WEIGHTED VERSIONS OF GL-FOM AND GL-GMRES FOR SOLVING GENERAL COUPLED LINEAR MATRIX EQUATIONS

Fatemeh Panjeh Ali Beik*,§ and Davod Khojasteh Salkuyeh†

§*Department of Mathematics, Vali-e-Asr University of Rafsanjan, P. O. Box 518, Rafsanjan, Iran*
e-mail: f.beik@vru.ac.ir, beik.fatemeh@gmail.com

†*Faculty of Mathematical Sciences, University of Guilan, P.O. Box 1914, Rasht, Iran*
e-mail: khojasteh@guialn.ac.ir, salkuyeh@gmail.com

Abstract. More recently, Beik and Salkuyeh [F. P. A. Beik and D. K. Salkuyeh, On the global Krylov subspace methods for solving general coupled matrix equations, *Computers and Mathematics with Applications*, 62 (2011) 4605–4613] have presented the Gl-FOM and Gl-GMRES algorithms for solving the general coupled linear matrix equations. In this paper, two new algorithms called weighted Gl-FOM (WGl-FOM) and weighted Gl-GMRES (WGl-GMRES) are proposed for solving the general coupled linear matrix equations. In order to accelerate the speed of convergence, a new inner product is used. Invoking the new inner product and a new matrix product, the weighted global Arnoldi algorithm is introduced which will be utilized for employing the WGl-FOM and WGl-GMRES algorithms to solve the linear coupled linear matrix equations. After introducing the weighted methods, some relations that link Gl-FOM (Gl-GMRES) to its weighted version are established. Numerical experiments are presented to illustrate the effectiveness of the new algorithms in comparison with Gl-FOM and Gl-GMRES for solving the linear coupled linear matrix equations.

Keywords: Linear matrix equation, Krylov subspace, Weighted methods, Global FOM, Global GMRES, Global Arnoldi.
AMS Subject Classification: Primely: 65F10, Secondly: 65F50 and 15A24.

1. Introduction

Linear matrix equations arise in many areas such as control theory, system theory, stability theory and some other fields of pure and applied mathematics. They also play a fundamental role in filtering theory for continues or discrete-time large-scale dynamical systems, image restoration and other problems; for more details see [2, 4, 11, 12, 17, 19, 20, 21] and the references therein. Before stating the main aim of this work, we briefly review some of the works which have been recently presented in the subject of linear matrix equations.

In [1], the authors have proposed the global full orthogonaliztion method (Gl-FOM) and the global generalized minimal residual (Gl-GMRES) method for solving the general coupled linear matrix equations

$$\sum_{j=1}^{p} A_{ij} X_j B_{ij} = C_i, \qquad i = 1, \ldots, p, \tag{1.1}$$

where $A_{ij} \in \mathbb{R}^{m\times m}, B_{ij} \in \mathbb{R}^{n\times n}$, and $C_i \in \mathbb{R}^{m\times n}$, $i, j = 1, 2, \ldots, p$, are large and sparse matrices, $X_i \in \mathbb{R}^{m\times n}$, $i = 1, 2, \ldots, p$, are the unknown matrices.

*Corresponding author

.

We would like to comment here that Eq. (1.1) includes many investigated matrix equations in the literature as special cases, see [4, 11, 12, 17, 19, 20, 21]. More precisely, Bouhamidi and Jbilou [2] have considered the generalized Sylvester matrix equation

$$\sum_{j=1}^{p} A_j X B_j = C, \tag{1.2}$$

and proposed a Krylov subspace method for solving (1.2).

In [6], Dehghan and Hajarian have presented an iterative method to solve the general coupled linear matrix equations (1.1) over generalized bisymmetric matrix group $(X_1, X_2, \ldots, X_p)$. Ding et al. [8] have presented a gradient based algorithm and a least-squares based iterative algorithm for solving (1.2). In [20], the authors have employed the CGNE [16] and Bi-CGSTAB [16] algorithms to solve (1.1).

Evidently, the general coupled linear matrix equations (1.1) is equivalent to

$$\sum_{j=1}^{p} (B_{ij}^T \otimes A_{ij}) \mathrm{vec}(X_j) = \mathrm{vec}(C_i), \qquad i = 1, \ldots, p, \tag{1.3}$$

where $\otimes$ denotes the Kronecker product operator and $\mathrm{vec}(Z) = (z_1^T, z_2^T, \ldots, z_m^T)^T$ for $Z = (z_1, z_2, \ldots, z_n) \in \mathbb{R}^{m\times n}$. Obviously, the coefficient matrix of the linear system (1.3) is of order pmn and can be solved by iterative methods such as the methods based on the Krylov subspace methods like the GMRES [15]. However, the size of the linear system (1.3) would be huge even for moderate values of m, n and p. Therefore, in application, we apply an iterative method for solving the original system (1.1) instead of the linear system (1.3). Evidently, the system (1.1) has a unique solution if and only if the coefficient matrix of the linear system (1.3) is nonsingular. Throughout this paper we assume that the system (1.1) has a unique solution.

The global Krylov subspace methods have been originally presented for solving linear system of equations with multiple right-hand sides by Jbilou et al. [11]. It is well-known that the global Krylov subspace methods outperform other iterative methods for solving such systems when the coefficient matrix is large and nonsymmetric. Moreover, they can be employed for solving large and sparse linear matrix equations; for more details see [2, 17, 14] and the references therein. On the other hand, for improving the rate of convergence of FOM and GMRES, the weighted Krylov subspace methods have been originally introduced by Essai [9] for solving nonsymmetric linear systems. Recently, in some aspects, there is a growing interest on the applications of weighted techniques for improving the speed of convergence. For instance, Jing and Huang [13] have been proposed a weighted version of FOM for solving shifted linear systems which the method is based on the weighted Arnoldi process introduced by Essai [9]. In [10], the weighted versions of global Krylov subspace methods have been introduced for solving the matrix equation $AX = B$ where $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times s}$ are given constant matrices, and $X \in \mathbb{R}^{n\times s}$ is the unknown matrix.

In the present work, we are interested to improve the speed of convergence of Gl-FOM and Gl-GMRES for solving (1.1). To this end, we extend the weighted Krylov subspace method presented in [10] to the new weighted methods for solving the general coupled linear matrix equations (1.1). For introducing these new methods, we need to define a new inner product and its corresponding matrix product. Then, we demonstrate how the weighted versions of Gl-FOM and Gl-GMRES can be applied for solving (1.1).

For simplicity, we define the linear operator $\mathcal{M}$ as follows

$$\mathcal{M} : \mathbb{R}^{m\times n} \times \cdots \times \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times pn},$$

$$X \to \mathcal{M}(X),$$

in which $X = (X_1, X_2, \ldots, X_p)$ and $\mathcal{M}(X) = (\mathcal{A}_1(X), \mathcal{A}_2(X), \ldots, \mathcal{A}_p(X)$ where

$$\mathcal{A}_i(X) = \sum_{j=1}^{p} A_{ij} X_j B_{ij}, \qquad i = 1, 2, \ldots, p.$$

Using the linear operator $\mathcal{M}$, we can rewrite Eq. (1.1) as

$$\mathcal{M}(X) = C, \tag{1.4}$$

where $C = (C_1, C_2, \ldots, C_p)$. We use the linear matrix operator $\mathcal{M}$ to propose WGl-FOM and WGl-GMRES algorithms for solving (1.4) which is equivalent to Eq. (1.1).

The outline of this paper is organized as follows. In Section 2, we first review some necessary definitions and principles which are used in next sections. Then, a new inner product and its corresponding matrix product are introduced which will be employed for analyzing the weighted versions of the global Krylov subspace methods for solving (1.1). In Section 3, we present and analyze the WGl-FOM and WGl-GMRES algorithms for solving (1.1). To demonstrate the better performance of the WGl-FOM and WGl-GMRES algorithms for solving (1.1), some numerical examples are presented in Section 4. More precisely, we numerically compare the application of the Gl-FOM (Gl-GMRES) with its weighted version in terms of both number of iterations and CPU-time (S). Finally, the paper is ended with a brief conclusion in Section 5.

2. Preliminaries

In this section, we recall some notations and definitions which are utilized throughout this paper. Some new concepts are also introduced which are useful for presenting the WGl-FOM and WGl-GMRES algorithms for solving (1.4).

Assume that $D = (d_{ij}) \in \mathbb{R}^{m\times n}$ has non-negative entries. Hence, we may define $D^{1/2} := (\sqrt{d_{ij}})$. Supposing that $U = [u_{ij}]$ and $V = [v_{ij}]$ belong to $\mathbb{R}^{m\times n}$, the Hadamard product of U and V is defined by $U \circ V = [u_{ij}v_{ij}]_{m\times n}$. For two matrices Y and Z in $\mathbb{R}^{m\times n}$, the inner product $< Y, Z >_F$ is defined as $< Y, Z >_F = \mathrm{tr}(Y^T Z)$, and the associate norm is the well-known Frobenius norm denoted by $\|.\|_F$. For $D \in \mathbb{R}^{m\times n}$, the D-scalar product is defined by $< Y, Z >_D = \mathrm{tr}(Y^T(D \circ Z))$. The scalar product $< Y, Z >_D$ is well defined if all the entries of D are non-negative [10]. Now, we define a new inner product as follows.

Definition 2.1. *Let Y and Z in $\mathbb{R}^{m\times n}$ and $D_k \in \mathbb{R}^{m\times n}$ is given such that all the entries of D_k are non-negative for $k = 1, 2$. The scalar product $< Y, Z >_{D_1,D_2}$ is defined by*

$$< Y, Z >_{D_1,D_2} = tr((D_1^{1/2} \circ Y)^T (D_2^{1/2} \circ Z)).$$

Remark 2.2. *Consider the matrices D_1 and D_2 as assumed in Definition 2.1, we set $D^*_{1,2} = D_1^{1/2} \circ D_2^{1/2}$. Evidently, $D^*_{1,2}$ is an $m \times n$ matrix with non-negative entries. It is not difficult to see that $< Y, Z >_{D_1,D_2} = < Y, Z >_{D^*_{1,2}}$. Hence, the scalar product introduced in Definition 2.1 is well defined.*

Definition 2.3. (Bouyouli et al. [3]). *Let $A = [A_1, A_2, \ldots, A_p]$ and $B = [B_1, B_2, \ldots, B_\ell]$ be matrices of dimensions $m \times pn$ and $m \times \ell n$, respectively, where A_i and B_j are $m \times n$ matrices. Then the matrix $A^T \Diamond B = [(A^T \Diamond B)_{ij}]_{p\times \ell}$ is defined by*

$$(A^T \Diamond B)_{ij} = < A_i, B_j >_F .$$

Now, as a natural way, we may define the weighted version of $\Diamond$ product as follows.

Definition 2.4. *Let $A = [A_1, A_2, \ldots, A_p]$ and $B = [B_1, B_2, \ldots, B_p]$ be matrices of dimensions $m \times pn$, respectively, where A_i and B_j are $m \times n$ matrices. Moreover, assume that $\mathcal{D} = [D_1, D_2, \ldots, D_p]$*

is an $m \times pn$, where each D_i is an $m \times n$ matrix with non-negative entries $(i = 1, 2, \ldots, p)$. Then the matrix $A^T \Diamond_{\mathcal{D}} B = [(A^T \Diamond_{\mathcal{D}} B)_{ij}]_{p\times p}$ is defined by

$$(A^T \Diamond_{\mathcal{D}} B)_{ij} =< A_i, B_j >_{D^*_{i,j}} .$$

In the following, we define a new inner product and its corresponding matrix norm which are used for deriving our further results in this paper.

Definition 2.5. *Assume that $\overline{X} = (\overline{X}_1, \overline{X}_2, \ldots, \overline{X}_p)$ and $\widetilde{X} = (\widetilde{X}_1, \widetilde{X}_2, \ldots, \widetilde{X}_p)$ are in $\mathbb{R}^{m\times pn}$. Let $\mathcal{D} = [D_1, D_2, \ldots, D_p]$ be an $m \times pn$, where each D_i is an $m \times n$ matrix with non-negative entries $(i = 1, 2, \ldots, p)$. We define the inner product $< ., . >_{\mathcal{D}}$ as follows:*

$$< \overline{X}, \widetilde{X} >_{\mathcal{D}} = \mathrm{tr}(\overline{X}^T \Diamond_{\mathcal{D}} \widetilde{X}). \tag{2.1}$$

Remark 2.6. *For $X = (\overline{X}_1, \overline{X}_2, \ldots, \overline{X}_p)$ in $\mathbb{R}^{m\times pn}$, the norm of X is defined by $\|X\|^2_{\mathcal{D}} = \mathrm{tr}(X^T \Diamond_{\mathcal{D}} X)$. Throughout this paper, a set of matrices in $\mathbb{R}^{m\times pn}$ is said to be orthonormal if it is orthonormal with respect to the scalar product (2.1).*

In the following, we use a new matrix product denoted by $\circledcirc_{\mathcal{D}}$ which is used for implementing WGl-FOM and WGl-GMRES for solving (1.4).

Definition 2.7. *Let $A = [A^{(1)}, A^{(2)}, \ldots, A^{(k)}]$, $B = [B^{(1)}, B^{(2)}, \ldots, B^{(\ell)}]$ be $m \times kpn$ and $m \times \ell pn$ matrices, respectively, where $A^{(i)} = [A^{(i)}_1, A^{(i)}_2, \ldots, A^{(i)}_p]$, $B^{(s)} = [B^{(s)}_1, B^{(s)}_2, \ldots, B^{(s)}_p]$ and $A^{(i)}_j, B^{(s)}_j \in \mathbb{R}^{m\times n}$ for $i = 1, 2, \ldots, k$, $s = 1, 2, \ldots, \ell$ and $j = 1, 2, \ldots, p$. Furthermore, suppose that $\mathcal{D} = [D_1, D_2, \ldots, D_p]$ be an $m \times pn$, where each D_i is an $m \times n$ matrix with non-negative entries $(i = 1, 2, \ldots, p)$. The $k \times \ell$ matrix $A^T \circledcirc_{\mathcal{D}} B$ is defined by:*

$$A^T \circledcirc_{\mathcal{D}} B = \begin{pmatrix} tr((A^{(1)})^T \Diamond_{\mathcal{D}} B^{(1)}) & tr((A^{(1)})^T \Diamond_{\mathcal{D}} B^{(2)}) & \ldots & tr((A^{(1)})^T \Diamond_{\mathcal{D}} B^{(\ell)}) \\ tr((A^{(2)})^T \Diamond_{\mathcal{D}} B^{(1)}) & tr((A^{(2)})^T \Diamond_{\mathcal{D}} B^{(2)}) & \ldots & tr((A^{(2)})^T \Diamond_{\mathcal{D}} B^{(\ell)}) \\ \vdots & \vdots & & \vdots \\ tr((A^{(k)})^T \Diamond_{\mathcal{D}} B^{(1)}) & tr((A^{(k)})^T \Diamond_{\mathcal{D}} B^{(2)}) & \ldots & tr((A^{(k)})^T \Diamond_{\mathcal{D}} B^{(\ell)}) \end{pmatrix}.$$

Remark 2.8. *It is not difficult to verify the following statements:*

- **(i)** *If $X = (X_1, X_2, \ldots, X_p) \in \mathbb{R}^{m\times pn}$, then $X^T \circledcirc_{\mathcal{D}} X = \|X\|^2_{\mathcal{D}}$.*
- **(ii)** *The matrix $A = (A^{(1)}, A^{(2)}, \ldots, A^{(k)})$ is called orthonormal if and only if*

$$A^T \circledcirc_{\mathcal{D}} A = I_k.$$

- **(iii)** *Let the matrices A, B be defined as before and $L \in \mathbb{R}^{k\times\ell}$. Then*

$$A^T \circledcirc_{\mathcal{D}} (B((L \otimes I_p) \otimes I_n)) = (A^T \circledcirc_{\mathcal{D}} B)L. \tag{2.2}$$

- **(iv)** *Let $A, B, C \in \mathbb{R}^{m\times kpn}$, then*
  - *(a) $(A + B)^T \circledcirc_{\mathcal{D}} C = A^T \circledcirc_{\mathcal{D}} C + B^T \circledcirc_{\mathcal{D}} C$.*
  - *(b) $A^T \circledcirc_{\mathcal{D}} (B + C) = A^T \circledcirc_{\mathcal{D}} B + A^T \circledcirc_{\mathcal{D}} C$.*
  - *(c) $(A^T \circledcirc_{\mathcal{D}} B)^T = B^T \circledcirc_{\mathcal{D}} A$.*

Remark 2.9. *In the case that all of entries of $\mathcal{D}$ are equal to one, the inner product $< ., . >_{\mathcal{D}}$, the norm $\|.\|_{\mathcal{D}}$ and $\circledcirc_{\mathcal{D}}$ product are respectively denoted by $< ., . >$, $\|.\|$ and $\circledcirc$ which have been already defined by Beik and Salkuyeh [1].*

3. Weighted versions of the global Krylov subspace methods

In this section, we introduce WGl-FOM and WGl-GMRES algorithms to solve Eq. (1.4). Suppose that $X^{(0)} = (X_1^{(0)}, X_2^{(0)}, \dots, X_p^{(0)})$ in $\mathbb{R}^{m\times pn}$ is a given initial approximate solution.

Consider the matrix Krylov subspace as follows:

$$\mathcal{K}_k(\mathcal{M}, R^{(0)}) = \text{span}\left\{R^{(0)}, \mathcal{M}(R^{(0)}), \dots, \mathcal{M}^{k-1}(R^{(0)})\right\}, \tag{3.1}$$

where $R^{(0)} = C - \mathcal{M}(X^{(0)})$.

3.1. **Global Arnoldi process.** In this subsection, we recall the global Arnoldi process which constructs an orthonormal, with respect to inner product $< .,. >$, basis for the matrix Krylov subspace defined by (3.1); for more details see [1].

Algorithm 1. *Global Arnoldi process.*
1. Set $V_1 = R^{(0)}/\left\|R^{(0)}\right\|$.
2. For $j = 1, 2, \dots, k$ Do
3. $\quad W := \mathcal{M}(V_j)$
4. $\quad$ For $i = 1, 2, \dots, j$ Do
5. $\qquad h_{ij} :=< W, V_i >$
6. $\qquad W := W - h_{ij}V_i$
7. $\quad$ End for
8. $\quad h_{j+1,j} := \|W\|$. If $h_{j+1,j} := 0$, then stop.
9. $\quad V_{j+1} := W/h_{j+1,j}$
10. End for

Suppose that $\mathcal{V}_k = [V_1, V_2, \dots, V_k]$ denotes the $m \times kpn$ matrix where $V_i = [V_1^{(i)}, V_2^{(i)}, \dots, V_p^{(i)}]$ is obtained by Algorithm 1 for $i = 1, 2, \dots, k$. Let $\overline{H}_k$ be a $(k+1)\times k$ upper Hessenberg matrix where its nonzero entries h_{ij} are calculated by Algorithm 1 and H_k is the $k \times k$ matrix obtained from $\overline{H}_k$ by deleting its last row. It is not difficult to see that the matrix $\mathcal{V}_k$ is an orthonormal basis for the $\mathcal{K}_k(\mathcal{M}, R^{(0)})$, i.e., $\mathcal{V}_k^T \circledcirc \mathcal{V}_k = I_k$.

Proposition 3.1. *Let $\mathcal{V}_k$, $\overline{H}_k$ and H_k be defined as before, then we have the following relations:*

(1) $[\mathcal{M}(V_1), \mathcal{M}(V_2), \dots, \mathcal{M}(V_k)] = \mathcal{V}_k((H_k \otimes I_p) \otimes I_n) + h_{k+1,k}[0_{m\times pn}, \dots, 0_{m\times pn}, V_{k+1}]$.
(2) $[\mathcal{M}(V_1), \mathcal{M}(V_2), \dots, \mathcal{M}(V_k)] = \mathcal{V}_{k+1}((\overline{H}_k \otimes I_p) \otimes I_n)$.

Proof. See [1]. □

3.2. **Weighted global Arnoldi process.** In this subsection, we presented a weighted version of the global Arnoldi process which produces an orthonormal basis for the matrix Krylov subspace defined by (3.1).

Algorithm 2. *Weighted global Arnoldi process.*
1. Set $\widetilde{V}_1 = R^{(0)}/\left\|R^{(0)}\right\|_{\mathcal{D}}$.
2. For $j = 1, 2, \dots, k$ Do
3. $\quad \widetilde{W} := \mathcal{M}(\widetilde{V}_j)$
4. $\quad$ For $i = 1, 2, \dots, j$ Do
5. $\qquad \widetilde{h}_{ij} :=< \widetilde{W}, \widetilde{V}_i >_{\mathcal{D}}$
6. $\qquad \widetilde{W} := \widetilde{W} - \widetilde{h}_{ij}\widetilde{V}_i$
7. $\quad$ End for
8. $\quad \widetilde{h}_{j+1,j} := \left\|\widetilde{W}\right\|_{\mathcal{D}}$. If $\widetilde{h}_{j+1,j} := 0$, then stop.
9. $\quad \widetilde{V}_{j+1} := \widetilde{W}/h_{j+1,j}$
10. End for

Remark 3.2. *In Algorithm 2, assume that the matrix $\mathcal{D} = [D_1, D_2, \ldots, D_p]$ is chosen such that all entries of $\mathcal{D}$ are equal to one. In this case, the weighted global Arnoldi process reduces to the global Arnoldi process which is presented in [1].*

In what follows, $\widetilde{\mathcal{V}}_k = [\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_k]$ represents the $m \times kpn$ matrix where $\widetilde{V}_i = [\widetilde{V}_1^{(i)}, \widetilde{V}_2^{(i)}, \ldots, \widetilde{V}_p^{(i)}]$ and is computed by Algorithm 2 for $i = 1, 2, \ldots, k$. Assume that $\overline{\widetilde{H}}_k$ is an $(k+1) \times k$ is an upper Hessenberg matrix where its nonzero entries h_{ij} are computed by Algorithm 2 and $\widetilde{H}_k$ is a $k \times k$ matrix obtained from $\overline{\widetilde{H}}_k$ by deleting its last row. Evidently, $\widetilde{\mathcal{V}}_k$ is an orthonormal basis for the $\mathcal{K}_k(\mathcal{M}, R^{(0)})$, i.e.,

$$\widetilde{\mathcal{V}}_k^T \circledcirc_{\mathcal{D}} \widetilde{\mathcal{V}}_k = I_k.$$

Proposition 3.3. *Let $\widetilde{\mathcal{V}}_k$, $\overline{\widetilde{H}}_k$ and $\widetilde{H}_k$ be defined as before, then we have the following relations:*

(1) $[\mathcal{M}(\widetilde{V}_1), \mathcal{M}(\widetilde{V}_2), \ldots, \mathcal{M}(\widetilde{V}_k)] = \widetilde{\mathcal{V}}_k((\widetilde{H}_k \otimes I_p) \otimes I_n) + \widetilde{h}_{k+1,k}[0_{m\times pn}, \ldots, 0_{m\times pn}, \widetilde{V}_{k+1}]$.

(2) $[\mathcal{M}(\widetilde{V}_1), \mathcal{M}(\widetilde{V}_2), \ldots, \mathcal{M}(\widetilde{V}_k)] = \widetilde{\mathcal{V}}_{k+1}((\overline{\widetilde{H}}_k \otimes I_p) \otimes I_n)$.

Now, we establish some relations between the matrices produced by the global Arnoldi [1] and weighted global Arnoldi processes.

Proposition 3.4. *Let the global Arnoldi and the weighted global Arnoldi processes do not break down before mth step. Moreover, assume that $\mathcal{V}_m = [V_1, V_2, \ldots, V_m]$ and $\widetilde{\mathcal{V}}_m = [\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_m]$ where V_i and $\widetilde{V}_i$ are respectively computed by the global Arnoldi and weighted global Arnoldi processes for $i = 1, 2, \ldots, m$. Then, there exists an upper triangular matrix U_m such that*

$$\widetilde{\mathcal{V}}_m = \mathcal{V}_m((U_m \otimes I_p) \otimes I_n), \tag{3.2}$$

$$U_m = \mathcal{V}_m^T \circledcirc \widetilde{\mathcal{V}}_m, \tag{3.3}$$

$$U_m^{-1} = \widetilde{\mathcal{V}}_m^T \circledcirc_{\mathcal{D}} \mathcal{V}_m, \tag{3.4}$$

$$\overline{\widetilde{H}}_m = U_{m+1}^{-1} \overline{H}_m U_m. \tag{3.5}$$

Proof. Invoking the fact that $\mathcal{V}_j$ and $\widetilde{\mathcal{V}}_j$ are two bases of the matrix Krylov subspace $\mathcal{K}_k(\mathcal{M}, R^{(0)})$, we may conclude (3.2) immediately. From Eq. (2.2) and the fact that $\mathcal{V}_m^T \circledcirc \mathcal{V}_m = I_m$, we can deduce (3.3). It is known that $\widetilde{\mathcal{V}}_m^T \circledcirc_{\mathcal{D}} \widetilde{\mathcal{V}}_m = I_m$. Hence from Eqs. (2.2) and (3.2), we get

$$I_m = \widetilde{\mathcal{V}}_m^T \circledcirc_{\mathcal{D}} \widetilde{\mathcal{V}}_m = \widetilde{\mathcal{V}}_m^T \circledcirc_{\mathcal{D}} (\mathcal{V}_m((U_m \otimes I_p) \otimes I_n)) = (\widetilde{\mathcal{V}}_m^T \circledcirc_{\mathcal{D}} \mathcal{V}_m) U_m,$$

which implies Eq. (3.4). Using (3.2), we get

$$[\mathcal{M}(\widetilde{V}_1), \mathcal{M}(\widetilde{V}_2), \ldots, \mathcal{M}(\widetilde{V}_m)] = [\mathcal{M}(V_1), \mathcal{M}(V_2), \ldots, \mathcal{M}(V_m)]((U_m \otimes I_p) \otimes I_n),$$

By using Proposition 3.3, we get

$$\begin{aligned}[\mathcal{M}(\widetilde{V}_1), \mathcal{M}(\widetilde{V}_2), \ldots, \mathcal{M}(\widetilde{V}_m)] &= \widetilde{\mathcal{V}}_{m+1}((\overline{\widetilde{H}}_m \otimes I_p) \otimes I_n) \\ &= \mathcal{V}_{m+1}((U_{m+1} \otimes I_p) \otimes I_n)((\overline{\widetilde{H}}_m \otimes I_p) \otimes I_n) \\ &= \mathcal{V}_{m+1}((U_{m+1}\overline{\widetilde{H}}_m \otimes I_p) \otimes I_n).\end{aligned}$$

Therefore,

$$[\mathcal{M}(V_1), \mathcal{M}(V_2), \ldots, \mathcal{M}(V_m)]((U_m \otimes I_p) \otimes I_n) = \mathcal{V}_{m+1}((U_{m+1}\overline{\widetilde{H}}_m \otimes I_p) \otimes I_n).$$

From Proposition 3.1, we get

$$\mathcal{V}_{m+1}((\overline{H}_m \otimes I_p) \otimes I_n)((U_m \otimes I_p) \otimes I_n) = \mathcal{V}_{m+1}((U_{m+1}\overline{\widetilde{H}}_m \otimes I_p) \otimes I_n).$$

Therefore, we derive

$$\mathcal{V}_{m+1}^T \circledcirc (\mathcal{V}_{m+1}((\overline{H}_m U_m \otimes I_p) \otimes I_n)) = \mathcal{V}_{m+1}^T \circledcirc (\mathcal{V}_{m+1}((U_{m+1}\overline{\widetilde{H}}_m \otimes I_p) \otimes I_n)).$$

Since $\mathcal{V}_{m+1}^T \circledcirc \mathcal{V}_{m+1} = I_{m+1}$, the Eq. (2.2) implies that

$$\overline{H}_m U_m = U_{m+1}\overline{\widetilde{H}}_m.$$

Now, we may conclude Eq. (3.5) by multiplying the above equality on the left side by U_{m+1}^{-1}. □

Based on the formula (3.5), Essai [9] has presented the following proposition.

Proposition 3.5. *Under the same assumptions as in Proposition 3.4, we can express $\widetilde{H}_m$ in terms of H_m by*

$$\widetilde{H}_m = U_m^{-1} H_m U_m + h_{m+1,m} u_{mm} g_{m+1} e_m^T,$$

where $g_{m+1} \in \mathbb{R}^m$ is obtained from column $m+1$ of the matrix U_{m+1}^{-1} by deleting its last component.

Proof. See [9]. □

The following remark has been established in [9].

Remark 3.6. *Under the same assumptions as in Proposition 3.4, we can express H_m in terms $\widetilde{H}_m$ by*

$$H_m = U_m^{-1} \widetilde{H}_m U_m + \frac{h_{m+1,m}}{u_{m+1,m+1}} u_{m+1} e_m^T,$$

where $u_{m+1} \in \mathbb{R}^m$ is obtained from column $m+1$ of the matrix U_{m+1} by deleting its last component.

3.3. **WGl-FOM for solving the general coupled linear matrix equations.** Assume that the initial guess $X^{(0)} \in \mathbb{R}^{m \times pn}$ is given with the corresponding residual $R^{(0)} = C - \mathcal{M}(X^{(0)})$. We recall that the Gl-FOM algorithm computes the approximate solution $X^{(k)}$ such that

$$X^{(k)} \in X^{(0)} + \mathcal{K}_k(\mathcal{M}, R^{(0)}),$$

and

$$R^{(k)} = C - \mathcal{M}(X^{(k)}) \perp \mathcal{K}_k(\mathcal{M}, R^{(0)}). \tag{3.6}$$

Considering the orthonormal basis $\mathcal{V}_k = [V_1, V_2, \ldots, V_k]$ for $\mathcal{K}_k(\mathcal{M}, R^{(0)})$, we get

$$X^{(k)} = X^{(0)} + \sum_{i=1}^{k} V_i y_i^{(k)} = X^{(0)} + \mathcal{V}_k((y^{(k)} \otimes I_p) \otimes I_n), \tag{3.7}$$

where the real vector $y^{(k)} = [y_1^{(k)}, y_2^{(k)}, \ldots, y_k^{(k)}]^T$ is obtained by imposing the orthogonality condition (3.6).

The WGl-FOM algorithm computes the approximate solution $\widetilde{X}^{(k)}$ such that

$$\widetilde{X}^{(k)} \in X^{(0)} + \mathcal{K}_k(\mathcal{M}, R^{(0)}),$$

and

$$R^{(k)} = C - \mathcal{M}(\widetilde{X}^{(k)}) \perp_{\mathcal{D}} \mathcal{K}_k(\mathcal{M}, R^{(0)}), \tag{3.8}$$

where $\mathcal{D} = [D_1, D_2, \ldots, D_p]$ is a given matrix with non-negative entries.
Consider the orthonormal basis $\widetilde{\mathcal{V}}_k = [\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_k]$ for $\mathcal{K}_k(\mathcal{M}, R^{(0)})$, we get

$$\widetilde{X}^{(k)} = X^{(0)} + \sum_{i=1}^{k} \widetilde{V}_i \widetilde{y}_i^{(k)} = X^{(0)} + \widetilde{\mathcal{V}}_k((\widetilde{y}^{(k)} \otimes I_p) \otimes I_n), \tag{3.9}$$

where the real vector $\widetilde{y}^{(k)} = [\widetilde{y}_1^{(k)}, \widetilde{y}_2^{(k)}, \dots, \widetilde{y}_k^{(k)}]^T$ is obtained by imposing the orthogonality condition (3.8).

Theorem 3.7. *The approximate solution $X^{(k)}$ produced by the Gl-FOM algorithm is given by $X^{(k)} = X^{(0)} + \mathcal{V}_k((y^{(k)} \otimes I_p) \otimes I_n)$ where $y^{(k)}$ is the solution of the following linear system*

$$H_k y = \beta e_1,$$

where $\beta = \left\| R^{(0)} \right\|$.

Proof. See [1]. □

With the same strategy employed in [1], we may establish the following theorem.

Theorem 3.8. *The approximate solution $\widetilde{X}^{(k)}$ produced by the WGl-FOM algorithm is given by $\widetilde{X}^{(k)} = X^{(0)} + \widetilde{\mathcal{V}}_k((\widetilde{y}^{(k)} \otimes I_p) \otimes I_n)$ where $\widetilde{y}^{(k)}$ is the solution of the following linear system*

$$\widetilde{H}_k \widetilde{y} = \widetilde{\beta} e_1,$$

where $\widetilde{\beta} = \left\| R^{(0)} \right\|_{\mathcal{D}}$.

The following propositions help us to obtain the residual $\left\| R^{(k)} \right\|$ ($\left\| R^{(k)} \right\|_{\mathcal{D}}$) without computing $X^{(k)}$ ($\widetilde{X}^{(k)}$).

Proposition 3.9. *The norm of residual $R^{(k)}$ corresponding to the approximate solution $X^{(k)}$ computed by the Gl-FOM algorithm satisfies the following equality*

$$\left\| R^{(k)} \right\| = h_{k+1,k} \left| y_k^{(k)} \right|,$$

where $y_k^{(k)}$ is the last component of the vector $y^{(k)}$.

Proof. See [1]. □

Analogues to the proof of Proposition 3.9, we may establish the following proposition.

Proposition 3.10. *The norm of residual $R^{(k)}$ corresponding to the approximate solution $\widetilde{X}^{(k)}$ computed by the WGl-FOM algorithm satisfies the following equality*

$$\left\| R^{(k)} \right\|_{\mathcal{D}} = \widetilde{h}_{k+1,k} \left| \widetilde{y}_k^{(k)} \right|,$$

where $\widetilde{y}_k^{(k)}$ is the last component of the vector $\widetilde{y}^{(k)}$.

To save memory and CPU-time requirements, the Gl-FOM and WGl-GMRES algorithms are utilized in a restarted mode. In fact, the algorithms are restarted every k inner iterations, where k is a given fixed integer and the corresponding algorithms is denoted by Gl-FOM (k) and WGl-FOM (k) which are summarized as follows:

Algorithm 3. *Gl-FOM(k) algorithm for* (1.1).
1. Choose $X^{(0)}$ and a tolerance ε. Compute $R^{(0)} = C - \mathcal{M}(X^{(0)})$ and $V_1 = R^{(0)}$.
2. Construct the orthonormal basis $V_1, V_2, \dots, V_k$ by Algorithm 1.
3. Find $y^{(k)}$ as the solution of the linear system

$$H_k y = \left\| R^{(0)} \right\| e_1.$$

4. Compute the residual $R^{(k)}$ and $\left\| R^{(k)} \right\|$ using Proposition 3.9.
5. If $\frac{\left\| R^{(k)} \right\|}{\left\| R^{(0)} \right\|} < \varepsilon$ Stop; else $R^{(0)} := R^{(k)}$, $V_1 := R^{(0)}$, Go to 2.

Now, we may present WGl-FOM algorithm as follows.

Algorithm 4. *WGl-FOM(k) algorithm for* (1.1).
1. Choose $X^{(0)}$ and a tolerance ε. Compute $R^{(0)} = C - \mathcal{M}(X^{(0)})$ and $\widetilde{V}_1 = R^{(0)}$.
2. Choose the matrix $\mathcal{D} = [D_1, D_2, \ldots, D_p]$ such that $\|\mathcal{D}\| = \sqrt{mnp}$.
3. Construct the orthonormal basis $\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_k$ by Algorithm 2.
4. Find $\widetilde{y}^{(k)}$ as the solution of the linear system

$$\widetilde{H}_k \widetilde{y} = \left\| R^{(0)} \right\|_{\mathcal{D}} e_1.$$

5. Compute the residual $R^{(k)}$ and $\left\| R^{(k)} \right\|_{\mathcal{D}}$ using Proposition 3.10.
6. If $\frac{\|R^{(k)}\|_{\mathcal{D}}}{\|R^{(0)}\|_{\mathcal{D}}} < \varepsilon$ Stop; else $R^{(0)} := R^{(k)}$, $\widetilde{V}_1 := R^{(0)}$, Go to 2.

Note that under restriction $\|\mathcal{D}\| = \sqrt{mnp}$, the WGl-FOM includes Gl-FOM as an especial case, i.e., when of of entries of $\mathcal{D}$ is equal to one the WGl-FOM reduces to Gl-FOM for solving (1.1). In the following proposition, we show that Algorithm 4 has a scaling-invariant property in the absence of round of errors.

Proposition 3.11. *If the matrix $\mathcal{D}$ is replaced with $\alpha\mathcal{D}$ where α is a given positive scalar. Then, the solution of WGl-FOM will not change.*

Proof. In the kth step of the weighted global Arnoldi process, assume that the matrix $\widehat{\mathcal{D}} = \alpha\mathcal{D}$ is used instead of $\mathcal{D}$. Let $\widehat{\widetilde{\mathcal{V}}}_k = [\widehat{\widetilde{V}}_1, \widehat{\widetilde{V}}_2, \ldots, \widehat{\widetilde{V}}_k]$, $\widehat{\overline{\widetilde{H}}}_k = (\widehat{\widetilde{h}}_{ij})_{m+1\times m}$ and $\widehat{\widetilde{H}}_k = (\widehat{\widetilde{h}}_{ij})_{m\times m}$ be the corresponding matrices to $\widetilde{\mathcal{V}}_k = [\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_k]$, $\overline{\widetilde{H}}_k = (\widetilde{h}_{ij})_{m+1\times m}$ and $\widetilde{H}_k = (\widetilde{h}_{ij})_{m\times m}$. Using the mathematical induction, we may show that $\widehat{\widetilde{V}}_i = \frac{1}{\sqrt{\alpha}}\widetilde{V}_i$, and hence $\widehat{\widetilde{W}} = \mathcal{M}(\widehat{\widetilde{V}}_i) = \frac{1}{\sqrt{\alpha}}\mathcal{M}(\widetilde{V}_i) = \frac{1}{\sqrt{\alpha}}\widetilde{W}$. Now, it is not difficult to verify that

$$\widehat{\widetilde{h}}_{ij} =< \widehat{\widetilde{W}}, \widehat{\widetilde{V}}_i >_{\widehat{\mathcal{D}}}= \frac{1}{\alpha} < \widetilde{W}, \widetilde{V}_i >_{\widehat{\mathcal{D}}}=< \widetilde{W}, \widetilde{V}_i >_{\mathcal{D}}= \widetilde{h}_{ij},$$

which implies that $\widehat{\widetilde{\mathcal{V}}}_k = \frac{1}{\sqrt{\alpha}}\widetilde{\mathcal{V}}_k, \quad \widehat{\overline{\widetilde{H}}}_k = \overline{\widetilde{H}}_k \quad \text{and} \quad \widehat{\widetilde{H}}_k = \widetilde{H}_k.$

Suppose that $\widetilde{X}_k$ and $\widehat{\widetilde{X}}_k$ be the corresponding approximate solution to $\mathcal{D}$ and $\widehat{\mathcal{D}}$. Straightforward computations show that

$$\begin{aligned}
\widehat{\widetilde{X}}_k &= X_0 + \widehat{\widetilde{\mathcal{V}}}_k((\widehat{\widetilde{H}}_k^{-1} \|R^{(0)}\|_{\widehat{\mathcal{D}}} e_1 \otimes I_p) \otimes I_n) \\
&= X_0 + \tfrac{1}{\sqrt{\alpha}}\widetilde{\mathcal{V}}_k((\widetilde{H}_k^{-1}(\sqrt{\alpha}\|R^{(0)}\|_{\mathcal{D}})e_1 \otimes I_p) \otimes I_n) \\
&= \widetilde{\mathcal{V}}_k((\widetilde{H}_k^{-1}\|R^{(0)}\|_{\mathcal{D}} e_1 \otimes I_p) \otimes I_n) = \widetilde{X}_k,
\end{aligned}$$

which completes the proof. □

3.4. **A link between Gl-FOM and WGl-FOM for solving** (1.1)**.** In this subsection, we demonstrate how we may compute the $k$the iterate $\widetilde{X}_k$ of WGl-FOM using the global Arnoldi process [1] instead of using the weighted global Arnoldi process.

From Eq.(3.9), we have

$$\widetilde{X}^{(k)} - X^{(0)} = \widetilde{\mathcal{V}}_k((\widetilde{y}^{(k)} \otimes I_p) \otimes I_n).$$

Now, Eq. (3.2) implies that

$$\begin{aligned}
\widetilde{X}^{(k)} - X^{(0)} &= \mathcal{V}_k((U_k \otimes I_p) \otimes I_n)((\widetilde{y}^{(k)} \otimes I_p) \otimes I_n) \\
&= \mathcal{V}_k((U_k\widetilde{y}^{(k)} \otimes I_p) \otimes I_n) \\
&= \mathcal{V}_k((\widehat{\widetilde{y}}^{(k)} \otimes I_p) \otimes I_n),
\end{aligned}$$

where $\widehat{\widetilde{y}}^{(k)} \in \mathbb{R}^k$ and $\widetilde{y}^{(k)} = U_k^{-1}\widehat{\widetilde{y}}^{(k)}$.

Using Remark 3.6 and some straightforward computations, we may show that $\widehat{\widetilde{y}}^{(k)}$ is the solution of the following linear system

$$\widehat{H}_k y = \beta e_1,$$

in which $\beta = \|R^{(0)}\|$ and

$$\widehat{H}_k = H_k - \frac{h_{k+1,k}}{u_{k+1,k+1}} u_{k+1} e_k^T. \tag{3.10}$$

From Eq. (3.3), we derive that

$$u_{k+1} = \left(\begin{array}{cccc} tr(V_1^T \lozenge \widetilde{V}_{k+1}), & tr(V_2^T \lozenge \widetilde{V}_{k+1}), & \ldots, & tr(V_{k+1}^T \lozenge \widetilde{V}_{k+1}) \end{array} \right)^T, \tag{3.11}$$

where V_i and $\widetilde{V}_i$, $i = 1, 2, \ldots, k+1$, are respectively computed by the global Arnoldi and weighted global Arnoldi processes. Invoking Eqs. (3.10) and (3.11), we may compute $\widetilde{X}_k$ by the global Arnoldi process with the knowledge of the entries of $\widetilde{V}_{k+1}$. This would help us to switch between the Gl-FOM and WGl-FOM at each iterate.

3.5. **WGl-GMRES for solving the general coupled linear matrix equations.** In this section, we first briefly review Gl-GMRES for solving (1.1); for more details see [1]. In Gl-GMRES algorithm, the kth approximate solution X_k belongs to $X^{(0)} + \mathcal{K}_k(\mathcal{M}, R^{(0)})$. Meanwhile, the vector $y^{(k)}$ in Eq. (3.7) is obtained by imposing the following orthogonality condition

$$R^{(k)} = C - \mathcal{M}(X_k) \quad \perp \quad \mathcal{K}_k(\mathcal{M}, \mathcal{M}(R_0)). \tag{3.12}$$

Finding $X^{(k)}$ by (3.12) is equivalent to solve the following minimization problem

$$\min_{X - X^{(0)} \in \mathcal{K}_k(\mathcal{M}, R^{(0)})} \|C - \mathcal{M}(X)\|. \tag{3.13}$$

Now, we may present the WGl-GMRES method. In the WGl-GMRES algorithm

$$\widetilde{X}_k \in X^{(0)} + \mathcal{K}_k(\mathcal{M}, R^{(0)}).$$

Therefor,

$$\widetilde{X}_k = X^{(0)} + \widetilde{\mathcal{V}}_k((\widetilde{y}^{(k)} \otimes I_p) \otimes I_n)).$$

The vector $\widetilde{y}^{(k)} \in \mathbb{R}^k$ is computed such that

$$R^{(k)} = C - \mathcal{M}(\widetilde{X}_k) \quad \perp_{\mathcal{D}} \quad \mathcal{K}_k(\mathcal{M}, \mathcal{M}(R_0)). \tag{3.14}$$

Like the Gl-GMRES method, we may show that imposing the orthogonality condition (3.14) is equivalent to solve the following minimization problem

$$\min_{X - X^{(0)} \in \mathcal{K}_k(\mathcal{M}, R^{(0)})} \|C - \mathcal{M}(X)\|_{\mathcal{D}}. \tag{3.15}$$

Theorem 3.12. *The approximate solution $X^{(k)}$ computed by the Gl-GMRES algorithm is presented by $X^{(k)} = X^{(0)} + \mathcal{V}_k((y^{(k)} \otimes I_p) \otimes I_n)$ where $y^{(k)}$ is the solution of the following least-squares problem*

$$\min_{y \in \mathbb{R}^k} \left\| \left\| R^{(0)} \right\| e_1 - \overline{H}_k y \right\|_2. \tag{3.16}$$

Proof. See [1]. □

The proof of the following theorem is the same as the proof of Theorem 3.12.

Theorem 3.13. *The approximate solution $\widetilde{X}^{(k)}$ computed by the WGl-GMRES algorithm is presented by $\widetilde{X}^{(k)} = X^{(0)} + \widetilde{\mathcal{V}}_k((\widetilde{y}^{(k)} \otimes I_p) \otimes I_n)$ where $\widetilde{y}^{(k)}$ is the solution of the following least-squares problem*

$$\min_{y \in \mathbb{R}^k} \left\| \left\| R^{(0)} \right\|_{\mathcal{D}} e_1 - \overline{\widetilde{H}}_k y \right\|_2 . \tag{3.17}$$

Let us consider the QR decompositions of the $(k+1) \times k$ matrices $\overline{H}_k$ and $\overline{\widetilde{H}}_k$. That is,

$$\overline{R}_k = Q_k \overline{H}_k, \qquad \overline{\widetilde{R}}_k = \widetilde{Q}_k \overline{\widetilde{H}}_k.$$

where $\overline{R}_k$ ($\overline{\widetilde{R}}_k$) and Q_k ($\widetilde{Q}_k$) are respectively upper triangular and unity matrices. We set

$$\overline{g}_k = \left\| R^{(0)} \right\| Q_k e_1 = (\gamma_1, \gamma_2, \ldots, \gamma_{k+1})^T, \quad \overline{\widetilde{g}}_k = \left\| R^{(0)} \right\|_{\mathcal{D}} \widetilde{Q}_k e_1 = (\widetilde{\gamma}_1, \widetilde{\gamma}_2, \ldots, \widetilde{\gamma}_{k+1})^T,$$

and R_k ($\widetilde{R}_k$) stands as the $k \times k$ matrix obtained from $\overline{R}_k$ ($\overline{\widetilde{R}}_k$) by removing its last row and g_k ($\widetilde{g}_k$) is the k-dimensional vector obtained from $\overline{g}_k$ ($\overline{\widetilde{g}}_k$) by deleting its last component. It is not difficult to see that $y^{(k)} = R_k^{-1} g_k$ ($\widetilde{y}^{(k)} = \widetilde{R}_k^{-1} \widetilde{g}_k$).

Theorem 3.14. *The residual $R^{(k)} = C - \mathcal{M}(X^{(k)})$ obtained by the Gl-GMRES algorithm for the general coupled matrix equations satisfies the following equalities*

$$R^{(k)} = \gamma_{k+1} \mathcal{V}_{k+1}((Q_k^T e_{k+1} \otimes I_p) \otimes I_n),$$

and

$$\left\| R^{(k)} \right\| = |\gamma_{k+1}| ,$$

where γ_{k+1} is the last component of the vector $\overline{g}_k$.

Proof. See [1]. □

Analogous to strategy employed in the proof of Theorem 3.14, we may prove the following theorem.

Theorem 3.15. *The residual $R^{(k)} = C - \mathcal{M}(\widetilde{X}^{(k)})$ obtained by the WGl-GMRES algorithm for the general coupled matrix equations satisfies the following equalities*

$$R^{(k)} = \widetilde{\gamma}_{k+1} \widetilde{\mathcal{V}}_{k+1}((\widetilde{Q}_k^T e_{k+1} \otimes I_p) \otimes I_n),$$

and

$$\left\| R^{(k)} \right\|_{\mathcal{D}} = |\widetilde{\gamma}_{k+1}| ,$$

where $\widetilde{\gamma}_{k+1}$ is the last component of the vector $\overline{\widetilde{g}}_k$.

In application, the Gl-GMRES and WGl-GMRES algorithms are restarted every k inner iterations, where k is a given fixed integer number.

Algorithm 5. *Gl-GMRES(k) algorithm for (1.1).*
1. Choose $X^{(0)}$, a tolerance ε. Compute $R^{(0)} = C - \mathcal{M}(X^{(0)})$, and $V_1 = R^{(0)}$.
2. Construct the orthonormal basis $V_1, V_2, \ldots, V_k$ by Algorithm 1.
3. Determine $y^{(k)}$ as the solution of the least-squares problem:

$$\min_{y \in \mathbb{R}^k} \left\| \|R^{(0)}\| e_1 - \overline{H}_k y \right\|_2 .$$

Compute $X^{(k)} = X^{(0)} + \mathcal{V}_k((y^{(k)} \otimes I_p) \otimes I_n)$.
4. Compute the residual $R^{(k)}$ and $\left\| R^{(k)} \right\|$ using Theorem 3.14.
5. If $\frac{\|R^{(k)}\|}{\|R^{(0)}\|} < \varepsilon$ Stop; else $R^{(0)} := R^{(k)}$, $V_1 := R^{(0)}$, Go to 2.

Now, we present the WGl-GMRES algorithm for solving the general coupled matrix equations (1.1).

Algorithm 6. *WGl-GMRES(k) algorithm for* (1.1).
1. Choose $X^{(0)}$, a tolerance ε. Compute $R^{(0)} = C - \mathcal{M}(X^{(0)})$, and $\widetilde{V}_1 = R^{(0)}$.
2. Choose the matrix $\mathcal{D} = [D_1, D_2, \ldots, D_p]$ such that $\|\mathcal{D}\| = \sqrt{mnp}$.
3. Construct the orthonormal basis $\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_k$ by Algorithm 2.
4. Determine $\widetilde{\widetilde{y}}^{(k)}$ as the solution of the least-squares problem:

$$\min_{y\in\mathbb{R}^k} \left\| \|R^{(0)}\|_{\mathcal{D}} e_1 - \overline{\widetilde{H}}_k y \right\|_2 .$$

Compute $\widetilde{X}^{(k)} = X^{(0)} + \widetilde{\mathcal{V}}_k((\widetilde{y}^{(k)} \otimes I_p) \otimes I_n)$.
5. Compute the residual $R^{(k)}$ and $\left\|R^{(k)}\right\|_{\mathcal{D}}$ using Theorem 3.15.
6. If $\frac{\|R^{(k)}\|}{\|R^{(0)}\|} < \varepsilon$ Stop; else $R^{(0)} := R^{(k)}$, $\widetilde{V}_1 := R^{(0)}$, Go to 2.

With a similar strategy employed in a proof of Proposition 3.11, we may show that Algorithm 6 has a scaling-invariant property in the exact arithmetic.

3.6. **A link between Gl-GMRES and WGl-GMRES.** It is known that the kth approximate solution $\widetilde{X}_k$ computed by WGl-GMRES satisfies

$$\widetilde{X}^{(k)} = X^{(0)} + \widetilde{\mathcal{V}}_k((\widetilde{y}^{(k)} \otimes I_p) \otimes I_n) = X^{(0)} + \mathcal{V}_k((\widehat{\widetilde{y}}^{(k)} \otimes I_p) \otimes I_n),$$

where $\widetilde{y}^{(k)} = U_k^{-1}\widehat{\widetilde{y}}^{(k)}$. Therefore, we can derive the $\widetilde{X}_k$ using the global Arnoldi process by computing $\widehat{\widetilde{y}}^{(k)}$. With the same techniques employed in [9], it can be shown that $\widehat{\widetilde{y}}^{(k)}$ is the solution of the following minimization problem

$$\min_{y\in\mathbb{R}^k} \left\| \left\|R^{(0)}\right\| e_1 - \overline{H}_k y \right\|_{U_{k+1}^{-T}U_{k+1}^{-1}} .$$

As seen, in order to calculate $\widetilde{X}_k$ in WGl-GMRES from the global Arnoldi process, we must have all the entries of $\widetilde{\mathcal{V}}_{k+1} = [\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_{k+1}]$. While, for computing $\widetilde{X}_k$ in WGl-FOM with the global Arnoldi process, we only need to know the entries of $\widetilde{V}_{k+1}$.

3.7. **Strategy for choosing weights.** As a natural way, we generalize the idea of choosing the weights by the authors in [10]. In the kth step of WGl-FOM (WGl-GMRES) algorithm, the matrix $\mathcal{D} = [D_1, D_2, \ldots, D_p]$ are chosen such that

$$(D_\ell)_{ij} = \sqrt{mnp}\frac{|(R_\ell^{(k)})_{ij}|}{\|R^{(k)}\|}, \quad i = 1, 2, \ldots, m, \quad j = 1, 2, \ldots, n, \quad \ell = 1, 2, \ldots, p, \tag{3.18}$$

in which $R^{(k)} = C - \mathcal{M}(\widetilde{X}_k) = [R_1^{(k)}, R_2^{(k)}, \ldots, R_p^{(k)}]$.

4. Numerical examples

In this section, some numerical examples are presented to illustrate the effectiveness of the WGl-GMRES(k) and WGl-FOM(k) algorithms and to compare with those of the Gl-GMRES(k) and Gl-FOM(k) algorithms, respectively. All the numerical experiments presented in this section were computed in double precision with some Matlab codes on a Pentium 4 PC, with a 3.06 GHz CPU and 1.00GB of RAM.

Example 4.1. We consider the general coupled matrix equations (see [1])

$$\begin{cases} AX_1 + X_2B = C_1, \\ BX_1 + X_2A = C_2, \end{cases}$$

Table 1. Numerical results for Example 4.1.

m	Gl-GMRES(5)	WGl-GMRES(5)	Gl-FOM(5)	WGl-FOM(5)
50	25(1.3)	17(1.1)	23(1.2)	17(1.1)
100	24(7.9)	17(5.2)	23(6.1)	20(6.1)
150	23(17.4)	17(14.7)	23(16.8)	18(16.1)
200	23(41.0)	17(34.3)	23(39.1)	18(35.8)

where

$$A = \begin{pmatrix} 4 & -1 & & -1 \\ -1 & 4 & \ddots & \\ & \ddots & \ddots & -1 \\ -1 & & -1 & 4 \end{pmatrix}, \quad \text{and} \quad B = \begin{pmatrix} 8 & -2 & & -2 \\ -2 & 8 & \ddots & \\ & \ddots & \ddots & -2 \\ -2 & & -2 & 8 \end{pmatrix},$$

are $m \times m$ matrices. The right-hand side of the corresponding system $\mathcal{M}(X) = C$ was taken such that $X = (X_1, X_2)$ is the exact solution of the system where $X_1 = \text{tridiag}(1, 1, 1)$ and $X_2 = \text{tridiag}(1, -1, 1)$. The initial guess was taken to be zero and the test was stopped as soon as

$$\frac{\|R^{(j)}\|}{\|R^{(0)}\|} < 10^{-8},$$

where $R^{(j)} = C - \mathcal{M}(X^{(j)})$. Gl-GMRES(5) and Gl-FOM(5) algorithms and their weighted versions have been used to solve the system for different values of m. The numerical results are given in Table 1. In this table, number of iterations needed for the convergence and the CPU times for computing the approximate solution are given (CPU times in seconds are given in parenthesis). As observed, the weighted versions of the algorithms are superior to primitive ones.

Example 4.2. Let

$$T_{d,k} = \text{tridiag}(-1 + \frac{10}{k+1}, d, -1 + \frac{10}{k+1}) \in \mathbb{R}^{k \times k}.$$

We consider the general coupled matrix equations

$$\begin{cases} A_{11}X_1B_{11} + A_{12}X_2B_{12} = C_1, \\ A_{21}X_1B_{21} + A_{22}X_2B_{22} = C_2, \end{cases}$$

where $B_{11} = B_{22} = T_{2,n}$, $B_{12} = B_{21} = T_{3,n}$ and $A_{11} = A_{12} = A_{21} = A_{22} = \text{NOS4}$, in which NOS4 has been downloaded from the Matrix-Market website [18]. Here we mention that NOS4 is a symmetric positive definite matrix of order 100 with 347 nonzero entries. The right-hand side of the corresponding system $\mathcal{M}(X) = C$ was taken such that $X = (X_1, X_2)$ is the exact solution of the system where

$$(X_1)_{ij} = \begin{cases} 1, & |i-j| \leq 1, \\ 0, & \text{otherwise}, \end{cases}$$

$$(X_2)_{ij} = \begin{cases} 1, & i = j, \\ 0, & \text{otherwise}. \end{cases}$$

All of other assumptions are as the previous example. The numerical results for different values of n are presented in Table 2. In this table, a dagger (†) shows that the method is not convergent in 2000 iterations. As seen, numerical results demonstrate that the WGl-GMRES(5) and WGl-FOM(5) algorithms are more effective than the Gl-GMRES(5) and Gl-FOM(5) algorithms, respectively.

Table 2. Numerical results for Example 4.2.

m	Gl-GMRES(5)	WGl-GMRES(5)	Gl-FOM(5)	WGl-FOM(5)
25	76(3.3)	49(2.6)	†	135(7.4)
50	99(10.5)	59(7.6)	†	214(27.8)
75	133(29.6)	75(16.9)	†	158(36.0)
100	173(51.5)	82(28.9)	†	184(65.7)

5. Conclusion

The weighted versions of Gl-FOM and Gl-GMRES have been proposed. Some links between Gl-FOM and Gl-GMRES algorithms and their weighted versions have been established. We presented some numerical experiments to show the good performance of weighted methods in terms of both number of iterations and CPU-time(s). The open problem of finding optimum weights is under investigation.

References

[1] F. P. A. Beik and D. K. Salkuyeh, On the global Krylov subspace methods for solving general coupled matrix equations, *Compute. Math. Appl.* **62** (2011) 4605–4613.
[2] A. Bouhamidi and K. Jbilou, A note on the numerical approximate solutions for generalized Sylvester matrix equations with applications, *Appl. Math. Comput.* **206** (2008) 687–694.
[3] R. Bouyouli, K. Jbilou, R. Sadaka and H. Sadok, Convergence properties of some block Krylov subspace methods, *J. Comput. Appl. Math.* **196** (2006) 498–511.
[4] X.W. Chang and J.S.Wang, The symmetric solution of the matrix equations $AX+YA=C$, $AXA^T+BYB^T=C$ and $(A^TXA, B^TXB)=(C,D)$, *Linear Algebra Appl.* **179** (1993) 171–189.
[5] Z. H. Cao and X. Y. Yu, A note on weighted FOM and GMRES for solving nonsymmetric linear systems,*Appl. Math. comput.* **151** (2004) 719–727.
[6] M. Dehghan and M. Hajarian, The general coupled matrix equations over generalized bisymmetric matrices, *Linear Algebra Appl.* **432** (2010) 1531–1552.
[7] F. Ding and T. Chen, On iterative solutions of general coupled matrix equations, *SIAM J. Control Optim.* **44** (2006) 2269–2284.
[8] F. Ding, P.X. Liu and J. Ding, Iterative solutions of the generalized Sylvester matrix equations by using the hierarchical identification principle, *Appl. Math. Comput.* **197** (2008) 41–50.
[9] A. Essai, Weighted FOM and GMRES for solving nonsymmetric linear systems, *Numerical Algorithms,* **18** (1998) 277–292.
[10] M. Heyouni and A. Essai, Matrix Krylov subspace methods for linaer systems with multiple right-hand sides, *Numerical Algorithms,* **40** (2005) 137–156.
[11] K. Jbilou, A. Messaudi and H. Sadok, Global FOM and GMRES algorithms for matrix equations, *Appl. Numer. Math.* **31** (1999) 49–63.
[12] K. Jbilou and A. J. Riquet, Projection methods for large Lyapunov matrix equations, *Linear Algebra Appl.* **415** (2006) 344–358.
[13] Y.-F. Jing and T. Z. Huang, Restarted weighted full orthogonalization method for shifted linear systems, *Compute. Math. Appl.,* **57** (2009) 1583–1591.
[14] Y.Q. Lin, Implicitly restarted global FOM and GMRES for nonsymmetric equations and Sylvester equations, *Appl. Math. comput.* **167** (2005) 1004–1025.
[15] Y. Saad and M.H. Schultz, *GMRES: A generalized minimal residual method for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput. 7 (1986), pp. 856-869.
[16] Y. Saad, Iterative Methods for Sparse linear Systems, PWS press, New York, 1995.
[17] D.K. Salkuyeh and F. Toutounian, New approaches for solving large Sylvester equations, *Appl. Math. comput.* **173** (2006) 9–18.
[18] Matrix Market, http://math.nist.gov/MatrixMarket (August 2005).
[19] Q.W. Wang, J.H. Sun and S.Z. Li, Consistency for bi(skew)symmetric solutions to systems of generalized Sylvester equations over a finite central algebra, *Linear Algebra Appl.* **353** (2002) 169–182.
[20] J.J. Zhang, A note on the iterative solutions of general coupled matrix equation, *Appl. Math. Comput.* **217** (2011) 9380–8386.
[21] B. Zhou and G.R. Duan, On the generalized Sylvester mapping and matrix equation, *Syst. Contro Lett.* **57 (3)** (2008) 200–208.