Bull. Iranian Math. Soc.
Vol. 40 (2014), No. 1, pp. 101–113
Online ISSN: 1735-8515

ON THE NUMERICAL SOLUTION OF GENERALIZED SYLVESTER MATRIX EQUATIONS

A. KAABI

(Communicated by Mohammad Asadzadeh)

Abstract. The global FOM and GMRES algorithms are among the effective methods to solve Sylvester matrix equations. In this paper, we study these algorithms in the case that the coefficient matrices are real symmetric (real symmetric positive definite) and extract two CG-type algorithms for solving generalized Sylvester matrix equations. The proposed methods are iterative projection methods onto matrix Krylov subspaces. Numerical examples are presented.

Keywords: Generalized Sylvester matrix equations, matrix Krylov subspace, global GMRES algorithm.
MSC(2010): Primary: 65F10; Secondary: 65F20.

1. Introduction

Consider the generalized Sylvester matrix equations (GSME)

$$\sum_{i=1}^{q} A_i X B_i = C, \tag{1.1}$$

where $A_i \in R^{n\times n};\ B_i \in R^{s\times s},\ i = 1, ..., q$; C and X $\in R^{n\times s}$.

This kind of matrix equations plays an important role in linear control and filtering theory for continuous or discrete-time large scale dynamical systems and image restoration. Also such problem arise in the solution of large eigenvalue problems, in boundary value problem and other problems [2, 3, 4, 7, 10]. As we know, the generalized Sylvester

Article electronically published on February 25, 2014.
Received: 16 November 2011, Accepted: 17 December 2012.

matrix equations (1.1) can be written as a big linear system of equations

$$\sum_{i=1}^{q}(B_i^T \otimes A_i)vec(X) = vec(C), \tag{1.2}$$

where $\otimes$ denote the Kronecker product, and for each matrix $L = [l_{ij}] \in R^{n\times s}$, the vector $vec(L)$ is define as follows:

$$vec(L) = (l_{11}, ..., l_{n1}, l_{12}, ..., l_{n2}, ..., l_{1s}, ..., l_{ns})^T.$$

Krylov subspace methods such as FOM and GMRES algorithms [9] could be used to solve linear system (1.2) however, for large problems this approach can not be applied directly. In [1], A. Bouhamidi and K. Jbilou presented a global approach for solving matrix equation (1.1) Their method uses the global generalized minimal residual (GLGMRES) method [6] which was originally introduced for solving linear systems with multiple right-hand sides. The conjugate gradient (CG) algorithm [5] is an extremely effective method to solve symmetric positive definite (SPD) linear system of equations. This algorithm can be extracted from the full orthogonalization method (FOM) [8] in the case that the coefficient matrix is an SPD matrix. Also, the GMRES algorithm [8, 9] results the conjugate residual (CR) algorithm [8] to solve symmetric linear system of equations not necessarily positive definite. It is well known that conjugate gradient and conjugate residual algorithms are less cost effective than FOM and GMRES algorithms, respectively. The aim of this paper is to present two CG-type methods for solving the linear matrix equations (1.1) in the case that the coefficient matrices are symmetric or symmetric positive definite.

Throughout this paper, we use the following notations. Let $\mathbb{E} = \mathbb{R}^{n\times s}$. For X and Y two matrices in $\mathbb{E}$, we define the following inner product $(X, Y)_F = tr(X^TY)$ where $tr(.)$ denotes the trace and X^T the transpose of the matrix X. The associated norm denoted by $\| \, . \, \|_F$. For $V \in \mathbb{E}$, the matrix Krylov subspace $K_m(A, V)$ is the subspace generated by the vectors (matrices) $V, AV, ..., A^{m-1}V$. Unless specified, the Frobenius norm will be used for matrices and vectors. A set of members of $\mathbb{E}$ is said to be F-orthogonal if it is orthogonal with respect to the scaler product $(., .)_F$.

This paper is organized as follows. In Section 2, we give a brief description of global-GMRES(GLGMRES) method for solving linear matrix equations. In Section 3, CG-type algorithms for solving generalized

Sylvester matrix equations are presented. In Section 4 we give some numerical experiments. Finally, Section 5 summarizes the main conclusion of this paper.

2. A brief description of GLGMRES method for Eq.(1.1)

In this section, we give a brief description of a numerical Krylov subspace method for solving the generalized Sylvester matrix equations (1.1). As we know, Eq.(1.1) has a unique solution if and only if the matrix $\sum_{i=1}^{q}(B_i^T \otimes A_i)$ is nonsingular. Throughout this paper, we suppose that this condition is verified. Also, we assume that all matrices

$$A_i,\ B_i,\ i = 1, ..., q,$$

are symmetric.

Let S be the operator from $\mathbb{R}^{n\times s}$ onto $\mathbb{R}^{n\times s}$ defined as follows:

$$S(X) = \sum_{i=1}^{q} A_i X B_i.$$

Thus, by this definition, the generalized Sylvester matrix equations (1.1) can be expressed as following equivalent form

$$S(X) = C. \tag{2.1}$$

The transpose of the operator S with respect to inner product $(.,.)_F$ is defined from $\mathbb{R}^{n\times s}$ onto $\mathbb{R}^{n\times s}$ by

$$S^T(X) = \sum_{i=1}^{q} A_i^T X B_i^T.$$

Thus S is a real symmetric (real symmetric positive definite) operator if and only if all matrices $A_i,\ B_i,\ i = 1, ..., q$, are real symmetric (real symmetric positive definite). Let V in $\mathbb{R}^{n\times s}$ be a rectangular matrix, as we see in [6], the global Arnoldi process construct an F-orthonormal basis $V_1, V_2, ..., V_m$, of matrix Krylov subspace

$$K_m(S, V) = \{V, S(V), ..., S^{m-1}(V)\},$$

i.e.

$$(V_i, V_j)_F = \delta_{ij},\ i, j = 1, ..., m,$$

where δ_{ij} denotes the classical Kronecker symbol and $S^i(V)$ is defined recursively as

$$S^i(V) = S(S^{i-1}(V)).$$

It is clear that, each U in $K_m(S,V)$ can be written by the following formula

$$U=\sum_{i=0}^{m-1}\alpha_i S^i(V),$$

where $\alpha_i,\ i=0,1,...,m-1$ are scalars. The modified global Arnoldi algorithm for constructing an F-orthogonal basis $V_1,V_2,...,V_m$, of the Krylov subspace $K_m(S,V)$ is as follows:

Algorithm 1. *Modified global Arnoldi algorithm.*

```
1- Choose a nonzero matrix $V \in \mathbb{R}^{n\times s}$. Set $V_1 = V/\parallel V \parallel_F$ .
2- For $j=1,...,m$ do:
$\tilde{V} = S(V_j)$,
for i=1,...,j, do
$h_{i},_{j} = (V_i,\tilde{V})_F$,
$\tilde{V} = \tilde{V} - h_{ij}V_i$,
enddo
$h_{j+1},_{j} =\parallel \tilde{V} \parallel_F$,
$V_{j+1} = \tilde{V}/h_{j+1,j}$
Enddo.
```

Let us collect the matrices V_i constructed by the Algorithm 1 in the $n\times ms$ and $n\times(m+1)s$ orthonormal matrices $W_m=[V_1,V_2,...,V_m]$ and $W_{m+1}=[W_m,V_{m+1}]$ and also we denote by H_m the upper $m\times m$ Hessenberg matrix whose entries are the scalars h_{ij} and the $(m+1)\times m$ matrix $\bar{H}_m$ is the same as H_m except for an additional row whose only nonzero element is $h_{m+1,m}$ in the $(m+1,m)$ position. In this paper, we use the notation $*$ for the following product:

$$W_k*\ \alpha=\sum_{i=1}^{k}\alpha_i V_i,$$

where $\alpha=(\alpha_1,\alpha_2,...,\alpha_k)^T$ is a vector in $\mathbb{R}^k$ and, by same way, we set

$$W_k*H_k=[W_k*H_k^{(1)},W_k*H_k^{(2)},...,W_k*H_k^{(k)}],$$

where $H_k^{(j)}$ denote the jth column of the matrix H_k. From Algorithm 1 we have following proposition [1]:

Proposition 2.1. *The following relations are satisfied:*
1.$[S(V_1),...,S(V_m)]=W_m(H_m\otimes I_s)+E_{m+1},$ *where*
$E_{m+1}=h_{m+1,m}[0_{n\times s},...,0_{n\times s},V_{m+1}].$

2.$[S(V_1), ..., S(V_m)] = W_{m+1}(\bar{H}_m \otimes I_s)$.
3.For any $M \in \mathbb{R}^{(m+1)\times s}$, *we have* $\| W_{m+1}(M \otimes I_s) \|_F = \| M \|_F$.

Let X_0 be an initial $n \times s$ matrix guess to the solution X of equation (1.1) and $R_0 = C - S(X_0)$ its associated residual. At the mth iterate of GLGMRES algorithm, a correction Z_m is determined in the matrix Krylov subspace $K_m = K_m(S, R_0)$ such that the new residual is F-orthogonal to K_m, i.e.,

$$X_m - X_0 = Z_m \in K_m(S, R_0),$$
$$R_m = C - S(Z_m) \perp_F K_m(S, S(R_0)),$$

where

$$K_m(S, S(R_0)) = span\{S(R_0), S^2(R_0), ..., S^m(R_0)\}.$$

Therefore, X_m can be obtained as a solution of the following minimization problem:

$$min_{X - X_0 \in K_m(S,R_0)} \| C - S(X) \|_F .$$

Any approximate solution X_m of the equation (1.1) in $X_0 + K_m$ can be written as

$$X_m = X_0 + W_m * y_m,$$

where $y_m \in \mathbb{R}^m$ is the solution of the following small least squares problem

$$min_{y \in \mathbb{R}^m} \| \| R_0 \|_F e_1 - \bar{H}_m y \|_2, \tag{2.2}$$

and e_1 is the first unit vector in $\mathbb{R}^{m+1}$. By consideration the QR decomposition of $\bar{H}_m$, we have:

$$\bar{R}_m = Q_m \bar{H}_m,$$

where $\bar{R}_m$ is upper triangular and Q_m is unitary. Thus, at step m, the residual R_m produced by the GLGMRES for for equation (1.1) has the following properties [1]:

$$R_m = \gamma_{m+1} W_{m+1}(Q_m^T e_{m+1} \otimes I_s) \tag{2.3}$$

and

$$\| R_m \|_F = | \gamma_{m+1} |, \tag{2.4}$$

where γ_{m+1} is the last component of the vector $g_m = \| R_0 \|_F Q_m e_1$ and e_{m+1} is the last unit vector in $\mathbb{R}^{m+1}$.

Thus, the restated GLGMRES algorithm for solving generalized Sylvester matrix equations (1.1) can be summarized as follows:

Algorithm 2. *Restarted global generalized minimal residual algorithm (GLGMRES) for Eq.*(1.1).

```
1. Choose an initial approximate solution $X_0$ and a tolerance $\epsilon$.
2. Compute $R_0 = C - S(X_0)$, $\beta =\| R_0 \|_F$ and $V_1 = R_0/\beta$.
3. If $\| R_0 \|_F < \epsilon$ then exit.
4. For $j = 1, ..., m$ apply Algorithm 1 to compute the F- orthonormal basis $V_1, V_2, ..., V_m$ of $K_m(S, R_0)$.
5. Determine $y_m$ the minimizer of the least square problem (2.2).
6. Compute $X_m = X_0 + W_m(y_m \otimes I_s)$.
7. Compute the residual $R_m$ and $\| R_m \|_F$ using relations (2.3) and (2.4).
8. If $\| R_m \|_F < \epsilon$ stop; else set $X_0 = X_m$, $\beta =\| R_0 \|_F$, $V_1 = R_0/\beta$ and Go to 4.
```

3. CG-type methods for solving Eq.(1.1)

In this section we present two CG-type numerical Krylov subspace algorithms for solving linear matrix equation (1.1) . These algorithms are extracted from global full orthogonalization (GLFOM) and global generalized minimal residual (GLGMRES) methods. First, we state the following proposition .

Proposition 3.1. *Assume that the global Arnoldi method is applied to a real symmetric operator S. Then the matrix H_m generated by the algorithm is tridiagonal and symmetric.*

Proof. As we know, the matrix H_m is a Hessenberg matrix and its elements are

$$h_{ij} = tr(V_i^T S(V_j)).$$

On the other hand, we have

$$h_{ji} = tr(V_j^T S(V_i)) = tr(V_i^T S(V_j)) = h_{ij}.$$

Hence, the matrix H_m is tridiagonal and symmetric. □

Thus, the matrix H_m can be written as follows:

$$H_m = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_2 & & \\ & & & & \\ & & \beta_{m-1} & \alpha_{m-1} & \beta_m \\ & & & \beta_m & \alpha_m \end{bmatrix}.$$

This leads the following form of Algorithm 1, namely the global Lanczos algorithm.

Algorithm 3. *Global Lanczos algorithm.*

```
1- Choose a matrix $V \in \mathbb{R}^{n\times s}$ such that $\| V \|_F = 1$. Set $\beta_1 = 0,\ V_0 = 0$.
2- For $j = 1, ..., m$ do:
$W_j = S(V_j) - \beta_j V_{j-1}$ (If $j = 0$ set $\beta_1 = 0,\ V_0 = 0$).
$\alpha_j = tr(V_j^T W_j)$.
$W_j = W_j - \alpha_j V_j$.
$\beta_{j+1} = \| W_j \|_F$. If $\beta_{j+1} = 0$ then stop.
$V_{j+1} = W_j / \beta_{j+1}$.
Enddo.
```

By using the global Lanczos and FOM algorithms we can drive a new algorithm for solving symmetric linear matrix equations (2.1).

Algorithm 4. *A new algorithm for solving Eq.*(2.1).

```
1. Choose an initial approximate solution $X_0$ and a tolerance $\epsilon$.
2. Compute $R_0 = C - S(X_0)$, $\beta = \| R_0 \|_F$ and $V_1 = R_0/\beta$.
3. If $\| R_0 \|_F < \epsilon$ then exit.
4. For j=1,...,m apply Algorithm 3 to compute the F- orthonormal basis
   $V_1, V_2, ..., V_m$ of $K_m(S, R_0)$.
5. Set $H_m = tridiag(\beta_i,\ \alpha_i,\ \beta_{i+1})$ and $W_m = [V_1, ..., V_m]$.
6. Compute $y_m = H_m^{-1}(\beta_1 e_1)$ and $X_m = X_0 + W_m * y_m$.
```

Now, let $A_i,\ B_i,\ i = 1, ..., q$, are real real symmetric positive definite matrices. Thus the matrix H_m has LU factorization of the form $H_m = L_m U_m$ which can be as follows:

$$H_m = L_m U_m = \begin{bmatrix} 1 & & & & \\ \lambda_2 & 1 & & & \\ & \lambda_3 & 1 & & \\ & & & & \\ & & & \lambda_m & 1 \end{bmatrix} \begin{bmatrix} \eta_1 & \beta_2 & & & \\ & \eta_2 & \beta_3 & & \\ & & & & \\ & & & \eta_{m-1} & \beta_m \\ & & & & \eta_m \end{bmatrix}.$$

Hence, the approximate solution X_m is given by

$$X_m = X_0 + W_m * (U_m^{-1} L_m^{-1} \beta e_1) = X_0 + (W_m * U_m^{-1}) * (L_m^{-1} \beta e_1).$$

By letting

$$Q_m = W_m * U_m^{-1},$$

and

$$z_m = L_m^{-1} \beta e_1,$$

we have

$$X_m = X_0 + Q_m * z_m,$$

where

$$Q_m = [P_1, ..., P_m].$$

It is easily seen that

$$Q_m * U_m = W_m,$$

Thus,

$$\beta_m P_{m-1} + \eta_m P_m = V_m,$$

or

$$P_m = \frac{1}{\eta_m}[V_m - \beta_m P_{m-1}]. \tag{3.1}$$

On the other hand from the LU-factorization of H_m, we have

$$\lambda_m = \frac{\beta_m}{\eta_{m-1}},$$

$$\eta_m = \alpha_m - \lambda_m \beta_m.$$

Now, by letting

$$z_m = [z_{m-1} \ \zeta_m]^T,$$

we have

$$\zeta_m = -\lambda_m \zeta_{m-1}.$$

Therefore we can conclude that

$$X_m = X_{m-1} + \zeta_m P_m,$$

where P_m is defined by (3.1). This gives the global D-Lanczos algorithm for solving Eq.(1.1) .

Algorithm 5. *Global D-Lanczos algorithm for solving Eq.*(1.1)*.*

```
1. Choose an initial approximate solution $X_0$.
2. Compute $R_0 = C - S(X_0)$, $\zeta_1 = \beta =\| R_0 \|_F$ and set $V_1 = R_0/\beta$.
3. For $m = 1, 2, ...,$ until convergence do:
4. Compute $W = S(V_m) - \beta_m V_{m-1}$ and set $\alpha_m = (W, V_m)_F$.
5. If $m > 1$ then compute $\lambda_m = \frac{\beta_m}{\eta_{m-1}}$ and $\zeta_m = -\lambda_m \zeta_{m-1}$.
6. $\eta_m = \alpha_m - \lambda_m \beta_m$.
7. $P_m = \eta_m^{-1}[V_m - \beta_m P_{m-1}]$.
8. $X_m = X_{m-1} + \zeta_m P_m$,
9. If $X_m$ has convergent then stop.
10.$W = W - \alpha_m V_m$.
11.$\beta_{m+1} =\| W \|_F, \ V_{m+1} = W/\beta_{m+1}$.
12.Enddo.
```

Proposition 3.2. *Let $R_m = C - S(X_m),\ m = 0, 1, ...,$ be the residual matrices and $P_m,\ m = 0, 1, ...,$ be the auxiliary matrices produced by the global D-Lanczos algorithm. Then,*
1. Each residual matrix R_m is such that $R_m = \sigma_m V_{m+1}$ where σ_m is a certain scaler. As a result the residual matrices are F-orthogonal to each other.
2. The the auxiliary matrices P_m are S-conjugate set with respect to inner product $(.,.)_F$, i.e., $(S(P_i), P_j)_F = 0$ for $i \neq j$.

Proof. It is obvious that the algorithms 4 and 5 are equivalent. Hence the first part of the proposition immediate consequence of the relation (2.3).
For the second part, let $[\delta_{1j}, ..., \delta_{jj}, 0, ..., 0]^T$ be the jth column of U_m^{-1}, then we have

$$P_j = \sum_{k=1}^{j} \delta_{kj} V_k.$$

Therefore,

$$(S(P_i), P_j)_F = (S(\sum_{l=1}^{i} \delta_{li} V_l), \sum_{k=1}^{j} \delta_{kj} V_k)_F = \sum_{l=1}^{i} \sum_{k=1}^{j} \delta_{li} \delta_{kj} (S(V_l), V_k)_F.$$

Since $(S(V_l), V_k)_F = 0$, for $l \neq k$ we conclude that $(S(P_i), P_j)_F = 0$, for $i \neq j$. □

By using this proposition we can obtain the GLCG algorithm. From the global D-Lanczos algorithm we have

$$X_{j+1} = X_j + \alpha_j P_j.$$

Therefore the residual matrix must satisfy the following recurrence

$$R_{j+1} = R_j + \alpha_j S(P_j). \tag{3.2}$$

From the F-orthogonality of R_j's we get

$$(R_j - \alpha_j S(P_j), R_j)_F = 0$$

and as a result

$$\alpha_j = \frac{(R_j, R_j)_F}{(S(P_j), R_j)_F}. \tag{3.3}$$

Also, it is known that the next search direction P_{j+1} is a linear combination of R_{j+1} and P_j, i.e.,

$$P_{j+1} = R_{j+1} + \beta_j P_j.$$

Since $S(P_j)$ is F-orthogonal to P_{j-1}, thus, from the above relation we have

$$(S(P_j), R_j)_F = (S(P_j), P_{j-1} - \beta_j P_{j-1})_F = (S(P_j), P_j)_F.$$

Hence the relation (3.3) becomes

$$\alpha_j = \frac{(R_j, R_j)_F}{(S(P_j), P_j)_F}.$$

From the F-orthogonality of P_{j+1} to AP_j yields

$$\beta_j = -\frac{(R_{j+1}, S(P_j))_F}{(P_j, S(P_j))_F}.$$

From the equation (3.2) we get

$$S(P_j) = -\frac{-1}{\alpha_j}(R_{j+1} - R_j),$$

and hence

$$\beta_j = -\frac{1}{\alpha_j}\frac{(R_{j+1}, R_{j+1} - R_j)_F}{(S(P_j), P_j)_F} = \frac{(R_{j+1}, R_{j+1})_F}{(R_j, R_j)_F}.$$

Thus, we can conclude the global conjugate gradient (GLCG) algorithm for solving generalized Sylvester matrix equations (1.1) as follows:

Algorithm 6. *Global conjugate gradient (GLCG) algorithm for solving Eq.*(1.1).

```
1. Choose an initial approximate solution $X_0$.
2. Compute $R_0 = C - S(X_0)$ and set $P_0 = R_0$.
3. For $j = 1, 2, ...,$ until convergence do:
4. $\alpha_j = (R_j, R_j)_F / (S(P_j), P_j)_F$.
5. $X_{j+1} = X_j + \alpha_j P_j$.
6. $R_{j+1} = R_j - \alpha_j S(P_j)$.
7. $\beta_j = (R_{j+1}, R_{j+1})_F / (R_j, R_j)_F$.
8. $P_{j+1} = R_{j+1} + \beta_j P_j$.
9. Enddo.
```

On the other hand, if the coefficient matrices

$$A_i,\ B_i,\ i = 1, ..., q,$$

are symmetric, we can extract the global conjugate residual (GLCR) algorithm from the GLGMRES method. In this case, we have

$$(S(P_i), S(P_j))_F = 0,\ i \neq j,$$

and

$$(S(R_i), R_j)_F = 0,\ i \neq j.$$

Also, we can conclude the global conjugate residual (GLCR) algorithm for solving generalized Sylvester matrix equations (1.1) as follows:

Algorithm 7. *Global conjugate residual (GLCR) algorithm for solving Eq.*(1.1)*.*

```
1. Choose an initial approximate solution $X_0$.
2. Compute $R_0 = C - S(X_0)$ and set $P_0 = R_0$.
3. For $j = 1, 2, ...,$ until convergence do:
4. $\alpha_j = (R_j, S(R_j))_F/(S(P_j), S(P_j))_F$.
5. $X_{j+1} = X_j + \alpha_j P_j$.
6. $R_{j+1} = R_j - \alpha_j S(P_j)$.
7. $\beta_j = (R_{j+1}, S(R_{j+1}))_F/(R_j, S(R_j))_F$.
8. $P_{j+1} = R_{j+1} + \beta_j P_j$.
9. $S(P_{j+1}) = S(R_{j+1}) + \beta_j S(P_j)$.
10.Enddo.
```

4. Numerical experiments

In this section, we present the results of numerical experiments. Computations were carried out using MATLAB 6.5 codes on a personal computer Pentium 3-800EB MHs. The tests were stopped as soon as

$$\frac{\| \sum_{i=1}^{2} A_i X_k B_i - C \|_F}{\| C \|_F} \leq 10^{-5},$$

or the maximum number of iterations reached to 1000. For all the experiments, the initial guess was $X_0 = 0$. The matrix C is defined so that a true solution of equation (1.1) is the matrix of all one. For the global conjugate gradient(GLCG) algorithm, we used the real symmetric positive definite matrices
$A_i = [(2^{-i}-1)I_n + diag(1, 2, ..., n) + U_n^T] + [(2^{-i}-1)I_n + diag(1, 2, ..., n) + U_n^T]^T$, and

$$B_i = [I_s + 2^{-i}U_s] + [I_s + 2^{-i}U_s]^T,$$

where $i = 1, 2$, U_n and U_s are the $n \times n$ and $s \times s$ matrices respectively with unit entries below the diagonal and all other entries zero. For the global conjugate residual(GLCR) algorithm, we used the real symmetric matrices

$A_i = tridiag(1+i/n, 2, 1+i/n)$ *and* $B_i = tridiag(-1-i/n, -2, -1-i/n)$,

Table 1. Number of iterations and CPU times.

n	Method	$s = 200$	$s = 300$	$s = 400$	$s = 500$
2000	GLFOM	3(18.61)	4(40.52)	4(65.51)	6(115.2)
	GLCG	3(9.751)	3(15.53)	4(21.22)	4(28.22)
2500	GLFOM	4(38.21)	4(60.31)	7(152.1)	8(229.3)
	GLCG	3(14.72)	3(22.84)	4(31.71)	5(41.42)

Table 2. Number of iterations and CPU times.

n	Method	$s = 200$	$s = 300$	$s = 400$	$s = 500$
2000	GLGMRES	15(101)	14(150)	13(196)	13(258)
	GLCR	14(55)	12(85)	11(117)	11(150)
2500	GLGMRES	15(153)	13(224)	13(290)	13(379)
	GLCR	12(82)	12(119)	12(170)	10(216)

where $i = 1, 2$. The results are reported in Table 1 and Table 2 with different values of n, s, i.e., $n = 2000,\ 2500,\ \ s = 200,\ 300,\ 400,\ 500$, and $m = 2$.

5. Conclusion

In this paper, we have extracted the global conjugate gradient (GLCG) and global conjugate residual (GLCR) algorithms from the global full orthogonalization and generalized minimal residual algorithms for solving generalized Sylvester matrix equations, respectively. The experiments presented in this paper show that the solution of generalized Sylvester matrix equations can be obtained with high accuracy by using GLCG and GLCR algorithms. Table 1 and Table 2 show that the number of iterations and CPU times of GLCG and GLCR algorithms are less than GLFOM and GLGMRES algorithms respectively.

References

[1] A. Bouhamidi and K. Jbilou, A note on the numerical approximate solutions for generalized Sylvester matrix equations with applications, *Appl. Math. Comput.* **206** (2008), no. 2, 687–694.

[2] A. Bouhamidi, K. Jbilou and Sylvester Tikhonov regularization methods in image restoration, *J. Comput. Appl. Math.* **206** (2007), no. 1, 86–98.

[3] F. Chatelin, Valeurs Propres de Matrices, Masson, Paris, 1998.

[4] B. N. Datta, Numerical Methods for Linear Control Systems Design and Analysis, Esevier Press, San Diego, 2003.

[5] M. R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems, *J. Research Nat. Bur. Standards* **49** (1952) 409–436.

[6] K. Jbilou, A. Messaoudi and H. Sadok, Global FOM and GMRES algorithms for matrix equations, *Appl. Numer. Math.* **31** (1999), no. 1, 49–63.

[7] K. Jbilou and A. Riquet, Projection methods for large Lyapunov matrix equations, *Linear Algebra Appl.* **415** (2006) 344–358.

[8] Y. Saad, Iterative Methods for Sparse linear Systems, PWS Press, New York, 1995.

[9] Y. Saad and M. H. Schultz, GMRES, A generalized minimal residual algorithm for nonsymmetric linear systems, *SIAM J. Sci. Statist. Comput.* **7** (1986), no. 3, 856–868.

[10] B. Zhou and G. R. Duan, An explicit solution to the matrix equation $AX - XF = BY$, *Linear Algebra Appl.* **402** (2005) 345–366.

(Amer Kaabi) Department of Basic Science, Khorramshahr University of Marine Science and Technology, P.O. Box 669, Khorramshahr, Iran

E-mail address: kaabi.amer@yahoo.com