



Block Krylov subspace methods for large-scale matrix computations in control

A.H. Refahi Sheikhani*, S. Kordrostami

Department of Mathematics, Faculty of Mathematical Sciences, Islamic Azad University, Lahijan Branch, Lahijan, Iran

Available online 4 August 2014

Abstract

In this paper we show how to improve the approximate solution of the large Sylvester equation obtained by an arbitrary method. Such problems appear in many areas of control theory such as the computation of Hankel singular values, model reduction algorithms and others. Moreover, we propose a new method based on refinement process and weighted block Arnoldi algorithm for solving large Sylvester matrix equation. The numerical tests report the effectiveness of these methods.

© 2014 Taibah University. Production and hosting by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/3.0/>).

MSC: 65F30; 65F50

Keywords: Matrix equations; Sylvester; Arnoldi; Refinement

1. Introduction

An important class of linear matrix equations is the Sylvester equation

$$XA + BX = C. \quad (1.1)$$

Since the fundamental work of Sylvester on the stability of the motion, these matrix equations have been widely used in stability theory of differential equations and play an important role in control and communications theory [1–5]. For small problems, direct methods for solving the Sylvester equation are attractive. The standard methods are based on the Hessenberg–Schur decomposition to transform the original equation into a form that can be easily solved. Iterative methods for solving large Sylvester matrix equations have been developed during the past years [6–8]. A class of classical methods known as the Krylov subspace methods, that include the block Arnoldi and weighted block Arnoldi, etc. have been found to be suitable for sparse matrix computations. In this paper, we extend the idea to propose a new projection method for solving (1.1) based on weighted block Krylov subspace method. The paper is organized as

* Corresponding author. Tel.: +98 9113475547.

E-mail addresses: ah_refahi@guilan.ac.ir (A.H. Refahi Sheikhani), krostami@guilan.ac.ir (S. Kordrostami).

Peer review under responsibility of Taibah University.



follows. In Section 2, we introduce the main minimal residual algorithm for iteratively solving the Sylvester matrix equation (1.1). Several numerical examples presented in Section 3. Finally, the conclusion is given in the last section.

2. Block refinement Arnoldi method

In this section we propose to show that the obtained approximate solution of the Sylvester equation by any method can be improved, in other words the accuracy can be increased. If this idea is applicable then we have found an iterative method for solving of the Sylvester equation. Therefore let the basis $V_m = [v_1, \dots, v_m]$ and $W_m = [w_1, \dots, w_m]$ constructed by the block Arnoldi process, thus we have

$$V_m^T V_m = I_m \quad W_m^T W_m = I_m$$

The square block Hessenberg matrices H_m and \hat{H}_m ($m = r * l$ where r and l are the dimensions of blocks) whose nonzero entries are the scalars h_{ij} and \hat{h}_{ij} , constructed by the block Arnoldi process can be expressed as

$$H_m = V_m^T A^T V_m \quad \hat{H}_m = W_m^T B W_m$$

Let $X^{(0)}$ be an initial approximate solution of the Sylvester equation $XA + BX = C$ where $A, B, C, X^{(0)} \in R^{n \times n}$.

Also introduce the residual matrix

$$R^{(0)} = C - (X^{(0)}A + BX^{(0)})$$

And let $Y_m \in R^{m \times m}$ be the solution of the Sylvester equation:

$$Y_m H_m^T + \hat{H}_m Y_m = W_m^T R^{(0)} V_m \tag{2.1}$$

If set

$$X^{(1)} = X^{(0)} + W_m Y_m V_m^T \tag{2.2}$$

then the corresponding residual $R^{(1)} = C - (X^{(1)}A + BX^{(1)})$ satisfies:

$$\begin{aligned} R^{(1)} &= C - ((X^{(0)} + W_m Y_m V_m^T)A + B(X^{(0)} + W_m Y_m V_m^T)) \\ &= R^{(0)} - W_m Y_m V_m^T A - B W_m Y_m V_m^T \\ &= R^{(0)} - W_m Y_m H_m^T V_m^T - W_m \hat{H}_m Y_m V_m^T \\ &= R^{(0)} - W_m (Y_m H_m^T + \hat{H}_m Y_m) V_m^T \end{aligned}$$

Since Y_m is the solution of (2.1) we have:

$$R^{(1)} = R^{(0)} - W_m W_m^T R^{(0)} V_m V_m^T = 0$$

According to the above results we can develop an iterative method for the solving of the Sylvester equation when the matrices A, B and C are large and sparse. For doing this idea if we choose $m < n$, then instead of solving $XA + BX = C$ we can solve (2.1). In other words in this method, first we transform the initial Sylvester equation to another Sylvester equation with less dimensions, then in each iteration step solve this matrix equation and extend the obtained solution to the solution of initial equation by (2.2). The algorithm is as follows:

Algorithm 1 (*Block refinement Arnoldi method (BRA)*). Choose an initial solution $X^{(0)}$, and a tolerance ϵ and select two numbers r and l for dimensions of block and set $m = r * l (m < n)$.

For $k = 0 \dots$ until Convergence.

$$R^{(k)} = C - (X^{(k)}A + BX^{(k)}).$$

Construct the orthonormal basis V_m and $W_m \in R^{n \times m}$ by the block Arnoldi process

$$H_m = V_m^T A^T V_m \quad \hat{H}_m = W_m^T B W_m.$$

Solve the reduced Sylvester equation

$$Y_m H_m^T + \hat{H}_m Y_m = W_m^T R^{(k)} V_m.$$

$$X^{(k+1)} = X^{(k)} + W_m Y_m V_m^T.$$

$$\text{if } \frac{\|X^{(k+1)} - X^{(k)}\|}{\|X^{(k)}\|} \leq \epsilon \text{ Stop}$$

else End.

Table 1
Implementation of iterative BRA method to solve the Sylvester equation with different values of m .

m	r	l	$\ V_m^T A^T V_m - H_m\ $	$\ W_m^T B W_m - \hat{H}_m\ $	Iteration	Time
4	2	2	8.63E-014	4.05E-014	278	5.83
8	2	4	1.57E-013	5.49E-014	156	4.68
10	2	5	2.81E-013	1.98E-014	95	3.56
20	2	10	2.84E-013	1.77E-013	58	2.68
30	2	15	3.05E-013	4.46E-014	36	1.74
40	2	20	9.77E-013	8.36E-014	21	0.7811
50	2	25	3.17E-012	4.68E-013	2	0.0918

Table 2
Implementation of new Iterative methods and Hessenberg–Schur method for solving the Sylvester equation.

n	Hessenberg–Schur method		Weighted Krylov method		BRA method		cond(B)
	Error	Time	Error	Time	Error	Time	
200	1.16E-010	0.6881	2.50E-012	0.4753	2.38E-014	0.2612	8.53E+003
400	3.89E-007	6.312	4.22E-008	4.642	6.15E-014	3.134	3.17E+004
600	0.0011	28.89	0.0033	65.76	6.95E-014	21.39	7.63E+005
800	8.931	85.74	13.01	174.32	8.37E-014	58.26	2.13E+007
1000	27.35	201.14	48.19	322.11	1.84E-013	121.53	1.50E+008

3. Numerical examples

In this section, we present some numerical examples to illustrate the effectiveness of algorithms described in this paper for large and sparse Sylvester equations. All numerical tests are performed in MATLAB software on a PC with 2.20 GHz with main memory 2 GB.

Example 3.1. Consider the Sylvester equation $XA + BX = C$ with $n = 100$. We apply the Iterative block refinement Arnoldi method for solving this matrix equation and take $\epsilon = 10^{-6}$. In Table 1, we report the results for different values of m .

$$A = \begin{pmatrix} 10 & 1.2 & .42 & .8 & 2.3 & .8 & 0 & \dots & \dots & 0 \\ 1.8 & 10 & 1.2 & .42 & .8 & 2.3 & .8 & 0 & \ddots & 0 \\ 1.6 & 1.8 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots \\ 1.64 & 1.6 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .8 & 0 \\ 1.3 & 1.64 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 2.3 & .8 \\ 1.61 & 1.3 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .8 & 2.3 \\ 0 & 1.61 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .42 & .8 \\ \vdots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 1.2 & .42 \\ \vdots & \vdots & 0 & 1.61 & 1.3 & 1.64 & 1.6 & 1.8 & 10 & 1.2 \\ 0 & 0 & \dots & 0 & 1.61 & 1.3 & 1.64 & 1.6 & 1.8 & 10 \end{pmatrix}_{n \times n}$$

$$B = \begin{pmatrix} 10 & 2.1 & .38 & .7 & 1.5 & .4 & 0 & \dots & \dots & 0 \\ 1.21 & 10 & 2.1 & .38 & .7 & 1.5 & .4 & 0 & \ddots & 0 \\ 1.9 & 1.21 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots \\ .64 & 1.9 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .4 & 0 \\ 1.9 & .64 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 1.5 & .4 \\ .87 & 1.9 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .7 & 1.5 \\ 0 & .87 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .38 & .7 \\ \vdots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 2.1 & .38 \\ \vdots & \vdots & 0 & .87 & 1.9 & .64 & 1.9 & 1.21 & 10 & 2.1 \\ 0 & 0 & \dots & 0 & .87 & 1.9 & .64 & 1.9 & 1.21 & 10 \end{pmatrix}_{n \times n}$$

$$C = \begin{pmatrix} .1 & 2.21 & 1.4 & 1.5 & .13 & 2.62 & 0 & \dots & \dots & 0 \\ 1.3 & .1 & 2.21 & 1.4 & 1.5 & .13 & 2.62 & 0 & \ddots & 0 \\ 2.6 & 1.3 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots \\ 1.7 & 2.6 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 2.62 & 0 \\ 2.3 & 1.7 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .13 & 2.62 \\ 2.6 & 2.3 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 1.5 & .13 \\ 0 & 2.6 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 1.4 & 1.5 \\ \vdots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 2.21 & 1.4 \\ \vdots & \vdots & 0 & 2.6 & 2.3 & 1.7 & 2.6 & 1.3 & .1 & 2.21 \\ 0 & 0 & \dots & 0 & 2.6 & 2.3 & 1.7 & 2.6 & 1.3 & .1 \end{pmatrix}_{n \times n}$$

Example 3.2. Now consider A and B and C are the same matrices that used in [Example 3.1](#). We apply two Iterative methods and Hessenberg–Schur method to solve the Sylvester equation when the dimension of the matrices are large. Results are shown in [Table 2](#).

4. Comments and conclusion

In this paper, we introduced a new method for computing low-rank approximate solutions to large Sylvester matrix equations. Moreover Refinement process presented in [Section 2](#) has the capability of improving the results obtained by an arbitrary method. For example in this paper we apply the refinement process with Hessenberg–Schur method. The experiments illustrate the advantages of the method for large sparse matrices.

References

- [1] B.N. Datta, *Numerical Methods for Linear Control Systems: Design and Analysis*, Elsevier Academic Press, Amsterdam, 2004.
- [2] B.N. Datta, K. Datta, Theoretical and computational aspects of some linear algebra problems in control theory, in: C.I. Byrnes, A. Lindquist (Eds.), *Computational and Combinatorial Methods in Systems Theory*, Elsevier, Amsterdam, 1986, pp. 201–212.

- [3] N.J. Higham, *Accuracy and Stability of Numerical Algorithms*, second ed., SIAM, Philadelphia, PA, 2002.
- [4] C. Hyland, D. Bernstein, The optimal projection equations for fixed-order dynamic compensation, *IEEE Trans. Autom. Control* 29 (1984) 1034–1037.
- [5] R.H. Bartels, G.W. Stewart, Solution of the matrix equation $AX + XB = C$, *Commun. ACM* 15 (1972) 820–826.
- [6] G.H. Golub, S. Nash, C.F. Van Loan, A Hessenberg–Schur method for the problem $AX + XB = C$, *IEEE Trans. Autom. Control* 24 (1979) 909–913.
- [7] G.H. Golub, C.F. Van Loan, *Matrix Computations*, third ed., Johns Hopkins U.P., Baltimore, 1996.
- [8] A.El. Guennouni, K. Jbilou, A.J. Riquet, Block Krylov subspace methods for solving large Sylvester equations, *Numer. Alg.* 29 (2002) 75–96.