Applied Mathematics and Computation 206 (2008) 687–694

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

A note on the numerical approximate solutions for generalized Sylvester matrix equations with applications

A. Bouhamidi *, K. Jbilou

L.M.P.A, Université du Littoral, 50 rue F. Buisson BP699, F-62228 Calais Cedex, France

ARTICLE INFO

Keywords:
Linear matrix equation
Global GMRES
Matrix Krylov subspace
Preconditioning

ABSTRACT

In the present paper, we propose a Krylov subspace method for solving large and sparse generalized Sylvester matrix equations. The proposed method is an iterative projection method onto matrix Krylov subspaces. As a particular case, we show how to adapt the ILU and the SSOR preconditioners for solving large Sylvester matrix equations. Numerical examples and applications to some PDE's will be given.

1. Introduction

The aim of this paper is to present numerical Krylov subspace methods for solving the linear matrix equation

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

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

Such problems arise in the solution of large eigenvalue problems [6] and in the boundary value problem. They play also an important role in linear control and filtering theory for continuous or discrete-time large-scale dynamical systems, image restoration and other problems; see [2–5,8,11–13,16–19] and the references therein. The matrix equation (1) contains the well-known Lyapunov, Sylvester and Stein matrix equations.

The linear matrix equation can be written as the following $np \times np$ linear system:

$$\left[\sum_{i=1}^{q} (B_i^{\mathrm{T}} \otimes A_i)\right] \mathrm{vec}(X) = \mathrm{vec}(C), \tag{2}$$

where $\mathrm{vec}(X)$ is the vector of $\mathbb{R}^{np}$ obtained by stacking the columns of the $n \times p$ matrix X and $\otimes$ denotes the Kronecker product; $(F \otimes G = [f_{ij}G])$. Krylov subspace methods such as the GMRES algorithm [15] could be used to solve the linear system (2). However, for large problems this approach cannot be applied directly.

In the present paper, we present a global approach for solving the matrix equation (1). Our method uses the global generalized minimal residual (GlGMRES) method [10] which was originally introduced for solving linear systems with multiple right-hand sides.

In Section 2, we recall the global generalized minimal residual (GlGMRES) method, show how to apply the GlGMRES method for solving the matrix equation (1) and give some theoretical results. In Section 3, we a left-right preconditioner

* Corresponding author.
E-mail addresses: bouhamidi@lmpa.univ-littoral.fr (A. Bouhamidi), jbilou@lmpa.univ-littoral.fr (K. Jbilou).

0096-3003/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.amc.2008.09.022

for large Sylvester matrix equations. Section 4 is devoted to the symmetric successive overrelaxation (SSOR) preconditioning in association with the GlGMRES algorithm for Sylvester matrix equations. In the last section we give some numerical experiments.

In this paper, we use the following notations. For X and Y two matrices in $\mathbb{R}^{n\times p}$, we define the following inner product $\langle X, Y\rangle_F = tr(X^{\mathrm{T}}Y)$ where $tr(.)$ denotes the trace and X^{T} the transpose of the matrix X. The associated norm is the well known Frobenius norm denoted by $\|\cdot\|_F$. For $V \in \mathbb{R}^{n\times p}$, the matrix Krylov subspace $\mathscr{K}_k(A, V)$ is the subspace generated by the vectors (matrices) $V, AV, \ldots, A^{k-1}V$. Unless specified, the Frobenius norm will be used for matrices and vectors. A system of matrices of $\mathbb{R}^{n\times p}$ is said to be F-orthogonal if it is orthogonal with respect to the scalar product $\langle ., .\rangle_F$.

2. The global-GMRES method for linear matrix equations

In this section, we present a numerical Krylov subspace method for solving the linear matrix equation (1). Eq. (1) has a unique solution if and only if the matrix $\sum_{i=1}^{q} B_i^{\mathrm{T}} \otimes A_i$ is nonsingular. Throughout this paper, we assume that this condition is verified.

Let $\mathscr{M}$ be the operator defined as follows:

$$\mathscr{M} : \mathbb{R}^{n\times p} \to \mathbb{R}^{n\times p},$$

$$X \to \sum_{i=1}^{q} A_i X B_i.$$

The transpose of the operator $\mathscr{M}$ with respect to the inner product $\langle ., .\rangle_F$ is defined from $\mathbb{R}^{n\times p}$ onto $\mathbb{R}^{n\times p}$ by $\mathscr{M}^{\mathrm{T}}(X) = \sum_{i=1}^{q} A_i^{\mathrm{T}} X B_i^{\mathrm{T}}$. Next, we show how to solve iteratively the problem (1) using Krylov subspace methods.

Let V be any $n \times p$ matrix and consider the matrix Krylov subspace associated to the pair $(\mathscr{M}, V)$ and an integer k defined by

$$\mathscr{K}_k(\mathscr{M}, V) = \mathrm{span}\{V, \mathscr{M}(V), \ldots, \mathscr{M}^{k-1}(V)\}.$$

We note that $\mathscr{M}^i(V)$ is defined recursively as $\mathscr{M}^i(V) = \mathscr{M}(\mathscr{M}^{i-1}(V))$. Remark that the matrix Krylov subspace $\mathscr{K}_k(\mathscr{M}, V)$ is a subspace of $\mathbb{R}^{n\times p}$.

The modified global Arnoldi algorithm [10] constructs an F-orthonormal basis $V_1, V_2, \ldots, V_k$ of the matrix Krylov subspace $\mathscr{K}_k(\mathscr{M}, V)$, i.e.

$$\langle V_i, V_j\rangle_F = \delta_{ij}, \quad \text{for } i, j = 1, \ldots, k,$$

where δ_{ij} denotes the classical Kronecker symbol. The algorithm is described as follows:

Algorithm 1 (*Modified Global Arnoldi algorithm*).

1. Set $V_1 = V/\|V\|_F$.
2. For $j = 1, \ldots, k$. do
  $\widetilde{V} = \mathscr{M}(V_j)$,
  for $i = 1, \ldots, j$. do
    $h_{ij} = \langle V_i, \widetilde{V}\rangle_F$,
    $\widetilde{V} = \widetilde{V} - h_{ij}V_i$,
  endfor
  $h_{j+1,j} = \|\widetilde{V}\|_F$,
  $V_{j+1} = \widetilde{V}/h_{j+1,j}$.
EndFor.

Let $\mathscr{V}_k$ be the $n \times kp$ matrix: $\mathscr{V}_k = [V_1, V_2, \ldots, V_k]$. $\widetilde{H}_k$ denotes the $(k+1) \times k$ upper Hessenberg matrix whose nonzero entries h_{ij} are defined by Algorithm 1 and H_k is the $k \times k$ matrix obtained from $\widetilde{H}_k$ by deleting its last row. Note that the block matrix $\mathscr{V}_k$ is F-orthonormal which means that the matrices $V_1, \ldots, V_k$ are orthonormal with respect to the scalar product $\langle \cdot, \cdot\rangle_F$.

It is not difficult to show the following proposition:

Proposition 1. *We have the following relations:*

1. $[\mathscr{M}(V_1), \ldots, \mathscr{M}(V_k)] = \mathscr{V}_k(H_k \otimes I_p) + E_{k+1}$, *where* $E_{k+1} = h_{k+1,k}[0_{n\times p}, \ldots, 0_{n\times p}, V_{k+1}]$.
2. $[\mathscr{M}(V_1), \ldots, \mathscr{M}(V_k)] = \mathscr{V}_{k+1}(\widetilde{H}_k \otimes I_p)$.
3. *For any* $(k+1) \times s$ *matrix* G, *we have* $\|\mathscr{V}_{k+1}(G \otimes I_p)\|_F = \|G\|_F$.

Starting from an initial guess $X_0 \in \mathbb{R}^{n\times p}$ and the corresponding residual $R_0 = C - \mathscr{M}(X_0)$, the global GMRES method defines, at step k, the approximate solution X_k as follows:

$$X_k = X_0 + Z_k \quad \text{with } Z_k \in \mathscr{K}_k(\mathscr{M}, R_0) \tag{3}$$

and

$$R_k = C - \mathscr{M}(X_k) \perp_F \mathscr{K}_k(\mathscr{M}, \mathscr{M}(R_0)) \tag{4}$$

Observe that the residual $R_k = C - \mathscr{M}(X_k)$ is obtained by projecting orthogonally R_0 onto the matrix Krylov subspace

$$\mathscr{K}_k(\mathscr{M}, \mathscr{M}(R_0)) = \text{span}\{\mathscr{M}(R_0), \mathscr{M}^2(R_0), \ldots, \mathscr{M}^k(R_0)\}.$$

This shows that X_k can be obtained as the solution of the minimization problem

$$\min_{X - X_0 \in \mathscr{K}_k(\mathscr{M}, R_0)} \|C - \mathscr{M}(X)\|_F. \tag{5}$$

We have the following result:

Theorem 1. *At step k, the approximate solution X_k produced by the global-GMRES method is given by $X_k = X_0 + \mathscr{V}_k(y_k \otimes I_p)$ where y_k is the solution of the following small least-squares problem*

$$\min_{y \in \mathbb{R}^k} \| \|R_0\|_F e_1 - \widetilde{H}_k y\|_2 \tag{6}$$

where e_1 is the first unit vector of $\mathbb{R}^{k+1}$.

Proof. From the relation (3), the approximation X_k can be written as $X_k = X_0 + \mathscr{V}_k(y \otimes I_p)$ where $y \in \mathbb{R}^k$. Hence replacing in (5), using item 2 of Proposition 1 and the fact that $R_0 = \|R_0\|_F \mathscr{V}_{k+1}(e_1 \otimes I_p)$, we obtain the following minimization problem:

$$\min_{y \in \mathbb{R}^k} \|\mathscr{V}_{k+1}((\|R_0\|_F e_1 - \widetilde{H}_k y_k) \otimes I_p)\|_F. \tag{7}$$

Using item 3 of Proposition 1, the result follows.

To solve the problem (6) we consider the QR decomposition of the $(k+1) \times k$ matrix $\widetilde{H}_k$:

$$\widetilde{R}_k = Q_k \widetilde{H}_k,$$

where $\widetilde{R}_k$ is upper triangular and Q_k is unitary. Then if $g_k = \|R_0\|_F Q_k e_1$ and if R_1 denotes the $k \times k$ matrix obtained from $\widetilde{R}_k$ by deleting its last row, y_k is given by $R_1 y_k = g_k$.

As k increases it is interesting to compute the k-th residual without having to compute extra matrix-matrix products with the matrices A and B. This is given in the following theorem. □

Theorem 2. *At step k, the residual $R_k = C - \mathscr{M}(X_k)$ produced by the global GMRES for the linear matrix equation satisfies the following properties*

$$R_k = \gamma_{k+1} \mathscr{V}_{k+1}(Q_k^T e_{k+1} \otimes I_p) \tag{8}$$

and

$$\|R_k\|_F = |\gamma_{k+1}|, \tag{9}$$

where γ_{k+1} is the last component of the vector $g_k = \|R_0\|_F Q_k e_1$ and $e_{k+1} = (0, 0, \ldots, 1)^T \in \mathbb{R}^{k+1}$.

Proof. At step k the residual is given by $R_k = C - \mathscr{M}(X_k)$ where $\mathscr{M}(X_k) = X_k - AX_kB$ with $X_k = X_0 + \mathscr{V}_k(y_k \otimes I_p)$. Then R_k can be expressed as

$$R_k = R_0 - [\mathscr{M}(V_1), \ldots, \mathscr{M}(V_1)](y_k \otimes I_p).$$

Invoking item 2 of Proposition 1, it follows that

$$R_k = R_0 - \mathscr{V}_{k+1}(\widetilde{H}_k \otimes I_p)(y_k \otimes I_p).$$

Then

$$R_k = R_0 - \mathscr{V}_{k+1}(\widetilde{H}_k y_k \otimes I_p).$$

Now using the fact that $R_0 = \|R_0\|_F \mathscr{V}_{k+1}(e_1 \otimes I_p)$, we obtain

$$R_k = \mathscr{V}_{k+1}[(\|R_0\|_F e_1 - \widetilde{H}_k y_k \otimes I_p).$$

Using the QR decomposition $\widetilde{R}_k = Q_k, \widetilde{H}_k$, we get

$$R_k = \mathscr{V}_{k+1}[(Q_k^T Q_k \otimes I_p)(\|R_0\|_F e_1 - \widetilde{H}_k y_k \otimes I_p) = \mathscr{V}_{k+1}[(Q_k^T \otimes I_p)((\|R_0\|_F Q_k e_1 - Q_k \widetilde{H}_k y_k) \otimes I_p),$$

therefore $R_k = \mathscr{V}_{k+1}[(Q_k^{\mathrm{T}} \otimes I_p)((\|R_0\|_F Q_k e_1 - \widetilde{R}_k y_k) \otimes I_p)$. Now as y_k solves the problem (6), it follows that $R_k = \gamma_{k+1}\mathscr{V}_{k+1}(Q_k^{\mathrm{T}} e_{k+1} \otimes I_p)$, where γ_{k+1} is the last component of the vector $g_k = \|R_0\|_F Q_k e_1$, which shows the relation (8). To show (9), we use the fact that for $z \in \mathbb{R}^{k+1}$ we have $\|\mathscr{V}_{k+1}(z \otimes I_p)\|_F = \|z\|_2$. Hence using this remark and the relation (8), it follows that

$$\|R_k\|_F = |\gamma_{k+1}|\|Q_k^{\mathrm{T}} e_{k+1}\|_2.$$

Therefore since Q_k is unitary we obtain $\|R_k\|_F = |\gamma_{k+1}|$. □

To save memory and CPU-time requirements, the Global GMRES method will be used in a restarted mode. This means that we have to restart the algorithm every k inner iterations, where k is a fixed integer. The restarted Global GMRES algorithm for solving the linear matrix Eq. (1), denoted by GlGMRES(k), is summarized as follows:

Algorithm 2. GlGMRES(k) algorithm for the linear matrix equation (1)).

1. Choose X_0 a tolerance ε and set $iter = 0$.
Compute: $R_0 = C - \mathscr{M}(X_0)$, $\beta = \|R_0\|_F$, and $V_1 = R_0/\beta$.
2. Construct the F-orthonormal basis $V_1, V_2, \ldots, V_k$ by Algorithm 1 with $\mathscr{M}$.
3. Determine y_k as solution of the least square problem:

$$\min_{y \in \mathbb{R}^k} \|\|R_0\|_F e_1 - \widetilde{H}_k y\|_2$$

Compute: $X_k = X_0 + \mathscr{V}_k(y_k \otimes I_p)$
4. Compute the residual R_k and $\|R_k\|_F$ using Theorem 2.
5. If $\|R_k\|_F < \varepsilon$ Stop;
else $X_0 = X_k$, $R_0 = R_k$, $\beta = \|R_0\|_F$, $V_1 = R_0/\beta$, $iter = iter + 1$, Goto 2.

When $p = 1$, the global-GMRES algorithm reduces to the classical GMRES method [15] for solving linear systems of equations.

3. The ILU-Sylvester preconditioning

Consider the Sylvester matrix equation

$$AX - XB = C, \tag{10}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{p\times p}$, X and C are in $\mathbb{R}^{n\times p}$. We assume here that p is much smaller than n.

The Sylvester matrix equation (10) is a particular case of (1) with $q = 2$, $A_1 = A$, $A_2 = I_p$, $B_1 = I_p$ and $B_2 = I_n$.

Direct methods for solving (10) as those proposed in [1,7] are attractive if the matrices are of small size. Several block based iterative methods have been proposed for the approximate solution to large Sylvester and Lyapunov matrix equations; see [9,11,12]. These methods use Galerkin projection to produce low-dimensional Sylvester equation which is solved by using direct methods. We note that no preconditioning have been used for these methods.

The eigenvalues of the linear operator $\mathscr{A}$ defined by $\mathscr{A}(X) = AX - XB$ are $\lambda_i(A) - \lambda_j(B)$, $i = 1, \ldots, n$ and $j = 1, \ldots, p$ where $\lambda_i(A)$ denotes the ith eigenvalue of the matrix A. Therefore when $p \ll n$ the distribution of the eigenvalues of the linear operator $\mathscr{M}$ is dominated by the distribution of the eigenvalues of the matrix A. So our idea for preconditioning is to act only on A.

Consider left–right preconditioner (M_1, M_2) of the matrix A, i.e; $M_1^{-1} A M_2^{-1} \approx I_n$. A popular left–right preconditioner is the incomplete LU factorization (ILU) [14].

Multiplying from the left the Sylveter matrix equation by M_1^{-1}, it follows:

$$M_1^{-1}AX - M_1^{-1}XB = M_1^{-1}C, \tag{11}$$

which is equivalent to

$$M_1^{-1}AM_2^{-1}M_2X - M_1^{-1}M_2^{-1}M_2XB = M_1^{-1}C \tag{12}$$

setting $Y = M_2X$ we get

$$A_1Y - A_2YB = C_1, \quad X = M_2^{-1}Y \tag{13}$$

with $A_1 = M_1^{-1}AM_2^{-1}$, $A_2 = M_1^{-1}M_2^{-1}$ and $C_1 = M_1^{-1}C$. The obtained matrix equation (13) is a particular case of (1) and can be solved iteratively by the global GMRES (Algorithm 2). We notice that when using this technique of preconditioning, we do not need to inverse matrices but only solve linear systems with the lower and the upper matrices L and U respectively.

4. The SSOR-Sylvester preconditioning

The matrix equation (10) is theoretically equivalent to the $np \times np$ linear system

$$\widetilde{\mathscr{A}}\tilde{x} = \tilde{c}, \tag{14}$$

where $\widetilde{\mathscr{A}} = I_p \otimes A - B^{\mathrm{T}} \otimes I_n$; $\tilde{c} = \mathrm{vec}(C)$ and $\tilde{x} = \mathrm{vec}(X)$.

Let the splitting of A and B be as

$$\begin{cases} A = D_A - E_A - F_A, \\ B = D_B - E_B - F_B, \end{cases}$$

where D_A is the diagonal of A, $-E_A$ its strict lower part and $-F_A$ its strict upper part. Then the splitting of the matrix $\widetilde{\mathscr{A}}$ is given as

$$\widetilde{\mathscr{A}} = I_s \otimes (D_A - E_A - F_A) - (D_B - E_B^{\mathrm{T}} - F_B^{\mathrm{T}}) \otimes I_N = \widetilde{\mathscr{D}}_{\widetilde{\mathscr{A}}} - \widetilde{\mathscr{E}}_{\widetilde{\mathscr{A}}} - \widetilde{\mathscr{F}}_{\widetilde{\mathscr{A}}}$$

with

$$\begin{cases} \widetilde{\mathscr{D}}_{\widetilde{\mathscr{A}}} = I_s \otimes D_A - D_B \otimes I_N, \\ \widetilde{\mathscr{E}}_{\widetilde{\mathscr{A}}} = I_s \otimes E_A - F_B^{\mathrm{T}} \otimes I_N, \\ \widetilde{\mathscr{F}}_{\widetilde{\mathscr{A}}} = I_s \otimes F_A - E_B^{\mathrm{T}} \otimes I_N. \end{cases}$$

Now instead of solving the original problem (10), we will apply the GLGMRES algorithm to the preconditioned system

$$\mathscr{M}^{-1}(\mathscr{A}(X)) = \mathscr{M}^{-1}(C), \tag{15}$$

which is equivalent to solve the linear system

$$\widetilde{\mathscr{M}}^{-1}\widetilde{\mathscr{A}}\tilde{x} = \widetilde{\mathscr{M}}^{-1}\tilde{c}, \tag{16}$$

where $\widetilde{\mathscr{M}}$ is the matrix of the linear preconditioner $\mathscr{M}$. We note that the $np \times np$ matrix $\widetilde{\mathscr{A}}$ is not used explicitly. We use only the action of the operator $\mathscr{A}$ on a matrix $V \in \mathbb{R}^{n\times p}$: $\mathscr{A}(V) = AV - VB$. Here we use the SSOR preconditioner defined by

$$\widetilde{\mathscr{M}}_{\mathrm{SSOR}} = \frac{1}{\omega(2-\omega)}(\widetilde{\mathscr{D}}_{\widetilde{\mathscr{A}}} - \omega\widetilde{\mathscr{E}}_{\widetilde{\mathscr{A}}})\widetilde{\mathscr{D}}_{\widetilde{\mathscr{A}}}^{-1}(\widetilde{\mathscr{D}}_{\widetilde{\mathscr{A}}} - \omega\widetilde{\mathscr{F}}_{\widetilde{\mathscr{A}}}). \tag{17}$$

As in the GlGMRES algorithm we use only matrix-by- vector products, then when using the SSOR preconditioner we have to compute, for a given $V \in \mathbb{R}^{n\times p}$, the matrix $W \in \mathbb{R}^{n\times p}$ such that

$$\tilde{w} = \widetilde{\mathscr{M}}_{\mathrm{SSOR}}^{-1}\widetilde{\mathscr{A}}\tilde{v} \quad \text{with } \tilde{w} = \mathrm{vec}(W) \quad \text{and} \quad \tilde{v} = \mathrm{vec}(V).$$

This can be solved as follows:

- $\tilde{r} = \widetilde{\mathscr{A}}\tilde{v} \Longleftrightarrow R = AV - VB$ with $\tilde{r} = \mathrm{vec}(R)$,
- $\tilde{w} = \widetilde{\mathscr{M}}_{\mathrm{SSOR}}^{-1}\tilde{r} \Longleftrightarrow \widetilde{\mathscr{M}}_{\mathrm{SSOR}}\tilde{w} = \tilde{r}$.

To compute W such that $\tilde{w} = \widetilde{\mathscr{M}}_{\mathrm{SSOR}}^{-1}\tilde{r}$, we have to solve the following problems:

1. solve $(D_A - \omega E_A)Y - Y(D_B - \omega F_B) = R$,
2. compute $D_A Y - Y D_B = Z$,
3. solve $(D_A - \omega F_A)W - W(D_B - \omega E_B) = Z$.

The matrix equations (1) and (3) are also Sylvester matrix equations. But since the matrices involved in these equations are triangular, they are solved easily. For Eq. (1), the matrix Y is computed from left to right and from top to bottom in each column, this corresponds to backward solve. Eq. (3) is solved in the opposite sense and this corresponds to forward solve.

5. Numerical examples

The tests reported in this section were run on SUN Microsystems workstations using Matlab. For all the experiments, the initial guess X_0 was taken to be the zero $n \times p$ matrix. The tests were stopped as soon as $\|R_k\|_F/\|R_0\|_F \leqslant 10^{-7}$.

Example 1. As a first test problem, we consider the convection diffusion equation with the Dirichlet boundary conditions

$$\begin{cases} L(u) := -\Delta u + 2v\frac{\partial u}{\partial x} + 2v\frac{\partial u}{\partial y} = f \quad \text{on } \Omega, \\ u = g \quad \text{on } \partial\Omega. \end{cases} \tag{18}$$

Here Ω is the rectangular domain $\Omega = [a,b] \times [c,d]$. The operator L was discretized using central finite differences on Ω, with mesh sizes $h = (b-a)/(n+1)$ in the "x" direction and $k = (d-c)/(p+1)$ in the "y" direction. The nodes of the discretization are $x_i = a + (i-1)h$, for $i = 1,\ldots,n+2$ and $y_j = c + (j-1)k$, for $j = 1,\ldots,p+2$. This yields two tridiagonal matrices A and B given by

$$A = -\frac{1}{h^2}\text{tridiag}((1+vh), -2, (1-vh)), \quad B = \frac{1}{k^2}\text{tridiag}((1+vk), -2, (1-vk)).$$

The right hand side matrix C was

$$\begin{aligned}
&C(i,j) = f(x_{i+1}, y_{j+1}) \quad \text{for } i = 2,\ldots,n-1,\ j = 2,\ldots,p-1,\\
&C(1,1) = f(x_2, y_2) + \tfrac{1+yh}{h^2}g(a, y_2) + \tfrac{1+yk}{k^2}g(x_2, c),\\
&C(1,p) = f(x_2, y_{p+1}) + \tfrac{1+yh}{h^2}g(a, y_{p+1}) + \tfrac{1-yk}{k^2}g(x_2, d),\\
&C(n,1) = f(x_{n+1}, y_2) + \tfrac{1-yh}{h^2}g(b, y_2) + \tfrac{1+yk}{k^2}g(x_{n+1}, c),\\
&C(n,p) = f(x_{n+1}, y_{p+1}) + \tfrac{1-yh}{h^2}g(b, y_{p+1}) + \tfrac{1-yk}{k^2}g(x_{n+1}, d),\\
&C(1,j) = f(x_2, y_{j+1}) + \tfrac{1+yh}{h^2}g(a, y_{j+1}) \quad \text{for } j = 2,\ldots,p-1,\\
&C(n,j) = f(x_{n+1}, y_{j+1}) + \tfrac{1-yh}{h^2}g(b, y_{j+1}) \quad \text{for } j = 2,\ldots,p-1,\\
&C(i,1) = f(x_{i+1}, y_2) + \tfrac{1+yk}{k^2}g(x_{i+1}, c) \quad \text{for } i = 2,\ldots,n-1,\\
&C(i,p) = f(x_{i+1}, y_{p+1}) + \tfrac{1-yk}{k^2}g(x_{i+1}, d) \quad \text{for } i = 2,\ldots,n-1.
\end{aligned}$$

In our example, the functions f and g were chosen such that the exact solution is

$$u(x,y) = x\mathrm{e}^{-x^2-y^2}$$

on the domain $\Omega = [0,1] \times [0,1]$.

In Fig. 1, we plotted the true residual norms, in a logarithmic scale, versus the number of iterations for GlGMRES(5) (dotted line) and the SSOR-GlGMRES(5) (solid line). These results were obtained for $v = 100$, $n = 600$ and $p = 300$; for this experiment, the SSOR preconditioner was used with the parameter $\omega = 0.9$. The relative error norm was

$$E_r(u-\tilde{u}) = \sqrt{\frac{\sum_{i=1}^{n+2}\sum_{j=1}^{p+2}|u(x_i,y_j) - \tilde{u}(x_i,y_j)|^2}{\sum_{i=1}^{n+2}\sum_{j=1}^{p+2}|u(x_i,y_j)|^2}} = 8.9 \times 10^{-5},$$

where u is the exact solution and $\tilde{u}$ is the obtained approximate solution on the domain Ω.

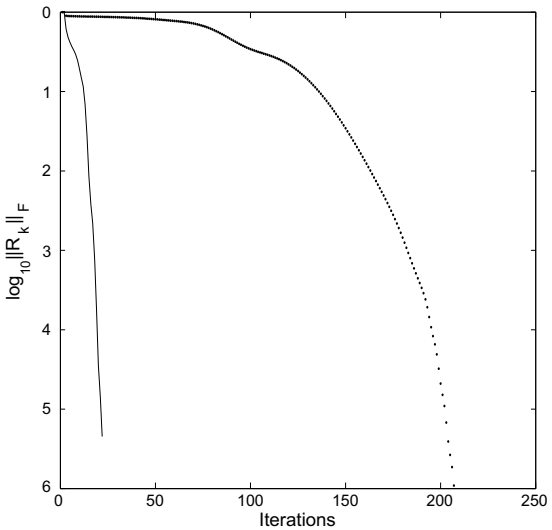

Fig. 1. Residual norms for GlGMRES(5) and SSOR-GlGMRES(5).

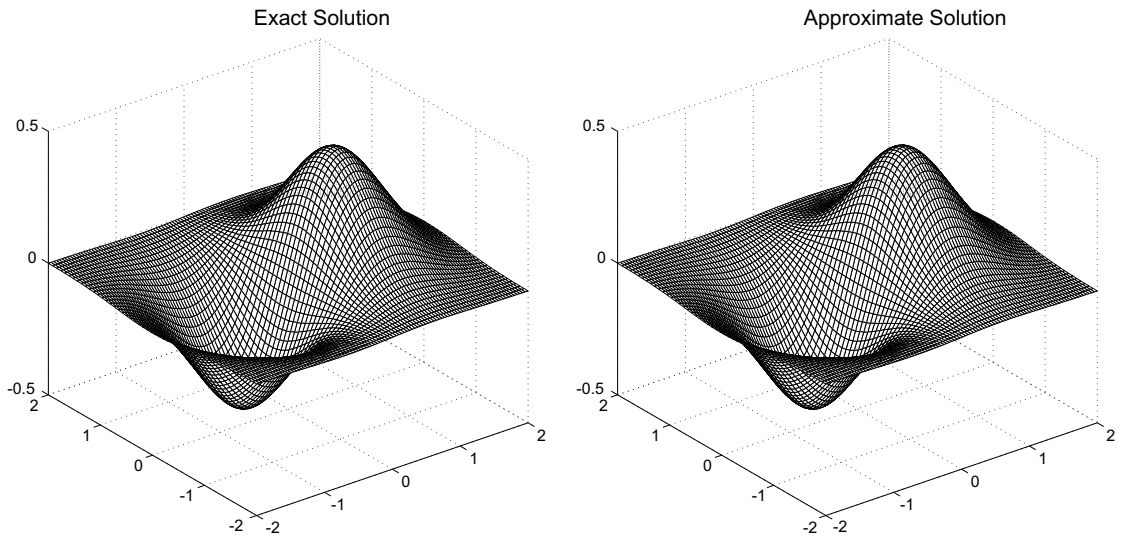

Fig. 2. Exact solution (left); Approximate solution (right).

Fig. 2 shows the exact solution and the approximate solution obtained by using the GlGMRES(5) with the SSOR preconditioner for $n = 100$, $p = 50$ and $\omega = 0.5$.

Example 2. In this experiment, we compared the performances of the GlGMRES(5) and the ILU-GlGMRES(5) algorithms for solving the Sylvester matrix Eq. (10). The matrices A and B are generated from the 5-point discretization of the operators L_1 and L_2, respectively, on the unit square $[0,1] \times [0,1]$ with homogeneous Dirichlet boundary conditions:

$$L_i(u) = \Delta u - f_{1,i}(x,y)\frac{\partial u}{\partial x} - f_{2,i}(x,y)\frac{\partial u}{\partial y} - f_{3,i}(x,y)u, \quad i = 1,2.$$

We set $f_{1,1}(x,y) = e^{x^2+y}$, $f_{1,2}(x,y) = 2xy$, $f_{1,3}(x,y) = \cos(xy)$, $f_{2,1}(x,y) = \sin(x+2y)$, $f_{2,2}(x,y) = e^{xy}$ and $f_{2,3}(x,y) = xy$.

The entries of the matrix C were random values uniformly distributed on $[0,1]$. The dimensions of the matrices A and B are $n = n_0^2$ and $p = p_0^2$ respectively, where n_0 and p_0 are the number of inner grid points in each direction. For this experiment we used $n = 16.000$, $p = 25$, which corresponds to a very large linear system of dimension $4.10^5 \times 4.10^5$. A maximum of 300 iterations was allowed to the two algorithms. In Fig. 3, we plotted the residual norms (in a logarithm scale) obtained by GlGMRES(5) (dotted line) and ILU-GlGMRES(5) (solid line).

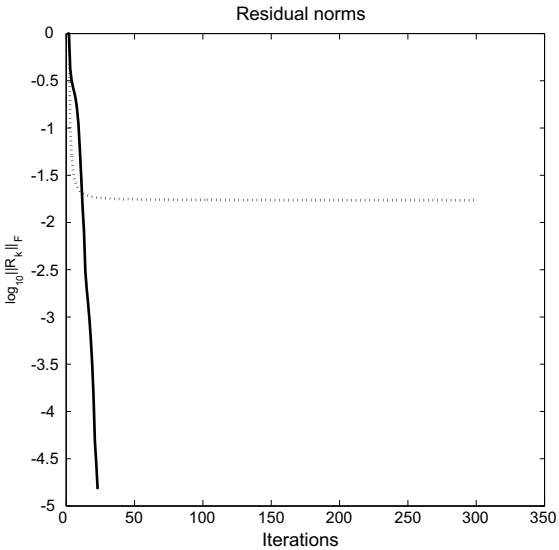

Fig. 3. Residual norms for GlGMRES(5) and ILU-GlGMRES(5).

6. Conclusion

In this paper, we proposed an iterative projection method for solving generalized Sylvester matrix equations. We also applied the ILUT and the SSOR preconditioners to accelerate the convergence of our proposed algorithm. Numerical examples for some PDE's show that the method is interesting.

Acknowledgements

We thank the anonymous referees for their constructive comments and helpful suggestions.

References

[1] R.H. Bartels, G.W. Stewart, Algorithm 432: Solution of the matrix equation $AX + XB = C$, Circ. Syst. Signal Proc. 13 (1994) 820–826.
[2] A. Bouhamidi, K. Jbilou, Sylvester Tikhonov-regularization methods in image restoration 206 (2007) 86–98.
[3] D. Calvetti, L. Reichel, Application of ADI iterative methods to the restoration of noisy images, SIAM J. Matrix Anal. Appl. 17 (1996) 165–186.
[4] D. Calvetti, N. Levenberg, L. Reichel, Iterative methods for $X - AXB = C$, J. Comput. Appl. Math. 86 (1997) 73–101.
[5] E.B. Castelan, V. Gomes da Silva, On the solution of a Sylvester equation appearing in descriptor systems control theory, Syst. Control Lett. 54 (2) (2005) 109–117.
[6] F. Chatelin, Valeurs Propres de Matrices, Masson, Paris, 1988.
[7] G.H. Golub, S. Nash, C. Van Loan, A Hessenberg–Schur method for the problem $AX + XB = C$, IEEC Trans. Autom. Contr. AC-24 (1979) 909–913.
[8] B.N. Datta, Numerical Methods for Linear Control Systems Design and Analysis, Elsevier Press, 2003.
[9] D.Y. Hu, L. Reichel, Krylov subspace methods for the sylvester equation, Linear Algebra Appl. 174 (1992) 283–314.
[10] K. Jbilou, A. Messaoudi, H. Sadok, Global FOM and GMRES algorithms for matrix equations, Appl. Numer. Math. 31 (1999) 49–63.
[11] K. Jbilou, A. Riquet, Projection methods for large Lyapunov matrix equations, Linear Algebra Appl. 415 (2006) 344–358.
[12] K. Jbilou, Low rank approximate solutions to large Sylvester matrix equations, Appl. Math. Comput. 177 (2006) 365–376.
[13] B.C. Moore, Principal component analysis in linear systems: controllability, observability and modessssssml reduction, IEEE Trans. Auto. Control AC 26 (1981) 17–31.
[14] Y. Saad, Iterative Methods for Sparse Linear Systems, PWS Press, New York, 1995.
[15] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (1986) 856–869.
[16] P. Van Dooren, Gramian Based Model Reduction of Large-scale Dynamical Systems, In Numerical Analysis, Chapman and Hall/CRC Press London, 2000. pp. 231–247.
[17] B. Zhou, G.R. Duan, An explicit solution to the matrix equation $AX - XF = BY$, Linear Algebra Appl. 402 (1) (2005) 345–366.
[18] B. Zhou, G.R. Duan, A new solution to the generalized Sylvester matrix equation $AV - EVF = BW$, Syst. Control Lett. 55 (3) (2006) 193–198.
[19] B. Zhou, G.R. Duan, On the generalized Sylvester mapping and matrix equations, Syst. Control Lett. 57 (3) (2008) 200–208.