International Electronic Journal of Pure and Applied Mathematics

Volume 2 No. 2 2010, 47-67

A NEW WEIGHTED GLOBAL FULL ORTHOGONALIZATION METHOD FOR SOLVING NONSYMMETRIC LINEAR SYSTEMS WITH MULTIPLE RIGHT-HAND SIDES

M. Mohseni Moghadam[1], F. Panjeh Ali Beik[2 §]

[1,2]Centre of Mathematics and Computer Science
Department of Mathematics
Shahid Bahonar University of Kerman
Kerman, IRAN
[1]e-mail: mohseni@mail.uk.ac.ir
[2]e-mail: panjeh_ali_beik@yahoo.com

Abstract: In the present work, we will introduce a new weighted global FOM (WG-FOM) method, which is a variant of global FOM (G-FOM) method, for solving nonsymmetric linear systems. Using the Schur complement formulae and a new matrix product, we give some new results for the WG-FOM method when applied to multiple linear systems. These results can be used to derive new convergence properties for the global FOM (G-FOM) method and its weighted version which will be presented in this paper. Finally, experimental results are presented to show the good performance of the new method compared to G-FOM. The WG-FOM can provide accelerating convergence rate with respect to the number of restarts in compared with G-FOM. Also, the associated CPU time is reduced as shown by the numerical experiments. Moreover, our new method is able to solve certain systems which the G-FOM cannot handle sometimes.

AMS Subject Classification: 65F10
Key Words: nonsymmetric multiple linear systems, global FOM, iterative methods, Schur complement

1. Introduction

Many applications require the solution of the following multiple linear system of equations

$$AX = B, \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$ is a large and spars matrix, B and X are $n \times s$ rectangular real

Received: July 20, 2010

§Correspondence author

matrices with $s \ll n$.

For nonsymmetric problems, recently, some block Krylov subspace methods have been developed; see [13, 15, 18, 22, 23] and [26] and the references therein.

The global full orthogonalization method (G-FOM) was derived in [15]. Also, some convergent properties for global methods for solving multiple linear systems were given in [2, 3, 25].

The weighted Arnoldi process has been developed by Essai in [14]. Recently, the application of the weighted techniques for the accelerating purpose in some aspects has been a subject of growing interest. For example, in [16], Yan Fei Jing and Ting-Zhu Huang employed weighted full orthogonalization method (WFOM) for solving shifted linear systems. Furthermore, Cao and Yu [7] discussed the performance of the preconditioned weighted FOM and GMRES methods.

In this work, first, we will present a new method called WG-FOM, for solving multiple nonsymmetric linear systems. Moreover, the weighted global Arnoldi process will be introduced for implementing this method. By studding the numerical comparison results between the FOM and WFOM algorithms, it is conjectured that in general the norm of the residual vector of the WFOM are not lower than the FOM in all steps. Note that the problem of finding optimum weights for WFOM algorithm is still an open problem, for more details see [7, 14]. As we will see, it is clear that the WG-FOM is a general form of WFOM, so the problem of finding optimum weights for WG-FOM algorithm is an open problem too. In this paper, for the special case, when all the weights are chosen equal, the new expression will help us to give a relation between the Frobenius norms of the residual matrices corresponding to the G-FOM and WG-FOM methods. The relation can be useful for finding proper weights.

This paper is organized as follows. In Section 2, we state some necessary definitions and notations and recall the global Arnoldi process and the G-FOM algorithm. In Section 3, we will introduce new weighted global Arnoldi process and we will prove some relations between global Arnoldi and weighted global Arnoldi processes. In Section 4, we will present new WG-FOM, also some links between G-FOM and its weighted version will be given. Furthermore, new relations between the norms of the residual matrices corresponding to the G-FOM and WG-FOM are given. In Section 5, experimental results are presented to show the effectiveness and good performance of the WG-FOM algorithm compared to G-FOM. Finally, the paper is ended with a brief conclusion in Section 6.

2. Preliminaries

We first recall the definition of Schur complement [24] and give some of their properties.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

Definition 1. Let M be a matrix partitioned into four blocks:

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix},$$

where the submatrix D is assumed to be square and nonsingular. The Schur complement of D in M, denote by (M/D), is defined by

$$(M/D) = A - BD^{-1}C.$$

If D is not a square matrix then pseudo-Schur complement of D in M can still be defined; for more details see [26, 27, 28]. Generalization and properties of the Schur complements are found in [1, 4, 6, 8, 10-12, 20, 21].

Proposition 1. (see A. Messaoudi et al [20]) *Let us assume that the matrix D is nonsingular, then*

$$\left(\begin{bmatrix} A & B \\ C & D \end{bmatrix} \Big/ D\right) = \left(\begin{bmatrix} D & C \\ A & B \end{bmatrix} \Big/ D\right) = \left(\begin{bmatrix} B & A \\ D & C \end{bmatrix} \Big/ D\right) = \left(\begin{bmatrix} C & D \\ A & B \end{bmatrix} \Big/ D\right).$$

Proposition 2. (see A. Messaoudi et al [20]) *Assuming that the matrix D is nonsingular and E is a matrix such that the product EA is well defined, then*

$$\left(\begin{bmatrix} EA & EB \\ C & D \end{bmatrix} \Big/ D\right) = E\left(\begin{bmatrix} A & B \\ C & D \end{bmatrix} \Big/ D\right).$$

For a matrix $V = [v_{ij}] \in \mathbb{R}^{n\times s}$, we denote by $vec(V)$ the vector of $\mathbb{R}^{ns}$ defined by

$$vec(V) = \left[v(.,1)^T, v(.,2)^T, \ldots, v(.,s)^T\right]^T$$

where $v(.,j), j = 1, \ldots, s$, is the j-th column of V.

Notation. For an arbitrary $n \times ms$ matrix $\mathcal{V} = [V_1, V_2, \ldots, V_m]$ where each V_i, $i = 1, 2, \ldots, m$, is an $n \times s$ matrix. We associate a new $ns \times m$ matrix $\mathcal{V}^v$ which is defined as the following,

$$\mathcal{V}^v = [vec(V_1), vec(V_2), \ldots, vec(V_m)].$$

For a given matrices $A \in \mathbb{R}^{n\times m}$ and $B \in \mathbb{R}^{k\times l}$, the so called Kronecker product of the matrices A and B, denoted by $A \otimes B$, is defined by the following $nk \times ml$ matrix,

$$A \otimes B = [a_{ij}B].$$

Some properties of this product are given as follows (see [17])

$$(A \otimes B)(E \otimes F) = (AE \otimes BF). \tag{2}$$

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

(i) If A and B are nonsingular matrices of dimension $n\times n$ and $p\times p$, respectively, then $(A\otimes B)^{-1} = A^{-1}\otimes B^{-1}$.

(ii) If A and B are $n\times n$ and $p\times p$, matrices, then

$$\det(A\otimes B) = \det(A)^n \det(B)^p \quad \text{and} \quad tr(A\otimes B) = tr(A)tr(B),$$

$$vec(ABC) = (C^T \otimes A)vec(B), \tag{3}$$

$$vec(A)^T vec(B) = tr(A^T B). \tag{4}$$

(iii) If $A \in \mathbb{R}^{n\times m}, B \in \mathbb{R}^{m\times p}$ and $C \in \mathbb{R}^{p\times n}$, then

$$tr(ABC) = vec(A^T)^T (I_n \otimes B)vec(C) = vec(C)^T (I_n \otimes B^T)vec(A^T).$$

In the rest of this paper, we suppose that $D \in \mathbb{R}^{n\times n}$ is a positive definite diagonal matrix. If u and v are two vectors which belong to $\mathbb{R}^{ns}$, the inner product $< .,. >_{I_s\otimes D}$ defined as

$$< u, v >_{I_s\otimes D} = u^T (I_s \otimes D)v.$$

The norm associated with this inner product denoted by $\|.\|_{I_s\otimes D}$ and defined by

$$\| \ u \ \|^2_{I_s\otimes D} =< u, u >_{I_s\otimes D} = u^T \ (I_s \otimes D) \, u.$$

For two matrices Y and Z in $\mathbb{R}^{n\times s}$, we define the inner product $< .,. >_D$ as follows

$$< Y, Z >_D = tr(Y^T DZ), \tag{5}$$

$tr(Y^T DZ)$ denotes the trace of the matrix $Y^T DZ$. We introduce matrix norm $\|.\|_D$ associated to the inner product (5) by

$$\|X\|^2_D = tr(X^T DX) \quad \forall X \in \mathbb{R}^{n\times s} \ .$$

Note that if $D = I_{n\times n}$, then $< .,. >_D$ reduces to $< .,. >_F$ defined as follows

$$< Y, Z >_F = tr(Y^T Z).$$

Proposition 3. *Let $X, Z \in \mathbb{R}^{n\times s}$, then*

$$< X, Z >_D =< vec(X), vec(Z) >_{I_s\otimes D} = (vec(X))^T (I_s \otimes D)vec(Z).$$

Thus $\|X\|_D = \|vec(X)\|_{I_s\otimes D}$.

Proof. Suppose $X = [x_1, x_2, \dots, x_s]$ and $Z = [z_1, z_2, \dots, z_s]$, by property (iii) of the Kronecker product, we conclude that

$$< X, Z >_D = tr(X^T DZ) =< vec(X), vec(Z) >_{I_s\otimes D} \ .$$

Hence $\|X\|_D = \|vec(X)\|_{I_s\otimes D}$. □

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

In the following we introduce a new product denoted by $\diamondsuit_D$ defined as follows:

Definition 2. Let $A = [A_1, A_2, \ldots, A_p]$ and $B = [B_1, B_2, \ldots, B_\ell]$ be matrices of dimensions $n \times ps$ and $n \times \ell s$, respectively, where A_i and B_j are $n \times s$ matrices. Then the $p \times \ell$ matrix $A^T \diamondsuit_D B$ is defined by

$$A^T \diamondsuit_D B = \begin{pmatrix} < A_1, B_1 >_D & < A_1, B_2 >_D & \ldots & < A_1, B_\ell >_D \\ < A_2, B_1 >_D & < A_2, B_2 >_D & \ldots & < A_2, B_\ell >_D \\ \vdots & \vdots & \vdots & \vdots \\ < A_p, B_1 >_D & < A_p, B_2 >_D & \ldots & < A_p, B_\ell >_D \end{pmatrix}.$$

If we set $D = I_{n\times n}$ then the new product $\diamondsuit_D$ reduces to the product $\diamondsuit$ which is defined in [3].

Remarks. (1) The matrix $A = [A_1, A_2, \ldots, A_p]$ is F-orthonormal if and only if $A^T \diamondsuit A = I_p$.

(2) If $X \in \mathbb{R}^{n\times s}$, then $X^T \diamondsuit X = \|X\|_F^2$.

(3) Let $A, B \in \mathbb{R}^{n\times s}$, $L \in \mathbb{R}^{p\times p}$ then $A^T \diamondsuit (B(L \otimes I_s)) = (A^T \diamondsuit B)L$.

Proposition 4. (see R. Bouyouli et al [3]) *Let $A \in \mathbb{R}^{n\times s}, B \in \mathbb{R}^{n\times ks}, C \in \mathbb{R}^{k\times p}, G \in \mathbb{R}^{k\times k}$ and $E \in \mathbb{R}^{n\times s}$. If the matrix G is nonsingular matrix then*

$$E^T \diamondsuit \left(\begin{pmatrix} A & B \\ C \otimes I_s & G \otimes I_s \end{pmatrix} \Big/ G \otimes I_s \right) = \begin{pmatrix} E^T \diamondsuit A & E^T \diamondsuit B \\ C & G \end{pmatrix} \Big/ G.$$

Corollary 1. *Let $A \in \mathbb{R}^{n\times s}, B \in \mathbb{R}^{n\times ks}, C \in \mathbb{R}^{k\times p}, G \in \mathbb{R}^{k\times k}$ and $E \in \mathbb{R}^{n\times s}$. If the matrix D is nonsingular matrix then*

$$E^T \diamondsuit_D \left(\begin{pmatrix} A & B \\ C \otimes I_s & G \otimes I_s \end{pmatrix} \Big/ G \otimes I_s \right) = \begin{pmatrix} E^T \diamondsuit_D A & E^T \diamondsuit_D B \\ C & G \end{pmatrix} \Big/ G.$$

Proof. Suppose that X, Y are arbitrary real matrices, clearly $X \diamondsuit_D Y = X \diamondsuit (DY)$. Hence, the result follows immediately from Proposition 4. □

Proposition 5. *Let $A = [A_1, A_2, \ldots, A_p]$ and $B = [B_1, B_2, \ldots, B_p]$ be matrices of dimension $n \times ps$, where A_i and B_j are $n \times s$ matrices and $D = diag\{k, k, \ldots, k\}$. Then*

$$\det[A^T \diamondsuit_D B] = k \det[A^T \diamondsuit B].$$

Proof. Suppose that

$$\mathcal{A}^v = [vec(A_1), vec(A_2), \ldots, vec(A_p)], \mathcal{B}^v = [vec(B_1), vec(B_2), \ldots, vec(B_p)].$$

It is easy to see that

$$A^T \diamondsuit_D B = (\mathcal{A}^v)^T (I_s \otimes D)\mathcal{B}^v.$$

As a result, we get

$$A^T \diamondsuit B = (\mathcal{A}^v)^T \mathcal{B}^v.$$

Now, we can conclude the result immediately. □

Let $V \in \mathbb{R}^{n\times s}$, and consider the block Krylov subspace

$$\mathcal{K}_m(A,V) = span\left\{V, AV, \ldots, A^{m-1}V\right\},$$

generated by the columns of the matrices $V, AV, \ldots, A^{m-1}V$. Note that $\mathcal{K}_m(A,V)$ is a subspace of $\mathbb{R}^n$. In fact $\mathcal{K}_m(A,V)$ associates to the direct sum of m simple Krylov subspaces $K_j(A, v_j)$ where $v_j, j = 1, 2, \ldots, s$, is the j-th column of the matrix V.

The global Arnoldi process constructs an F-orthonormal basis $V_1, V_2, \ldots, V_m$ of the matrix block Krylov $\mathcal{K}_m(A,V)$, i.e., the matrices $V_1, V_2, \ldots, V_m$ satisfy in the following conditions

$$tr(V_i^T V_j) = 0, \quad tr(V_i^T V_i) = 1, \quad \text{for} \;\; i \neq j, \; i, j = 1, 2, \ldots, m. \tag{6}$$

Algorithm 1. (Global Arnoldi Process)
1. Choose an $n \times s$ matrix V. Set $\beta = \|V\|_F$, $\quad V_1 = V/\beta$,
2. For $j = 1, 2, \ldots, m$ Do:
3. $W = AV_j$
4. For $i = 1, 2, \ldots, j$ Do:
5. $h_{i,j} =< W, V_i >_F$
6. $W = W - h_{ij}V_i$
7. End Do
8. $h_{j+1,j} = \|W\|_F$. If $h_{j+1,j} = 0$ Stop
9. $V_{j+1} = W/h_{j+1,j}$
10. End Do.

Denote by $\mathcal{V}_m$, the $n \times ms$ block matrix with columns $V_1, V_2, \ldots, V_m$, $\overline{H}_m$, the $(m+1) \times m$ Hessenberg matrix whose nonzero entries $h_{ij}, i = 1, 2, \ldots, m+1, j = 1, \cdots$, are defined by Algorithm 1, and by H_m, the matrix obtained from $\overline{H}_m$ by deleting its last row.

It is known that the matrices constructed by the global Arnoldi process satisfy the following relations, for more details see [18],

$$A\mathcal{V}_m = \mathcal{V}_m(H_m \otimes I_s) + h_{m+1,m}\left[\begin{array}{cccc} 0_{n\times s}, & 0_{n\times s}, & \ldots, & V_{m+1} \end{array}\right], \tag{7}$$

$$A\mathcal{V}_m = \mathcal{V}_{m+1}(\overline{H}_m \otimes I_s). \tag{8}$$

The G-FOM method generates a new approximation $X_m \in \mathcal{K}_m(A, R_0)$ such that

$$X_m = X_0 + \mathcal{V}_m(y_m \otimes I_s), \tag{9}$$

and

$$R_m = R_0 - A\mathcal{V}_m(y_m \otimes I_s), \tag{10}$$

where X_0 is a given initial approximate matrix to the solution X of $AX = B$ and $R_0 = B - AX_0$ is the corresponding residual matrix and $y_m \in \mathbb{R}^m$.

The G-FOM algorithm requires the storage of $\mathcal{V}_m$. That is, in order to save the vector $\mathcal{V}_m$, we need an m dimensional vectors space whose entries are $n \times s$ matrices. To cure the storage problem, encountered also in FOM, the value of m is limited by storage constraint and by avoiding rounding errors. Hence, this algorithm can be restarted after m iterations. The corresponding algorithm is called the restarted G-FOM (m), see [22].

Algorithm 2. (G-FOM (m))

1. Choose X_0, m and a tolerance ε, compute $R_0 = B - AX_0$,$V = R_0$.

2. Construct the F-orthonormal basis $V_1, V_2, \ldots, V_m$ by the global Arnoldi process.

3. Find y_m as the solution of

$$H_m y_m = \beta e_1.$$

4. Compute the approximate solution $X_m = X_0 + \mathcal{V}_m(y_m \otimes I_s)$, compute $R_m = B - AX_m$.

5. If $\|R_m\|_F < \varepsilon$ Stop.

6. Set $X_0 = X_m$, $R_0 = R_m$, $V = R_0$, and go to 2.

3. Weighted Global Arnoldi Process

By a given $n \times n$ positive definite diagonal matrix D, we will present the new weighted global Arnoldi process. It constructs a D-orthonormal basis for the block Krylov subspace $\mathcal{K}_m(A, V)$.

Algorithm 3. (Weighted Global Arnoldi Process)

1. Choose an $n \times s$ matrix V. Set $\widetilde{\beta} = \|V\|_D$, $\widetilde{V}_1 = V/\widetilde{\beta}$,
2. For $j = 1, 2, \ldots, m$ Do:
3. $\widetilde{W} = A\widetilde{V}_j$
4. For $i = 1, 2, \ldots, j$ Do:
5. $\widetilde{h}_{i,j} =< \widetilde{W}, \widetilde{V}_i >_D$
6. $\widetilde{W} = \widetilde{W} - \widetilde{h}_{ij}\widetilde{V}_i$
7. End Do
8. $\widetilde{h}_{j+1,j} = \|\widetilde{W}\|_D$. If $\widetilde{h}_{j+1,j} = 0$ Stop
9. $\widetilde{V}_{j+1} = \widetilde{W}/\widetilde{h}_{j+1,j}$
10. End Do.

Note that if we set $D = I_n$, the weighted global Arnoldi reduces to the global Arnoldi process. In the rest of this section, we assume that Algorithms 1 and 3 do not break down before the m-th step. Suppose that $\widetilde{\mathcal{V}}_m$ denotes the $n \times ms$ matrix defined by $\widetilde{\mathcal{V}}_m = \left[\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_m\right]$. $\widetilde{\overline{H}}_m$ denotes the $(m+1) \times m$ Hessenberg matrix

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

with nonzero entries $\widetilde{h}_{ij}$ defined by Algorithm 3, and by $\widetilde{H}_m$, the matrix obtained from $\widetilde{\overline{H}}_m$ by removing its last row.

It is easy to see that $\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_m$, constructed by Algorithm 3, form a D-orthonormal basis for $\mathcal{K}_m(A, V)$, i.e.

$$tr((\widetilde{V}_i)^T D\widetilde{V}_j) = 0, \qquad i \neq j, \; i, j = 1, 2, \ldots, m, \tag{11}$$

$$tr((\widetilde{V}_i)^T D\widetilde{V}_i) = 1, \qquad i = 1, 2, \ldots, m. \tag{12}$$

The following theorem shows that relations (7) and (8) can be extended for the corresponding matrices obtained by weighted global Arnoldi process.

Theorem 1. *Let $\widetilde{\mathcal{V}}_m$, $\widetilde{H}_m$, and $\widetilde{\overline{H}}_m$ defined as before. Then the following relations hold*

$$A\widetilde{\mathcal{V}}_m = \widetilde{\mathcal{V}}_m(\widetilde{H}_m \otimes I_s) + [0_{n\times s}, \ldots, 0_{n\times s}, \widetilde{h}_{m+1,m}\widetilde{V}_{m+1}], \tag{13}$$

$$A\widetilde{\mathcal{V}}_m = \widetilde{\mathcal{V}}_m(\widetilde{H}_m \otimes I_s) + \widetilde{h}_{m+1,m}\widetilde{V}_{m+1}(e_m^T \otimes I_s), \tag{14}$$

$$A\widetilde{\mathcal{V}}_m = \widetilde{\mathcal{V}}_{m+1}(\widetilde{\overline{H}}_m \otimes I_s), \tag{15}$$

where $e_m^T = \begin{bmatrix} 0 & \ldots & 0 & 1 \end{bmatrix}_{1\times m}$.

Proof. The relation (13) follows from lines 3, 6 and 9 of Algorithm 3. The relations (14) and (15) are reformulation of (13). □

In the following, some relations between matrices generated by two processes, namely global Arnoldi and weighted global Arnoldi processes are given. Now, assume that $\mathcal{V}_m = [V_1, V_2, \ldots, V_m]$ and $\widetilde{\mathcal{V}}_m = \left[\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_m\right]$ are $n \times ms$ matrices corresponding to the two bases for $\mathcal{K}_m(A, V)$ constructed by global Arnoldi and weighted global Arnoldi processes, respectively. By considering the matrices $\mathcal{V}_m^v$ and $\widetilde{\mathcal{V}}_m^v$ consist the matrices $\mathcal{V}_m$ and $\widetilde{\mathcal{V}}_m$, we prove the following theorem.

Theorem 2. *Assume that $\mathcal{V}_m^v$ and $\widetilde{\mathcal{V}}_m^v$ are defined as before. Then:*

(1) The columns of $\mathcal{V}_m^v$ form basis for $\mathcal{K}_m((I_s \otimes A), vec(V))$ and

$$(\mathcal{V}_m^v)^T \mathcal{V}_m^v = I. \tag{16}$$

(2) The columns of $\widetilde{\mathcal{V}}_m^v$ form basis for $\mathcal{K}_m((I_s \otimes A), vec(V))$ and

$$(\widetilde{\mathcal{V}}_m^v)^T (I_s \otimes D)\widetilde{\mathcal{V}}_m^v = I. \tag{17}$$

Proof. It is known that the set of the columns of $\mathcal{V}_m(\widetilde{\mathcal{V}}_m)$ is a basis for $\mathcal{K}_m(A, V)$. Also, by the properties of Kronecker product, we can easily see that for $i = 1, 2, 3, \ldots,$

$$(I_s \otimes A)^i vec(V) = vec(A^i V). \tag{18}$$

Hence, we conclude that $\mathcal{V}_m^v$ ($\widetilde{\mathcal{V}}_m^v$) forms a basis for $\mathcal{K}_m((I_s \otimes A), vec(V))$.

The relations (16) and (17) follow from (6), (11), (12) and the following relations

$$tr(V_i^T V_j) = vec(V_i)^T vec(V_j),$$
$$tr(\widetilde{V}_i^T D V_j) = vec(\widetilde{V}_i)^T (I_s \otimes D) vec(\widetilde{V}_j). \qquad \square$$

Now, we establish the following useful proposition.

Proposition 6. *Consider $\mathcal{V}_m$ as defined earlier and assume that $B_m = [b_{ij}]$ is an arbitrary $m \times p$ matrix. If*

$$\mathcal{V}_m(B \otimes I_s) = 0_{n\times ps}, \tag{19}$$

then $B_m = 0_{m\times p}$.

Proof. Let b_j, for each j, $1 \le j \le p$, be the j-th column of the B_m. Evidently, (19) implies that $\mathcal{V}_m(b_j \otimes I_s) = 0_{n\times s}$, i.e.

$$\sum_{i=1}^{m} V_i b_{ij} = 0_{n\times s}, \qquad 1 \le j \le p,$$

which is equivalent to say that

$$\sum_{i=1}^{m} vec(V_i) b_{ij} = 0_{ns\times 1}, \quad 1 \le j \le p.$$

From the above relation, we can easily see that $\mathcal{V}_m^v b_j = 0_{ns\times 1} (1 \le j \le p)$, which implies that $\mathcal{V}_m^v B = 0_{ns\times p}$. Now, the result follows from the relation (16). $\square$

Now, we give some relations between matrices obtained by weighted global Arnoldi and global Arnoldi processes. These relations will be useful for presenting links between G-FOM and new WG-FOM algorithms. To this end, first, we establish the following theorem.

Theorem 3. *Suppose that Algorithms 1 and 3 do not break down before the m-th step. Then there exists an upper nonsingular triangular matrix $U_m \in \mathbb{R}^{m\times m}$such that*

$$\widetilde{\mathcal{V}}_m = \mathcal{V}_m(U_m \otimes I_s), \tag{20}$$
$$U_m = (\mathcal{V}_m^v)^T \widetilde{\mathcal{V}}_m^v, \tag{21}$$
$$U_m^{-1} = (\widetilde{\mathcal{V}}_m^v)^T (I_s \otimes D) \mathcal{V}_m^v, \tag{22}$$
$$\widetilde{\overline{H}}_m = U_{m+1}^{-1} \overline{H}_m U_m. \tag{23}$$

Proof. Since $\mathcal{V}_j^v$ and $\widetilde{\mathcal{V}}_j^v$ are two bases of Krylov subspace $\mathcal{K}_j((I_s \otimes A), vec(V))$ for all $j \in \{1, 2, \ldots, m\}$, we can express $\widetilde{\mathcal{V}}_m^v$ in terms of$\mathcal{V}_m^v$ as

$$\widetilde{\mathcal{V}}_m^v = \mathcal{V}_m^v U_m.$$

It is easy to see that the above relation equivalent to the relation (20). If we multiply the above relation on the left by $(\mathcal{V}_m^v)^T$, we get (21). Also, for obtaining (22), it is sufficient to multiply $\widetilde{\mathcal{V}}_m^v = \mathcal{V}_m^v U_m$ on the left by $(\widetilde{\mathcal{V}}_m^v)^T(I_s \otimes D)$. Using (8), (15) and the properties of the Kronecker product, we get

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

Or equivalently

$$\mathcal{V}_{m+1}((\overline{H}_m U_m - U_{m+1}\widetilde{\overline{H}}_m) \otimes I_s) = 0.$$

Now, the relation (23) follows from Proposition 6 by setting $B_m = \overline{H}_m U_m - U_{m+1}\widetilde{\overline{H}}_m$. □

The relation (23) is also valid for matrices $\overline{H}_m$ and $\widetilde{\overline{H}}_m$, constructed by the Arnoldi and weighted Arnoldi processes, respectively, for more details see [14]. Based on the relation (23) for the Hessenberg matrices produced by the Arnoldi and weighted Arnoldi processes, the following proposition established by Essai in [14]. Hence, we can immediately conclude that this proposition is also valid for matrices $\overline{H}_m$ and $\widetilde{\overline{H}}_m$, produced by the global Arnoldi and weighted global Arnoldi processes. We state the following proposition, for Hessenberg matrices constructed by global processes. Clearly, the proof is similar to the proof given in [14].

Proposition 7. *Under the same assumption as in Theorem 3, we can find the following relations between H_m and $\widetilde{H}_m$:*

$$\widetilde{H}_m = U_m^{-1} H_m U_m + h_{m+1,m} u_{m,m} \overline{u}_{m+1} e_m^T, \tag{24}$$

$$H_m = U_m^{-1} \widetilde{H}_m U_m + \frac{h_{m+1,m}}{u_{m+1,m+1}} u_{m+1} e_m^T, \tag{25}$$

where $\overline{u}_{m+1} \in \mathbb{R}^m$and $u_{m+1} \in \mathbb{R}^m$ are respectively obtained from the last columns of the matrices $U_{m+1}^{-1}$and U_{m+1}, where last entries are deleted.

Proof. See [14]. □

4. The New Method WG-FOM

In this section, we present new method called weighted global FOM (WG-FOM). Let X_0 be an $n \times s$ initial approximate matrix with the corresponding residual matrix $R_0 = B - AX_0$. Like G-FOM the approximation $\widetilde{X}_m$ in WG-FOM belongs to the Krylov subspace $\mathcal{K}_m(A, R_0)$. Hence, the WG-FOM method generates a new approximation $\widetilde{X}_m \in \mathcal{K}_m(A, R_0)$ such that

$$\widetilde{X}_m = X_0 + \widetilde{\mathcal{V}}_m(y_m^{WF} \otimes I_s), \tag{26}$$

and

$$\widetilde{R}_m = R_0 - A\widetilde{\mathcal{V}}_m(y_m^{WF} \otimes I_s). \tag{27}$$

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

For WG-FOM method, the approximation $\widetilde{X}_m$ is chosen such that the corresponding residual $\widetilde{R}_m = B - A\widetilde{X}_m$ satisfies in the following Petrov-Galerkin condition

$$\widetilde{R}_m \perp_D \mathcal{K}_m(A, R_0). \tag{28}$$

Algorithm 4. (WG-FOM (m))

1. Choose X_0, m and a tolerance ε, compute $R_0 = B - AX_0, V = R_0$.
2. Choose a $n \times n$ diagonal positive definite matrix D.
3. Construct the D-orthonormal basis $\widetilde{V}_1, \widetilde{V}_2, \ldots, \widetilde{V}_m$ by the global Arnoldi process.
4. Find $\widetilde{y}_m$ as the solution of

$$\widetilde{H}_m y_m^{WF} = \widetilde{\beta} e_1.$$

5. Compute the approximate solution $\widetilde{X}_m = X_0 + \widetilde{\mathcal{V}}_m(y_m^{WF} \otimes I_s)$, compute $\widetilde{R}_m = B - A\widetilde{X}_m$.
6. If $\left\|\widetilde{R}_m\right\|_F < \varepsilon$ Stop.
7. Set $X_0 = \widetilde{X}_m$, $R_0 = \widetilde{R}_m$, $V = R_0$, then go to 2.

Note that if we set $D = I_n$ for each restart, in Step 2, then the WG-FOM method reduces to the G-FOM. In Section 5, we will give a suggestion to choose weights in each step. The idea of choosing weights, by the relation (43), has been gotten from [14].

Now, we present a link between G-FOM and WG-FOM. Note that, $\widetilde{X}_m - X_0 \in \mathcal{K}_m(A, R_0)$, so we have

$$\begin{aligned} \widetilde{X}_m - X_0 &= \widetilde{\mathcal{V}}_m(y_m^{WF} \otimes I_s), \\ &= \widetilde{\mathcal{V}}_m(U_m^{-1} \otimes I_s)(U_m y_m^{WF} \otimes I_s), \\ &= \mathcal{V}_m(\widetilde{y}_m^{WF} \otimes I_s), \end{aligned}$$

where $y_m^{WF} = U_m^{-1}\widetilde{y}_m^{WF}$. It is known that, the vector y_m^{WF} is the solution of the following linear system,

$$\widetilde{H}_m y_m^{WF} = \widetilde{\beta} e_1.$$

We recall that $\widetilde{\beta} = \|R_0\|_D$.

By some easy computations on the above linear system, we get

$$U_m \widetilde{H}_m U_m^{-1} \widetilde{y}_m^{WF} = U_m \widetilde{\beta} e_1 = \beta e_1.$$

Note that $\beta = \|R_0\|_F$ and $\widetilde{y}_m^{WF} = U_m y_m^{WF}$.

From the relation (25), we deduce that

$$(H_m - \frac{h_{m+1,m}}{u_{m+1,m+1}} u_{m+1} e_m^T)\widetilde{y}_m^{WF} = \beta e_1. \tag{29}$$

By considering the properties of the basis $\mathcal{V}_m$, we can easily see that the nonzero entries of the matrix $U_{m+1} = [u_{ij}]_{m+1\times m+1}$, are obtained as follows

$$u_{ij} = tr(V_i^T \widetilde{V}_j), \quad j = 1, 2, \ldots, m+1, i = 1, 2, \ldots, j.$$

From (29) and the above relation we conclude that the knowledge of $\widetilde{V}_{m+1}$ allows us to construct the iterate $\widetilde{X}_m$ from the global Arnoldi process.

Proposition 8. *Suppose that* $\widetilde{X}_m \in X_0 + \mathcal{K}_m(A, R_0)$, *is the approximate solution computed by WG-FOM algorithm with the corresponding residual matrix* $\widetilde{R}_m$. *Then*

$$\widetilde{R}_m = -\widetilde{h}_{m+1,m}\widetilde{V}_{m+1}(e_m^T y_m^{WF} \otimes I_s).$$

Proof. It is well known that,

$$\widetilde{R}_m = R_0 - A\widetilde{\mathcal{V}}_m(y_m^{WF} \otimes I_s).$$

By substituting the relation (14) in the above relation, we get

$$\begin{aligned}\widetilde{R}_m = R_0 - \widetilde{\mathcal{V}}_m(\widetilde{H}y_m^{WF} \otimes I_s) - \widetilde{h}_{m+1,m}\widetilde{V}_{m+1}(e_m^T y_m^{WF} \otimes I_s) \\ = \widetilde{\beta}\widetilde{V}_1 - \widetilde{\mathcal{V}}_m(\widetilde{\beta}e_1 \otimes I_s) - \widetilde{h}_{m+1,m}\widetilde{V}_{m+1}(e_m^T y_m^{WF} \otimes I_s).\end{aligned}$$

Obviously, $\widetilde{\mathcal{V}}_m(\widetilde{\beta}e_1 \otimes I_s) = \widetilde{\beta}\widetilde{V}_1$. Now, the result follows immediately. □

It is clear that G-FOM is a special case of the WG-FOM algorithm, so we can conclude the following corollary.

Corollary 2. *Suppose that* $X_m \in X_0 + \mathcal{K}_m(A, R_0)$ *is the approximate solution computed by G-FOM algorithm with the corresponding residual matrix* R_m. *Then*

$$R_m = -h_{m+1,m}V_{m+1}(e_m^T y_m \otimes I_s).$$

In the following, first we will give a new expression for D-norm of the residual matrix $\widetilde{R}_m$. Then, for the case that all the weights are equal, we will obtain a relation between the Frobenius norm of the m-th residual matrices corresponding to the G-FOM and WG-FOM.

From Petrov-Galerkin condition, we deduce that

$$\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_m = 0.$$

The above relation and the property of the $\diamondsuit_D$ product imply that

$$\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0 = \widetilde{\mathcal{V}}_m^T \diamondsuit_D ((A\widetilde{\mathcal{V}}_m)(y_m^{WF} \otimes I_s)) = (\widetilde{\mathcal{V}}_m^T \diamondsuit_D A\widetilde{\mathcal{V}}_m)y_m^{WF}.$$

For simplicity, we set $\widetilde{\mathcal{W}}_m = A\widetilde{\mathcal{V}}_m$. Hence, we have the following results

$$\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0 = (\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m)y_m^{WF}. \quad (30)$$

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

Theorem 3. *Suppose that $\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m$, is a nonsingular matrix. Then the approximate solution $\widetilde{X}_m$ and the corresponding residual matrix $\widetilde{R}_m$ can be expressed by the following Schur complements*

$$\widetilde{X}_m = \begin{pmatrix} X_0 & -\widetilde{\mathcal{V}}_m \\ (\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0) \otimes I_S & (\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m) \otimes I_s \end{pmatrix} \Big/ ((\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m) \otimes I_s), \tag{31}$$

and, therefore,

$$\widetilde{R}_m = \begin{pmatrix} R_0 & \widetilde{\mathcal{W}}_m \\ (\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0) \otimes I_s & (\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m) \otimes I_s \end{pmatrix} \Big/ ((\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m) \otimes I_s), \tag{32}$$

where $\widetilde{R}_m = B - A\widetilde{X}_m$.

Proof. From (28), we have

$$\widetilde{X}_m = X_0 + \widetilde{\mathcal{V}}_m \left[[(\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m)^{-1} (\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0)] \otimes I_s \right].$$

By using the properties of the Kronecker product, we get

$$\begin{aligned} \widetilde{X}_m &= X_0 + \widetilde{\mathcal{V}}_m \left[[(\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m)^{-1} \otimes I_s][(\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0) \otimes I_s] \right] \\ &= X_0 + \widetilde{\mathcal{V}}_m [(\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m) \otimes I_s]^{-1} [(\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0) \otimes I_s]. \end{aligned}$$

From the above relation, we can easily conclude the results. □

Now, by using Theorem 3, we will obtain an expression for the D-norm of the residual matrix corresponding to the approximate solution $\widetilde{X}_m$.

Theorem 4. *Suppose that $\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m$, is a nonsingular matrix. The D-norm of the residual matrix $\widetilde{R}_m$, is given by*

$$\left\| \widetilde{R}_m \right\|_D^2 = \widetilde{h}_{m+1,m} \left\| R_0 \right\|_D \frac{\det[\widetilde{\mathcal{V}}_{m+1}^T \diamondsuit_D \widetilde{\overline{\mathcal{V}}}_{m+1}] \det[\widetilde{\overline{W}}_m]}{(\det[\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m])^2}, \tag{33}$$

where $\widetilde{\overline{\mathcal{V}}}_{m+1} = [\widetilde{V}_1, \widetilde{\mathcal{W}}_m]$ and $\widetilde{\overline{W}}_m = [\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0, \widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_{m-1}] = \widetilde{\mathcal{V}}_m^T \diamondsuit_D [R_0, \widetilde{\mathcal{W}}_{m-1}]$.

Proof. It is obvious that

$$\left\| \widetilde{R}_m \right\|_D^2 = \widetilde{R}_m^T \diamondsuit_D \widetilde{R}_m.$$

From Proposition 8, we conclude that

$$\begin{aligned} \widetilde{R}_m^T \diamondsuit_D \widetilde{R}_m &= (-\widetilde{h}_{m+1,m} (e_m^T y_m^{WF} \otimes I_s)^T \widetilde{V}_{m+1}^T) \diamondsuit_D \widetilde{R}_m \\ &= -\widetilde{h}_{m+1,m} (y_m^{WF})^{(m)} (\widetilde{V}_{m+1}^T \diamondsuit_D \widetilde{R}_m), \end{aligned} \tag{34}$$

where $(y_m^{WF})^{(m)}$ denotes the last component of the y_m^{WF}.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

Let us first compute $(\widetilde{V}_{m+1}^T \lozenge_D \widetilde{R}_m)$. By using the relation (32) and Corollary 1, we have

$$\widetilde{V}_{m+1}^T \lozenge_D \widetilde{R}_m = \begin{pmatrix} \widetilde{V}_{m+1}^T \lozenge_D R_0 & \widetilde{V}_{m+1}^T \lozenge_D \widetilde{\mathcal{W}}_m \\ \widetilde{\mathcal{V}}_m^T \lozenge_D R_0 & \widetilde{\mathcal{V}}_m^T \lozenge_D \widetilde{\mathcal{W}}_m \end{pmatrix} \Big/ (\widetilde{\mathcal{V}}_m^T \lozenge_D \widetilde{\mathcal{W}}_m).$$

Now, by using Proposition 1, we have

$$\widetilde{V}_{m+1}^T \lozenge_D \widetilde{R}_m = \|R_0\|_D \left[\begin{pmatrix} \widetilde{\mathcal{V}}_m^T \lozenge_D \widetilde{V}_1 & \widetilde{\mathcal{V}}_m^T \lozenge_D \widetilde{\mathcal{W}}_m \\ \widetilde{V}_{m+1}^T \lozenge_D \widetilde{V}_1 & \widetilde{V}_{m+1}^T \lozenge_D \widetilde{\mathcal{W}}_m \end{pmatrix} \Big/ (\widetilde{\mathcal{V}}_m^T \lozenge_D \widetilde{\mathcal{W}}_m) \right]. \qquad (35)$$

It is known that $\widetilde{\mathcal{V}}_{m+1} = [\widetilde{\mathcal{V}}_m, \widetilde{V}_{m+1}]$. By setting $\overline{\widetilde{\mathcal{V}}}_{m+1} = [\widetilde{V}_1, \widetilde{\mathcal{W}}_m]$, (35) can be rewrite as follows

$$\widetilde{V}_{m+1}^T \lozenge_D \widetilde{R}_m = \|R_0\|_D \left[(\widetilde{\mathcal{V}}_{m+1}^T \lozenge_D \overline{\widetilde{\mathcal{V}}}_{m+1}) \Big/ (\widetilde{\mathcal{V}}_m^T \lozenge_D \widetilde{\mathcal{W}}_m) \right].$$

$\widetilde{V}_{m+1}^T \lozenge_D \widetilde{R}_m$ is a scalar, therefore it follows that

$$\widetilde{V}_{m+1}^T \lozenge_D \widetilde{R}_m = (-1)^m \|R_0\|_D \frac{\det[\widetilde{\mathcal{V}}_{m+1}^T \lozenge_D \overline{\widetilde{\mathcal{V}}}_{m+1}]}{\det[\widetilde{\mathcal{V}}_m^T \lozenge_D \widetilde{\mathcal{W}}_m]}. \qquad (36)$$

On the other hand, $(y_m^{WF})^{(m)}$ can be computed from (30) by Cramer rule, as

$$(y_m^{WF})^{(m)} = (-1)^{m-1} \frac{\det[\widetilde{\overline{W}}_m]}{\det[\widetilde{\mathcal{V}}_m^T \lozenge_D \widetilde{\mathcal{W}}_m]}. \qquad (37)$$

Now, by substituting (36) and (37) in (34), the result follows immediately. □

As we have adumbrated earlier, G-FOM algorithm can be considered as a special case of the WG-FOM algorithm by setting $D = I_n$, where I_n denotes the $n \times n$ identity matrix. When $D = I_n$, D-norm equivalent to the well known Frobenius norm. Hence, we can state the above theorem for the residual matrix corresponding to the approximate solution obtained by G-FOM algorithm.

Corollary 3. *Suppose that $\mathcal{V}_m^T \lozenge \mathcal{W}_m$ is a nonsingular matrix. The Frobenius norm of the residual matrix R_m is given by*

$$\|R_m\|_F^2 = h_{m+1,m} \|R_0\|_F \frac{\det[\mathcal{V}_{m+1}^T \lozenge \overline{\mathcal{V}}_{m+1}] \det[\overline{W}_m]}{(\det[\mathcal{V}_m^T \lozenge \mathcal{W}_m])^2}, \qquad (38)$$

where $\overline{\mathcal{V}}_{m+1} = [V_1, \mathcal{W}_m]$ and $\overline{W}_m = [\mathcal{V}_m^T \lozenge R_0, \mathcal{V}_m^T \lozenge \mathcal{W}_{m-1}] = \mathcal{V}_m^T \lozenge [R_0, \mathcal{W}_{m-1}]$.

Now, we give a relation between the F-norm of the residual matrices $\widetilde{R}_m$ and R_m.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

Theorem 5. *Suppose that* $\mathcal{V}_m^T \diamondsuit \mathcal{W}_m$, $\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m$ *are nonsingular matrices and* $D = diag\{d, d, \ldots, d\}$. *Then*

$$\left\|\widetilde{R}_m\right\|_F^2 = \frac{\alpha}{d} \left\|R_m\right\|_F^2, \tag{39}$$

where $\alpha = \frac{\widetilde{h}_{m+1,m}}{h_{m+1,m}} \times \frac{u_{m+1,m+1}}{u_{m,m}}$.

Proof. It is known that $\widetilde{\widetilde{W}}_m = [\widetilde{\mathcal{V}}_m^T \diamondsuit_D R_0, \widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_{m-1}]$. Using (20), we get

$$\widetilde{\widetilde{W}}_m = [((U_m^T \otimes I_s)\mathcal{V}_m^T) \diamondsuit_D R_0, ((U_m^T \otimes I_s)\mathcal{V}_m^T) \diamondsuit_D (\mathcal{W}_{m-1}(U_{m-1} \otimes I_s))].$$

By using the properties of $\diamondsuit_D$ product, we have

$$\begin{aligned} \widetilde{\widetilde{W}}_m &= U_m^T[\mathcal{V}_m^T \diamondsuit_D R_0, (\mathcal{V}_m^T \diamondsuit_D \mathcal{W}_{m-1})U_{m-1}] \\ &= U_m^T(\mathcal{V}_m^T \diamondsuit_D ([R_0, \mathcal{W}_{m-1}]))\overline{U}_m, \end{aligned}$$

where $\overline{U}_m = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & U_{m-1} & \\ 0 & & & \end{pmatrix}_{m \times m}$.

Now, from Proposition 5, we conclude that

$$\begin{aligned} \det[\widetilde{\widetilde{W}}_m] &= d(\textstyle\prod_{i=1}^m u_{ii})(\prod_{i=1}^{m-1} u_{ii}) \det[\mathcal{V}_m^T \diamondsuit R_0, \mathcal{V}_m^T \diamondsuit \mathcal{W}_{m-1}] \\ &= d(\textstyle\prod_{i=1}^m u_{ii})(\prod_{i=1}^{m-1} u_{ii}) \det[\overline{W}_m]. \end{aligned} \tag{40}$$

On the other hand $\widetilde{\widetilde{\mathcal{V}}}_{m+1} = [\widetilde{V}_1, \widetilde{\mathcal{W}}_m]$, hence (20) implies that $\widetilde{\widetilde{\mathcal{V}}}_{m+1} = \overline{\mathcal{V}}_{m+1}(\overline{\overline{U}}_m \otimes I_S)$, where

$$\overline{\overline{U}}_m = \begin{pmatrix} u_{11} & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & U_m & \\ 0 & & & \end{pmatrix}_{m+1 \times m+1}.$$

Hence, we can easily see that

$$\det[\widetilde{\mathcal{V}}_{m+1}^T \diamondsuit_D \widetilde{\widetilde{\mathcal{V}}}_{m+1}] = du_{11}(\prod_{i=1}^{m+1} u_{ii})(\prod_{i=1}^{m} u_{ii}) \det[\mathcal{V}_{m+1}^T \diamondsuit \overline{\mathcal{V}}_{m+1}]. \tag{41}$$

From (20) and the properties of $\diamondsuit_D$ product, we get

$$\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m = U_m^T(\mathcal{V}_m^T \diamondsuit_D \mathcal{W}_m)U_m,$$

therefore,

$$\det[\widetilde{\mathcal{V}}_m^T \diamondsuit_D \widetilde{\mathcal{W}}_m] = d(\prod_{i=1}^{m} u_{ii})^2 \det[\mathcal{V}_m^T \diamondsuit \mathcal{W}_m]. \quad (42)$$

By substituting (40), (41), (42) and $u_{11} = \frac{\|R_0\|_F}{\|R_0\|_D}$ in (33), the result follows immediately. □

5. Numerical Experiments

In this section, we will present some numerical results to compare G-FOM algorithm with compose algorithm. The experiments were performed by *Mathematica 6*.

The matrix $D = diag\{d_1, d_2, \ldots, d_n\}$ is chosen such that

$$d_i = \sqrt{n}\frac{\left\|R_0^{i,*}\right\|_2}{\|R_0\|_F}, \quad i = 1, 2, \ldots, n. \quad (43)$$

$R_0^{i,*}, i = 1, 2, \ldots, n$, denotes the i-th row of the matrix R_0. In the following example, this choice will always be applied. The idea of choosing d_i, by the relation (43), is gotten from [14], i.e. it can be assumed as a generalized form of the weights chosen by the author in [14].

In the following examples, the matrix $B \in \mathbb{R}^{n\times 2}$, is chosen such that $AE = B$ where

$$E^T = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ 1.5 & 1.5 & \ldots & 1.5 \end{pmatrix}_{2\times n}.$$

Also, the initial guess is $X_0 = 0_{n\times 2}$.

The iteration process is terminated when the residual matrix is sufficiently small and the termination test, i.e.,

$$\frac{\|R_m\|_F}{\|R_0\|_F} \leq \varepsilon.$$

Now, the numerical comparison results for G-FOM and WG-FOM are given by two different way. The relation between the number of restarts as x-axis and the relative residual's logarithm base on 10 as y-axis are presented in the form of figures. Also, we give the comparison results in terms of both number of restarts and CPU-time(s) in Tables 1 and 2 for the corresponding experiments.

Example 1. (see [16, 27, 28]) This example considers an 100×100 upper bidiagonal matrix A, where the vector $k = [0.001, 0.002, 0.003, 0.004, 10, 11, 12, \ldots, 105]$ consists of the diagonal elements of the matrix A and supper-diagonal is the vector which all elements of equal to one ($\varepsilon = 0.5 \times 10^{-10}$).

The results of applying G-FOM (40) and WG-FOM (40) are shown in Table 1. The symbol "*" in Table 1 means the convergence is not reached. During the

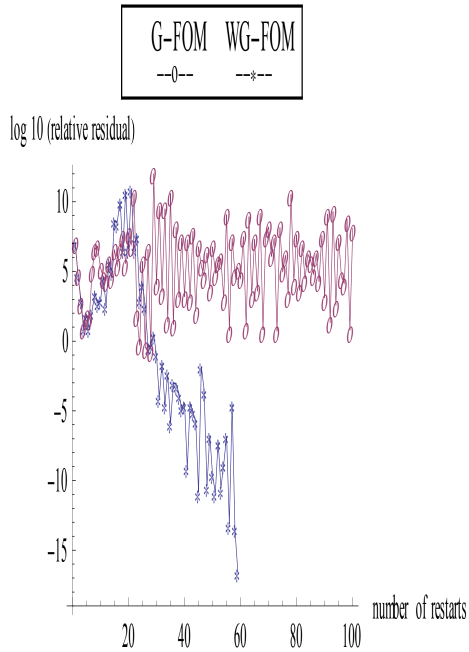

Figure 1: Example 1

performing Algorithm 2, in some steps, H_m becomes an ill-condition matrix which is caused significant numerical errors. Also, convergence doesn't reach even by increasing m. On the other hand, as the table 1 shows, our new new weighted algorithm is convergent. In Figure 1, we show the 100 steps of the G-FOM.

Example 1. Comparison in terms of both number of restarts and CPU consuming time in seconds.

Approach	G-FOM (40)	WG-FOM (40)
Iteration	*	59
CPU- time	*	6.973

Table 1

Example 2. (see [16, 27, 28]) Let the matrix A be a 200×200 matrix, employed in the corresponding literature, as follows ($\varepsilon = 0.5 \times 10^{-12}$)

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 2, No. 2 (2010)

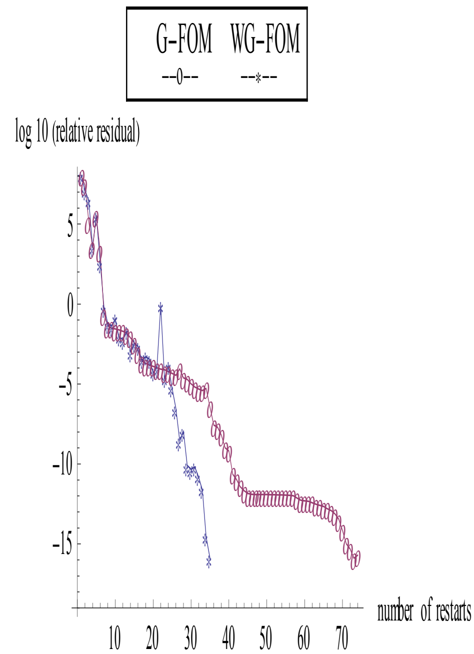

Figure 2: Example 2

$$A = \begin{pmatrix} \frac{1}{4} & 0.21 & 1.2 & 0 & 0.13 & 1.42 & & & & \\ 0.45 & \frac{2}{4} & 0.21 & 1.2 & 0 & 0.13 & 1.42 & & & \\ 0 & 0.45 & \frac{3}{4} & 0.21 & 1.2 & 0 & 0.13 & & & \\ 0.12 & 0 & 0.45 & \frac{4}{4} & 0.21 & & & & & \\ 0.11 & 0.12 & 0 & 0.45 & \ddots & \ddots & & & 0.13 & 1.42 \\ & 0.11 & 0.12 & 0 & \ddots & \ddots & \ddots & 1.2 & 0 & 0.13 \\ & & & & & \ddots & \frac{197}{4} & 0.12 & 1.2 & 0 \\ & & & & & 0 & 0.45 & \frac{198}{4} & 0.21 & 1.2 \\ & & & & 0.11 & 0.12 & 0 & 0.45 & \frac{199}{4} & 0.21 \\ & & & & & 0.11 & 0.12 & 0 & 0.45 & \frac{200}{4} \end{pmatrix}.$$

Example 2. Comparison in terms of both number of restarts and CPU consuming time in seconds.

Remark. As observed in Tables 1 and 2 and the figures significant progress has been made by WG-FOM not only in reduced number of restarts but also in decreased CPU computing time. Under such circumstances where G-FOM needs much more number of restarts to converge than WG-FOM, WG-FOM is much preferred.

Approach	G-FOM (40)	WG-FOM (40)
Iteration	121	55
CPU- time	65.348	31.917

Table 2

6. Conclusion

In this paper, we presented new weighted global FOM method. Moreover, using Schur complement formulae, we derived new expressions of the approximations and corresponding residual norms. In a special case, a relation between residual norms corresponding to the G-FOM and WG-FOM was obtained.

The open problem of finding optimal weighting matrix D is under investigation.

Acknowledgments

This work has been partially supported by the Mahani Mathematical Research Center, Shahid Bahonar University of Kerman.

References

[1] T. Ando, Generalized Schur complements, *Linear Algebra Appl.*, **27** (1979), 173-186.

[2] R. Bouyouli, K. Jbilou, R. Sadaka, H. Sadok, Convergence properties of some block Krylov subspace methods for multiple linear systems, *J. Comput. Appl. Math.*, **196** (2006), 498-511.

[3] R. Bouyouli, K. Jbilou, A. Messaoudi, H. Sadok, On block minimal residual methods, *Appl. Math. Lett.*, **20** (2007), 284-289.

[4] C. Brezinski, Other manifestations of the Schur complement, *Linear Algebra Appl.*, **111** (1988), 231-247.

[5] C. Brezinski, M. Redivo Zaglia, A Schur complement approach to a general extrapolation algorithm, *Linear Algebra Appl.*, **368** (2003), 279-301.

[6] R.A. Brualdi, H. Schneider, Determinantal identities: Gauss, Schur, Cauchy, Sylvester, Kronecker, Jacobi, Binet, Laplace, Muir and Cayley, *Linear Algebra Appl.*, **52-53** (1983), 769-791.

[7] Z.H. Cao, X.Y. Yu, A note on weighted FOM and GMRES for solving nonsymmetric linear systems, *Appl. Math. Comput.*, **151** (2004), 719-727.

[8] D. Carlson, What are Schur complements, anyway? *Linear Algebra Appl.*, **74** (1986), 257-275.

[9] D. Carlson, E.V. Haynsworth, T. Markham, A generalization of the Schur complement by means of the Moore-Penrose inverse, *SIAM J. Appl. Math.*, **26** (1974), 169-175.

[10] G. Corach, A. Maestripieri, D. Stojanoff, Generalized Schur complements and oblique projections, *Linear Algebra Appl.*, **341** (2002), 259-272.

[11] R.W. Cottle, Manifestations of the Schur complement, *Linear Algebra Appl.*, **8** (1974), 189-211.

[12] D.E. Crabtree, E.V. Haynsworth, An identity for the Schur complement of a matrix, *Proc. Amer. Math. Soc.*, **22** (1969), 364-366.

[13] A. El Guennouni, K. Jbilou, H. Sadok, A block version of BiCGSTAB for linear systems with multiple right-hand sides, *Elect. Trans. Numer. Anal.*, **16** (2003), 129-142.

[14] A. Essai, Weighted FOM and GMRES for solving nonsymmetric linear systems, *Numer. Algorithms*, **18** (1998), 277-292.

[15] K. Jbilou, A. Messaoudi, H. Sadok, Global FOM et GMRES algorithms for matrix equations, *Appl. Numer. Math.*, **31** (1999), 43-49.

[16] Y.-F. Jing, T.-Z. Huang, Restarted weighted full orthogonalization method for shifted linear systems, *Computers and Mathematics with Application*, **57** (2009), 1583-1591.

[17] P. Lancaster, *Theory of Matrix*, Academic Press, New York (1969).

[18] Y.-Q. Lin, Implicitly restarted global FOM and GMRES for nonsymmetric matrix equations and Sylvester equations, *Appl. Math. Comput.*, **167** (2005), 1004-1025.

[19] G. Marsaglia, G.P.H. Styan, Equalities and inequalities for ranks of matrices, *Linear Multilinear Algebra*, **2** (1974), 269-292.

[20] A. Messaoudi, Matrix recursive projection and interpolation algorithms, *Linear Algebra Appl.*, **202** (1994), 71-89.

[21] D.V. Ouellette, Schur complements and statistics, *Linear Algebra Appl.*, **36** (1981), 187-295.

[22] Y. Saad, *Iterative Methods for Spars Linear System*, PWS Press, New York (1995).

[23] M. Sadkane, Block Arnoldi and Davidson methods for unsymmetrical large eigenvalue problems, *Numer. Math.*, **64** (1993), 687-706.

[24] I. Schur, Potenzreihn im Innern des Einheitskreises, *J. Reine Angew. Math.*, **147** (1917), 205-232.

[25] H. Sadok, Analysis of the convergence of the minimal and orthogonal residual methods, *Numer. Alg.*, **40** (2005), 201-216.

[26] V. Simoncini, E. Gallopoulos, Convergence properties of block GMRES and matrix polynomial, *Linear Algebra Appl.*, **247** (1996), 97-119.

[27] V. Simoncini, Restarted full orthogonalization method for shifted linear system, *BIT (Numer. Math.)*, **43** (2003), 459-466.

[28] V. Simoncini, On the convergence of restarted Krylov subspace methods, *SIAM. J. Matrix Anal. Appl.*, **22** (2000), 430-452.