Block Algorithms for State Estimation and Functional Observers

Biswa Nath Datta, Fellow of IEEE [1], e-mail: dattab@math.niu.edu
URL: http://www.math.niu.edu/~dattab
Daniil Sarkissian, e-mail: sarkiss@math.niu.edu
Fax: (815) 753-1112
Department of Mathematical Sciences
Northern Illinois University
DeKalb, IL, 60115

Abstract

This paper presents a new block algorithm for the Sylvester-observer equation, which seems to be computationally more efficient than the Van Dooren's algorithm and is suitable for high performance computing using LAPACK.

Also is presented in this paper a new algorithm for the design of functional observer in which the vector $Kx(t)$, required in the implementation of the feedback law $u(t) = Kx(t)$, can be directly estimated without estimating the vector $x(t)$.

Numerical examples are provided to illustrate the algorithms and results on comparison with the well-known Van Dooren's algorithm for reduced-order state estimation are displayed using graphs.

1 Introduction

Let $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{r\times n}$ (full rank) and

$$\left\{ \begin{array}{rcl} \dot{x}(t) & = & Ax(t) + Bu(t) \\ y(t) & = & Cx(t) \end{array} \right. , \tag{1}$$

where the unknown states $x(t) \in \mathbb{R}^n$ has to be estimated (observed) using the measured output $y(t) \in \mathbb{R}^r$ and known input $u(t) \in \mathbb{R}^m$.

One of the well-known approaches (see *e.g.* [4]) for estimating the states is via the Sylvester-observer equation

$$XA - FX = GC\,, \tag{2}$$

where A and C are given and the matrices $G \in \mathbb{R}^{n-r\times r}$, $F \in \mathbb{R}^{n-r\times n-r}$ and $X \in \mathbb{R}^{n-r\times n}$ are to be found. Once the solution X is found, the estimate of $x(t)$ can be obtained by solving the differential equation (*Luenberger observer*):

$$\dot{z}(t) = Fz(t) + Gy(t) + XBu(t)\,. \tag{3}$$

It can be shown that $z(t) \to Xx(t)$ as $t \to \infty$ if F is stable, that is, if all eigenvalues of F have negative real parts. The estimated states $\hat{x}(t)$ are then given by

$$\hat{x}(t) \quad = \quad \begin{pmatrix} C \\ X \end{pmatrix}^{-1} \begin{pmatrix} y(t) \\ z(t) \end{pmatrix}. \tag{4}$$

The well-known algorithms for solving the Sylvester-observer equations include

1. A multi-output recursive method by Van Dooren (1984) ([6]).
2. A single-input recursive algorithm by Datta (1989) ([2]).
3. A parallel block algorithm by Bischof, Datta and Purkayastha (1996) ([1]).
4. An Arnoldi method by Datta and Saad (1991) ([3]).

In this paper, we present a new block algorithm for the Sylvester-observer equation, which seems to be computationally more efficient than the Van Dooren's algorithm and is suitable for high performance computing using LAPACK.

Also, we propose a new algorithm for the design of functional observer in which the vector $Kx(t)$, required in the implementation of the feedback law $u(t) = Kx(t)$, can be directly estimated without estimating the vector $x(t)$.

The proposed algorithms are equally applicable to discrete-time systems. In case of discrete-time system, the equation (3) has to be replaced with

$$z_{t+1} = Fz_t + Gy_t + XBu_t \tag{5}$$

[1] The corresponding author.

with F as a discrete-stable matrix; that is, F having its eigenvalues inside the unit disk.

2 A Multi-output Block Scheme for Reduced-order Observer

When more than one output is observed (that is $r > 1$) then the simple recursive single-input scheme (see [2]) can be generalized to use the additional information available to us. To this end, we partition the matrices X, F and G into conforming blocks as follows:

$$X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_k \end{pmatrix}, \; F = \begin{pmatrix} F_{11} & F_{12} & & 0 \\ & F_{22} & \ddots & \\ & & \ddots & F_{k-1,k} \\ 0 & & & F_{kk} \end{pmatrix}$$

and

$$G = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ G_k \end{pmatrix},$$

where the number k will be determined in the algorithm. Then the equation (2) can be written as

$$\begin{array}{rcl} X_k A - F_{kk} X_k & = & G_k C \\ & \vdots & \\ X_2 A - F_{22} X_2 & = & F_{23} X_3 \\ X_1 A - F_{11} X_1 & = & F_{12} X_2 . \end{array} \qquad (6)$$

The idea now is to compute $X_k, X_{k-1}, \ldots, X_1$ recursively, normalizing each Y_i, $i = k, k-1, \ldots, 1$ in such a way that each row of Y_i is orthogonal to the other rows in the same Y_i.

To achieve this we note that if

$$X_{i+1} = Y_i A - \tilde{F}_{ii} Y_i$$

and if the SVD of Y_i^T is

$$Y_i^T = \begin{pmatrix} U & * \end{pmatrix} \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V^T \\ * \end{pmatrix}$$

then we have

$$U^T A - (\Sigma^{-1} V^T \tilde{F}_{ii} V \Sigma) U^T \;=\; (\Sigma^{-1} V^T) X_{i+1} \, . \qquad (7)$$

Comparing this with i-th equation of (6), we obtain

$$X_i := U^T \,, \; F_{ii} := \Sigma^{-1} V^T \tilde{F}_{ii} V \Sigma \text{ and } F_{i,i+1} := \Sigma^{-1} V^T .$$

Moreover, looking into the main diagonal of the matrix Σ we could check if $\mathrm{rank}(Y_i) = \mathrm{rank}(X_{i+1})$.

The above discussion yields itself to the following algorithm

Algorithm 1 A block Algorithm for Reduced-order State Estimation

Input:
$A \in \mathbb{R}^{n\times n}$, $C \in \mathbb{R}^{r\times n}$, set of scalars $\{\lambda_1, \ldots, \lambda_{n-r}\}$

Output:
$X \in \mathbb{R}^{n-r\times n}$, $G \in \mathbb{R}^{n-r\times r}$ and $F \in \mathbb{R}^{n-r\times n-r}$ such that spectrum of F is $\{\lambda_1, \ldots, \lambda_{n-r}\}$ and

$$XA - FX = GC$$

Assumptions:
(A, C) is observable, C has full rank, set $\{\lambda_1, \ldots, \lambda_{n-r}\}$ can be broken into a disjoint union of self-conjugate subsets of appropriate sizes, specified in Step 2.1 of the algorithm, and Step 2.2 goes through.

Step 1
Set $k = n - r$, $\tilde{r} = r$, $F = 0_{n-r\times n-r}$,
$X = \begin{pmatrix} 0_{n-r\times n} \\ C \end{pmatrix}$.

Step 2
While $k > 0$ do

Step 2.1
Set $\tilde{r} = \min(\tilde{r}, k)$, select any matrix $\tilde{F}$ with eigenvalues $\{\lambda_{k-\tilde{r}+1}, \ldots, \lambda_k\}$, and solve for Y

$$YA - \tilde{F}Y = X(k+1 : k+\tilde{r}, :)$$

Step 2.2
Compute the economy size *SVD of Y^T, $Y^T = U\Sigma V^T$. If the number of nonzero singular values is less then $\tilde{r}$ then restart the algorithm with different set of the eigenvalues $\lambda_1, \ldots, \lambda_{n-r}$.*

Step 2.3
Set $k = k - \tilde{r}$ and

$$\begin{array}{rcl} X(k+1 : k+\tilde{r}, :) & = & U^T, \\ F(k+1 : k+\tilde{r}, k+1 : k+\tilde{r}) & = & \Sigma^{-1} V^T \tilde{F} V \Sigma . \end{array}$$

If $k + \tilde{r} < n - r$ then set

$$F(k+1 : k+\tilde{r}, k+\tilde{r}+1 : k+2\tilde{r}) = \Sigma^{-1} V^T$$

else set

$$G = \begin{pmatrix} 0 \\ \Sigma^{-1} V^T \end{pmatrix}.$$

Step 3
$X = X(1 : n-r, :)$.

3 Functional Observer

The one step that gives us significant numerical errors is the inversion of the matrix $Z = \begin{pmatrix} C \\ X \end{pmatrix}$ in (4).

We show in this section, how this inversion can be avoided by estimating the feedback law $u(t) = Kx(t)$ directly without estimating $x(t)$ explicitly. This is known as *functional observer.*

This means that we will construct the *Luenberger Observer* with additional condition that there exists matrix G_1 such that

$$K = X_1 + G_1 C \,, \text{ where } X = \begin{pmatrix} X_1 \\ * \end{pmatrix} . \tag{8}$$

Note that we *do not* require the matrix Z to be of full rank, since we do not need to invert it to estimate $Kx(t)$.

For example, if we construct the Luenberger observer (3) with F, G and X obtained from Algorithm 2 and denote the the first m elements of vector $z(t)$ by $z_{1:m}(t)$, where m is the number of columns in matrix K, then

$$z_{1:m}(t) + G_1 y(t) \to Kx(t) \text{ as } t \to \infty. \tag{9}$$

For simplicity of the exposition, we will not use any scaling techniques and assume that $n = r(k+1)$, where n and r are the numbers of states and outputs of (1), respectively.

Let us conformably partition the block matrices X, F and G as follows:

$$F = \begin{pmatrix} F_{11} & I & & 0 \\ & F_{22} & \ddots & \\ & & \ddots & I \\ 0 & & & F_{kk} \end{pmatrix},$$

$$G = \begin{pmatrix} G_2 \\ \vdots \\ G_k \\ G_{k+1} \end{pmatrix} \text{ and } X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_k \end{pmatrix} .$$

Then equations (2) and (8) becomes

$$\begin{aligned} X_1 &= K - G_1 C \\ X_2 &= X_1 A - F_{11} X_1 - G_2 C \\ &\vdots \\ X_k &= X_{k-1} A - F_{k-1,k-1} X_{k-1} - G_k C \\ 0 = X_{k+1} &= X_k A - F_{kk} X_k - G_{k+1} C \,. \end{aligned}$$

Successively eliminating the $X_1, \ldots, X_{i-1}$ from the above equations it is seen that

$$X_i = K_i - \sum_{j=0}^{i-1} G_{ij} C A^j \,, \; 1 \le i \le k+1 \,, \tag{10}$$

where

$$K_1 = K \,, \; G_{k+1} = G_{k+1,0} \tag{11}$$

and for each $i = 1, 2, \ldots, k$

$$\begin{aligned} &K_{i+1} = K_i A - F_{ii} K_i \,, \\ &G_i = G_{i0} \,, \; G_{i,i-1} = G_{i+1,1} \,, \\ &G_{i,j-1} = G_{i+1,j} - F_{ii} G_{ij} \,, \; j = 1, 2, \ldots, k \,. \end{aligned} \tag{12}$$

Since the matrix pair (A, C) is observable and because of our assumptions that $\text{rank}(C) = r$ and $n = r(k+1)$, the observability matrix

$$O_M = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^k \end{pmatrix}$$

is non-singular. Therefore, the coefficients $G_{k+1,0}$, $G_{k+1,1}$, …, $G_{k+1,k}$ can be determined from

$$0 = X_{k+1} = K_{k+1} - (G_{k+1,0}, G_{k+1,1}, \ldots, G_{k+1,k}) O_M$$

and the equations (11) and (12) can be used recursively to compute $G_1, G_2, \ldots, G_{k+1}$.

The above discussion yields to the following algorithm

Algorithm 2 A Block Algorithm for Functional Observer

Input:
$A \in \mathbb{R}^{n \times n}$, $C \in \mathbb{R}^{r \times n}$, $K \in \mathbb{R}^{m \times n}$, set of scalars $\{\lambda_1, \ldots, \lambda_{km}\}$, where $k = \text{ceiling}\left(\frac{n-r}{r}\right)$.

Output:
$X \in \mathbb{R}^{km \times n}$, $G \in \mathbb{R}^{km \times r}$ and $F \in \mathbb{R}^{km \times km}$ such that spectrum of F is $\{\lambda_1, \ldots, \lambda_{km}\}$ and

$$XA - FX = GC$$

Assumptions:
(A, C) is observable, C has full rank, and the set $\{\lambda_1, \ldots, \lambda_{km}\}$ can be broken into a union of k self-conjugate disjoint subsets each of size m.

Step 1
Set $\tilde{K} = K$ and select matrices $F_{ii} \in \mathbb{R}^{m \times m}$ such that union of the spectra of F_{11}, …, F_{kk} is $\{\lambda_1, \ldots, \lambda_{km}\}$

Step 2
For $i = 1, 2, \ldots, k$ do
$\tilde{K} = \tilde{K} A - F_{ii} \tilde{K}$
End

Step 3
Solve for $G_1, G_2, \ldots, G_{k+1}$

$$\tilde{K} = (G_1, G_2, \ldots, G_{k+1}) \begin{pmatrix} C \\ CA \\ \vdots \\ CA^k \end{pmatrix}$$

Step 4
For $i = k, k-1, \ldots, 1$ *do*
For $j = 2, 3, \ldots, i+1$ *do*
$G_j = G_j + F_{ii} G_{j-1}$
End
End

Step 5
Set $X_1 = K - G_1 C$

Step 6
For $i = 1, 2, \ldots, k-1$ *do*
$X_{i+1} = X_i A - F_{ii} X_i - G_{i+1} C$
End

Step 7
Form the solution matrices

$$F = \begin{pmatrix} F_{11} & I & & 0 \\ & F_{22} & \ddots & \\ & & \ddots & I \\ 0 & & & F_{kk} \end{pmatrix},$$

$$G = \begin{pmatrix} G_2 \\ \vdots \\ G_k \\ G_{k+1} \end{pmatrix} \text{ and } X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_k \end{pmatrix}.$$

Note that matrix X recovered in the Steps 5 and 6 is not needed to construct the functional observer.

4 Illustrative Numerical Example

Consider a discrete-time state-space model of the steam power system (see [5]) with the system matrices

$$A = 10^{-1} \begin{pmatrix} 9.15 & 0.51 & 0.38 & 0.15 & 0.38 \\ -0.3 & 8.89 & -0.005 & 0.46 & 1.11 \\ -0.06 & 4.68 & 2.47 & 0.14 & 0.48 \\ -7.15 & -0.22 & -0.211 & 2.4 & -0.24 \\ -1.48 & -0.03 & -0.04 & 0.9 & 0.26 \end{pmatrix},$$

$$B = \begin{pmatrix} 0.0098 & 0.112 & 0.036 & 0.562 & 0.115 \end{pmatrix}^T$$

and

$$C = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Since $\text{rank}(C) = 2$, we design a reduced-order observer of order 3 with discrete-stable matrix F having eigenvalues $0.1444, -0.1893$ and -0.2719.

The relative errors between the true states and the observed states obtained by Algorithm 1 and Van Dooren's algorithm are plotted in Figure 1.

Then a functional observer of order 2 is constructed with the feedback matrix

$$K = \begin{pmatrix} 92.079 & -1.203 & 5.151 & 1.672 & 3.044 \end{pmatrix}.$$

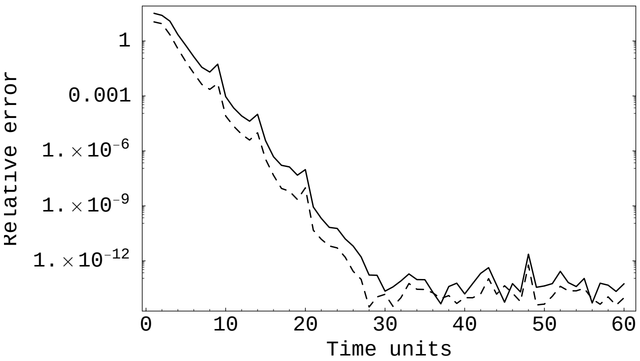

Figure 1: The relative errors between the true states and observed states (solid line - Van Dooren's algorithm, dashed line - Algorithm 1).

The true $Kx(t)$ and the estimated $Kx(t)$ obtained by Algorithm 2, Van Dooren's algorithm and Algorithm 1 are plotted in Figure 2.

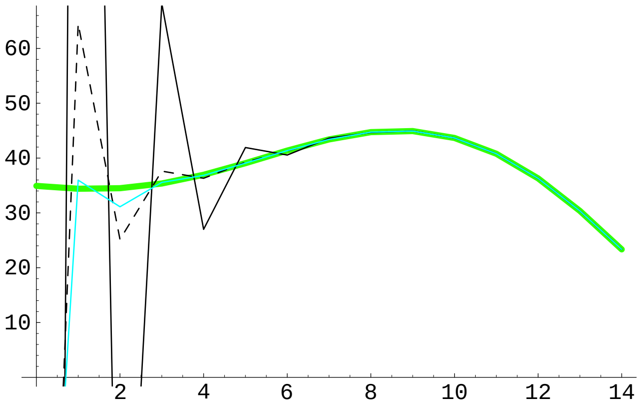

Figure 2: The graphs of true $Kx(t)$ and the estimated $Kx(t)$ (thick gray line - exact, solid line - Van Dooren's algorithm, dashed line - Algorithm 1, gray line - Algorithm 2).

Finally, the relative errors between true $Kx(t)$ and the estimated $Kx(t)$ obtained by these three algorithms are plotted in Figure 3.

5 Conclusions

In this paper, we have presented two new block algorithms - one for the reduced-order observer and the other for the functional observer. Both the algorithms are compared with the well-known Van Dooren's algorithm for reduced-order state estimation.

¿From the numerical experiments performed by us, it is seen that both the algorithms perform better than Van Dooren's algorithm. Some further studies and numerical experiments on high performance computers with the new algorithms are in order.

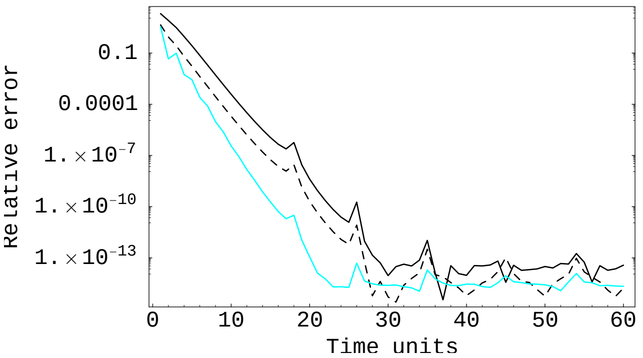

Figure 3: The relative errors between exact $Kx(t)$ and the estimated $Kx(t)$ (solid line - Van Dooren's algorithm, dashed line - Algorithm 1, gray line - Algorithm 2).

References

[1] Bischof, C. H., Datta, B. N. and Purkayastha, A., *A parallel algorithm for the Sylvester observer equation*, SIAM J. Sci. Comput., **17** (1996), no. 3, pp. 686–698.

[2] Datta, B. N., *Parallel and large-scale matrix computations in control: Some ideas*, Linear Algebra Appl., **121** (1989), pp. 243–264.

[3] Datta, B. N. and Saad, Y., *Arnoldi methods for large Sylvester-like observer matrix equations, and an associated algorithm for partial spectrum assignment*, Linear Algebra Appl., **154/156** (1991), pp. 225–244.

[4] Datta, B. N., *Numerical Methods for Linear Control Systems Design and Analysis*, Academic Press (2000).

[5] Mahmoud, M. S., *Structural properties of discrete systems with slow and fast modes*, Large Scale Systems, **3** (1982), no. 4, 227–236.

[6] Van Dooren, P., *Reduced order observers: A new algorithm and proof*, Systems Control Lett., **4** (1984), pp. 243–251.