \Psi_s(k) \end{bmatrix},$$

$$X_3(k+1) = X(k) + \mu_1 \left[M_{b,1} + \tau\Delta_b,\ \ldots, M_{b,r} + \tau\Delta_b\right] \left[\Psi_1(k), \ldots\ ,\ \Psi_s(k)\right]^H,$$

and

$$X_4(k+1) = X(k) + \mu_2 \left[G_{b,1} + N_{b,1} + \gamma\Gamma_b,\ \ldots, G_{b,r} + N_{b,r} + \gamma\Gamma_b\right] \left[\Psi_1(k), \ldots,\ \Psi_s(k)\right]^H.$$

The aforementioned equations lead to the conclusion that:

$$\begin{aligned} X(k+1) &= X(k) + \frac{1}{4} \begin{bmatrix} \mu_1 (M_{a,1} + \tau\Delta_a) + \mu_2 (G_{a,1} + N_{a,1} + \gamma\Gamma_a) \\ \vdots \\ \mu_1 (M_{a,r} + \tau\Delta_a) + \mu_2 (G_{a,r} + N_{a,r} + \gamma\Gamma_a) \end{bmatrix}^H \begin{bmatrix} \Psi_1(k) \\ \vdots \\ \Psi_s(k) \end{bmatrix} \\ &+ \frac{1}{4} \left[\mu_1 (M_{b,1} + \tau\Delta_b) + \mu_2 (G_{b,1} + N_{b,1} + \gamma\Gamma_b),\ \ldots\ , \mu_1 (M_{b,r} + \tau\Delta_b) + \mu_2 (G_{b,r} + N_{b,r} + \gamma\Gamma_b)\right] \\ &\qquad \times \left[\Psi_1(k), \ldots, \Psi_s(k)\right]^H, \end{aligned} \tag{53}$$

or

$$\begin{aligned} X(k+1) &= X(k) + \frac{1}{4}\Bigg\{\mu_1 \begin{bmatrix} A_1 \\ \vdots \\ A_r \end{bmatrix}^H + (\mu_2 - \mu_1) \begin{bmatrix} G_{a,1} + N_{a,1} \\ \vdots \\ G_{a,r} + N_{a,r} \end{bmatrix}^H + \mu_1\tau \begin{bmatrix} \Delta_a \\ \vdots \\ \Delta_a \end{bmatrix}^H \\ &+ \mu_2\gamma \begin{bmatrix} \Gamma_a \\ \vdots \\ \Gamma_a \end{bmatrix}^H \Bigg\} \begin{bmatrix} \Psi_1(k) \\ \vdots \\ \Psi_s(k) \end{bmatrix} + \frac{1}{4}\Big\{\mu_1 \left[\ B_1, \ldots, B_r\ \right] \\ &+ (\mu_2 - \mu_1) \left[\ G_{b,1} + N_{b,1}, \ldots, G_{b,r} + N_{b,r}\ \right] + \mu_1\tau[\Delta_b, \ldots,\ \Delta_b] + \mu_2\gamma[\Gamma_b, \ldots,\ \Gamma_b]\Big\} \left[\Psi_1(k), \ldots, \Psi_s(k)\right]^H. \end{aligned}$$

Therefore,

$$X(k+1) = X(k) + \frac{\mu_1}{4}\left\{\begin{bmatrix} A_1 \\ \vdots \\ A_r \end{bmatrix}^H \begin{bmatrix} \Psi_1(k) \\ \vdots \\ \Psi_s(k) \end{bmatrix} + \begin{bmatrix} B_1, \ldots, B_r \end{bmatrix} [\Psi_1(k), \ldots, \Psi_s(k)]^H \right\}$$

$$+ \frac{\mu_2 - \mu_1}{4}\left\{\begin{bmatrix} G_{a,1} + N_{a,1} \\ \vdots \\ G_{a,r} + N_{a,r} \end{bmatrix}^H \begin{bmatrix} \Psi_1(k) \\ \vdots \\ \Psi_s(k) \end{bmatrix} + [G_{b,1} + N_{b,1}, \ldots, G_{b,r} + N_{b,r}] [\Psi_1(k), \ldots, \Psi_s(k)]^H \right\}$$

$$+ \frac{\mu_1 \tau}{4}\left\{\begin{bmatrix} \Delta_a \\ \vdots \\ \Delta_a \end{bmatrix}^H \begin{bmatrix} \Psi_1(k) \\ \vdots \\ \Psi_s(k) \end{bmatrix} + [\Delta_b, \ldots, \Delta_b] [\Psi_1(k), \ldots, \Psi_s(k)]^H \right\}$$

$$+ \frac{\mu_2 \gamma}{4}\left\{\begin{bmatrix} \Gamma_a \\ \vdots \\ \Gamma_a \end{bmatrix}^H \begin{bmatrix} \Psi_1(k) \\ \vdots \\ \Psi_s(k) \end{bmatrix} + [\Gamma_b, \ldots, \Gamma_b] [\Psi_1(k), \ldots, \Psi_s(k)]^H \right\}.$$

Therefore

$$X(k+1) = X(k) + \frac{\mu_1}{4}\sum_{i=1}^{r}\left(A_i^H \Psi_i(k) + B_i \Psi_i(k)^H\right)$$
$$+ \frac{\mu_2 - \mu_1}{4}\sum_{i=1}^{r}\left((G_{a,i} + N_{a,i})^H \Psi_i(k) + (G_{b,i} + N_{b,i})\Psi_i(k)^H\right)$$
$$+ \frac{\mu_1 \tau}{4}\sum_{i=1}^{r}\left(\Delta_a^H \Psi_i(k) + \Delta_b \Psi_i(k)^H\right) + \frac{\mu_2 \gamma}{4}\sum_{i=1}^{r}\left(\Gamma_a^H \Psi_i(k) + \Gamma_b \Psi_i(k)^H\right). \quad (54)$$

As a final step, we establish the following iterative algorithm to solve equations (3).

Algorithm 1. To begin, select a matrix $X(1) \in \mathbb{C}^{n \times n}$ and real parameters $\mu_1, \mu_2, \tau, \gamma$. Then, for each $k = 1, 2, ...$, perform the following computation:

$$\Psi_i(k) = F_i - A_i X(k) - X(k)^H B_i, \quad i = 1, 2, \ldots, r,$$

$$X(k+1) = X(k) + \frac{\mu_1}{4}\sum_{i=1}^{r}\left(A_i^H \Psi_i(k) + B_i \Psi_i(k)^H\right) + \frac{\mu_2 - \mu_1}{4}\sum_{i=1}^{r}\left((N_{a,i} - G_{a,i})\Psi_i(k) + (N_{b,i} + G_{b,i})\Psi_i(k)^H\right)$$
$$+ \frac{\mu_1 \tau}{4}\sum_{i=1}^{r}\left(\Delta_a^H \Psi_i(k) + \Delta_b \Psi_i(k)^H\right) + \frac{\mu_2 \gamma}{4}\sum_{i=1}^{r}\left(\Gamma_a^H \Psi_i(k) + \Gamma_b \Psi_i(k)^H\right). \quad (55)$$

Remark 3.5. *In Algorithm 1 suppose $\mu_1 = \mu_2 = \mu$, $\tau = \gamma = 0$, then the following iterative method will be obtained*

$$X(k+1) = X(k) + \frac{\mu}{4}\sum_{i=1}^{r}\left(A_i^H \Psi_i(k) + B_i \Psi_i(k)^H\right) \quad (56)$$

that is the gradient iterative algorithm (GI) as described in [42].

This section aims to examine the convergence characteristics of Algorithm 1.

Theorem 3.6. *If we have coupled Sylvester matrix equations* (3) *with a unique solution X, the solution $X(k)$ derived from Algorithm 1 will converge to X^* provided that the inequality*

$$\sum_{i=1}^{r}\Bigg[\left\|\frac{I}{r}-\frac{\mu_2-\mu_1}{4}(N_{a,i}-G_{a,i})A_i-\frac{\mu_2-\mu_1}{4}(N_{b,i}+G_{b,i})B_i^H-\frac{\mu_1}{4}(A_i^HA_i+B_iB_i^H)\right\|$$
$$+\left\|\frac{\mu_1}{4}A_i^H+\frac{\mu_2-\mu_1}{4}(N_{a,i}-G_{a,i})\right\|\|B_i\|$$
$$+\left\|\frac{\mu_1}{4}B_i+\frac{\mu_2-\mu_1}{4}(N_{a,i}+G_{a,i})\right\|\|A_i\|+\frac{\mu_1\tau}{4}\{\|A_i\|+\|B_i\|\}\{\|\Delta_a\|+\|\Delta_b\|\}$$
$$+\frac{\mu_2\gamma}{4}\{\|A_i\|+\|B_i\|\}\{\|\Gamma_a\|+\|\Gamma_b\|\}\Bigg]<1, \quad (57)$$

holds for the parameters μ_1,μ_2,τ and γ, where $\|.\|$ is a matrix norm.

Proof. We start by defining the error matrix $\mathcal{E}(k)=X(k)-X^*$, where X^* is the true solution. Subsequently, applying Algorithm 1 leads to the following outcome (similar to the approach in [7]):

$$\mathcal{E}(k+1)=\mathcal{E}(k)+\frac{\mu_1}{4}\sum_{i=1}^{r}\left(A_i^H\Psi_i(k)+B_i\Psi_i(k)^H\right)+\frac{\mu_2-\mu_1}{4}\sum_{i=1}^{r}\left((N_{a,i}-G_{a,i})\Psi_i(k)+(N_{b,i}+G_{b,i})\Psi_i(k)^H\right)$$
$$+\frac{\mu_1\tau}{4}\sum_{i=1}^{r}\left(\Delta_a^H\Psi_i(k)+\Delta_b\Psi_i(k)^H\right)+\frac{\mu_2\gamma}{4}\sum_{i=1}^{r}\left(\Gamma_a^H\Psi_i(k)+\Gamma_b\Psi_i(k)^H\right). \quad (58)$$

On the other hand

$$\Psi_i(k)=-(A_i\mathcal{E}(k)+\mathcal{E}(k)^HB_i),\ i=1,...,r. \quad (59)$$

Hence the expression below can be obtained by combining equations (78) and (79):

$$\mathcal{E}(k+1)=\mathcal{E}(k)-\frac{\mu_1}{4}\sum_{i=1}^{r}\left(A_i^HA_i\mathcal{E}(k)+A_i^H\mathcal{E}(k)^HB_i+B_i(\mathcal{E}(k)^HA_i^H+B_i^H\mathcal{E}(k))\right)$$
$$-\frac{\mu_2-\mu_1}{4}\sum_{i=1}^{r}\left((N_{a,i}-G_{a,i})(A_i\mathcal{E}(k)+\mathcal{E}(k)^HB_i)+(N_{b,i}+G_{b,i})(\mathcal{E}(k)^HA_i^H+B_i^H\mathcal{E}(k))\right)$$
$$-\frac{\mu_1\tau}{4}\sum_{i=1}^{r}\left(\Delta_a^H(A_i\mathcal{E}(k)+\mathcal{E}(k)^HB_i)+\Delta_b(\mathcal{E}(k)^HA_i^H+B_i^H\mathcal{E}(k))\right)$$
$$-\frac{\mu_2\gamma}{4}\sum_{i=1}^{r}\left(\Gamma_a^H(A_i\mathcal{E}(k)+\mathcal{E}(k)^HB_i)+\Gamma_b(\mathcal{E}(k)^HA_i^H+B_i^H\mathcal{E}(k))\right)$$

$$=\mathcal{E}(k)-\frac{\mu_1}{4}\sum_{i=1}^{r}\left(A_i^HA_i\mathcal{E}(k)+A_i^H\mathcal{E}(k)^HB_i+B_i\mathcal{E}(k)^HA_i^H+B_iB_i^H\mathcal{E}(k)\right)$$
$$-\frac{\mu_2-\mu_1}{4}\sum_{i=1}^{r}((N_{a,i}-G_{a,i})A_i\mathcal{E}(k)+(N_{a,i}-G_{a,i})\mathcal{E}(k)^HB_i+(N_{b,i}+G_{b,i})\mathcal{E}(k)^HA_i^H+(N_{b,i}+G_{b,i})B_i^H\mathcal{E}(k))$$
$$-\frac{\mu_1\tau}{4}\sum_{i=1}^{r}\left(\Delta_a^HA_i\mathcal{E}(k)+\Delta_a^H\mathcal{E}(k)^HB_i+\Delta_b\mathcal{E}(k)^HA_i^H+\Delta_bB_i^H\mathcal{E}(k)\right)$$
$$-\frac{\mu_2\gamma}{4}\sum_{i=1}^{r}\left(\Gamma_a^HA_i\mathcal{E}(k)+\Gamma_a^H\mathcal{E}(k)^HB_i+\Gamma_b\mathcal{E}(k)^HA_i^H+\Gamma_bB_i^H\mathcal{E}(k)\right)$$

$$= \mathcal{E}(k) - \frac{\mu_1}{4}\sum_{i=1}^{r}\left(A_i^H A_i\mathcal{E}(k) + B_iB_i^H\mathcal{E}(k)\right) - \frac{\mu_1}{4}\sum_{i=1}^{r}\left(A_i^H\mathcal{E}(k)^H B_i + B_i\mathcal{E}(k)^H A_i^H\right)$$
$$- \frac{\mu_2-\mu_1}{4}\sum_{i=1}^{r}((N_{a,i}-G_{a,i})A_i\mathcal{E}(k) + (N_{b,i}+G_{b,i})B_i^H\mathcal{E}(k))$$
$$- \frac{\mu_2-\mu_1}{4}\sum_{i=1}^{r}((N_{a,i}-G_{a,i})\mathcal{E}(k)^H B_i + (N_{b,i}+G_{b,i})\mathcal{E}(k)^H A_i^H)$$
$$- \frac{\mu_1\tau}{4}\sum_{i=1}^{r}\left(\Delta_a^H A_i\mathcal{E}(k) + \Delta_a^H\mathcal{E}(k)^H B_i + \Delta_b\mathcal{E}(k)^H A_i^H + \Delta_b B_i^H\mathcal{E}(k)\right)$$
$$- \frac{\mu_2\gamma}{4}\sum_{i=1}^{r}\left(\Gamma_a^H A_i\mathcal{E}(k) + \Gamma_a^H\mathcal{E}(k)^H B_i + \Gamma_b\mathcal{E}(k)^H A_i^H + \Gamma_b B_i^H\mathcal{E}(k)\right)$$

$$= \sum_{i=1}^{r}\left(\frac{I}{r} - \frac{\mu_2-\mu_1}{4}(N_{a,i}-G_{a,i})A_i - \frac{\mu_2-\mu_1}{4}(N_{b,i}+G_{b,i})B_i^H\right.$$
$$\left. - \frac{\mu_1}{4}(A_i^H A_i + B_iB_i^H)\right)\mathcal{E}(k) - \sum_{i=1}^{r}\left(\frac{\mu_1}{4}A_i^H + \frac{\mu_2-\mu_1}{4}(N_{a,i}-G_{a,i})\right)\mathcal{E}(k)^H B_i$$
$$-\sum_{i=1}^{r}\left(\frac{\mu_1}{4}B_i + \frac{\mu_2-\mu_1}{4}(N_{a,i}+G_{a,i})\right)\mathcal{E}(k)^H A_i^H - \frac{\mu_1\tau}{4}\sum_{i=1}^{r}\left(\Delta_a^H A_i\mathcal{E}(k) + \Delta_a^H\mathcal{E}(k)^H B_i + \Delta_b\mathcal{E}(k)^H A_i^H + \Delta_b B_i^H\mathcal{E}(k)\right)$$
$$- \frac{\mu_2\gamma}{4}\sum_{i=1}^{r}\left(\Gamma_a^H A_i\mathcal{E}(k) + \Gamma_a^H\mathcal{E}(k)^H B_i + \Gamma_b\mathcal{E}(k)^H A_i^H + \Gamma_b B_i^H\mathcal{E}(k)\right).$$

Furthermore, through the application of matrix norm to each side of above equation, one can derive:

$$\|\mathcal{E}(k+1)\| \leq \sum_{i=1}^{r}\left\|\frac{I}{r} - \frac{\mu_2-\mu_1}{4}(N_{a,i}-G_{a,i})A_i - \frac{\mu_2-\mu_1}{4}(N_{b,i}+G_{b,i})B_i^H\right.$$
$$\left. - \frac{\mu_1}{4}(A_i^H A_i + B_iB_i^H)\right\| \|\mathcal{E}(k)\| + \sum_{i=1}^{r}\left\|\frac{\mu_1}{4}A_i^H + \frac{\mu_2-\mu_1}{4}(N_{a,i}-G_{a,i})\right\| \|B_i\| \, \|\mathcal{E}(k)\|$$
$$+ \sum_{i=1}^{r}\left\|\frac{\mu_1}{4}B_i + \frac{\mu_2-\mu_1}{4}(N_{a,i}+G_{a,i})\right\| \|A_i\| \, \|\mathcal{E}(k)\|$$
$$+ \frac{\mu_1\tau}{4}\sum_{i=1}^{r}\left\|\Delta_a^H A_i\mathcal{E}(k) + \Delta_a^H\mathcal{E}(k)^H B_i + \Delta_b\mathcal{E}(k)^H A_i^H + \Delta_b B_i^H\mathcal{E}(k)\right\|$$
$$+ \frac{\mu_2\gamma}{4}\sum_{i=1}^{r}\left\|\Gamma_a^H A_i\mathcal{E}(k) + \Gamma_a^H\mathcal{E}(k)^H B_i + \Gamma_b\mathcal{E}(k)^H A_i^H + \Gamma_b B_i^H\mathcal{E}(k)\right\|, \quad (60)$$

or

$$\|\mathcal{E}(k+1)\| \leq \Bigg[\sum_{i=1}^{r}\left\|\frac{I}{r} - \frac{\mu_2-\mu_1}{4}(N_{a,i}-G_{a,i})A_i - \frac{\mu_2-\mu_1}{4}(N_{b,i}+G_{b,i})B_i^H\right.$$
$$\left. - \frac{\mu_1}{4}(A_i^H A_i + B_iB_i^H)\right\| + \sum_{i=1}^{r}\left\|\frac{\mu_1}{4}A_i^H + \frac{\mu_2-\mu_1}{4}(N_{a,i}-G_{a,i})\right\| \|B_i\|$$

$$+ \sum_{i=1}^{r} \left\| \frac{\mu_1}{4} B_i + \frac{\mu_2 - \mu_1}{4}(N_{a,i} + G_{a,i}) \right\| \|A_i\| + \frac{\mu_1 \tau}{4} \sum_{i=1}^{r} \{\|A_i\| + \|B_i\|\}\{\|\Delta_a\| + \|\Delta_b\|\}$$
$$+ \frac{\mu_2 \gamma}{4} \sum_{i=1}^{r} \{\|A_i\| + \|B_i\|\}\{\|\Gamma_a\| + \|\Gamma_b\|\} \Big] \|\mathcal{E}(k)\|. \quad (61)$$

It is apparent that the equation presented above can be rephrased as:

$$\|\mathcal{E}(k+1)\| \leq \sum_{i=1}^{r} \Big[\Big\| \frac{I}{r} - \frac{\mu_2 - \mu_1}{4}(N_{a,i} - G_{a,i})A_i - \frac{\mu_2 - \mu_1}{4}(N_{b,i} + G_{b,i})B_i^H$$
$$- \frac{\mu_1}{4}(A_i^H A_i + B_i B_i^H) \Big\| + \left\| \frac{\mu_1}{4} A_i^H + \frac{\mu_2 - \mu_1}{4}(N_{a,i} - G_{a,i}) \right\| \|B_i\|$$
$$+ \left\| \frac{\mu_1}{4} B_i + \frac{\mu_2 - \mu_1}{4}(N_{a,i} + G_{a,i}) \right\| \|A_i\| + \frac{\mu_1 \tau}{4} \{\|A_i\| + \|B_i\|\}\{\|\Delta_a\| + \|\Delta_b\|\}$$
$$+ \frac{\mu_2 \gamma}{4} \{\|A_i\| + \|B_i\|\}\{\|\Gamma_a\| + \|\Gamma_b\|\} \Big] \|\mathcal{E}(k)\|. \quad (62)$$

From equation (62), we conclude that if (57) is satisfied, then $\lim_{k\to\infty} \mathcal{E}(k) = 0$, that gives

$$\lim_{k\to\infty} X(k) = X^*,$$

which completes the proof immediately. □

Remark 3.7. *Note that even if the condition (57) is not satisfied, Algorithm 1 can be employed. This is because, during the proof, we observe that the control inequality serves as a sufficient criterion rather than a necessary one.*

Remark 3.8. *Let A_i, B_i, $F_i \in \mathbb{C}^{n\times n}$ $(i = 1, \dots, r)$ and there exists a_i such that $A_i^H A_i + B_i B_i^H = a_i I$ $(i = 1, \dots, r)$(for example let A_i and B_i be unitary matrices). Suppose $\mu = \mu_1 = \mu_2$. Then the inequality of (57) will be as follows:*

$$\sum_{i=1}^{r} \left[\left\| \frac{I}{r} - \frac{\mu a_i}{4} I \right\| + \frac{\mu}{2} \|A_i\| \|B_i\| + \frac{\mu\tau}{4} \{\|A_i\| + \|B_i\|\}\{\|\Delta_a\| + \|\Delta_b\|\} + \frac{\mu\gamma}{4} \{\|A_i\| + \|B_i\|\}\{\|\Gamma_a\| + \|\Gamma_b\|\} \right] < 1. \quad (63)$$

Then we have

$$\sum_{i=1}^{r} \left\| \frac{I}{r} - \frac{\mu a_i}{4} I \right\| + \mu\theta_1 + \mu\tau\theta_2 + \mu\gamma\theta_3 < 1,$$

where

$$\theta_1 = \frac{1}{2} \sum_{i=1}^{r} \|A_i\| \|B_i\|, \quad (64)$$

$$\theta_2 = \frac{1}{4}(\|\Delta_a\| + \|\Delta_b\|) \sum_{i=1}^{r} (\|A_i\| + \|B_i\|), \quad (65)$$

$$\theta_3 = \frac{1}{4}(\|\Gamma_a\| + \|\Gamma_b\|) \sum_{i=1}^{r} (\|A_i\| + \|B_i\|). \quad (66)$$

Now suppose we use Euclidean norm, then

$$\sum_{i=1}^{r} \left| \frac{1}{r} - \frac{\mu a_i}{4} \right| + \mu\theta_1 + \mu\tau\theta_2 + \mu\gamma\theta_3 < 1.$$

It can be easily seen that a solution for the above inequality is as follows

$$\left|4 - r\mu a_i\right| < 1, \ (i = 1, \dots, r), \quad \mu\gamma < \frac{1}{4\theta_3}, \quad \mu\tau < \frac{1}{4\theta_2}, \quad \mu < \frac{1}{4\theta_1}.$$

4. Determining the generalized reflexive and anti-reflexive solutions

This section aims to derive solutions for matrix equations (3), encompassing both generalized reflexive and anti-reflexive cases, while also examining the convergence characteristics of these techniques.

4.1. Generalized reflexive solution

By applying Lemma 3.3, we can obtain the generalized reflexive solution of system (3) by solving the following equations:

$$\begin{cases} A_1X + X^HB_1 = F_1, \\ A_1PXQ + (PXQ)^HB_1 = F_1, \\ \vdots \\ A_rX + X^HB_r = F_r, \\ A_rPXQ + (PXQ)^HB_r = F_r. \end{cases} \tag{67}$$

By following a similar approach as in the preceding section, we can obtain equations:

$$(M_{a,i} + \tau\Delta_a)PXQ = (\tau\Delta_a - G_{a,i} - N_{a,i})\,PXQ - (PXQ)^HB_i + F_i, \tag{68}$$

$$(G_{a,i} + N_{a,i} + \gamma\Gamma_a)PXQ = (\gamma\Gamma_a - M_{a,i})\,PXQ - (PXQ)^HB_i + F_i, \tag{69}$$

$$(PXQ)^H(M_{b,i} + \tau\Delta_b) = (PXQ)^H\,(\tau\Delta_b - G_{b,i} - N_{b,i}) - A_iPXQ + F_i, \tag{70}$$

$$(PXQ)^H(G_{b,i} + N_{b,i} + \gamma\Gamma_b) = (PXQ)^H\,(\gamma\Gamma_b - M_{b,i}) - A_iPXQ + F_i, \tag{71}$$

for $i = 1, 2, ..., r$. The implementation of the Hierarchical identification principle to the above-mentioned equations results in the following outcomes:

$$X_5(k+1) = X(k) + \mu_1P^H\begin{bmatrix} M_{a,1} + \tau\Delta_a \\ \vdots \\ M_{a,r} + \tau\Delta_a \end{bmatrix}^H\begin{bmatrix} F_1 - A_1PX(k)Q - (PX(k)Q)^HB_1 \\ \vdots \\ F_r - A_rPX(k)Q - (PX(k)Q)^HB_r \end{bmatrix}Q^H,$$

$$X_6(k+1) = X(k) + \mu_2P^H\begin{bmatrix} G_{a,1} + N_{a,1} + \gamma\Gamma_a \\ \vdots \\ G_{a,r} + N_{a,r} + \gamma\Gamma_a \end{bmatrix}^H\begin{bmatrix} F_1 - A_1PX(k)Q - (PX(k)Q)^HB_1 \\ \vdots \\ F_r - A_rPX(k)Q - (PX(k)Q)^HB_r \end{bmatrix}Q^H,$$

$$X_7(k+1) = X(k) + \mu_1P^H\left[M_{b,1} + \tau\Delta_b,\ \ldots, M_{b,r} + \tau\Delta_b\right]\Big[F_1 - A_1(PX(k)Q) - (PX(k)Q)^HB_1,$$
$$\ldots\ ,\ F_r - A_r(PX(k)Q) - (PX(k)Q)^HB_r\Big]^H Q^H,$$

and

$$X_8(k+1) = X(k) + \mu_2P^H\left[G_{b,1} + N_{b,1} + \gamma\Gamma_b,\ \ldots\ , G_{b,r} + N_{b,r} + \gamma\Gamma_b\right]$$
$$\times\Big[F_1 - A_1(PX(k)Q) - (PX(k)Q)^HB_1, \ldots\ ,\ F_r - A_r(PX(k)Q) - (PX(k)Q)^HB_r\Big]^H Q^H.$$

To derive an iterative gradient algorithm, we can compute the mean of $X_i(k)$ for $i = 1, 2, ..., 8$:

$$X(k+1) = \frac{\sum_{i=1}^{8} X_i(k)}{8}, \tag{72}$$

where $X_1(k), X_2(k), X_3(k)$ and $X_4(k)$ are defined in Eqs. (49)-(52) and

$$X_5(k+1) = X(k) + \mu_1P\begin{bmatrix} M_{a,1} + \tau\Delta_a \\ \vdots \\ M_{a,r} + \tau\Delta_a \end{bmatrix}^H\begin{bmatrix} F_1 - A_1PX(k)Q - (PX(k)Q)^HB_1 \\ \vdots \\ F_r - A_r(PX(k)Q) - (PX(k)Q)^HB_r \end{bmatrix}Q, \tag{73}$$

$$X_6(k+1) = X(k) + \mu_2 P \begin{bmatrix} G_{a,1} + N_{a,1} + \gamma\Gamma_a \\ \vdots \\ G_{a,r} + N_{a,r} + \gamma\Gamma_a \end{bmatrix}^H \begin{bmatrix} F_1 - A_1(PX(k)Q) - (PX(k)Q)^H B_1 \\ \vdots \\ F_r - A_r(PX(k)Q) - (PX(k)Q)^H B_r \end{bmatrix} Q, \tag{74}$$

$$\begin{aligned} X_7(k+1) &= X(k) + \mu_1 P\left[M_{b,1} + \tau\Delta_b,\ \ldots, M_{b,r} + \tau\Delta_b\right]\Big[F_1 - A_1(PX(k)Q) - (PX(k)Q)^H B_1, \\ &\ldots,\ F_r - A_r(PX(k)Q) - (PX(k)Q)^H B_r\Big]^H Q, \end{aligned} \tag{75}$$

and

$$\begin{aligned} X_8(k+1) &= X(k) + \mu_2 P\left[G_{b,1} + N_{b,1} + \gamma\Gamma_b,\ \ldots\ , G_{b,r} + N_{b,r} + \gamma\Gamma_b\right] \\ &\times \Big[F_1 - A_1(PX(k)Q) - (PX(k)Q)^H B_1, \ldots,\ F_r - A_r(PX(k)Q) - (PX(k)Q)^H B_r\Big]^H Q. \end{aligned} \tag{76}$$

It is easy to check that if $X(1) \in \mathbb{C}_r^{n\times n}(P, Q)$ then

$$F_i - A_i(PX(k)Q) - (PX(k)Q)^H B_i = F_i - A_i X(k) - X(k)^H B_i,\ i = 1, ..., r,\ k = 1, 2, ...$$

Finally, from (49)-(52), (72), (73)-(76), the following iterative algorithm is determined to solve equations (3) over generalized reflexive matrices.

Algorithm 2. Choose an initial matrix $X(1) \in \mathbb{C}_r^{n\times n}(P, Q)$ and real parameters $\mu_1, \mu_2, \tau, \gamma$. For $k = 1, 2, ...,$ compute:

$$\Psi_i(k) = F_i - A_i X(k) - X(k)^H B_i, \quad i = 1, 2, \ldots, r.$$

$$\begin{aligned} X(k+1) &= X(k) + \frac{\mu_1}{8}\sum_{i=1}^{r}\Big(A_i^H\Psi_i(k) + B_i\Psi_i(k)^H + PA_i^H\Psi_i(k)Q + PB_i\Psi_i(k)^H Q\Big) \\ &+ \frac{\mu_2 - \mu_1}{8}\sum_{i=1}^{r}\Big((N_{a,i} - G_{a,i})\Psi_i(k) + (N_{b,i} + G_{b,i})\Psi_i(k)^H + P(N_{a,i} - G_{a,i})\Psi_i(k)Q + P(N_{b,i} + G_{b,i})\Psi_i(k)^H Q\Big) \\ &+ \frac{\mu_1\tau}{8}\sum_{i=1}^{r}\Big(\Delta_a^H\Psi_i(k) + \Delta_b\Psi_i(k)^H + P\Delta_a^H\Psi_i(k)Q + P\Delta_b\Psi_i(k)^H Q\Big) \\ &+ \frac{\mu_2\gamma}{8}\sum_{i=1}^{r}\Big(\Gamma_a^H\Psi_i(k) + \Gamma_b\Psi_i(k)^H + P\Gamma_a^H\Psi_i(k)Q + P\Gamma_b\Psi_i(k)^H Q\Big). \end{aligned} \tag{77}$$

The focus of this section is to analyze the convergence properties of the Algorithm 2.

Theorem 4.1. *Given coupled Sylvester matrix equations* (3) *with a unique generalized reflexive solution* $X^* \in \mathbb{C}_r^{n\times n}(P, Q)$*, the iterative solution* $X(k)$ *obtained from Algorithm 2 converges to* X^* *when inequality* (57) *holds for the parameters* μ_1, μ_2, τ *and* γ*, where* $\|.\|$ *is a matrix norm.*

Proof. We can apply a similar approach as demonstrated in [7] to prove this theorem. Initially, let the matrix of errors given by

$$\mathcal{E}(k) = X(k) - X^*.$$

After that, applying Algorithm 2 will yield the following result:

$$\begin{aligned} \mathcal{E}(k+1) &= \mathcal{E}(k) + \frac{\mu_1}{8}\sum_{i=1}^{r}\Big(A_i^H\Psi_i(k) + B_i\Psi_i(k)^H + PA_i^H\Psi_i(k)Q + PB_i\Psi_i(k)^H Q\Big) \\ &+ \frac{\mu_2 - \mu_1}{8}\sum_{i=1}^{r}\Big((N_{a,i} - G_{a,i})\Psi_i(k) + (N_{b,i} + G_{b,i})\Psi_i(k)^H + P(N_{a,i} - G_{a,i})\Psi_i(k)Q + P(N_{b,i} + G_{b,i})\Psi_i(k)^H Q\Big) \end{aligned}$$

$$+\frac{\mu_1\tau}{8}\sum_{i=1}^{r}\Big(\Delta_a^H\Psi_i(k)+\Delta_b\Psi_i(k)^H+P\Delta_a^H\Psi_i(k)Q+P\Delta_b\Psi_i(k)^HQ\Big)$$
$$+\frac{\mu_2\gamma}{8}\sum_{i=1}^{r}\Big(\Gamma_a^H\Psi_i(k)+\Gamma_b\Psi_i(k)^H+P\Gamma_a^H\Psi_i(k)Q+P\Gamma_b\Psi_i(k)^HQ\Big). \quad (78)$$

On the other hand we have

$$\Psi_i(k)=-(A_i\mathcal{E}(k)+\mathcal{E}(k)^HB_i),\ i=1,\dots,r. \quad (79)$$

Therefore, by combining equations (78) and (79), we can derive

$$\mathcal{E}(k+1)=\mathcal{E}(k)-\frac{\mu_1}{8}\sum_{i=1}^{r}\Big(A_i^HA_i\mathcal{E}(k)+A_i^H\mathcal{E}(k)^HB_i+B_i\mathcal{E}(k)^HA_i^H+B_iB_i^H\mathcal{E}(k)$$
$$+PA_i^HA_i\mathcal{E}(k)Q+PA_i^H\mathcal{E}(k)^HB_iQ+PB_i\mathcal{E}(k)^HA_i^HQ+PB_iB_i^H\mathcal{E}(k)Q\Big)$$
$$-\frac{\mu_2-\mu_1}{8}\sum_{i=1}^{r}\Big((N_{a,i}-G_{a,i})A_i\mathcal{E}(k)+(N_{a,i}-G_{a,i})\mathcal{E}(k)^HB_i$$
$$+(N_{b,i}+G_{b,i})\mathcal{E}(k)^HA_i^H+(N_{b,i}+G_{b,i})B_i^H\mathcal{E}(k)+P(N_{a,i}-G_{a,i})A_i\mathcal{E}(k)Q+P(N_{a,i}-G_{a,i})\mathcal{E}(k)^HB_iQ$$
$$+P(N_{b,i}+G_{b,i})\mathcal{E}(k)^HA_i^HQ+P(N_{b,i}+G_{b,i})B_i^H\mathcal{E}(k)Q\Big)$$
$$-\frac{\mu_1\tau}{8}\sum_{i=1}^{r}\Big(\Delta_a^HA_i\mathcal{E}(k)+\Delta_a^H\mathcal{E}(k)^HB_i+\Delta_b\mathcal{E}(k)^HA_i^H+\Delta_bB_i^H\mathcal{E}(k)$$
$$+P\Delta_a^HA_i\mathcal{E}(k)Q+P\Delta_a^H\mathcal{E}(k)^HB_iQ+P\Delta_b\mathcal{E}(k)^HA_i^HQ+P\Delta_bB_i^H\mathcal{E}(k)Q\Big)$$
$$-\frac{\mu_2\gamma}{8}\sum_{i=1}^{r}\Big(\Gamma_a^HA_i\mathcal{E}(k)+\Gamma_a^H\mathcal{E}(k)^HB_i+\Gamma_b\mathcal{E}(k)^HA_i^H+\Gamma_bB_i^H\mathcal{E}(k)$$
$$+P\Gamma_a^HA_i\mathcal{E}(k)Q+P\Gamma_a^H\mathcal{E}(k)^HB_iQ+P\Gamma_b\mathcal{E}(k)^HA_i^HQ+P\Gamma_bB_i^H\mathcal{E}(k)Q\Big)$$
$$=\sum_{i=1}^{r}\Big(\frac{I}{2r}-\frac{\mu_2-\mu_1}{8}(N_{a,i}-G_{a,i})A_i-\frac{\mu_2-\mu_1}{8}(N_{b,i}+G_{b,i})B_i^H-\frac{\mu_1}{8}A_i^HA_i-\frac{\mu_1}{8}B_iB_i^H\Big)\mathcal{E}(k)$$
$$-\sum_{i=1}^{r}\Big(\frac{\mu_1}{8}A_i^H+\frac{\mu_2-\mu_1}{8}(N_{a,i}-G_{a,i})\Big)\mathcal{E}(k)^HB_i-\sum_{i=1}^{r}\Big(\frac{\mu_1}{8}B_i+\frac{\mu_2-\mu_1}{8}(N_{a,i}+G_{a,i})\Big)\mathcal{E}(k)^HA_i^H$$
$$-\frac{\mu_1\tau}{8}\sum_{i=1}^{r}\Big(\Delta_a^HA_i\mathcal{E}(k)+\Delta_a^H\mathcal{E}(k)^HB_i+\Delta_b\mathcal{E}(k)^HA_i^H+\Delta_bB_i^H\mathcal{E}(k)\Big)$$
$$-\frac{\mu_2\gamma}{8}\sum_{i=1}^{r}\Big(\Gamma_a^HA_i\mathcal{E}(k)+\Gamma_a^H\mathcal{E}(k)^HB_i+\Gamma_b\mathcal{E}(k)^HA_i^H+\Gamma_bB_i^H\mathcal{E}(k)\Big)$$
$$+\sum_{i=1}^{r}P\Big(\frac{I}{2r}-\frac{\mu_2-\mu_1}{8}(N_{a,i}-G_{a,i})A_i-\frac{\mu_2-\mu_1}{8}(N_{b,i}+G_{b,i})B_i^H-\frac{\mu_1}{8}A_i^HA_i-\frac{\mu_1}{8}B_iB_i^H\Big)\mathcal{E}(k)Q$$
$$-\sum_{i=1}^{r}P\Big(\frac{\mu_1}{8}A_i^H+\frac{\mu_2-\mu_1}{8}(N_{a,i}-G_{a,i})\Big)\mathcal{E}(k)^HB_iQ-\sum_{i=1}^{r}P\Big(\frac{\mu_1}{8}B_i+\frac{\mu_2-\mu_1}{8}(N_{a,i}+G_{a,i})\Big)\mathcal{E}(k)^HA_i^HQ$$
$$-\frac{\mu_1\tau}{8}\sum_{i=1}^{r}P\Big(\Delta_a^HA_i\mathcal{E}(k)+\Delta_a^H\mathcal{E}(k)^HB_i+\Delta_b\mathcal{E}(k)^HA_i^H+\Delta_bB_i^H\mathcal{E}(k)\Big)Q$$

$$- \frac{\mu_2\gamma}{8}\sum_{i=1}^{r} P\Big(\Gamma_a^H A_i\mathcal{E}(k) + \Gamma_a^H\mathcal{E}(k)^H B_i + \Gamma_b\mathcal{E}(k)^H A_i^H + \Gamma_b B_i^H\mathcal{E}(k)\Big)Q.$$

Taking the matrix norm of both sides of the last equation yields:

$$\begin{aligned}
\|\mathcal{E}(k+1)\| \le{}& 2\sum_{i=1}^{r}\Big\|\frac{I}{2r} - \frac{\mu_2-\mu_1}{8}(N_{a,i}-G_{a,i})A_i - \frac{\mu_2-\mu_1}{8}(N_{b,i}+G_{b,i})B_i^H \\
&- \frac{\mu_1}{8}A_i^H A_i - \frac{\mu_1}{8}B_iB_i^H\Big\|\,\|\mathcal{E}(k)\| + 2\sum_{i=1}^{r}\Big\|\frac{\mu_1}{8}A_i^H + \frac{\mu_2-\mu_1}{8}(N_{a,i}-G_{a,i})\Big\|\,\|B_i\|\,\|\mathcal{E}(k)\| \\
&+ 2\sum_{i=1}^{r}\Big\|\frac{\mu_1}{8}B_i + \frac{\mu_2-\mu_1}{8}(N_{a,i}+G_{a,i})\Big\|\,\|A_i\|\,\|\mathcal{E}(k)\| \\
&+ \frac{\mu_1\tau}{4}\sum_{i=1}^{r}\Big\|\Delta_a^H A_i\mathcal{E}(k) + \Delta_a^H\mathcal{E}(k)^H B_i + \Delta_b\mathcal{E}(k)^H A_i^H + \Delta_b B_i^H\mathcal{E}(k)\Big\| \\
&+ \frac{\mu_2\gamma}{4}\sum_{i=1}^{r}\Big\|\Gamma_a^H A_i\mathcal{E}(k) + \Gamma_a^H\mathcal{E}(k)^H B_i + \Gamma_b\mathcal{E}(k)^H A_i^H + \Gamma_b B_i^H\mathcal{E}(k)\Big\|. \qquad (80)
\end{aligned}$$

As a result of the analysis, it can be concluded that the sequence $\{X(k)\}$ converges provided that condition (57) holds. □

Remark 4.2. *If we set $P = Q$ in Algorithm 2, the resulting algorithm provides the reflexive solution of matrix equations* (3)*, and it can be simplified as follows:*

Algorithm 3. Choose an initial matrix $X(1) \in \mathbb{C}_r^{n\times n}(P)$ and real parameters $\mu_1, \mu_2, \tau, \gamma$. For $k = 1, 2, ...,$ compute:

$$\Psi_i(k) = F_i - A_iX(k) - X(k)^H B_i, \quad i = 1, 2, \dots, r.$$

$$\begin{aligned}
X(k+1) ={}& X(k) + \frac{\mu_1}{8}\sum_{i=1}^{r}\Big(A_i^H\Psi_i(k) + B_i\Psi_i(k)^H + PA_i^H\Psi_i(k)P + PB_i\Psi_i(k)^H P\Big) \\
&+ \frac{\mu_2-\mu_1}{8}\sum_{i=1}^{r}\Big((N_{a,i}-G_{a,i})\Psi_i(k) + (N_{b,i}+G_{b,i})\Psi_i(k)^H + P(N_{a,i}-G_{a,i})\Psi_i(k)P + P(N_{b,i}+G_{b,i})\Psi_i(k)^H P\Big) \\
&+ \frac{\mu_1\tau}{8}\sum_{i=1}^{r}\Big(\Delta_a^H\Psi_i(k) + \Delta_b\Psi_i(k)^H + P\Delta_a^H\Psi_i(k)P + P\Delta_b\Psi_i(k)^H P\Big) \\
&+ \frac{\mu_2\gamma}{8}\sum_{i=1}^{r}\Big(\Gamma_a^H\Psi_i(k) + \Gamma_b\Psi_i(k)^H + P\Gamma_a^H\Psi_i(k)P + P\Gamma_b\Psi_i(k)^H P\Big). \qquad (81)
\end{aligned}$$

Theorem 4.3. *Given coupled Sylvester matrix equations* (3) *with a unique reflexive solution $X^* \in \mathbb{C}_r^{n\times n}(P)$, the iterative solution $X(k)$ obtained from Algorithm 3 converges to X^* when the inequality* (57) *holds for the parameters μ_1, μ_2, τ and γ.*

Proof. The proof of this theorem follows a similar approach to Theorem 4.1 and is omitted for brevity. □

4.2. Generalized ant-reflexive solution

Using Lemma 3.4, we can derive the generalized anti-reflexive solution of system (3) by solving a set of equations given by

$$\begin{cases} A_1X + X^HB_1 = F_1, \\ A_1PXQ + (PXQ)^HB_1 = -F_1, \\ \vdots \\ A_rX + X^HB_r = -F_r, \\ A_rPXQ + (PXQ)^HB_r = F_r. \end{cases} \tag{82}$$

Following a similar approach as in the previous section, we can obtain equations as follows:

$$X_5(k+1) = X(k) + \mu_1 P^H \begin{bmatrix} M_{a,1} + \tau\Delta_a \\ \vdots \\ M_{a,r} + \tau\Delta_a \end{bmatrix}^H \begin{bmatrix} -F_1 - A_1PX(k)Q - (PX(k)Q)^HB_1 \\ \vdots \\ -F_r - A_rPX(k)Q - (PX(k)Q)^HB_r \end{bmatrix} Q^H,$$

$$X_6(k+1) = X(k) + \mu_2 P^H \begin{bmatrix} G_{a,1} + N_{a,1} + \gamma\Gamma_a \\ \vdots \\ G_{a,r} + N_{a,r} + \gamma\Gamma_a \end{bmatrix}^H \begin{bmatrix} -F_1 - A_1PX(k)Q - (PX(k)Q)^HB_1 \\ \vdots \\ -F_r - A_rPX(k)Q - (PX(k)Q)^HB_r \end{bmatrix} Q^H,$$

$$X_7(k+1) = X(k) + \mu_1 P^H \left[M_{b,1} + \tau\Delta_b,\ \ldots, M_{b,r} + \tau\Delta_b\right] \Big[-F_1 - A_1(PX(k)Q) - (PX(k)Q)^HB_1,$$
$$\ldots\ ,\ -F_r - A_r(PX(k)Q) - (PX(k)Q)^HB_r\Big]^H Q^H,$$

and

$$X_8(k+1) = X(k) + \mu_2 P^H \left[G_{b,1} + N_{b,1} + \gamma\Gamma_b,\ \ldots\ , G_{b,r} + N_{b,r} + \gamma\Gamma_b\right]$$
$$\times \Big[-F_1 - A_1(PX(k)Q) - (PX(k)Q)^HB_1, \ldots\ ,\ -F_r - A_r(PX(k)Q) - (PX(k)Q)^HB_r\Big]^H Q^H.$$

Similar to Algorithm 2, the following iterative algorithm is determined to solve equations (3) over generalized anti-reflexive matrices.

Algorithm 4. Choose an initial matrix $X(1) \in \mathbb{C}_a^{n\times n}(P,Q)$ and real parameters $\mu_1, \mu_2, \tau, \gamma$. For $k = 1, 2, ...,$ compute:

$$\Psi_i(k) = F_i - A_iX(k) - X(k)^HB_i, \quad i = 1, 2, \ldots, r.$$

$$\begin{aligned} X(k+1) = X(k) &+ \frac{\mu_1}{8}\sum_{i=1}^{r}\Big(A_i^H\Psi_i(k) + B_i\Psi_i(k)^H - PA_i^H\Psi_i(k)Q - PB_i\Psi_i(k)^HQ\Big) \\ &+ \frac{\mu_2-\mu_1}{8}\sum_{i=1}^{r}\Big((N_{a,i} - G_{a,i})\Psi_i(k) + (N_{b,i} + G_{b,i})\Psi_i(k)^H - P(N_{a,i} - G_{a,i})\Psi_i(k)Q - P(N_{b,i} + G_{b,i})\Psi_i(k)^HQ\Big) \\ &+ \frac{\mu_1\tau}{8}\sum_{i=1}^{r}\Big(\Delta_a^H\Psi_i(k) + \Delta_b\Psi_i(k)^H - P\Delta_a^H\Psi_i(k)Q - P\Delta_b\Psi_i(k)^HQ\Big) \\ &+ \frac{\mu_2\gamma}{8}\sum_{i=1}^{r}\Big(\Gamma_a^H\Psi_i(k) + \Gamma_b\Psi_i(k)^H - P\Gamma_a^H\Psi_i(k)Q - P\Gamma_b\Psi_i(k)^HQ\Big). \end{aligned} \tag{83}$$

Theorem 4.4. *Given coupled Sylvester matrix equations* (3) *with a unique generalized ant-reflexive solution* $X^* \in \mathbb{C}_a^{n\times n}(P,Q)$*, the iterative solution* $X(k)$ *obtained from Algorithm 4 converges to* X^* *when inequality* (57) *holds for the parameters* μ_1, μ_2, τ *and* γ.

Proof. The proof of this theorem follows a similar approach to Theorem 4.1 and is omitted for brevity. □

Remark 4.5. *If we set $P = Q$ in Algorithm 4, the resulting algorithm provides the ant-reflexive solution of matrix equations* (3)*, and it can be simplified as follows:*

Algorithm 5. Choose an initial matrix $X(1) \in \mathbb{C}_a^{n\times n}(P)$ and real parameters $\mu_1, \mu_2, \tau, \gamma$. For $k = 1, 2, ...,$ compute:

$$\Psi_i(k) = F_i - A_i X(k) - X(k)^H B_i, \quad i = 1, 2, \ldots, r. \tag{84}$$

$$\begin{aligned} X(k+1) = X(k) &+ \frac{\mu_1}{8}\sum_{i=1}^{r}\left(A_i^H\Psi_i(k) + B_i\Psi_i(k)^H - PA_i^H\Psi_i(k)P - PB_i\Psi_i(k)^H P\right) \\ &+ \frac{\mu_2-\mu_1}{8}\sum_{i=1}^{r}\Big((N_{a,i}-G_{a,i})\Psi_i(k) + (N_{b,i}+G_{b,i})\Psi_i(k)^H - P(N_{a,i}-G_{a,i})\Psi_i(k)P - P(N_{b,i}+G_{b,i})\Psi_i(k)^H P\Big) \\ &+ \frac{\mu_1\tau}{8}\sum_{i=1}^{r}\left(\Delta_a^H\Psi_i(k) + \Delta_b\Psi_i(k)^H - P\Delta_a^H\Psi_i(k)P - P\Delta_b\Psi_i(k)^H P\right) \\ &+ \frac{\mu_2\gamma}{8}\sum_{i=1}^{r}\left(\Gamma_a^H\Psi_i(k) + \Gamma_b\Psi_i(k)^H - P\Gamma_a^H\Psi_i(k)P - P\Gamma_b\Psi_i(k)^H P\right). \end{aligned} \tag{85}$$

Theorem 4.6. *Given coupled Sylvester matrix equations* (3) *with a unique ant-reflexive solution $X^* \in \mathbb{C}_a^{n\times n}(P)$, the iterative solution $X(k)$ obtained from Algorithm 5 converges to X^* when the inequality* (57) *holds for the parameters μ_1, μ_2, τ and γ.*

Proof. The proof of this theorem follows a similar approach to Theorem 4.1 and is omitted for brevity. □

5. Numerical reports

In this section, we provide numerical experiments to demonstrate the effectiveness of our proposed algorithms. The initial matrices are set to $X(1) = \mathbf{O}_n$, where $\mathbf{O}_n$ denotes an $n \times n$ zero matrix. We consider $E(k)$ to be a zero matrix if $\|E(k)\| < \epsilon$, where ϵ is a small positive number. Additionally, we measure the relative error using :

$$RES(k) := \sqrt{\frac{\sum_{i=1}^{r}\|A_iX(k) + X(k)^H B_i - F_i\|_2^2}{\sum_{i=1}^{r}\|F_i\|_2^2}}, \quad k = 1, 2, \ldots.$$

In addition, the matrices $G_{a,i}, G_{b,i}, N_{a,i}, N_{b,i}, i = 1, 2, ..., r$ are selected according to the procedure described in (21).

The numerical experiments were conducted using MATLAB (R2015a) software on a system with an Intel (R) Pentium 29 (R) CPU N3700 and 4 GB of RAM.

Example 5.1. *The system of matrix equations we are studying in this section is given by [18]*

$$\begin{cases} A_1X + X^T B_1 = F_1, \\ A_2X + X^T B_2 = F_2, \end{cases}$$

with the following parameters:

$$A_1 = \begin{bmatrix} 3 & 5 & -2 \\ 10 & 2 & 2 \\ -11 & -6 & 18 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 14 & 4 & -1 \\ -6 & 0 & 0 \\ 16 & 4 & 8 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 8 & -6 & 3 \\ 8 & 4 & 6 \\ 4 & 9 & 4 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -1 & 5 & -4 \\ 2 & -5 & -14 \\ -3 & -5 & 8 \end{bmatrix},$$

$$F_1 = \begin{bmatrix} 13 & 24 & 45 \\ 110 & 108 & 139 \\ 120 & 56 & 18 \end{bmatrix} \text{ and } \quad F_2 = \begin{bmatrix} 12 & 61 & 123 \\ -23 & -58 & -70 \\ 39 & 106 & 70 \end{bmatrix}.$$

The solution to the coupled matrix equations that satisfies the generalized reflexive property can be expressed in the following manner:

$$X^* = \begin{bmatrix} 2 & 3 & 6 \\ -1 & 5 & 5 \\ 2 & 6 & 3 \end{bmatrix} \in \mathbb{R}_r^{3\times 3}(P, Q),$$

with

$$P = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \text{ and } \quad Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

We will use Algorithm 2 to solve this problem. In this algorithm, we set $\Delta_a = \Delta_b = \Gamma_a = \Gamma_b = I$.

The convergence curves for the iterative method (77) *with different parameters are shown in Figures 1 and 2. Based on the results presented in these figures, we can determine the optimal parameters as:*

$$\tau = \gamma = 3, \ \mu_1 = 10^{-3}, \ \mu_2 = 8 \times 10^{-3}.$$

Interestingly, we observe that for these parameters, increasing the number of iterations improves the accuracy of solution.

Table 1 displays the solution obtained after each iteration, and the final solution achieved after 130 iterations is given below:

$$X(130) = \begin{bmatrix} 2.000000000000001 & 3.000000000000000 & 5.999999999999999 \\ -1.000000000000002 & 4.999999999999997 & 4.999999999999997 \\ 2.000000000000001 & 5.999999999999999 & 3.000000000000000 \end{bmatrix},$$

such that

$$PX(130)Q = \begin{bmatrix} 2.000000000000001 & 3.000000000000000 & 5.999999999999999 \\ -1.000000000000002 & 4.999999999999997 & 4.999999999999997 \\ 2.000000000000001 & 5.999999999999999 & 3.000000000000000 \end{bmatrix} = X(130),$$

and $RES(130) = 2.8684 \times 10^{-16}$. *The results obtained from our study demonstrate that the algorithms utilized provide efficient and reliable approaches for computing the solutions to linear matrix equations* (3) *with generalized reflexive properties.*

Example 5.2. *Consider the system of matrix equations*

$$\begin{cases} A_1X + X^HB_1 = F_1, \\ A_2X + X^HB_2 = F_2, \end{cases}$$

with the following matrices:

$$A_1 = \begin{bmatrix} -4+2i & -6-4i & -6+2i & -6-5i & 5-i & 1+8i \\ 14+4i & 4+6i & 1+3i & 13-3i & 1+3i & 8-4i \\ -2+3i & -3-2i & -3+6i & 2i & 6 & 4-2i \\ 7 & -3+i & 9+9i & -4-7i & -8-5i & 2+3i \\ -5-2i & 7+4i & -1+8i & -12-5i & 4-5i & 4-i \\ 2+i & -1-6i & 12i & -4+6i & -4+5i & 2+i \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 3+7i & 1+2i & -5+3i & -2+7i & -i & 2+3i \\ 3i & 4+3i & 4-4i & -2+9i & 3-8i & -13+2i \\ 3+i & -6-7i & -3+2i & 4+5i & 3-6i & -5-8i \\ 6 & 5i & -2-4i & -4+i & -5-6i & -6-9i \\ -7+2i & -5+i & -2+5i & 6+3i & 13-2i & 1-i \\ -1+4i & 4-7i & 1+2i & 6+2i & 6-6i & 6+4i \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 7-13i & -2i & -6-4i & -1+4i & 1+9i & 7-3i \\ -2-8i & 3+4i & 3+2i & -2+5i & 4+8i & 4-2i \\ -7+4i & 6+8i & 2+2i & -4-4i & 1-2i & 9-4i \\ 1+4i & 6+6i & -11 & -1+12i & -1+i & 2+3i \\ -4+6i & 7-7i & 7-i & 13+i & 4i & -4-9i \\ -2-4i & 9+i & -3-2i & -2+6i & 3-6i & 2+2i \end{bmatrix},$$

$$B_2 = \begin{bmatrix} -5-2i & 3-3i & -6-i & -2+7i & -3+2i & 7-4i \\ -1-7i & 14-i & 3+8i & 10+2i & -1+i & 6+2i \\ 2-7i & 6+7i & 3-4i & 2-2i & 3-2i & 4+9i \\ -2+5i & 6-i & -3+3i & -2-6i & -4+i & -5-3i \\ -2+i & 3 & 4-i & 2 & 1+i & 7+8i \\ 2+3i & 7-6i & 5+3i & -6+3i & -7+3i & 0 \end{bmatrix},$$

$$F_1 = \begin{bmatrix} 0 & -5+10i & 0 & -1-5i & 0 & 0 \\ -9 & 24+8i & -1 & -1+11i & 4-8i & 11-2i \\ 0 & 1+4i & 0 & 3-2i & 0 & 0 \\ -6-2i & 21+9i & 10+i & 2i & 4+12i & -11i \\ 0 & 3+7i & 0 & 11-i & 0 & 0 \\ 0 & 2+13i & 0 & -5-i & 0 & 0 \end{bmatrix},$$

and

$$F_2 = \begin{bmatrix} 0 & -3+6i & 0 & 1+i & 0 & 0 \\ 4-4i & 4-i & 8-i & 3-4i & -4+i & 4+9i \\ 0 & -8-6i & 0 & -3-13i & 0 & 0 \\ -3-6i & 9-14i & 7+7i & 7+i & 2i & 13+10i \\ 0 & -1+4i & 0 & 8-i & 0 & 0 \\ 0 & 7+6i & 0 & 10-13i & 0 & 0 \end{bmatrix}.$$

The generalized anti-reflexive solution for this problem can be expressed as follows:

$$X^* = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \in \mathbb{R}_a^{6\times 6}(P,Q),$$

with

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \text{ and } Q = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

To solve this problem, we will utilize Algorithms 1, 4, and the gradient iterative algorithm (GI) as described in (56) *(refer to [42]). In the Algorithms 1 and 4, we set $\Delta_a = \Delta_b = \Gamma_a = \Gamma_b = I$. Moreover the optimal parameters for these algorithms have been determined through experimental analysis and are presented below:*

- *For GI method,* $\mu = 8.4 \times 10^{-3}$,
- *For Algorithm 1,* $\tau = 5,\ \gamma = 2.5,\ \mu_1 = \mu_2 = 8 \times 10^{-3}$,
- *For Algorithm 4,* $\tau = 0,\ \gamma = 1,\ \mu_1 = 10^{-3}$ *and* $\mu_2 = 4 \times 10^{-3}$.

After 200 iterations, we obtained the following results for Algorithms 1, 4 and GI method:

- *For GI method: we obtain* $\|X^* - X(200)\|_2 = 3.5715 \times 10^{-4}$.
- *For Algorithms 1: we obtain* $\|X^* - X(200)\|_2 = 9.8080 \times 10^{-6}$.
- *For Algorithms 4: we obtain* $\|X^* - X(200)\|_2 = 7.1612 \times 10^{-14}$ *and*

$$\|X(200) + PX(200)Q\|_2 = 9.0288 \times 10^{-14}.$$

The above results clearly show that Algorithm 4 has been able to approximate the generalized anti-reflexive solution of matrix equations (3) *with high accuracy. Also the convergence curves for the mentioned iterative methods with optimal parameters can be seen in Figure 3. From the obtained results, it is evident that the new algorithms are effective for computing the approximate solution of linear matrix equations* (3).

Example 5.3. *Consider complex matrix equation* $AX + X^H B = F$ *with full matrices*

$$A = \begin{bmatrix} 0.6131 & 0.8473 & 0.2959 & 0.6509 & 0.2356 & 0.6432 & 0.6293 & 0.3993 & 0.1779 & 0.2723 \\ 0.3942 & 0.6478 & 0.5305 & 0.3530 & 0.4784 & 0.6912 & 0.8806 & 0.9087 & 0.8787 & 0.0472 \\ 0.7456 & 0.4458 & 0.2843 & 0.3944 & 0.0438 & 0.4308 & 0.1943 & 0.9715 & 0.3739 & 0.7673 \\ 0.1140 & 0.4232 & 0.5516 & 0.8707 & 0.3489 & 0.4579 & 0.1387 & 0.4346 & 0.4680 & 0.9173 \\ 0.2353 & 0.4334 & 0.7044 & 0.9824 & 0.2325 & 0.7228 & 0.9972 & 0.3735 & 0.6448 & 0.0053 \\ 0.9717 & 0.6235 & 0.2554 & 0.3713 & 0.8292 & 0.3848 & 0.7507 & 0.4350 & 0.4229 & 0.3609 \\ 0.2326 & 0.1909 & 0.4913 & 0.7410 & 0.7807 & 0.4794 & 0.0374 & 0.1989 & 0.3645 & 0.2590 \\ 0.6295 & 0.8275 & 0.6904 & 0.8247 & 0.5668 & 0.1442 & 0.1316 & 0.7997 & 0.5318 & 0.7539 \\ 0.8265 & 0.7521 & 0.6249 & 0.0962 & 0.6033 & 0.3725 & 0.7198 & 0.8524 & 0.1920 & 0.1377 \\ 0.3634 & 0.9164 & 0.1316 & 0.3789 & 0.5338 & 0.0589 & 0.6721 & 0.5005 & 0.5004 & 0.2085 \end{bmatrix}$$

$$+i \begin{bmatrix} 0.3568 & 0.6470 & 0.2634 & 0.9521 & 0.4893 & 0.8238 & 0.7189 & 0.7653 & 0.8854 & 0.1352 \\ 0.4413 & 0.8922 & 0.8522 & 0.5248 & 0.9394 & 0.3946 & 0.6562 & 0.6699 & 0.7116 & 0.8109 \\ 0.1926 & 0.3257 & 0.6759 & 0.5855 & 0.3990 & 0.3146 & 0.7015 & 0.8561 & 0.4686 & 0.3071 \\ 0.5048 & 0.3553 & 0.6082 & 0.9870 & 0.7646 & 0.3654 & 0.0349 & 0.2680 & 0.0659 & 0.8459 \\ 0.4123 & 0.3886 & 0.7170 & 0.9371 & 0.5987 & 0.1208 & 0.0902 & 0.2423 & 0.0704 & 0.4265 \\ 0.8234 & 0.9436 & 0.9265 & 0.3459 & 0.8117 & 0.8025 & 0.2854 & 0.5299 & 0.7353 & 0.2734 \\ 0.6676 & 0.9677 & 0.4373 & 0.1711 & 0.5403 & 0.3118 & 0.5744 & 0.2756 & 0.6845 & 0.1982 \\ 0.5859 & 0.2253 & 0.1143 & 0.3462 & 0.6682 & 0.0367 & 0.9581 & 0.7315 & 0.5405 & 0.5196 \\ 0.4153 & 0.7224 & 0.3837 & 0.1402 & 0.2151 & 0.2155 & 0.8935 & 0.7004 & 0.0277 & 0.9618 \\ 0.8056 & 0.7515 & 0.5754 & 0.7516 & 0.4350 & 0.6653 & 0.2497 & 0.7074 & 0.0748 & 0.8542 \end{bmatrix},$$

$$B = \begin{bmatrix} 20.5819 & 0.6225 & 0.0378 & 0.9454 & 0.1020 & 0.8390 & 0.4574 & 0.8020 & 0.1840 & 0.8214 \\ 0.4078 & 20.9269 & 0.5325 & 0.0443 & 0.1267 & 0.5751 & 0.9508 & 0.8459 & 0.6778 & 0.2392 \\ 0.6676 & 0.2811 & 21.0757 & 0.6038 & 0.1613 & 0.9635 & 0.9825 & 0.3289 & 0.7972 & 0.7159 \\ 0.0419 & 0.7268 & 0.4211 & 20.1158 & 0.9749 & 0.6093 & 0.9241 & 0.2335 & 0.3658 & 0.0973 \\ 0.1354 & 0.4253 & 0.9918 & 0.7724 & 20.4130 & 0.9329 & 0.0970 & 0.8366 & 0.9296 & 0.6283 \\ 0.0673 & 0.7526 & 0.6673 & 0.3517 & 0.0587 & 21.2604 & 0.1445 & 0.9229 & 0.4875 & 0.2865 \\ 0.7068 & 0.2701 & 0.8899 & 0.5803 & 0.5727 & 0.0655 & 21.3558 & 0.6689 & 0.2810 & 0.6470 \\ 0.9337 & 0.5670 & 0.3570 & 0.1843 & 0.5094 & 0.4608 & 0.6794 & 21.2115 & 0.8227 & 0.2556 \\ 0.6415 & 0.8565 & 0.0977 & 0.2513 & 0.2559 & 0.4793 & 0.5453 & 0.9653 & 21.0154 & 0.6173 \\ 0.0873 & 0.0900 & 0.7263 & 0.3664 & 0.1212 & 0.6901 & 0.3937 & 0.1833 & 0.4632 & 21.1009 \end{bmatrix}$$

$$+i\begin{bmatrix} 0.8184 & 0.6171 & 0.5050 & 0.6428 & 0.7586 & 0.3889 & 0.8986 & 0.0243 & 0.2773 & 0.9953 \\ 0.9301 & 0.6878 & 0.8816 & 0.3380 & 0.0621 & 0.0128 & 0.9476 & 0.9767 & 0.6544 & 0.3477 \\ 0.0834 & 0.9042 & 0.9108 & 0.9301 & 0.2808 & 0.1352 & 0.3014 & 0.1753 & 0.5523 & 0.5704 \\ 0.5812 & 0.0100 & 0.1737 & 0.7091 & 0.5614 & 0.8175 & 0.8519 & 0.2409 & 0.3022 & 0.3658 \\ 0.6080 & 0.9309 & 0.2582 & 0.6126 & 0.8900 & 0.7595 & 0.9293 & 0.8054 & 0.0470 & 0.9245 \\ 0.6197 & 0.4284 & 0.6775 & 0.3930 & 0.8893 & 0.5638 & 0.2082 & 0.0226 & 0.6778 & 0.7640 \\ 0.9067 & 0.0377 & 0.1329 & 0.7449 & 0.5160 & 0.4195 & 0.7897 & 0.3111 & 0.3210 & 0.4391 \\ 0.9357 & 0.5332 & 0.6655 & 0.7413 & 0.5711 & 0.3828 & 0.9329 & 0.6841 & 0.9267 & 0.7210 \\ 0.1299 & 0.3483 & 0.9487 & 0.1122 & 0.3621 & 0.5298 & 0.8866 & 0.5398 & 0.0785 & 0.2999 \\ 0.8795 & 0.7002 & 0.8913 & 0.2815 & 0.2540 & 0.7200 & 0.6258 & 0.1757 & 0.4425 & 0.6240 \end{bmatrix},$$

and

$$F=\begin{bmatrix} 0.5221 & 0.2033 & 0.5592 & 0.2201 & 0.6078 & 0.6520 & 0.8380 & 0.1276 & 0.2296 & 0.3966 \\ 0.8317 & 0.6864 & 0.7144 & 0.6059 & 0.6259 & 0.4903 & 0.9974 & 0.1558 & 0.7329 & 0.1167 \\ 0.5317 & 0.6293 & 0.0189 & 0.8091 & 0.6988 & 0.7587 & 0.0179 & 0.6091 & 0.8682 & 0.8325 \\ 0.7645 & 0.4162 & 0.5598 & 0.7006 & 0.7275 & 0.4748 & 0.4374 & 0.8568 & 0.9860 & 0.0245 \\ 0.2291 & 0.4993 & 0.9951 & 0.6629 & 0.0754 & 0.5650 & 0.6868 & 0.7530 & 0.7761 & 0.1673 \\ 0.0572 & 0.1156 & 0.4693 & 0.2150 & 0.3528 & 0.0734 & 0.5885 & 0.8216 & 0.0460 & 0.4064 \\ 0.1870 & 0.8088 & 0.7003 & 0.5908 & 0.2469 & 0.9095 & 0.1579 & 0.3178 & 0.3449 & 0.7072 \\ 0.5438 & 0.4490 & 0.8813 & 0.8698 & 0.1582 & 0.0378 & 0.3157 & 0.2062 & 0.8334 & 0.7675 \\ 0.9127 & 0.5398 & 0.6453 & 0.4095 & 0.6658 & 0.6173 & 0.6110 & 0.2813 & 0.6373 & 0.2179 \\ 0.4281 & 0.9437 & 0.0904 & 0.3394 & 0.2769 & 0.5433 & 0.1073 & 0.5529 & 0.7768 & 0.3949 \end{bmatrix}.$$

By using Lemma 3.1, the solution of the above system of matrix equations can be given:

$$\begin{bmatrix} \mathrm{Col}[X] \\ \mathrm{Col}[\overline{X}] \end{bmatrix} = \left(\begin{bmatrix} I\otimes A & \left(B^T\otimes I\right)P_{10} \\ \left(B^H\otimes I\right)P_{10} & I\otimes \overline{A} \end{bmatrix}^H \begin{bmatrix} I\otimes A & \left(B^T\otimes I\right)P_{10} \\ \left(B^H\otimes I\right)P_{10} & I\otimes \overline{A} \end{bmatrix} \right)^{-1}$$

$$\times \begin{bmatrix} I\otimes A & \left(B^T\otimes I\right)P_{10} \\ \left(B^H\otimes I\right)P_{10} & I\otimes \overline{A} \end{bmatrix}^H \begin{bmatrix} \mathrm{Col}[F] \\ \mathrm{Col}[\overline{F}] \end{bmatrix}.$$

Hence

$$X=\begin{bmatrix} 0.0197 & 0.0327 & 0.0203 & 0.0300 & 0.0022 & -0.0041 & 0.0037 & 0.0202 & 0.0369 & 0.0155 \\ 0.0027 & 0.0233 & 0.0218 & 0.0111 & 0.0139 & -0.0007 & 0.0321 & 0.0134 & 0.0164 & 0.0379 \\ 0.0185 & 0.0251 & -0.0090 & 0.0175 & 0.0400 & 0.0151 & 0.0250 & 0.0334 & 0.0229 & -0.0025 \\ 0.0027 & 0.0190 & 0.0322 & 0.0261 & 0.0243 & 0.0037 & 0.0225 & 0.0343 & 0.0101 & 0.0105 \\ 0.0245 & 0.0222 & 0.0274 & 0.0281 & -0.0058 & 0.0124 & 0.0070 & -0.0000 & 0.0261 & 0.0087 \\ 0.0260 & 0.0156 & 0.0280 & 0.0147 & 0.0203 & -0.0003 & 0.0373 & -0.0060 & 0.0212 & 0.0205 \\ 0.0322 & 0.0372 & -0.0090 & 0.0094 & 0.0215 & 0.0215 & -0.0018 & 0.0034 & 0.0200 & -0.0027 \\ -0.0011 & -0.0022 & 0.0196 & 0.0302 & 0.0274 & 0.0350 & 0.0077 & 0.0008 & 0.0045 & 0.0190 \\ 0.0041 & 0.0265 & 0.0337 & 0.0394 & 0.0284 & -0.0053 & 0.0090 & 0.0323 & 0.0216 & 0.0307 \\ 0.0114 & -0.0042 & 0.0320 & -0.0077 & 0.0005 & 0.0141 & 0.0283 & 0.0283 & 0.0015 & 0.0125 \end{bmatrix}$$

$$+i\begin{bmatrix} 0.0088 & 0.0107 & 0.0098 & 0.0090 & 0.0076 & 0.0087 & 0.0088 & 0.0086 & 0.0093 & 0.0089 \\ 0.0100 & 0.0114 & 0.0095 & 0.0091 & 0.0083 & 0.0099 & 0.0103 & 0.0097 & 0.0094 & 0.0090 \\ 0.0096 & 0.0112 & 0.0095 & 0.0106 & 0.0098 & 0.0088 & 0.0099 & 0.0101 & 0.0088 & 0.0111 \\ 0.0114 & 0.0126 & 0.0092 & 0.0101 & 0.0100 & 0.0111 & 0.0093 & 0.0102 & 0.0089 & 0.0083 \\ 0.0107 & 0.0115 & 0.0108 & 0.0095 & 0.0080 & 0.0087 & 0.0080 & 0.0076 & 0.0095 & 0.0083 \\ 0.0051 & 0.0065 & 0.0077 & 0.0059 & 0.0048 & 0.0040 & 0.0043 & 0.0066 & 0.0065 & 0.0050 \\ 0.0086 & 0.0120 & 0.0104 & 0.0116 & 0.0089 & 0.0086 & 0.0079 & 0.0081 & 0.0098 & 0.0107 \\ 0.0056 & 0.0083 & 0.0068 & 0.0080 & 0.0069 & 0.0066 & 0.0057 & 0.0057 & 0.0059 & 0.0076 \\ 0.0074 & 0.0086 & 0.0071 & 0.0071 & 0.0077 & 0.0091 & 0.0086 & 0.0071 & 0.0065 & 0.0078 \\ 0.0095 & 0.0105 & 0.0086 & 0.0088 & 0.0069 & 0.0084 & 0.0085 & 0.0075 & 0.0091 & 0.0082 \end{bmatrix}.$$

Algorithm 1 can be applied to solve this problem by setting the parameters Δ_a, Δ_b, Γ_a, and Γ_b to be equal to the identity matrix I. The residual RES(k) as a function of the iteration number is shown in Figure 4 for $\tau = \gamma = 8$. Additionally, Figure 5 illustrates the variation of the residual RES(k) with respect to the iteration number for specific values of $\mu_1 = 10^{-3}$ and $\mu_2 = 4 \times 10^{-3}$. Notably, Figure 5 (bottom) demonstrates that when $\gamma = 8$, the parameter τ has no discernible effect on the convergence speed. Hence, to minimize computational requirements, it is advisable to select $\tau = 0$.

Therefore based on the results depicted in the figures, we have determined the optimal parameters as follows:

$$\tau = 0,\ \gamma = 8,\ \mu_1 = 10^{-3},\ \mu_2 = 4 \times 10^{-3}.$$

Notably, it is intriguing to observe that as the number of iterations increases, the accuracy improves.

It can be seen that for method in [7]; the optimum parameters are

$$\tau = \gamma = 8,\ \mu_1 = \mu_2 = 7.1 \times 10^{-4}.$$

The residual RES(k) as a function of the iteration number is shown in Figure 6 for method in [7] and method (55) by optimum parameters $\tau = 0,\ \gamma = 8,\ \mu_1 = 10^{-3},\ \mu_2 = 4 \times 10^{-3}$. This figure shows that method (55) is much faster than method in [7]. The obtained numerical results demonstrate the effectiveness and reliability of new algorithms in computing the approximate solution of linear matrix equations (3). These algorithms offer efficient and dependable methods for obtaining these solutions.

Example 5.4. *Let us consider the coefficient matrices with dimensions of 100×100 as given below:*

$$A_1 = rand(100), \quad A_2 = rand(100),$$

$$B_1 = diag(40 + diag(rand(100))) + rand(100), \quad B_2 = diag(40 + diag(rand(100))) + rand(100),$$

where rand(.) and diag(.) are functions in MATLAB. Also consider two cases for right-hand side matrices F_1 and F_2.

Case I: *Let*

$$F_1 = A_1X_1 + X_1^TB_1, \quad F_2 = A_2X_1 + X_1^TB_2,$$

where X_1 represents a 100×100 matrix, and all its elements are set to 1.

Case II: *Let*

$$F_1 = A_1X_2 + X_2^TB_1, \quad F_2 = A_2X_2 + X_2^TB_2,$$

where $X_2 = (x_{i,j})$ with

$$x_{i,j} = \frac{1}{\sin(x_i) + \cos(y_j) + 2.1}, \quad i, j = 1, 2, ..., 100, \tag{86}$$

and $x_i = -6 + \frac{4(i-1)}{33}$ and $y_j = -6 + \frac{4(j-1)}{33}$, $\quad i, j = 1, 2, ..., 100$.

To solve this problem, Algorithm 1 is applied with the following settings:

$$\Delta_a = \Delta_b = \Gamma_a = \Gamma_b = I.$$

The optimal parameters for this method have been determined through experimental analysis. The convergence curve of the iterative method with various parameters is depicted in Figure 7, which helps identify the optimal parameters as follows:

$$\tau = 0,\ \gamma = 30,\ \mu_1 = 0,\ \mu_2 = 2 \times 10^{-4},$$

for iterative method (55) and

$$\tau = \gamma = 30,\ \mu_1 = \mu_2 = 1 \times 10^{-4},$$

for method in [7].

Table 2 provides the values of the error RES(.) and the corresponding execution times for different numbers of iterations 50, 100, 150 and 200. It can be observed that the error RES(.) decreases with the increase in iterations, although the running time also increases. This table shows that method (55) *is much faster than method in [7].*

Figure 8 displays the exact solution as well as several approximations of the exact solution X_2 *achieved through different iterations of Algorithm 1 with parameters* $\tau = 0$, $\gamma = 30$, $\mu_1 = 0$, *and* $\mu_2 = 2 \times 10^{-4}$. *By examining the figure, it becomes evident that increasing the number of iterations in this algorithm results in a closer approximation to the exact solution. Notably, for k=50 iterations, the approximate solution closely matches the exact solution. Our findings demonstrate that the algorithms utilized in this study provide efficient and dependable approaches for computing the approximate solutions of linear matrix equations* (3).

5.1. *An application to the palindromic eigenvalue problem*

The content of this section is taken from reference [26]. Interested readers can see this reference for more details. Consider the palindromic eigenvalue problem expressed as

$$Ax = \lambda A^H x,$$

where $A \in \mathbb{C}^{n\times n}$. In [26], a method is presented to address this problem by first reducing matrix A to an anti-Hessenberg-triangular form. Subsequently, an anti-Hessenberg-triangular matrix, which may not be in its unreduced state, can be deflated to obtain unreduced eigenvalue problems of smaller dimensions. In [26], it is demonstrated that any matrix in anti-Hessenberg form can be transformed to an anti-Hessenberg-triangular form using a unitary transformation.
This transformation can be easily achieved when $a_{n-p,p} = 0$ for some $p = 1, ..., n - \left\lfloor \frac{n-1}{2} \right\rfloor - 1$. In such cases, matrix A can be partitioned as:

$$A = \begin{array}{c} \\ p \\ n-2p \\ p \end{array} \begin{array}{c} \begin{array}{ccc} p & n-2p & p \end{array} \\ \begin{bmatrix} & & A_{13} \\ & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \end{array}.$$

Consequently, the eigenvalues of (A, A^H) can be obtained from the generalized eigenvalue problem (A_{31}, A_{13}^H) and the palindromic eigenvalue problem (A_{22}, A_{22}^H). Let us consider the transformation of matrix A as [26]:

$$A = \begin{array}{c} \\ p \\ m \\ p \end{array} \begin{array}{c} \begin{array}{ccc} p & m & p \end{array} \\ \begin{bmatrix} & & A_{13} \\ & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \end{array}.$$

It is important to note that $m = 1$ if n is even and $m = 0$ otherwise. However, the discussion below applies to general cases of m. Exchanging the bulges consists of finding a unitary matrix Q such that [26]:

$$\tilde{A} = Q^H A Q = \begin{array}{c} \\ p \\ m \\ p \end{array} \begin{array}{c} \begin{array}{ccc} p & m & p \end{array} \\ \begin{bmatrix} & & \tilde{A}_{13} \\ & \tilde{A}_{22} & \tilde{A}_{23} \\ \tilde{A}_{31} & \tilde{A}_{32} & \tilde{A}_{33} \end{bmatrix} \end{array},$$

where $\Lambda(\tilde{A}_{31}, \tilde{A}_{13}^H) = \Lambda(A_{13}, A_{31}^H)$. Note that if Y, Z satisfy the conditions [26]:

$$\begin{cases} A_{31}Y + Z^H A_{22} = -A_{32}, \\ A_{13}^H Y + Z^H A_{22}^H = -A_{23}^H, \end{cases} \tag{87}$$

and $X \in \mathbb{C}^{p\times p}$ solves the equation:

$$A_{31}X + X^H A_{13} = F, \tag{88}$$

where

$$F = -\left(A_{33} + A_{32}Z + Z^H A_{23} + Z^H A_{22} Z\right),$$

then the following transformation can be employed to get[26]:

$$\begin{bmatrix} X^H & Z^H & I \\ Y^H & I & \\ I & & \end{bmatrix} \begin{bmatrix} & & A_{13} \\ & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} X & Y & I \\ Z & I & \\ I & & \end{bmatrix} = \begin{bmatrix} & & A_{31} \\ & A_{22} & \\ A_{13} & & \end{bmatrix}.$$

To achieve a unitary transformation, let us consider the QR factorization [26]:

$$\begin{bmatrix} X & Y & I \\ Z & I & \\ I & & \end{bmatrix} = Q \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ & R_{22} & R_{23} \\ & & R_{33} \end{bmatrix},$$

where $R_{ii}, i = 1, 2, 3$ are non-singular since the left-hand matrix is. Thus,

$$\tilde{A} = Q^H \begin{bmatrix} & & A_{13} \\ & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} Q = \begin{bmatrix} & & R_{11}^{-H} A_{31} R_{33}^{-1} \\ & R_{22}^{-H} A_{22} R_{22}^{-1} & \tilde{A}_{23} \\ R_{33}^{-H} A_{13} R_{11}^{-1} & \tilde{A}_{32} & \tilde{A}_{33} \end{bmatrix},$$

which accomplishes the desired exchange. Equations (87) and (88) correspond to the linear matrix equations that arise in the context of the palindromic eigenvalue problem discussed here. These equations demonstrate the practical application and relevance of the issues explored and analyzed in this paper. For a comprehensive understanding and further information, refer to [26].

6. Conclusions

In this paper, we considered the problem of computing the generalized reflexive and anti-reflexive solutions of a coupled Sylvester-conjugate transpose matrix equations by introducing new splittings of the coefficient matrices and utilizing the hierarchical identification principle. We presented some iterative algorithms, namely Algorithms 1, 2, 3, 4, and 5, to solve this problem. Convergence analysis was performed to show that the algorithms converge to the desired solutions under certain conditions. We also provided numerical examples to demonstrate the effectiveness of the proposed algorithms. Our results showed that new algorithms offer efficient and reliable methods for computing the generalized reflexive and anti-reflexive solutions of linear matrix equations (3), which have various applications in fields such as engineering and science.

Conflicts of interest

The authors have no conflict of interest to declare.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments

The authors wish to thank anonymous reviewers for careful reading and valuable comments and suggestions which improved the quality of this paper.

References

[1] R. Ali, K. Pan, The new iteration methods for solving absolute value equations, Appl Math., 68 (2023) 109-122.
[2] S. Bittanti, P. Colaneri, Analysis of discrete-time linear periodic systems, in: C. T. Leondes (Ed.), Digital Control and Signal Processing Systems and Techniques, Vol. 78 of Control and Dynamic Systems, Academic Press, 1996.
[3] Z.-Z. Bai, On Hermitian and skew-Hermitian splitting iteration methods for continuous Sylvester equations, J. Comput. Math., 29 (2011) 185–198.
[4] P. Chansangiam, Closed forms of general solutions for rectangular systems of coupled generalized Sylvester matrix differential equations, Communications in Mathematics and Applications, 11 (2020) 311–324.
[5] H.C. Chen, Generalized reflexive matrices: special properties and applications, SIAM J. Matrix Anal. Appl., 19 (1998) 140–153.
[6] O. L. V. Costa, M. D. Fragoso, Stability results for discrete-time linear systems with Markovian jumping parameters, J. Math. Anal. Appl., 179 (1993) 154–178.
[7] M. Dehghan, M. Hajarian, Construction of an iterative method for solving generalized coupled Sylvester matrix equations, Tranr. Inst. Mear. Control, 35 (2013) 961–970.
[8] M. Dehghan, M. Hajarian, Solving the system of generalized Sylvester matrix equations over the generalized centro-symmetric matrices, J. Vib. Control, 20 (2014) 838–846.
[9] M. Dehghan, M. Hajarian, An iterative method for solving the generalized coupled Sylvester matrix equations over generalized bisymmetric matrices, Appl. Math. Modell., 34 (2010) 639–654.
[10] M. Dehghan, M. Hajarian, The general coupled matrix equations over generalized bisymmetric matrices, Linear Algebra Appl., 432 (2010) 1531–1552.
[11] M. Dehghan, R. Mohammadi-Arani, Generalized product-type methods based on bi-conjugate gradient (GPBiCG) for solving shifted linear systems, Comput. Appl. Math., 36 (2017) 1591–1606.
[12] M. Dehghan, A. Shirilord, Solving complex Sylvester matrix equation by accelerated double- step scale splitting (ADSS) method, Eng. Comput., 37 (2021) 489–508.
[13] M. Dehghan, A. Shirilord, A generalized modified Hermitian and skew-Hermitian splitting (GMHSS) method for solving complex Sylvester matrix equation, Appl. Math. Comput., 348 (2019) 632–651.
[14] M. Dehghan, G. Karamali, A. Shirilord, An iterative scheme for a class of generalized Sylvester matrix equations, AUT J. Math. Comput., 5 (2024) 195–215.
[15] F. Ding, P.-X. Liu, J. Ding, Iterative solutions of the generalized Sylvester matrix equation by using the hierarchical identification principle, Appl. Math. Comput., 197 (2008) 41–50.
[16] X.-B. Feng, K. A. Loparo, Y.-D. Ji, H. J. Chizeck, Stochastic stability properties of jump linear systems, IEEE Trans. Autom. Control, 37 (1992) 38–53.
[17] K. Glover, D.J.N. Limebeer, J.C. Doyl, E.M. Kasenally, M.G. Safonov, A characterisation of all solutions to the four block general distance problem, SIAM J Control Optim., 29 (1991) 283–324.
[18] M. Hajarian, Convergence of ADGI methods for solving systems of linear matrix equations, Eng. Comput., 31 (2014) 681–690.
[19] M. Hajarian, Matrix iterative methods for solving the Sylvester-transpose and periodic Sylvester matrix equations, J. Franklin Inst., 350 (2013) 3328–3341.
[20] M. Hajarian, Computing symmetric solutions of general Sylvester matrix equations via Lanczos version of biconjugate residual algorithm, Comput. Math. Appl., 76 (2018) 686–700.
[21] Z.H. He, O.M. Agudelo, Q.W. Wang, B. De Moor, Two-sided coupled generalized Sylvester matrix equations solving using a simultaneous decomposition for fifteen matrices, Linear Algebra Appl., 496 (2016) 549–593.
[22] K. Jbilou, A.J. Riquet, Projection methods for large Lyapunov matrix equations, Linear Algebra Appl., 415 (2006) 344–358.
[23] B. Kágström, L. Westin, Generalized Schur methods with condition estimators for solving the generalized Sylvester equation, IEEE Trans. Autom. Contr., 34 (1989) 745–751.
[24] Y.-F. Ke, C.-F. Ma, Alternating direction method for generalized Sylvester matrix equation $AXB+CYD = E$, Appl. Math. Comput., 260 (2015) 106–125.
[25] Y.F. Ke, C.F. Ma, The alternating direction methods for solving the Sylvester-type matrix equation $AXB + CX^TD = E$, J. Comput. Math., 35 (2017) 620–641.
[26] D. Kressner, C. Schröder, D. S. Watkins, Implicit QR algorithms for palindromic and even eigenvalue problems, Numerical Algorithms, 51 (2009) 209–238.
[27] H.-Y. Li, Z.-S. Gao, D. Zhao, Least squares solutions of the matrix equation $AXB + CYD = E$ with the least norm for symmetric arrowhead matrices, Appl. Math. Comput. 226 (2014) 719–724.
[28] L.-L. Lv, L. Zhang, On the periodic Sylvester equations and their applications in periodic Luenberger observers design, J. Franklin Inst., 353 (2016) 1005–1018.
[29] Z.Y. Li, Y. Wang, B. Zhou, G.R. Duan, Least squares solution with the minimum-norm to general matrix equations via iteration, Appl. Math. Comput., 215 (2010) 3547–3562.
[30] M. Mariton, Almost sure and moments stability of jump linear systems, Syst. Control Lett., 11 (1988) 393–397.
[31] B.C. Moore, Principal component analysis in linear systems: controllability, observability and model reduction, IEEE Trans. Automat. Contr. 26 (1981) 17–31.
[32] Z.-H. Peng, X.-Y. Hu, L. Zhang, An efficient algorithm for the least-squares reflexive solution of the matrix equation $A_1XB_1 = C_1$, $A_2XB_2 = C_2$, Appl. Math. Comput. 181 (2006) 988–999.
[33] Z.-Y. Peng, Y.-X. Peng, An efficient iterative method for solving the matrix equation $AXB + CYD = E$, Numer. Linear Algebra Appl., 13 (2006) 473–485.
[34] X.-P. Sheng, G.-L. Chen, An iterative method for the symmetric and skew symmetric solutions of a linear matrix equation $AXB + CYD = E$, J. Comput. Appl. Math., 233 (2010) 3030–3040.

[35] V. Simoncini and Y. Hao, Analysis of the truncated conjugate gradient method for linear matrix equations, SIAM J. Matrix Anal., 44 (2023) 359–381.

[36] W. Wang, G. Qu, C. Song, Cyclic gradient based iterative algorithm for a class of generalized coupled sylvester-conjugate matrix equations, J. Franklin Inst., 360 (2023) 7206–7229.

[37] Q.W. Wang, A. Rehman, Z.H. He, Y.Zhang, Constraint generalized Sylvester matrix equations, Automatica, 69 (2016) 60–64.

[38] Y.-N. Wu, M.-L. Zeng, On ADMM-Based Methods for solving the nearness symmetric solution of the system of matrix equations $A_1XB_1 = C_1$ and $A_2XB_2 = C_2$, J. Appl. Anal. Comput., 11 (2021) 227–241.

[39] A.G. Wu, G. Feng, G.R. Duan, W.J. Wu Closed-form solutions to Sylvester-conjugate matrix equations, Comput. Math. Appl., 60 (2010) 95–111.

[40] A.G. Wu, G. Feng, J.Q. Hu, G.R. Duan, Closed-form solutions to the nonhomogeneous Yakubovich-conjugate matrix equation, Appl. Math. Comput., 214 (2009) 442–450.

[41] A.G. Wu, G.R. Duan, Solution to the generalised Sylvester matrix equation $AV + BW = EVF$, IET Control Theor. Appl., 1 (2007) 402–408.

[42] A.G. Wu, L. Lv, Guang-Ren Duan, Iterative algorithms for solving a class of complex conjugate and transpose matrix equations, Appl. Math. Comput., 217 (2011) 8343–8353.

[43] A.-G. Wu, G. Feng, G.-R. Duan, W.-Q. Liu, Iterative solutions to the Kalman–Yakubovich-conjugate matrix equation, Appl. Math. Comput. 217 (2011) 4427–4438.

[44] L. Xie, J. Ding, F. Ding, Gradient based iterative solutions for general linear matrix equations, Comput. Math. Appl., 58 (2009) 1441–1448.

[45] Y.-J. Xie, N. Huang, C.-F. Ma, Iterative method to solve the generalized coupled Sylvester-conjugate transpose linear matrix equations over reflexive or anti-reflexive matrix, Comput. Math. Appl., 67 (2014) 2071–2084.

[46] B. Zhou, J. Lam, G.-R. Duan, Convergence of gradient-based iterative solution of coupled Markovian jump Lyapunov equations, Comput. Math. Appl.,56 (2008) 3070–3078.

Table 1: The numerical solution versus iterations number for Example 5.1.

Iteration$(= k)$	x_{11}	x_{12}	x_{13}	x_{21}	x_{22}	x_{23}	x_{31}	x_{32}	x_{33}
1	5.9524	5.5750	7.9749	-0.8751	4.4313	4.4313	5.9524	7.9749	5.5750
2	0.3577	0.8639	4.5828	-1.9008	3.1864	3.1864	0.3577	4.5828	0.8639
3	3.7693	4.6230	6.7616	-1.0354	4.5081	4.5081	3.7693	6.7616	4.6230
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
38	2.0000	3.0000	6.0000	-1.0000	4.9999	4.9999	2.0000	6.0000	3.0000
39	2.0000	3.0000	6.0000	-1.0000	4.9999	4.9999	2.0000	6.0000	3.0000
40	2.0000	3.0000	6.0000	-1.0000	5.0000	5.0000	2.0000	6.0000	3.0000
Exact solution	2	3	6	-1	5	5	2	6	3

Table 2: The error $RES(k)$ and running time (in seconds) versus iterations number for Example 5.4.

	k(*Iteration*)	50	100	150	200
$\tau = \gamma = 30,\ \mu_1 = \mu_2 = 1 \times 10^{-4}$	$RES(k)$	1.2070×10^{-4}	1.7334×10^{-6}	5.3809×10^{-8}	1.3606×10^{-9}
	CPU Time (s)	1.340353	2.499115	3.783740	5.000741
$\tau = 0,\ \gamma = 30,\ \mu_1 = 0,\ \mu_2 = 2 \times 10^{-4}$	$RES(k)$	7.4864×10^{-7}	1.4491×10^{-10}	2.5430×10^{-14}	6.4201×10^{-17}
	CPU Time (s)	1.082153	1.100686	2.139891	3.226895

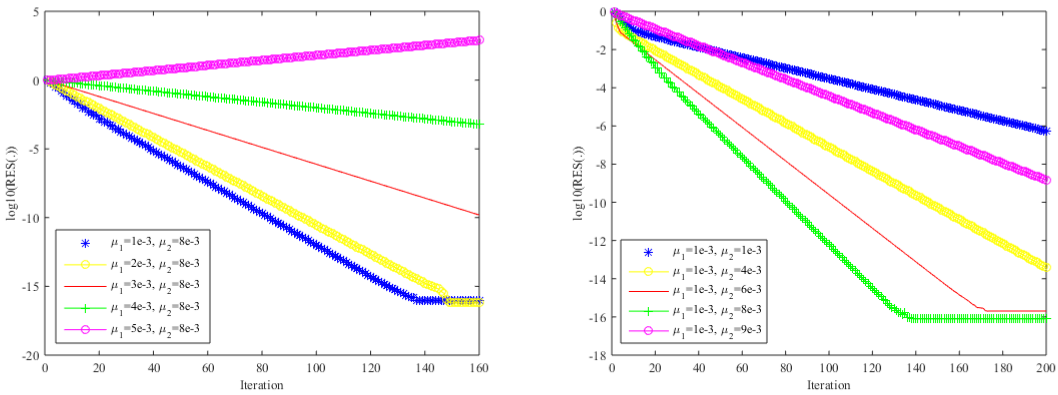

Figure 1: Example 5.1; The convergence curves for iterative method (77) by $\tau = \gamma = 3$.

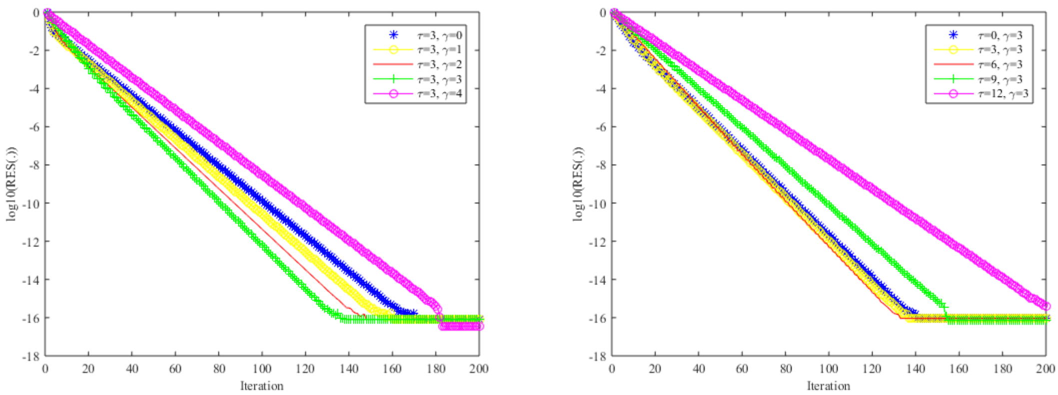

Figure 2: Example 5.1; The convergence curves for iterative method (77) by $\mu_1 = 10^{-3},\ \mu_2 = 8 \times 10^{-3}$.

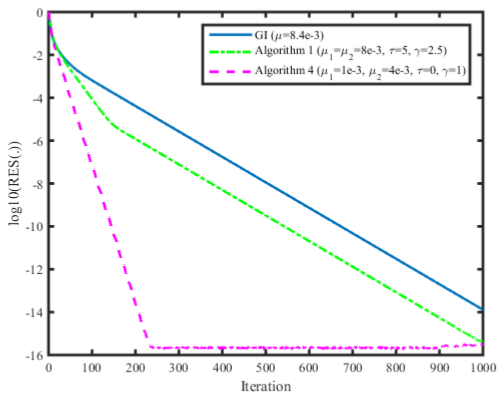

Figure 3: Example 5.2; The convergence curves for Algorithms 1, 4 and GI method with optimal parameters.

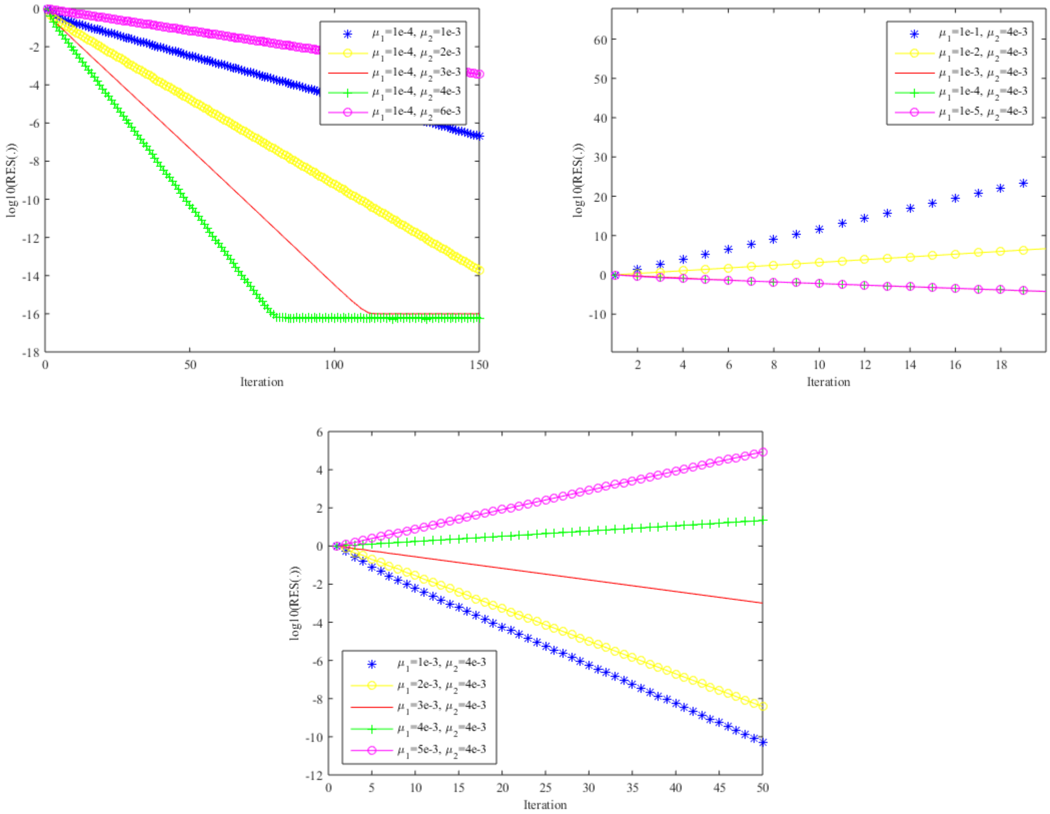

Figure 4: Example 5.3; The convergence curves for iterative method (55) by $\tau = \gamma = 8$.

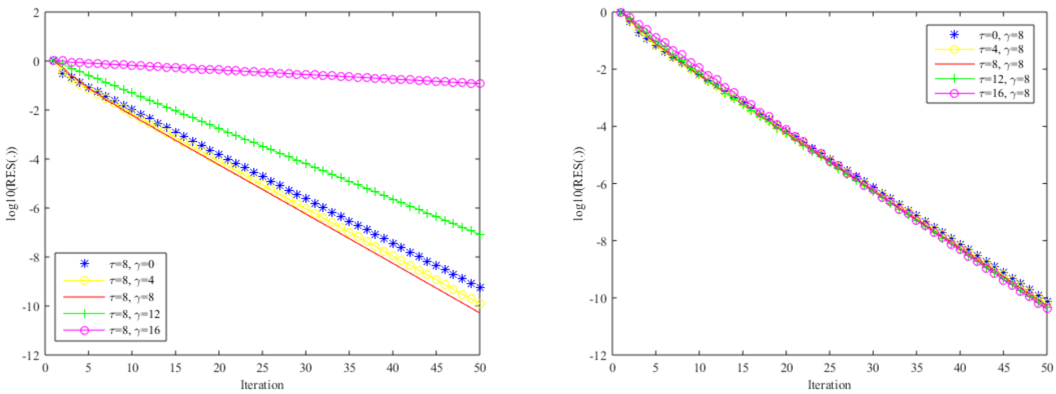

Figure 5: Example 5.3; The convergence curves for iterative method (55) by $\mu_1 = 10^{-3}, \ \mu_2 = 4 \times 10^{-3}$.

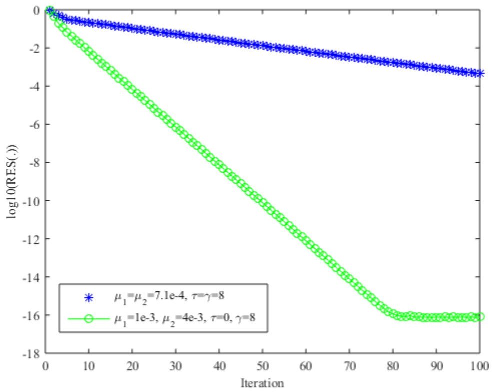

Figure 6: Example 5.3; The convergence curves for iterative method (55) (when $\tau = 0,\ \gamma = 8,\ \mu_1 = 10^{-3},\ \mu_2 = 4 \times 10^{-3}$) and method in [7] (when $\tau = \gamma = 8,\ \mu_1 = \mu_2 = 7.1 \times 10^{-4}$).

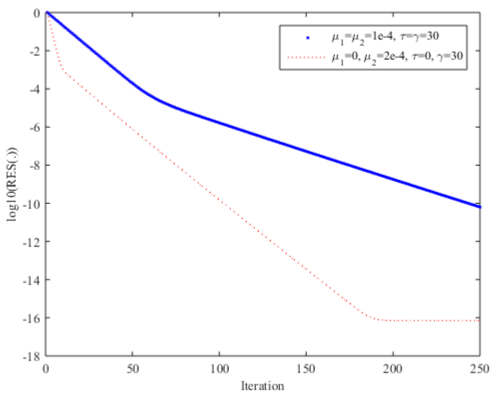

Figure 7: Example 5.4; The convergence curves for the iterative method (55) (when $\tau = 0,\ \gamma = 30,\ \mu_1 = 0,\ \mu_2 = 2 \times 10^{-4}$) and method in [7] (when $\tau = \gamma = 30,\ \mu_1 = \mu_2 = 1 \times 10^{-4}$).

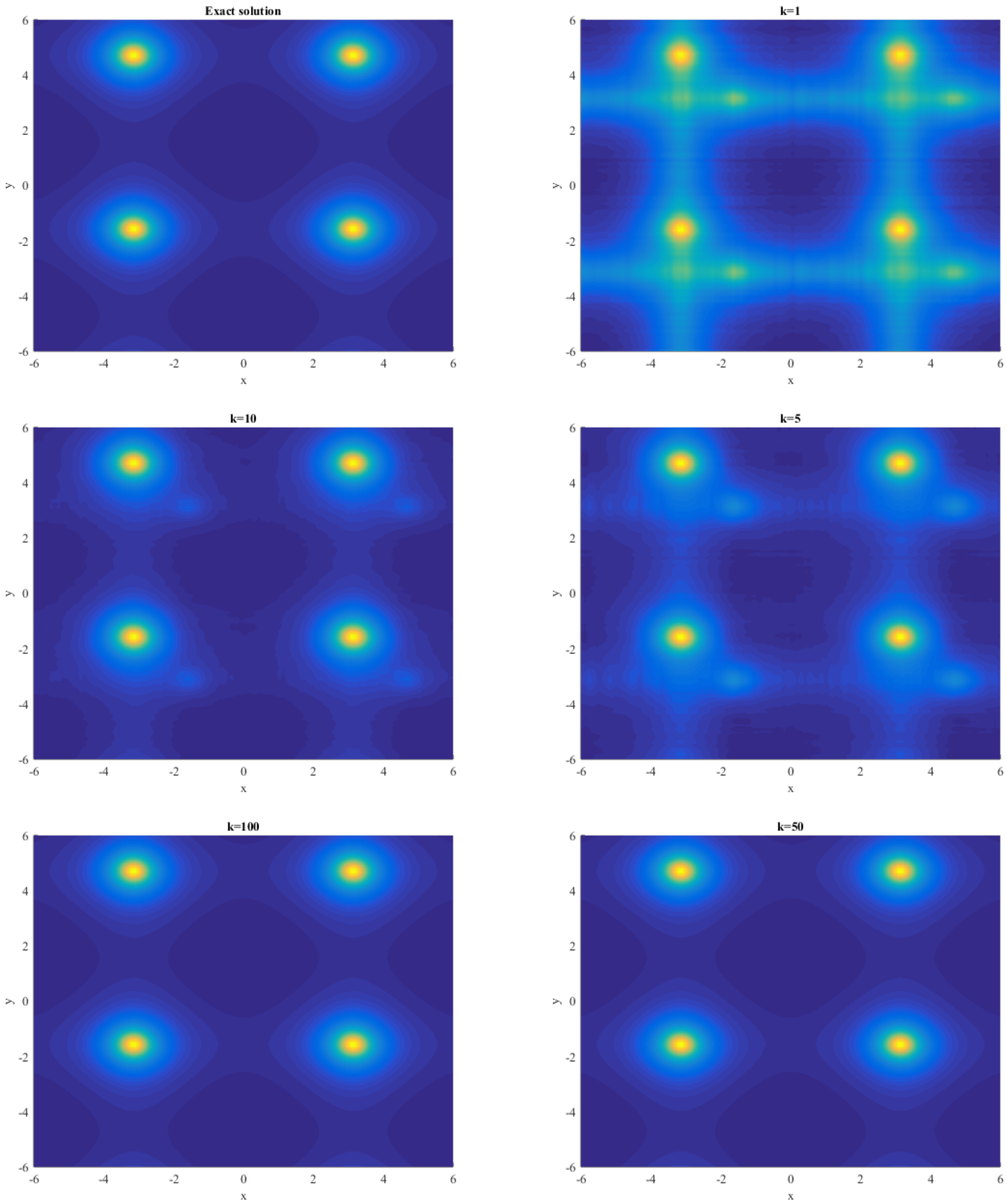

Figure 8: Example 5.4; Approximations of the exact solution X_2 in different iterations by $\tau = 0,\ \gamma = 30,\ \mu_1 = 0,\ \mu_2 = 2 \times 10^{-4}$.