Hindawi Publishing Corporation
Advances in Numerical Analysis
Volume 2011, Article ID 184314, 15 pages
doi:10.1155/2011/184314

Research Article

Refinement Methods for State Estimation via Sylvester-Observer Equation

H. Saberi Najafi and A. H. Refahi Sheikhani

Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Guilan, P.O. Box 1914, Rasht, Iran

Correspondence should be addressed to H. Saberi Najafi, hnajafi@guilan.ac.ir

Received 6 December 2010; Revised 26 April 2011; Accepted 14 June 2011

Academic Editor: William J. Layton

We present new iterative methods based on refinement process for solving large sparse Sylvester-observer equations applied in state estimation of a continuous-time system. These methods use projection methods to produce low-dimensional Sylvester-observer matrix equations that are solved by the direct methods. Moreover, the refinement process described in this paper has the capability of improving the results obtained by any other methods. Some numerical results will be reported to illustrate the efficiency of the proposed methods.

1. Introduction

Consider the continuous-time linear system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \tag{1.1}$$

where $A \in R^{n\times n}$, $B \in R^{n\times m}$, and $C \in R^{r\times n}$.

We well know that all the state-feedback problems, such as the feedback stabilization, the LQR, and the state-feedback H_∞-control problems, require the state vector $x(t)$ explicitly [1]. In most practical situations, the states are not fully accessible and the designer only knows the output and the input vectors. The unavailable states somehow need to be estimated accurately from the knowledge of the matrices A, B, and C, the output vector $y(t)$, and the input vector $u(t)$. There are two common procedures for state estimation: one via eigenvalue assignment (EVA) and the other via solution of the Sylvester-observer equation.

The main step in state estimation via solution of the Sylvester-observer equation is solving the Sylvester-observer equation of the form

$$XA - FX = GC, \tag{1.2}$$

where X, A, $F \in R^{n\times n}$, $G \in R^{n\times r}$, and $C \in R^{r\times n}$. Sylvester-observer equations (1.2) play vital roles in a number of applications such as control and communications theory [1], model reduction [2–4], numerical solution of matrix differential Riccati equations [5], and image processing [6].

The analytical solution of the matrix equation (1.2) has been considered by many authors; see [7, 8]. Direct methods for solving the matrix equation (1.2) are attractive if the matrices are of small size. These methods are based on the Schur decomposition, by which the original equation is transformed into a form that is easy to be solved by a forward substitution. Moreover, the matrices of the large practical problem are very sparse. Since, the standard methods for solving the Sylvester equations destroy the sparsity of the problems, they are only applicable for the matrices of small size; see [9–11]. Iterative projection methods for solving large Sylvester-observer matrix equations have been developed during the past years; see [12–16]. These methods use the Arnoldi process to compute an orthonormal basis of certain Krylov subspace. In this paper, we extend the idea to propose a new projection method for solving (1.1) based on Weighted block Krylov subspace process. These methods are based on the reduction of the large sparse Sylvester-observer equation to the low-dimensional problem by orthogonal similarity that is solved by the direct methods. Moreover, refinement process presented in Sections 4 and 5 has the capability of improving the results obtained by any other methods.

The remainder of the paper is organized as follows. In Section 2, we describe some fundamental results about control theory. Then, we discuss how the states of a continuous-time system can be estimated in Section 3. In Sections 4 and 5, we introduce two new iterative methods for solving large sparse Sylvester-observer equation. These methods are based on the reduction of the large sparse Sylvester-observer equation to the low-dimension problem. Section 6 is devoted to some numerical tests. Some concluding remarks are given in Section 7.

2. Some Fundamental Results

The two basic concepts in control theory are controllability and observability of a control system. In this section, we will state some well-known facts about controllability and observability for convenient use later in the paper. For an excellent account of controllability and observability and related results, see [1].

Definition 2.1. The system (1.1) is said to be controllable if, starting from any initial state $x(0)$, the system can be driven to any final state x_f in some finite time t_f, choosing the input vector $u(t)$, $0 \le t \le t_f$, appropriately.

Observability is the dual concept of controllability.

Definition 2.2. The system (1.1) is said to be observable if there exists $t_1 > 0$ such that the initial state $x(0)$ can be uniquely determined from the knowledge of $u(t)$ and $y(t)$ for all $0 \le t \le t_1$.

Remark 2.3. Since matrix C does not have any role in the definition of controllability, the controllability of the system (1.1) (see [1]), is often referred to as the controllability of the pair (A, B). Similarly, since matrix B does not have any role in the definition of observability, the observability of the system (1.1) is often referred to as the observability of the pair (A, C).

Some well-known criteria of controllability and observability are now stated in the next two theorems. The proofs of Theorems 2.4 and 2.5 can be found in [1]. In the following, A is $n \times n$, B is $n \times m$ $(m \leq n)$, and C is $r \times n$ $(r \leq n)$.

Theorem 2.4. *The pair (A, B) is controllable if and only if any one of the following equivalent conditions holds.*

(1) *The controllability matrix*

$$C_M = \left(B, AB, A^2B, \ldots, A^{n-1}B\right), \tag{2.1}$$

has rank n.

(2) *Rank $(A - \lambda I, B) = n$ for every eigenvalue λ of A.*

(3) *Let (λ, x) be an eigenpair of A, that is, $Ay = \lambda y$; then $Cy \neq 0$.*

Proof. See [1]. □

Theorem 2.5. *The pair (A, C) is observable if and only if any one of the following equivalent conditions holds*

(1) *The observability matrix*

$$O_M = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix} \tag{2.2}$$

has rank n.

(2) *The matrix $\binom{\lambda I - A}{C}$ has rank n for every eigenvalue λ of A.*

(3) *Let (λ, y) be an eigenpair of A, that is, $Ay = \lambda y$; then $Cy \neq 0$.*

Proof. See [1]. □

Definition 2.6. A matrix A is called a stable matrix if all of the eigenvalues of A have negative real parts.

3. State Estimation

In this section, we discuss how the states of a continuous-time linear system (1.1) can be estimated. The discussions here apply equally to the discrete-time systems possibly with some minor changes. So we concentrate on the continuous-time case only. We describe two common procedures for state estimation: one via eigenvalue assignment (EVA) and the other,via solution of the Sylvester-observer equation.

3.1. State Estimation via Eigenvalue Assignment

Now consider the linear time-invariant system (1.1). Let $\hat{x}(t)$ be an estimate of the state vector $x(t)$. Obviously, we would like to construct the vector $\hat{x}(t)$ in such a way that the error $e(t) = x(t) - \hat{x}(t)$ approaches zero as fast as possible, for all initial states $x(0)$ and for every input $u(t)$. The following theorem shows that the problem of state estimation can be solved by finding a matrix K such that the matrix $A - KC$ has a suitable desired spectrum. See [1].

Theorem 3.1. *If (A, C) is observable, then the states $x(t)$ of the system* (1.1) *can be estimated by*

$$\dot{\hat{x}}(t) = (A - KC)\hat{x}(t) + Ky(t) + Bu(t), \tag{3.1}$$

where K is constructed such that $A - KC$ is a stable matrix. The error $e(t) = x(t) - \hat{x}(t)$ is governed by

$$\dot{e}(t) = (A - KC)e(t), \tag{3.2}$$

and $e(t) \to 0$ as $t \to \infty$.

Proof. See [1]. □

3.2. State Estimation via Sylvester Equation

There is another approach for state estimation. Knowing A, B, C, $u(t)$, and $y(t)$, let us construct the system

$$\dot{z}(t) = Fz(t) + Gy(t) + Pu(t), \tag{3.3}$$

where F is $n \times n$, G is $n \times r$, and P is $n \times m$, in such a way that for some constant $n \times n$ nonsingular matrix X the error vector $e(t) = z(t) - Xx(t) \to 0$ for all $x(0)$, $z(0)$, and for every input $u(t)$. The vector $z(t)$ will then be an estimate of $Xx(t)$. The system (3.3) is then said to be the *state observer* for the system (1.1). D. Luenberger originated the idea and is hence referred to in control theory as the Luenberger observer; see [1].

We now show that the system (3.3) will be a state observer if the matrices X, F, G, and P satisfy certain requirements.

Theorem 3.2. *The system* (3.3) *is a state observer of the system* (1.1)*, that is,* $z(t)$ *is an estimate of* $Xx(t)$ *in the sense that the error* $e(t) = z(t) - Xx(t) \to 0$ *as* $t \to \infty$ *for any initial conditions* $x(0)$*,* $z(0)$*, and* $u(t)$ *if*

(1) $XA - FX = GC$,

(2) $P = XB$,

(3) F *is stable.*

Proof. See [1]. □

Definition 3.3. The matrix equation

$$XA - FX = GC, \tag{3.4}$$

where X, A, $F \in R^{n\times n}$, $G \in R^{n\times r}$, and $C \in R^{r\times n}$, is called the *Sylvester-observer* equation.

Theorem 3.2 suggests the following method for the observer design; see [1].

Algorithm 3.4 (full-order observer design via Sylvester-observer equation). One has the following.

Inputs

The system matrices A, B, and C of order $n \times n$, $n \times m$, and $r \times n$, respectively.

Output

An estimate $\hat{x}(t)$ of the state vector $x(t)$.

Assumptions 1. (A, C) is observable.

Step 1. Find a nonsingular solution of the Sylvester-observer equation (1.2) by choosing F as a stable matrix and choosing G in such a way that the resulting solution X is nonsingular.

Step 2. Compute $P = XB$.

Step 3. Construct the observer $z(t)$ by solving the system of differential equations

$$\hat{z}(t) = Fz(t) + Gy(t) + Pu(t), \quad z(0) = z_0. \tag{3.5}$$

Step 4. Find an estimate $\hat{x}(t)$ of $x(t)$: $\hat{x}(t) = X^{-1}z(t)$.

3.3. Characterization of Nonsingular Solutions of the Sylvester Equation

In this section, we describe some necessary conditions for a unique solution of the Sylvester equation to have such properties. The following theorem was proved by Bhattacharyya and de Souza. The proof here has been taken from [1].

Theorem 3.5. *Let A, F, G, and C, respectively, be of order $n \times n$, $n \times n$, $n \times r$, and $r \times n$. Let X be a unique solution of the Sylvester-observer equation* (1.2)*. Then, necessary conditions for X to be nonsingular are that (A, C) is observable and (F, G) is controllable.*

Proof. See [1]. □

Corollary 3.6. *If G is $n \times 1$ and C is $1 \times n$, then necessary and sufficient conditions for the unique solution X of* (1.2) *to be nonsingular are that (F, G) is controllable and (A, C) is observable.*

Proof. See [1]. □

Remark 3.7. According to Theorem 3.5 and Corollary 3.6, the controllability of (F, G) and observability of (A, C) guarantee the existence of a nonsingular solution of the Sylvester-observer equation (1.2) in Step 1 of Algorithm 3.4. Moreover, there are other choices for F and G in Step 1 of Algorithm 3.4 provided (F, G) is controllable. Also, we can use Theorems 2.4 and 2.5 for analyzing the controllability of (F, G) and the observability of (A, C).

Example 3.8. In this example we show how the Sylvester-observer equation (1.2) can be applied in state estimation of a continuous time system (1.1). In this sense, at first we use the MATLAB function *ode23* for directly solving (1.1) with $u(t)$ as the unit step function and $x(0) = (6, 0)^T$. Then, we apply Algorithm 3.4 for computing the estimate $\widehat{x}(t)$. Also, the differential equation (3.12) was solved with $z(0) = 0$ and MATLAB function *ode23*. The comparison of the state $x(t)$ and estimate $\widehat{x}(t)$ for this example is shown in Figure 1. The solid line corresponds to the exact state and the dotted line corresponds to the estimate state. Let

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \end{pmatrix}. \tag{3.6}$$

According to criteria 1 of Theorem 2.5, (A, C) is observable. Thus, we can use Algorithm 3.4 for state estimation of (1.1).

Step 5. Choose

$$G = \begin{pmatrix} 1 \\ 3 \end{pmatrix}, \qquad F = \operatorname{diag}(-1, -3). \tag{3.7}$$

According to criteria 1 of Theorem 2.4, (F, G) is controllable. Thus, by Corollary 3.6, the nonsingular solution X of $XA - FX = GC$ is

$$X = \begin{pmatrix} 0.6667 & -0.3333 \\ 0.8000 & -0.2000 \end{pmatrix}, \tag{3.8}$$

(computed by MATLAB function *lyap*).

Step 6. One has

$$P = XB = \begin{pmatrix} 0.6667 \\ 0.8000 \end{pmatrix}. \tag{3.9}$$

Step 7. An estimate $\hat{x}(t)$ of $x(t)$ is

$$\hat{x}(t) = X^{-1}z(t) = \begin{pmatrix} -1.5 & 2.5 \\ -6 & 5 \end{pmatrix} \begin{pmatrix} z_1(t) \\ z_2(t) \end{pmatrix} = \begin{pmatrix} -1.5z_1 + 2.5z_2 \\ -6z_1 + 5z_2 \end{pmatrix}, \tag{3.10}$$

where

$$z(t) = \begin{pmatrix} z_1(t) \\ z_2(t) \end{pmatrix}, \tag{3.11}$$

is given by

$$\hat{z}(t) = \begin{pmatrix} -1 & 0 \\ 0 & -3 \end{pmatrix} z(t) + \begin{pmatrix} 1 \\ 3 \end{pmatrix} y(t) + \begin{pmatrix} 0.6667 \\ 0.8000 \end{pmatrix} u(t), \quad z(0) = z_0. \tag{3.12}$$

Remark 3.9. According to Algorithm 3.4 the most important step is Step 1. In the case that n is small, there are many reliable algorithm s for solving the Sylvester-observer equation and the states of a continuous-time system can be estimated. However, for large and sparse systems solving the Sylvester-observer equation by the available methods can be costly. In Sections 4 and 5, we introduce two iterative refinement methods for solving large sparse Sylvester-observer equations.

4. Block Refinement Method

As we already mentioned, so far many numerical methods have been developed by different authors; see [1, 13, 17]. For example, the Hessenberg-Schur method is now widely used as an effective computational method for the Sylvester-observer equation. But numerical stability of this method has not been investigated. As the iterative methods are very efficient for the solution of computational problems, we thought it will be good idea to create an iterative method for solving the Sylvester-observer equation $XA - FX = GC$ where $A, F, X \in R^{n\times n}$, $G \in R^{n\times r}$, and $C \in R^{r\times n}$. In this section we propose to show that the obtained approximate solution of the Sylvester-observer 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-observer equation.

Theorem 4.1. *Let X_0 be the approximate solution obtained by an arbitrary method for the matrix equation* (1.2)*, and let $R(X_0) = GC - (X_0A - FX_0)$ be the corresponding residual. If m steps of the*

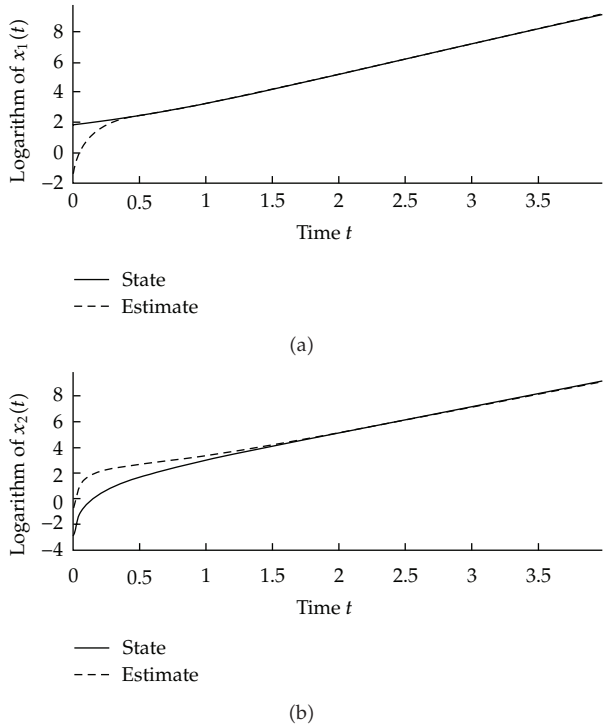

Figure 1: The (a) first and (b) second variables of the state $x(t)$ and estimate $\hat{x}(t)$ for Example 3.8.

block Arnoldi process for matrices A^T and F have been run ($m \ll n$) and $Y_m \in R^{m\times m}$ is the solution of the low-dimensional Sylvester equation

$$Y_m \widehat{A}_m^T - \widehat{F}_m Y_m = W_m^T R(X_0) V_m, \tag{4.1}$$

then

$$R(X_1) = R\left(X_0 + W_m Y_m V_m^T\right) = 0. \tag{4.2}$$

Proof. Let $V_m = [v_1, \ldots, v_m]$ and $W_m = [w_1, \ldots, w_m]$ be two orthogonal bases constructed by the block Arnoldi process for the matrices A^T and F, respectively. Thus, we have

$$V_m^T V_m = I_m, \qquad W_m^T W_m = I_m. \tag{4.3}$$

Also, the square block Hessenberg matrices $\widehat{A}_m$ and $\widehat{F}_m$ ($m = r\ell$, where r and ℓ are the dimensions of blocks) whose nonzero entries are the scalars $\widehat{a}_{ij}$ and $\widehat{f}_{ij}$, constructed by the Block Arnoldi process, can be expressed as

$$\widehat{A}_m = V_m^T A^T V_m, \qquad \widehat{F}_m = W_m^T F W_m. \tag{4.4}$$

If we set

$$X_1 = X_0 + W_m Y_m V_m^T, \tag{4.5}$$

then the corresponding residual $R(X_1) = GC - (X_1 A - F X_1)$ satisfies

$$\begin{aligned} R(X_1) &= GC - \left(\left(X_0 + W_m Y_m V_m^T\right) A - F\left(X_0 + W_m Y_m V_m^T\right)\right), \\ &= R(X_0) - W_m Y_m V_m^T A + F W_m Y_m V_m^T, \\ &= R(X_0) - W_m Y_m \widehat{A}_m^T V_m^T + W_m \widehat{F}_m Y_m V_m^T, \\ &= R(X_0) - W_m \left(Y_m \widehat{A}_m^T - \widehat{F}_m Y_m\right) V_m^T. \end{aligned} \tag{4.6}$$

Since Y_m is the solution of (4.1) and by using (4.3), we have

$$R(X_1) = R(X_0) - W_m W_m^T R_0 V_m V_m^T = 0. \tag{4.7}$$

□

According to Theorem 4.1, we can develop an iterative method for solving the Sylvester-observer equation when the matrices A, F, G, and C are large and sparse. For achieving this idea, if we choose $m < n$, then instead of solving $XA - FX = GC$ we can solve (4.1). In other words, in this method, first we transform the initial Sylvester-obsever equation to another Sylvester equation with less dimensions and then in each iteration step solve this matrix equation and extend the obtained solution to the solution of the initial equation by (4.5). The algorithm is as follows.

Algorithm 4.2 (block refinement method). (1) Start: choose an initial approximate solution X_0, and a tolerance ε.

(2) Select two numbers r and ℓ for dimensions of block and set $m = r\ell (m < n)$.

(3) Compute $R(X_0) = GC - (X_0 A - F X_0)$.

(4) Construct the orthonormal bases V_m and $W_m \in R^{n\times m}$ by the block Arnoldi process, such that

$$\widehat{A}_m = V_m^T A^T V_m, \qquad \widehat{F}_m = W_m^T F W_m. \tag{4.8}$$

(5) Solve the low-dimensional Sylvester-observer equation $Y_m \widehat{A}_m^T - \widehat{F}_m Y_m = W_m^T R(X_0) V_m$ with the Hessenberg-Schur method.

(6) Set $X_1 = X_0 + W_m Y_m V_m^T$.

(7) Compute $R(X_1) = GC - (X_1 A - F X_1)$.

(8) If $\|R(X_1)\|_F / \|R(X_0)\|_F \le \epsilon$ stop, otherwise set $X_0 = X_1$, $R(X_0) = R(X_1)$ and go to step (3).

Remark 4.3. By choosing $m < n$, Algorithm 4.2 reduces the original large sparse Sylvester-observer equation (1.2) to a low-dimensional Sylvester-observer equation (4.1). In step (5), we solve this low-dimensional matrix equation by any direct method such as the Hessenberg-Schur method. Then, in step (6) by using relation (4.5), we extend the obtained solution to the solution of the original matrix equation. Also, according to Theorem 4.1, Algorithm 4.2 is the convergence for any initial matrix X_0.

5. Weighted Block Refinement Method

In this section we discuss a new iterative method based upon a modified block refinement method. The new process uses instead of the Euclidean scalar product another one, denoted by $(\cdot,\cdot)_D$ where D is a chosen diagonal matrix. The idea of changing the inner product is to accelerate the convergence of the components of the residual which are far away from zero. To achieve this, an appropriate weight is associated to each term of the inner product. A natural choice of these weights is the entries of the first residual. The following method is based on reduction of A and F to the Hessenberg matrix with the use of weighted block Arnoldi process. Before giving a complete description of the new algorithm, let us define the D-scalar product.

If u and v are two vectors of R^n, their D-scalar product is

$$(u,v)_D = v^T D u = \sum_{i=1}^{n} d_i u_i v_i, \tag{5.1}$$

where $D = \operatorname{diag}(d_1, d_2, \ldots, d_n)$ is a diagonal matrix.

This inner product is well defined if and only if the matrix D is positive definite, that is, $d_i > 0$, for all $i \in \{1, \ldots, n\}$.

In this case, we can define the D-norm $\|\cdot\|_D$ associated with this inner product by

$$\|u\|_D = \sqrt{(u,u)_D} = \sqrt{u^T \mathrm{D} u} = \sqrt{\sum_{i=1}^{n} d_i u_i^2} \quad \forall u \in R^n. \tag{5.2}$$

Theorem 5.1. *Let X_0 be the approximate solution obtained by an arbitrary method for the matrix equation (1.2), and let $R(X_0) = GC - (X_0 A - F X_0)$ be the corresponding residual. If m ($m \ll n$) steps of the weighted block Arnoldi process by the diagonal matrices D and $\widehat{D}$, respectively, for matrices A and F have been run and $Y_m \in R^{m\times m}$ is the solution of the low-dimensional Sylvester equation*

$$Y_m \widehat{A}_m - \widehat{F}_m^T Y_m = V_m^T R(X_0) U_m, \tag{5.3}$$

then

$$V_m^T R(X_1) U_m = V_m^T R\Big(X_0 + \widehat{D} V_m Y_m U_m^T D\Big) U_m = 0. \tag{5.4}$$

Proof. By using the weighted Arnoldi process, we generate the bases $U_m = [u_1, \ldots, u_m]$ and $V_m = [v_1, \ldots, v_m]$ that are, respectively, D-orthonormal and $\widehat{D}$-orthonormal; thus it holds that

$$U_m^T D U_m = I_m, \qquad V_m^T \widehat{D} V_m = I_m, \tag{5.5}$$

where $U_m, V_m \in R^{n\times m}$ and $D, \widehat{D} \in R^{n\times n}$ are two diagonal matrices.

Moreover, the square Hessenberg matrices $\widehat{A}_m$ and $\widehat{F}_m$ whose nonzero entries are the scalars $\widehat{a}_{ij}$ and $\widehat{f}_{ij}$, constructed by the weighted Arnoldi process, can be expressed in the form

$$\widehat{A}_m = U_m^T D A U_m, \qquad \widehat{F}_m = V_m^T \widehat{D} F V_m. \tag{5.6}$$

Now, we set

$$X_1 = X_0 + E_0, \tag{5.7}$$

where $E_0 = \widehat{D} V_m Y_m U_m^T D$ and $Y_m \in R^{m\times m}$ is the solution of the Sylvester-obsever equation (5.3). Thus, the new residual matrix becomes

$$\begin{aligned} R(X_1) &= GC - (X_1 A - F X_1), \\ &= GC - ((X_0 + E_0)A - F(X_0 + E_0)), \\ &= R(X_0) - (E_0 A - F E_0), \\ &= R(X_0) - \left(\widehat{D} V_m Y_m U_m^T D A - F \widehat{D} V_m Y_m U_m^T D\right). \end{aligned} \tag{5.8}$$

Multiplying the above relation on the left by V_m^T and on the right by U_m, we have

$$V_m^T R(X_1) U_m = V_m^T R(X_0) U_m - \left(V_m^T \widehat{D} V_m Y_m U_m^T D A U_m - V_m^T F \widehat{D} V_m Y_m U_m^T D U_m\right). \tag{5.9}$$

Now, by using (5.3), (5.5), and (5.6) we get

$$V_m^T R(X_1) U_m = V_m^T R(X_0) U_m - \left(Y_m \widehat{A}_m - \widehat{F}_m^T Y_m\right) = 0. \tag{5.10}$$

□

In order to get $Y_m \in R^{m\times m}$, we need to solve the low-dimensional Sylvester equation (5.3). According to the results, we can develop an iterative method for solving of the Sylvester-observer equation. The algorithm is as follows.

Algorithm 5.2 (weighted block refinement (WBR) method). (1) Start: choose an initial solution X_0, new dimension m lesser than n and a tolerance ϵ.

(2) Compute $R(X_0) = GC - (X_0 A - F X_0)$.

(3) Construct the D-orthonormal basis $U_m \in R^{n\times m}$ and $\widehat{D}$-orthonormal basis $V_m \in R^{n\times m}$ by the weighted Arnoldi process, such that

$$\widehat{A}_m = U_m^T D A U_m, \qquad \widehat{F}_m = V_m^T \widehat{D} F V_m. \tag{5.11}$$

(4) Solve the low-dimensional Sylvester-observer equation $Y_m \widehat{A}_m - \widehat{F}_m^T Y_m = V_m^T R(X_0) U_m$ with the Hessenberg-Schur method.

(5) Set $X_1 = X_0 + \widehat{D} V_m Y_m U_m^T D$.

(6) Compute residual matrix $R(X_1) = GC - (X_1 A - F X_1)$.

(7) If $\|R(X_1)\|_F / \|R(X_0)\|_F \leq \epsilon$ stop, otherwise set $X_0 = X_1$, $R(X_0) = R(X_1)$ and go to step (2).

Remark 5.3. By choosing $m < n$, Algorithm 5.2 reduces the original large sparse Sylvester-observer equation (1.2) to a low-dimensional Sylvester-observer equation (5.3). In step (4), we solve this low-dimensional matrix equation by any direct method such as the Hessenberg-Schur method. Also, according to Theorem 5.1, Algorithm 5.2 is the convergence for any initial matrix X_0.

6. Numerical Experiments

In this section, we present some numerical examples to illustrate the effectiveness of our new iterative methods for solving large sparse Sylvester-obsever equation. In Examples 6.1 and 6.2, we apply Algorithms 2 and 3 for solving matrix equation (1.2). In Example 6.3, we compare the Hessenberg-Schur method described in [1] with our new algorithms for solving large sparse Sylvester-obsever equation. In order to show the efficiency of our algorithms, we choose the matrices A and C arbitrary in these three examples. But in Example 6.4, we use four matrices from MATLAB matrix collection with the large estimation of condition numbers.

The initial approximate solution is $X_0 = 0_{n\times n}$. The error is monitored by means of the test

$$\frac{\|R(X_1)\|_F}{\|R(X_0)\|_F} \leq \epsilon, \tag{6.1}$$

with the value of ϵ depending on the examples. The time is given in seconds for all examples. All numerical tests are performed in MATLAB software on a PC with 2.20 GHz with main memory 2 GB.

Example 6.1. For the first test we use two arbitrary matrices A and C. We choose the matrices F and G completely satisfying the controllability requirement of the pair (G, C). Now We apply block refinement method for solving Sylvester-observer equation $XA - FX = GC$ with $n = 200$. Also, we take $\epsilon = 10^{-6}$. In Table 1, we report the results for different values of m. In Table 1, the results show that by increasing the values of m and l, the number of iterations decreases. The last column of Table 1 also shows the decreasing of time consumption. Note that the fourth and fifth columns of this table are the errors of the orthogonalization method.

Table 1: Implementation of block refinement method to solve the Sylvester equation with different values of m.

m	r	ℓ	$\|V_m^T A^T V_m - \widehat{A}_m\|$	$\|W_m^T F W_m - \widehat{F}_m\|$	Iteration	Time
10	2	5	$4.11E-014$	$3.71E-016$	482	12.03
20	2	10	$6.32E-014$	$5.97E-015$	391	10.08
30	2	15	$1.16E-013$	$7.34E-015$	319	8.26
40	2	20	$3.65E-013$	$8.89E-014$	225	6.03
50	2	25	$2.01E-012$	$9.39E-014$	171	4.47
60	2	30	$4.73E-012$	$1.44E-013$	67	2.46

The desired accuracy has been chosen as 10^{-6}, but the model works well with any choice of 10^{-t}:

$$A = \begin{pmatrix} 10 & 1.2 & .42 & .8 & 2.3 & .8 & 0 & \dots & \dots & 0 \\ 1.8 & 10 & 1.2 & .42 & .8 & 2.3 & .8 & 0 & \ddots & 0 \\ 1.6 & 1.8 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots \\ 1.64 & 1.6 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .8 & 0 \\ 1.3 & 1.64 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 2.3 & .8 \\ 1.61 & 1.3 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .8 & 2.3 \\ 0 & 1.61 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & .42 & .8 \\ \vdots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 1.2 & .42 \\ \vdots & \vdots & 0 & 1.61 & 1.3 & 1.64 & 1.6 & 1.8 & 10 & 1.2 \\ 0 & 0 & \dots & 0 & 1.61 & 1.3 & 1.64 & 1.6 & 1.8 & 10 \end{pmatrix}_{n\times n} \qquad C = (1, 0, 0, \dots, 0). \tag{6.2}$$

Example 6.2. Consider Example 6.1 again. We apply the weighted Block refinement method for solving $XA - FX = GC$ and take $\epsilon = 10^{-6}$. In Table 2, we report the results for different values of m.

Example 6.3. According to the results in Tables 1 and 2, we see that the weighted block refinement method in comparison with block refinement method works better. Now, consider that A and C are the same matrices used in Example 6.1. We apply our two iterative methods with 2 iterations and the Hessenberg-Schur method to solve the Sylvester-observer equation when the dimensions of the matrices are large. Results are shown in Table 3.

Table 2: Implementation of weighted block refinement method to solve the Sylvester-observer equation with different values of m.

m	r	ℓ	$\|U_m^T DAU_m - \widehat{A}_m\|$	$\|V_m^T \widehat{D} F V_m - \widehat{F}_m\|$	Iteration	Time
10	2	5	$9.34E-016$	$2.27E-015$	311	7.34
20	2	10	$1.97E-015$	$2.98E-015$	174	5.89
30	2	15	$3.17E-015$	$3.12E-015$	106	4.48
40	2	20	$5.22E-015$	$6.05E-015$	67	3.37
50	2	25	$7.74E-015$	$9.59E-015$	40	2.19
60	2	30	$8.82E-015$	$1.11E-014$	11	0.984

Table 3: Implementation of new Iterative methods with 2 iterations and the Hessenberg-Schur method for solving the Sylvester equation.

n	Hessenberg-Schur method		Block refinement		Weighted block refinement		Cond (F)
	Error	Time	Error	Time	Error	Time	
400	$3.87E-07$	4.84	$4.12E-09$	3.58	$1.78E-015$	1.81	$3.17E+004$
800	$6.12E-002$	61.89	$3.55E-004$	48.24	$7.32E-013$	32.11	$2.13E+007$
1200	11.01	139.54	0.043	96.25	$4.13E-012$	71.43	$1.74E+010$
1600	26.29	208.24	11.01	134.32	$1.22E-09$	112.55	$1.27E+15$
2000	71.98	347.14	39.91	242.11	$5.67E-05$	194.78	$7.12E+021$

Example 6.4. In this example we show that the convergence of our proposed algorithms independent of the matrices structure. In this sense, we use four matrices from MATLAB collection for the matrix A. The first matrix is a sparse, random finite element matrix with the condition number $1.84E+03$. The second matrix is a symmetric, positive semidefinite (SPD) Toeplitz matrix that is composed of the sum of 800 rank 2 SPD Toeplitz matrices with the condition number $6.18E+04$. The third matrix is a row diagonally dominant matrix with the condition number $1.35E+010$. The last matrix is a sparse singular, row diagonally dominant matrix resulting from discrediting the Neumann problem with the usual five-point operator on a regular mesh. The estimated condition number is $5.56E+017$. For all of these examples, the matrix C is $C = \text{sprand}(1, n, d)$, where is a random, sparse matrix with approximately $d \cdot n \cdot n$ uniformly distributed nonzero entries with $d = 0.5$. We choose the matrices F and G completely satisfying the controllability and observability requirement of the pairs (F, G) and (G, C). We apply the Hessenberg-Schur method and weighted block refinement method (Algorithm 5.2) with 3 iterations for solving the Sylvester-observer equation $XA - FX = GC$. The results are shown in Table 4.

It is also obvious from Table 4 that the performance of the weighted block refinement method is much better than that of the Hessenberg-Schur method, specifically for the ill-conditioned matrices.

7. Comments and Conclusion

In this paper, we propose two new iterative algorithm s for solving the large sparse Sylvester-obsever matrix equations. The existing projection methods use the Arnoldi process, but the methods described in this paper are based on the weighted block Arnoldi process. Moreover,

Table 4: Effectiveness of the Hessenberg-Schur method and WBR algorithm with 3 iterations for randomly generated matrices.

Matrix	Hessenberg-Schur		WBR method	
	Error	CPU-time	Error	CPU-time
A = *gallery ("wathen", 736, 736)*	$2.19E-06$	66.24	$1.01E-016$	49.95
A = *gallery ("toeppd", 800, 800, r and (1,800)*	$6.01E-05$	72.19	$4.98E-016$	54.81
A = *gallery ("dorr", 1200, 0.01)*	$7.93E-01$	137.113	$6.12E-014$	101.02
A = *gallery ("neumann", 1156)*	$3.24E-01$	122.71	$9.21E-015$	94.29

the refinement process presented in Sections 4 and 5 has the capability of improving the results obtained by an arbitrary method. The numerical examples show the efficiency of the proposed schemes.

References

[1] B. N. Datta, *Numerical Methods for Linear Control Systems: Design and Analysis*, Elsevier Academic Press, San Diego, Calif, USA, 2004.

[2] A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems*, vol. 6 of *Advances in Design and Control*, SIAM, Philadelphia, Pa, USA, 2005.

[3] U. Baur and P. Benner, "Cross-Gramian based model reduction for data-sparse systems," *Electronic Transactions on Numerical Analysis*, vol. 31, pp. 256–270, 2008.

[4] D. C. Sorensen and A. C. Antoulas, "The Sylvester equation and approximate balanced reduction," *Linear Algebra and Its Applications*, vol. 351/352, pp. 671–700, 2002.

[5] W. H. Enright, "Improving the efficiency of matrix operations in the numerical solution of stiff ordinary differential equations," *ACM Transactions on Mathematical Software*, vol. 4, no. 2, pp. 127–136, 1978.

[6] D. Calvetti and L. Reichel, "Application of ADI iterative methods to the restoration of noisy images," *SIAM Journal on Matrix Analysis and Applications*, vol. 17, no. 1, pp. 165–186, 1996.

[7] J. Z. Hearon, "Nonsingular solutions of $TA - BT = C$," vol. 16, no. 1, pp. 57–63, 1977.

[8] E. de Souza and S. P. Bhattacharyya, "Controllability, observability and the solution of $AX - XB = C$," *Linear Algebra and its Applications*, vol. 39, pp. 167–188, 1981.

[9] G. H. Golub, S. Nash, and C. Van Loan, "A Hessenberg-Schur method for the problem AX + XB = C," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 909–913, 1979.

[10] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, Md, USA, 3rd edition, 1996.

[11] A. El Guennouni, K. Jbilou, and A. J. Riquet, "Block Krylov subspace methods for solving large Sylvester equations," *Numerical Algorithms*, vol. 29, no. 1–3, pp. 75–96, 2002.

[12] L. Bao, Y. Lin, and Y. Wei, "Krylov subspace methods for the generalized Sylvester equation," *Applied Mathematics and Computation*, vol. 175, no. 1, pp. 557–573, 2006.

[13] L. Bao, Y. Lin, and Y. Wei, "A new projection method for solving large Sylvester equations," *Applied Numerical Mathematics*, vol. 57, no. 5–7, pp. 521–532, 2007.

[14] D. Calvetti, B. Lewis, and L. Reichel, "On the solution of large Sylvester-observer equations," *Numerical Linear Algebra with Applications*, vol. 8, no. 6-7, pp. 435–451, 2001.

[15] K. Jbilou, "ADI preconditioned Krylov methods for large Lyapunov matrix equations," *Linear Algebra and its Applications*, vol. 432, no. 10, pp. 2473–2485, 2010.

[16] M. Robbé and M. Sadkane, "A convergence analysis of GMRES and FOM methods for Sylvester equations," *Numerical Algorithms*, vol. 30, no. 1, pp. 71–89, 2002.

[17] P. Benner, R.-C. Li, and N. Truhar, "On the ADI method for Sylvester equations," *Journal of Computational and Applied Mathematics*, vol. 233, no. 4, pp. 1035–1045, 2009.