International Journal of Pure and Applied Mathematics

Volume 53 No. 1 2009, 55-68

SYLVESTER MATRIX DIFFERENTIAL EQUATIONS: ANALYTICAL AND NUMERICAL SOLUTIONS

Laurene V. Fausett

Department of Mathematics
Texas A&M University-Commerce
Commerce, TX 75429, USA
email: Laurene_Fausett@TAMU-Commerce.edu

Abstract: This paper considers the relationships between the general solution of the first-order matrix Sylvester ODE:

$$\mathbf{X}' = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{X}(t)\mathbf{B}(t)$$

in terms of two fundamental matrix solutions of $\mathbf{T}' = \mathbf{AT}$ and $\mathbf{T}' = \mathbf{B}^*\mathbf{T}$, and the numerical solution of such equations which can be obtained using standard *Matlab* routines for the solution of vector ODE.

Theoretical results regarding the solution of a Sylvester control system

$$\mathbf{X}' = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{X}(t)\mathbf{B}(t) + \mathbf{C}(t)\mathbf{U}(t)$$

are used to compute the control signal $\mathbf{U}(t)$, a particular solution, and the total controlled solution.

AMS Subject Classification: 34H05, 93B05, 65L99
Key Words: matrix differential equations, numerical solutions of ODE, fundamental matrix solutions

1. Introduction

There is a close connection between scalar differential equations, vector differential equations and matrix differential equations. For equations with constant coefficients, we have the following familiar results.

Received: March 30, 2009

The scalar ODE $x' = ax(t)$ has a solution of the form $x(t) = ce^{at}$.

If $\mathbf{A}$ has distinct real eigenvalues, the vector ODE $\mathbf{x}' = \mathbf{A}\mathbf{x}(t)$ has a solution of the form

$$\mathbf{x} = c_1 e^{\lambda_1 t}\mathbf{v}_1 + \ldots + c_n e^{\lambda_n t}\mathbf{v}_n \,.$$

The matrix ODE $\mathbf{X}' = \mathbf{A}\mathbf{X}(t)$ has a solution of the form $\mathbf{X}(t) = \mathbf{C}e^{\mathbf{A}t}$.

The matrix Sylvester ODE, with constant, square coefficient matrices,

$$\mathbf{X}' = \mathbf{A}\mathbf{X}(t) + \mathbf{X}(t)\mathbf{B}$$

has a solution of the form:

$$\mathbf{X}(t) = e^{\mathbf{A}t}\mathbf{C}e^{\mathbf{B}t} \,.$$

A Sylvester ODE with $\mathbf{A} = \mathbf{B}$ is generally known as a Lyapunov ODE. A Sylvester matrix ODE is a special case of the matrix Riccati ODE

$$\mathbf{X}' = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{B}(t) - \mathbf{X}(t)\mathbf{C}(t)\mathbf{X}(t) - \mathbf{X}(t)\mathbf{D}(t)$$

and of the nonhomogeneous ODE for a matrix control system

$$\mathbf{X}' = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{X}(t)\mathbf{B}(t) + \mathbf{C}(t)\mathbf{U}(t) \,.$$

2. Sylvester Matrix ODE: Constant Coefficients

2.1. Analytic Solutions

We begin by considering the matrix Sylvester ODE

$$\mathbf{X}' = \mathbf{A}\mathbf{X}(t) + \mathbf{X}(t)\mathbf{B} \,, \tag{2.1}$$

where $\mathbf{X}(t)$, $\mathbf{A}$, and $\mathbf{B}$ are $n \times n$ matrices.

Let $\mathbf{Y}$ be a fundamental matrix solution of $\mathbf{T}' = \mathbf{A}\mathbf{T}$ and $\mathbf{Z}$ be a fundamental matrix solution of $\mathbf{T}' = \mathbf{B}^*\mathbf{T}$ (a fundamental matrix of the ODE is a matrix whose columns form a fundamental set of solutions to the ODE), see, e.g., Edwards and Penney [4].

The solution of the Sylvester ODE can be expressed as

$$\mathbf{X}(t) = \mathbf{Y}(t)\mathbf{C}\mathbf{Z}^*(t) \,,$$

where $\mathbf{C}$ is a constant $n \times n$ matrix.

Example 1a. Consider the differential equation above, with

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix}, \text{ and } \quad \mathbf{B} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} .$$

Find the required fundamental matrices by solving

$$\mathbf{y}' = \mathbf{A}\mathbf{y} \quad \text{and} \quad \mathbf{z}' = \mathbf{B}^T\mathbf{z}\,.$$

Since the eigenvalues of $\mathbf{A}$ are $2, 1$ with corresponding eigenvectors $\mathbf{v}_1 = [1, 2]^T$ and $\mathbf{v}_2 = [1, 1]^T$, we have

$$\mathbf{Y} = \begin{bmatrix} e^{2t} & e^t \\ 2e^{2t} & e^t \end{bmatrix}.$$

Similarly, using the eigenvalues and eigenvectors of $\mathbf{B}$ we find

$$\mathbf{Z} = \begin{bmatrix} e^t & e^{-t} \\ e^t & -e^{-t} \end{bmatrix}.$$

Then

$$\mathbf{X}(t) = \begin{bmatrix} e^{2t} & e^t \\ 2e^{2t} & e^t \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e^t & e^t \\ e^{-t} & -e^{-t} \end{bmatrix}.$$

Example 1b. Fundamental matrix solutions of **y'** = $\mathbf{A}\mathbf{y}$ and **z'** = $\mathbf{B}^T\mathbf{z}$ can also be found in terms of matrix exponentials. There are many ways to compute the matrix exponential. One of the polynomial methods is based on Lagrange interpolation. If the distinct eigenvalues of $\mathbf{A}$ are $\lambda_i, i = 1...n$, the Lagrange interpolation method gives

$$e^{\mathbf{A}x} = \sum_{j=1}^{n} e^{\lambda_j x}\mathbf{A}_j\,,$$

where

$$\mathbf{A}_j = \prod_{k=1,\, k\neq j}^{n} \frac{(\mathbf{A} - \lambda_k I)}{(\lambda_j - \lambda_k)}.$$

The solution of the Sylvester ODE of the previous example is then given in terms of the fundamental matrix solutions $\mathbf{Y} = e^{\mathbf{A}t}$ and $\mathbf{Z} = e^{\mathbf{B}t}$ as

$$\mathbf{Y} = \begin{bmatrix} -e^{2t} + 2e^t & e^{2t} - e^t \\ -2e^{2t} + 2e^t & 2e^{2t} - e^t \end{bmatrix}$$

and

$$\mathbf{Z} = \begin{bmatrix} 0.5e^t + 0.5e^{-t} & 0.5e^t - 0.5e^{-t} \\ 0.5e^t - 0.5e^{-t} & 0.5e^t + 0.5e^{-t} \end{bmatrix}.$$

Thus, the solution to the Sylvester ODE can also be expressed as

$$\mathbf{X}(t) = e^{\mathbf{A}t}\mathbf{X}(0)e^{\mathbf{B}t}\,,$$

or

$$\mathbf{X} = \begin{bmatrix} -e^{2t} + 2e^t & e^{2t} - e^t \\ -2e^{2t} + 2e^{2t} & 2e^{2t} - e^t \end{bmatrix} \mathbf{X}(0) \begin{bmatrix} 0.5e^t + 0.5e^{-t} & 0.5e^t - 0.5e^{-t} \\ 0.5e^t - 0.5e^{-t} & 0.5e^t + e^{-t} \end{bmatrix}.$$

It is easy to find the correspondence between the constants in the two forms if an initial condition is given.

2.2. Numerical Computation of Fundamental Matrices

The simple, closed form solutions in terms of matrix exponentials, or matrices using the eigenvalues and eigenvectors of the coefficient matrices, are convenient for many applications. Simple *Matlab* routines, using built-in functions for computing eigenvalues and eigenvectors, provide efficient numerical solution to Sylvester ODE, with constant coefficients.

```
% Example 1a;  X' = A X + X B
A = [ 0  1
     -2  3];
B = [ 0  1
      1  0];
[vA, eA] = eig(A);            [vB, eB] = eig(B);
t = 0:0.1:1                  % time interval for solution
Y1 =vA(:,1)*exp(eA(1,1)*t);  Y2 =vA(:,2)*exp(eA(2,2)*t);
Z1 =vB(:,1)*exp(eB(1,1)*t);  Z2 =vA(:,2)*exp(eB(2,2)*t);
```

The solution of this example can also be computed in terms of matrix exponentials. *Matlab* has four built-in functions to compute the matrix exponential. The following script uses the function `expm`, which uses Pade approximation with scaling and squaring. The other functions are: `expm1`, which is based on an algorithm from Golub and Van Loan [6]; `expm2`, which is based on a Taylor series approach; and `expm1`, which is based on eigenvalues and eigenvectors. For a comprehensive discussion of numerical approaches to computing the matrix exponential see Moler [8].

```
% Matlab script for Example 1b;
% to compute the solution to X' = A X + X B
clear
A = [ 0  1
     -2  3];
B = [ 0  1
      1  0];
t = 0:0.1:1
```

```
% use Matlab to compute the matrix exponential
% layer matrices for each time step into a 3-dim array
Y = expm(A*t(1));              Z = expm(B*t(1));
YY = Y;                        ZZ = Z;
for j = 2:length(t)
   Yt = expm(A*t(j));          Zt = expm(B*t(j));
   YY = cat(3, YY, Yt);        ZZ = cat(3, ZZ, Zt);
end
% YY is the solution to T' = A T;
% ZZ is the solution to T' = B*T
% take X(0) = identity and
% compute X = YY*ZZ at each time step
for k = 1:length(t)
   YYt = YY(:, :, k);          ZZt = ZZ(:, :, k)
   X = YYt*ZZt
end
```

Note that some caution is required in interpreting matrix exponential solutions in cases where there are complex eigenvalues. As is well known from the study of ODE, the fundamental solutions for a vector ODE which involve complex exponentials can be expressed in terms of sine and cosine functions, together with the exponential of the real part of the complex eigenvalue pair. However, the matrix exponential formed using the Lagrange method described above will give complex components when the eigenvalues are complex. Plotting these components using *Matlab* will plot only the real parts, and will therefore not show the desired results.

3. Sylvester Matrix Control System

This section presents several basic results concerning the first-order matrix Sylvester system

$$\mathbf{T}' = \mathbf{A}(t)\mathbf{T}(t) + \mathbf{T}(t)\mathbf{B}(t) + \mathbf{C}(t)\mathbf{U}(t)\,, \tag{3.1}$$

$$\mathbf{X}(t) = \mathbf{D}(t)\mathbf{T}(t)\,, \tag{3.2}$$

with control (or input) $\mathbf{U}(t)$, and output signal $\mathbf{X}(t)$. Matrix $\mathbf{D}(t)$ provides the filter between the computed state variables in matrix $\mathbf{T}(t)$ and those values that can actually be observed in the output signal $\mathbf{X}(t)$. All matrices are $n \times n$.

We can express the solution to this control system in terms of the fundamental matrix solutions of $\mathbf{T}' = \mathbf{A}(t)\mathbf{T}(t)$, which we denote as $\mathbf{Y}(t)$, and the fundamental matrix solution of $\mathbf{T}' = \mathbf{B}^*(t)\mathbf{T}(t)$, denoted $\mathbf{Z}(t)$.

Thus,

$$\mathbf{Y}' = \mathbf{A}(t)\mathbf{Y}(t) \quad \text{and} \quad \mathbf{Z}' = \mathbf{B}^*(t)\mathbf{Z}(t)\,.$$

In addition, we define

$$\Phi(t, t_0) = \mathbf{Y}(t)\mathbf{Y}^{-1}(t_0) \quad \text{and} \quad \Psi^*(t_0, t) = \mathbf{Z}^{*-1}(t_0)\mathbf{Z}^*(t)\,.$$

The basic results regarding the Sylvester control system are that:

— Any solution of the homogeneous equation

$$\mathbf{T}'(t) = \mathbf{A}(t)\mathbf{T}(t) + \mathbf{T}(t)\mathbf{B}(t)$$

is of the form $\mathbf{T}(t) = \mathbf{Y}(t)\mathbf{C}_1\mathbf{Z}^*(t)$ where $\mathbf{C}_1$ is a constant matrix.

— Any solution of the nonhomogeneous system (3.1):

$$\mathbf{T}' = \mathbf{A}(t)\mathbf{T}(t) + \mathbf{T}(t)\mathbf{B}(t) + \mathbf{C}(t)\mathbf{U}(t)$$

is of the form

$$\mathbf{T}(t) = \mathbf{Y}(t)\mathbf{C}_1\mathbf{Z}^*(t) + \mathbf{T}_p(t)\,,$$

where $\mathbf{T}_p(t)$ is a particular solution of (3.1).

— A particular solution of (3.1) is given by

$$\mathbf{T}_p(t) = \mathbf{Y}(t)[\int_a^t \mathbf{Y}^{-1}(s)\mathbf{C}(s)\mathbf{U}(s)\mathbf{Z}^{*-1}(s)ds\ ]\ \mathbf{Z}^*(t)\,.$$

— Any solution $\mathbf{T}(t)$ of the initial value problem (3.1) satisfying $\mathbf{T}(t_0) = \mathbf{T}_0$ is given by

$$\mathbf{T}(t) = \Phi(t, t_0)\mathbf{T}_0\Psi^*(t_0, t) + \Phi(t, t_0)[\int_{t_0}^t \Phi(t_0, s)\mathbf{C}(s)\mathbf{U}(s)\Psi^*(s, t_0)ds\ ]\ \Psi^*(t_0, t)\,.$$

These results are established in Murty and Fausett [9].

Example 2. Consider again the differential equation

$$\mathbf{Y}' = \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix} \mathbf{Y}$$

introduced in Example 1. We found

$$\mathbf{Y}(t) = \begin{bmatrix} e^{2t} & e^t \\ 2e^{2t} & e^t \end{bmatrix}, \text{ so } \quad \mathbf{Y}^{-1} = \begin{bmatrix} -e^{-2t} & e^{-2t} \\ 2e^{-t} & -e^{-t} \end{bmatrix}.$$

Then (for $t_0 = 0$)

$$\Phi(t, t_0) = \mathbf{Y}(t)\mathbf{Y}^{-1}(t_0) = \begin{bmatrix} -e^{2t} + 2e^t & e^{2t} - e^t \\ -2e^{2t} + 2e^t & 2e^{2t} - e^t \end{bmatrix}$$

and

$$\Phi(t_0,t) = \mathbf{Y}(t_0)\mathbf{Y}^{-1}(t) = \begin{bmatrix} 2e^{-t} - e^{-2t} & e^{-2t} - e^{-t} \\ -2e^{-2t} + 2e^{-t} & 2e^{-2t} - e^{-t} \end{bmatrix}.$$

Similarly, for

$$\mathbf{Z}' = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \mathbf{Z}$$

we have

$$\mathbf{Z} = \begin{bmatrix} e^t & e^{-t} \\ e^t & -e^{-t} \end{bmatrix}, \ \mathbf{Z}^* = \begin{bmatrix} e^t & e^t \\ e^{-t} & -e^{-t} \end{bmatrix}, \ \text{and} \quad \mathbf{Z}^{*-1} = \frac{1}{2} \begin{bmatrix} e^{-t} & e^t \\ e^{-t} & -e^t \end{bmatrix}.$$

Then, for $t_0 = 0$, we have

$$\Psi^*(t_0,t) = \mathbf{Z}^{*-1}(t_0)\mathbf{Z}^*(t) = \frac{1}{2} \begin{bmatrix} e^t + e^{-t} & e^t - e^{-t} \\ e^t - e^{-t} & e^t + e^{-t} \end{bmatrix}$$

and

$$\Psi^*(t,t_0) = \mathbf{Z}^{*-1}(t)\mathbf{Z}^*(t_0) = \frac{1}{2} \begin{bmatrix} e^t + e^{-t} & -e^t + e^{-t} \\ -e^t + e^{-t} & e^t + e^{-t} \end{bmatrix}.$$

These functions will be used in Section 3.2 to compute numerical solutions for this example.

3.1. Controllability

In this section, we address the fundamental concept of controllability of the system (3.1)

$$\mathbf{T}' = \mathbf{A}(t)\mathbf{T}(t) + \mathbf{T}(t)\mathbf{B}(t) + \mathbf{C}(t)\mathbf{U}(t).$$

The linear system above is completely controllable on $[t_0, t_1]$, if for any initial time t_0 and any initial state $\mathbf{T}(t_0) = \mathbf{T}_0$, there exists a continuous input signal $\mathbf{U}(t)$ such that the corresponding solution satisfies $\mathbf{T}(t_1) = \mathbf{T}_1$.

The basic results regarding controllability are:

1. The linear state equation is completely controllable on $[t_0, t_1]$ if and only if the $(n \times n)$ symmetric matrix, the controllability Grammian, defined as

$$\mathbf{W}(t_0,t_1) = \int_{t_0}^{t_1} \Phi(t_0,s)\mathbf{C}(s)\mathbf{C}^*(s)\Phi^*(t_0,s)ds,$$

is non-singular. In this case, the control $\mathbf{U}(t)$ is given by

$$-\mathbf{C}^*(t)\Phi^*(t_0,t)\mathbf{W}^{-1}(t_0,t_1)[\mathbf{T}_0 - \Phi(t_0,t_1)\mathbf{T}_1\Psi^*(t_1,t_0)]\Psi^*(t_0,t).$$

2. The time-invariant linear state equation

$$\mathbf{T}' = \mathbf{A}\mathbf{T}(t) + \mathbf{T}(t)\mathbf{B} + \mathbf{C}\mathbf{U}(t)$$

is completely controllable on $[t_0, t_1]$ if and only if

$$rank[\mathbf{C}, \mathbf{AC}, ...\mathbf{A}^{n-1}\mathbf{C}] = n, \text{ or } rank[\mathbf{C}, \mathbf{CB}, ...\mathbf{CB}^{n-1}] = n\,.$$

These are established in Murty and Fausett [9].

3.2. Numerical Solutions

We continue the previous script for solving the problem in Examples 1 and 2. For simplicity, we take **C** to be the identity matrix.

```
% script to compute the solution to X' = A X + X B
clear
A = [ 0  1
     -2  3];
B = [ 0  1
      1  0];
t0 = 0;     dt = 0.05;     t1 = 1;
t = t0 : dt : t1;          Lt = length(t);
% compute fundamental matrix solutions
% using matrix exponential functions
Y = expm(A*t(1));          Z = expm(B*t(1));
YY = Y;                    ZZ = Z;
for j = 2:Lt
    Yt = expm(A*t(j));     Zt = expm(B*t(j));
    YY = cat(3, YY, Yt);   ZZ = cat(3, ZZ, Zt);
end
for k = 1:Lt
    YYt = YY(:, :, k);     ZZt = ZZ(:, :, k);
    X = YYt*ZZt
end
% compute quantities for control problem.
C = eye(2);          R = [C, A*C];       r = rank(R);
% compute controllability Gramian
Phi_t_t0 = eye(2);   Phi_t0_t = eye(2);  W = 0;
% using analytic form of Phi_t_t0 and Phi_t0_t (Ex 2)
for k = 2:Lt
  P1=[-exp(2*t(k))+2*exp(t(k)), exp(2*t(k))-exp(t(k))
  -2*exp(2*t(k))+2*exp(t(k)),2*exp(2*t(k))-exp(t(k))];

  P2=[2*exp(-t(k))-exp(-2*t(k)), exp(-2*t(k))-exp(-t(k))
```

```
  -2*exp(-2*t(k))+2*exp(-t(k)),2*exp(-2*t(k))-exp(-t(k))];

  % Phi_t_t0 and Phi_t0_t layered into 3-dim arrays
  Phi_t_t0  = cat(3, Phi_t_t0, P1);
  Phi_t0_t  = cat(3, Phi_t0_t, P2);
  W  = W + dt*P2*P2';
end
W,        W_inv = inv(W),
% define initial state T0
T0 = [ 1  0
       0  1];
% define desired final state T1
T1 = [ 0  0
       0  0];
% compute Psi*
Psi_t0_t = eye(2);   Psi_t_t0 = eye(2);
for k = 2:Lt
  S1 = 0.5*[ exp(t(k))+exp(-t(k)),exp(t(k))-exp(-t(k))
             exp(t(k))-exp(-t(k)),exp(t(k))+exp(-t(k))];
  S2 = 0.5*[ exp(t(k))+exp(-t(k)),-exp(t(k))+exp(-t(k))
            -exp(t(k))+exp(-t(k)), exp(t(k))+exp(-t(k))];
  Psi_t0_t  = cat(3, Psi_t0_t, S1);
  Psi_t_t0  = cat(3, Psi_t_t0, S2);
end
% compute the control U
U = zeros(2);
for k = 2:Lt
    Br = (T0 - Phi_t0_t(:, :, Lt)*T1*Psi_t_t0(:, :, Lt));
    Uk = -Phi_t0_t(:, :, k)'*W_inv*Br*Psi_t0_t(:, :, k);
    U = cat(3, U, Uk);
end
```

To form the complete solution to this control problem, we must compute the particular solution, $\mathbf{T}_p$. At each step, the fundamental solution $\mathbf{Y}$ is given in `YY(:, :, k)`. The control $\mathbf{U}$ is given in `U(:, :, k)`. the fundamental solution $\mathbf{Z}^*$ is given in `ZZ(:, :, k)`.

```
Int = 0;     Tp = zeros(2);
% this uses the analytic form of Phi = P from Example 2
for k = 2:Lt
```

```
  Int=Int+dt*Phi_t0_t(:,:,k)*U(:,:,k)*Psi_t_t0(:,:,k);
  Tpk = Phi_t_t0(:,:,k)*Int*Psi_t0_t(:,:,k);
  Tp  = cat(3, Tp, Tpk);
end
Tp
% form total solution at each time step
TT = T0;
for k = 2:Lt
    TTk =  YY(:, :, k)*ZZ(:, :, k) + Tp(:, :, k);
    TT  =  cat(3, TT, TTk);
end
TT(:, :, Lt)
```

Finally, we add commands to plot the components of the controlled system. The graphs are shown in Figure 1.

```
% plot components of solution
for k = 1:Lt
    T11(k) = TT(1, 1, k);
    T12(k) = TT(1, 2, k);
    T21(k) = TT(2, 1, k);
    T22(k) = TT(2, 2, k);
end
subplot(2, 2, 1);  plot(t, T11);    title('T(1,1)');
subplot(2, 2, 2);  plot(t, T12);    title('T(1,2)');
subplot(2, 2, 3);  plot(t, T21);    title('T(2,1)');
subplot(2, 2, 4);  plot(t, T22);    title('T(2,2)');
```

3.3. Observability

We now consider the observability of the linear state equation. The linear state equation is

$$\mathbf{T}' = \mathbf{A}(t)\mathbf{T}(t) + \mathbf{T}(t)\mathbf{B}(t) + \mathbf{C}(t)\mathbf{U}(t)\,,$$

$$\mathbf{X}(t) = \mathbf{D}(t)\mathbf{T}(t)\,.$$

The linear state equation is said to be completely observable on $[t_0, t_1]$ if any initial state $\mathbf{T}(t_0) = \mathbf{T}_0$ is uniquely determined by the corresponding response $\mathbf{X}(t)$ for $t \in [t_0, t_1]$.

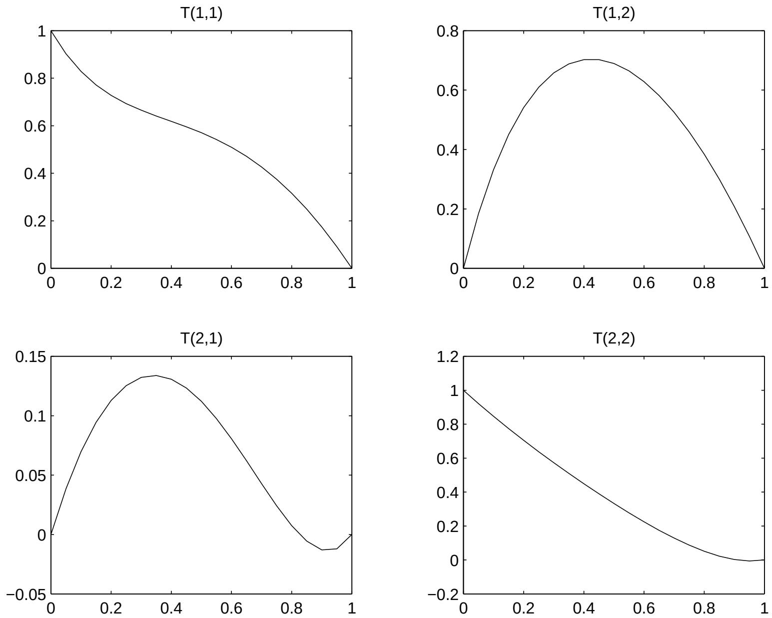

Figure 1: Controlled Sylvester system

Without loss of generality we consider the case with $\mathbf{U}(t) = 0$ on $[t_0, t_1]$:

$$\mathbf{T}' = \mathbf{A}(t)\mathbf{T}(t) + \mathbf{T}(t)\mathbf{B}(t)\,, \qquad \mathbf{X}(t) = \mathbf{D}(t)\mathbf{T}(t)\,.$$

The basic results are:

— The linear state equation is completely observable if and only if the observability Grammian $\mathbf{M}$ is non-singular.

$$\mathbf{M}(t_0, t_1) = \int_{t_0}^{t_1} \Phi^*(s, t_0)\mathbf{D}^*(s)\mathbf{D}(s)\Phi(s, t_0)ds\,.$$

— A more convenient criteria (assuming sufficient smoothness) is based on the following sequence of matrices:

$$\mathbf{L}_0(t) = \mathbf{D}(t)\,,$$

$$\mathbf{L}_j(t) = -\mathbf{A}(t)\mathbf{L}_{j-1}(t) - \mathbf{L}_{j-1}(t)\mathbf{B}(t) + \mathbf{L}'_{j-1}(t) \ (j = 1, 2, \dots, n-1)\,.$$

Then the linear state equation is completely observable on $[t_0, t_1]$ if for some $t_c \in [t_0, t_1]$

$$rank[\mathbf{L}_0(t_c), \mathbf{L}_1(t_c), ...\mathbf{L}_{n-1}(t_c)] = n\,.$$

These are established in Murty and Fausett [9].

Numerical solutions follow the same general form as that given in the scripts given for the controllability problem.

4. Sylvester Matrix ODE: Variable Coefficients

We now consider the numerical solution of the matrix Sylvester ODE

$$\mathbf{X}' = \mathbf{A}(t)\mathbf{X}(t) + \mathbf{X}(t)\mathbf{B}(t)$$

using standard *Matlab* routines for solving vector ODE.

```
% example of solving a matrix ODE
%A =  [     0       1          0
%           0       0          1
%           0      -t^(-2)     3*t^(-1)  ];
tspan = 1:0.1:2
[t,x1] = ode23('f_mat_ode_var_1', tspan,  [ 1  0  0]')
v1 = [ 1  3-sqrt(3)  9-5*sqrt(3)]';
[t,x2] = ode23('f_mat_ode_var_1', tspan, v1 )
v2 = [ 1  3+sqrt(3)  9+5*sqrt(3)]';
[t,x3] = ode23('f_mat_ode_var_1', tspan, v2 )
% construct 3 dim array, time is third dimension.
w1 = [x1(1, :)', x2(1, :)', x3(1, :)']
w = w1
for k = 2:length(tspan)
    wt = [x1(k, :)', x2(k, :)', x3(k, :)']
    w  = cat(3, w, wt)
end
```

This example comes from an Euler ODE; fundamental solutions may be taken to be $u_1 = 1; u_2 = t^{3-\sqrt{3}}; u_3 = t^{3+\sqrt{3}}$

5. Summary

In this paper we have demonstrated that theoretical results regarding solutions to the Sylvester matrix ODE and the Sylvester matrix control system can be used together with standard *Matlab* functions and programming techniques to compute numerical solutions to these problems. Although the methods were demonstrated using simple problems for which analytic solutions can be found,

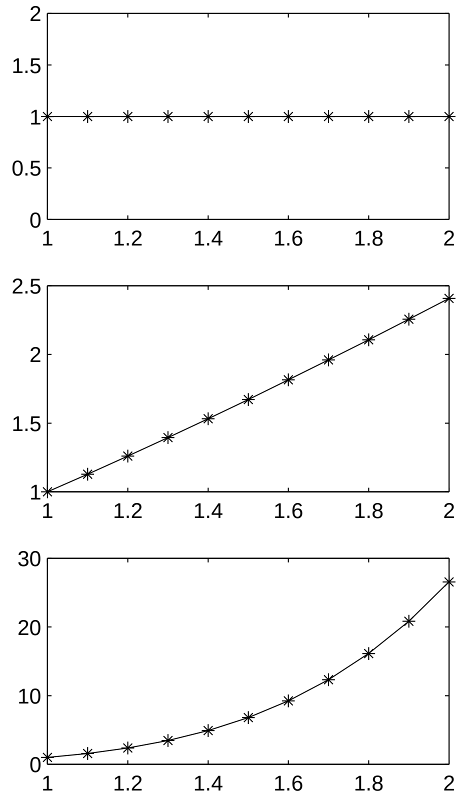

Figure 2: Computed and exact solutions

numerical solution of the fundamental matrix solutions could easily be used in the process. The procedure for applying built-in *Matlab* functions for solving vector ODE was demonstrated for a matrix ODE with variable coefficients.

Acknowledgements

The author would like to thank the Office of Graduate Studies and Research at Texas A&M University-Commerce for their support of this research. Thanks also to William Pleasant, an undergraduate mathematics student, who helped on this project, and to Gulcin Tugcu for her work on the Lyapunov method for matrix exponentials, which was part of her M.S. Thesis at Georgia Southern University. Finally, thanks go to Dr. K.N. Murty for his leading work on the

theoretical foundations for this research.

References

[1] S. Barnett, R.G. Cameron, *Introduction to Mathematical Control Theory*, Second Edition, Oxford University Press (1985).

[2] S. Barnett, Matrix differential equations and Kronecker products, *SIAM Journal on Applied Mathematics*, **24** (1973), 1-5.

[3] R. Bellman, *Introduction to the Theory of Control Process*, Academic Press (1967).

[4] C.H. Edwards, D.E. Penney, *Differential Equations and Linear Algebra* (2001).

[5] D.W. Fausett, L.V. Fausett, K.N. Murty, K.R. Prasad, Multipoint boundary value problems associated with a first order nonlinear Lyapunov system, *International Journal of Nonlinear Differential Equations, Theory, Methods and Applications*, **7** (2002), 175-192.

[6] G.H. Golub, C.F. Van Loan, *Matrix Computations*, Third Edition, Johns Hopkins University Press (1996)

[7] V. Lakshmikantham, D. Trigiante, *Theory of Difference Equations: Numerical Methods and Applications*, Academic Press (1988)

[8] C. Moler, Nineteen Dubious ways to compute the exponential of a matrix, twenty-five years later, *SIAM Review*, **45** (2003), 3-49.

[9] K.N. Murty, L.V. Fausett, Some fundamental results on controllability, observability and realizability of first-order matrix Lyapunov systems, *Mathematical Sciences Research Journal*, **6**, No. 3 (2002), 147-160.