

Cleveland State University  
Department of Electrical and Computer Engineering

EEC 510 Linear Systems  
Fall 2004

### Homework Suggestions

These homework assignments are taken from John Bay's book *Fundamentals of Linear State Space Systems*. You are expected to solve each HW problems. HW will NOT be collected or graded regularly. Students may be randomly selected to answer questions based on HW. Volunteers are welcome to demonstrate solutions of the difficult problems in the chapter at the beginning of each class.

For each chapter, homework should be organized as *two separate assignments*:

- (1) The written exercises, with the problems in order and the pages stapled
- (2) The Matlab exercises, with the problems in order and the pages stapled.

The record of all your HW should be kept throughout the semester.

### Chapter 1 Homework

#### *Written Exercises*

1.1

1.2 – Hint: The forces acting on the automobile are  $B(y'-x')$ ,  $K(y-x)$ , and  $Mg$ . The states of the system are  $z_1 = x$ ,  $z_2 = x'$ , and  $z_3 = y$ . The inputs to the system are  $y'$  and  $g$ .

1.6

1.13 – Assign the states as  $x_1 = I$ ,  $x_2 = \theta$ , and  $x_3 = \theta'$ .

A – Consider a pendulum with length  $L$ . The angle between the vertical reference and the pendulum is  $\theta$  (positive angle measured counterclockwise). The mass at the end of the pendulum is  $m$ . A horizontal force  $u$  is applied to the pendulum (positive force from left to right).

(1) Write the nonlinear differential equations for this system, where state  $x_1 = \theta$  and  $x_2 = \theta'$ .

(2) Linearize the equations with the assumption that  $\theta$  is very small to obtain a two-state linear differential equation of the form  $\dot{x}' = Ax + Bu$ .

#### *Matlab Exercise*

Use Matlab to plot simulation results for Example 1.2 in the text. Assign the states as  $x_1 = i$  and  $x_2 = v_C$ . Assign the output  $y = v_L$ , and input  $u = v$ . Use  $C = 1$ ,  $L = 2$ ,  $R_1 = 3$ ,  $R_2 = 4$ , and final time = 10 seconds. Hand in the following:

1. 1. Zero-state output plot due to a sine(t) input (use Matlab's *lsim* function)
2. 2. Zero-input output plot due to  $x(0) = [0.1 \ 0.2]'$  (use Matlab's *initial* function)
3. 3. Total output plot due to a sine(t) input and  $x(0) = [0.1 \ 0.2]'$  (obtained by superimposing the previous two results)

4. 4. Total output plot due to a sine(t) input and  $x(0) = [0.1 \ 0.2]'$  (obtained by writing your own simulation loop using rectangular integration). Increase your simulation step size until your simulation no longer converges. At what value of simulation step size does your simulation stop converging?

5. 5. Matlab program listing

Hint: You can use [Chen220.m](#) as a template for this assignment.

## Chapter 2 Homework

### Written Exercises

2.1, 2.3, 2.5, 2.8, 2.10, 2.14, 2.16, 2.22

Note: In 2.10, you can consider “degeneracy” to mean the same thing as “nullity.”

### Matlab Exercises

1. Use Matlab’s *rank* command to verify your answers for 2.10.
2. Use Matlab’s *det* command to verify your answers for 2.10(a) and (b).
3. Use Matlab’s *orth* command to solve 2.14. (Note that this will probably give you a different answer than you obtain manually.) Use Matlab’s *dot* command and *norm* command to verify the orthonormality of the vectors that resulted from the *orth* command. What is the two-norm of the matrix that results from augmenting the *orth* results into a 4 x 3 matrix?

## Chapter 3 Homework

### Written Exercises

3.1

3.2 – Hint: Start by calculating  $d(AA^{-1}) / dt$ .

3.3 – Prove your answer (that is, don’t just answer yes or no).

3.4, 3.6, 3.13, 3.17, 3.18

3.19 – For your particular solution, choose  $x_1 = 0$ . For your homogenous solution, choose  $x_3 = 1$ .

### Matlab Exercise

- 3.17 – Use Matlab’s backslash operator (or *mldivide* function) to verify your hand-calculated answer. Also, use Matlab’s *pinv* command to verify your answer.
- 3.19 – Use Matlab’s *null* command to verify your hand-calculated answer. (Note that Matlab’s answer will not be the same as your answer; it will be a multiple of your answer.)
- 3.21 – Hand in your Matlab program and a plot of  $g1[k]$  and  $g2[k]$ , where  $k$  is the number of data points that have been processed. Note that your Matlab program should not have any matrix inversions except possibly a 2 x 2 inversion at the beginning of the RLS process.

## Chapter 4 Homework

### Written Exercises

- 4.1 – No need to go through a lot of calculations. Just think about the definition and physical meaning of eigendata.
- 4.5 – Do these problems by hand (not using Matlab). However, you can use Matlab's *roots* function to find the roots of polynomials if you need to. Show your work.
- 4.9
- 4.13
- 4.14 – Use  $L = 1/2$ ,  $R_2 = 2$ ,  $C = 1$ , and  $R_1 = 1$ . First assign the states as  $x_1 = i$  and  $x_2 = v_C$ . The input is  $v$ . Next define a new state  $z = V^{-1}x$ , where  $V$  is the matrix consisting of the orthonormalized eigenvectors of the  $A$  matrix. Give the state equation using this new state.

#### Matlab Exercises

1. Use Matlab's *eig* function to compute the eigendata and verify that your answers for Problem 4.5 are correct. The eigenvalues that Matlab gives you should be the same as those you computed by hand. However, the eigenvectors should be different.
2. Use Matlab's *svd* function to find the SVD of the matrices in Problem 4.5. Verify that  $A = U\Sigma V^T$ . Use Matlab's *cond* function to find the condition number of the matrices. Verify that the condition number is equal to  $\sigma_{\max} / \sigma_{\min}$ .

### Chapter 5 Homework

#### Written Exercises

- 5.4
- 5.6 – Use both the Cayley-Hamilton method and the Jordan form method. It's ok to use Matlab to get the eigendata.
- 5.9(a),(c) – Use both the Cayley-Hamilton method and the Jordan form method. It's ok to use Matlab to get the eigendata.

#### Matlab Exercises

1. Use Matlab's *expm* function to verify your answer for Problem 5.6.
2. Use Matlab's *expm1*, *expm2*, and *expm3* functions to verify your answers for Problem 5.9(a) when  $t = 1$ . Why doesn't *expm3* work in this situation?

### Chapter 6 Homework

#### Written Exercises

- 6.1 – Give a closed-form solution for  $x(t)$ , assuming that  $x(0) = 0$ .
- 6.6
- 6.9 – Assume that  $x(0) = 0$ .
- 6.10
- 6.12 – Give a closed-form solution for  $A_d$  and  $B_d$ .

#### Matlab Exercises

1. Use Matlab's *lsim* function to simulate the system of Problem 6.1. Plot the two states, and also plot the two states that you obtained by hand. Hand in your Matlab code and the plots.
2. Use Matlab's *c2d* function to verify your answer to Problem 6.12.

## Chapter 7 Homework

### Written Exercises

Problem 7.3(a), 7.4, 7.10 (assume that  $y = x$ ), 7.13, 7.14 (assume that this is a continuous time system), 7.17

### Matlab Exercises

(A) Simulate the system in Problem 7.13 with  $L = C = I$  and  $R = \infty$ . To run the simulation, create a state space model in Matlab (e.g., using the *tf2ss* or *ss* function), and then use the *lsim* function to plot the zero-state output of the system for the following two cases:

- (1)  $u(t) = 1$  – What can you conclude from the output plot about the BIBO stability of the system?
- (2)  $u(t) = \sin(t)$  – What can you conclude from the output plot about the BIBO stability of the system?

Hand in your Matlab program, your plots, and your conclusion about BIBO stability.

(B) Simulate the system in Problem 7.17(a). To run the simulation, create a state space model in Matlab, and then use the *initial* function to plot the zero-input *state* response of the system. Note: do not plot the *output*, instead plot the *states*. Do this for the following two cases:

- (1)  $x(0) = [0 ; 1]^T$  – What can you conclude from the output plot about the asymptotic stability of the system?
- (2)  $x(0) = [1 ; 0]^T$  – What can you conclude from the output plot about the asymptotic stability of the system?

Hand in your Matlab program, your plots, and your conclusion about asymptotic stability.

## Chapter 8 Homework

### Written Exercises

Problems 8.1 and 8.8

Problem 8.A – Consider the system described in Problem 8.1(d). Compute  $e^{At}$ . Then compute the controllability Grammian (as a function of  $t$ ). (To compute the controllability Grammian you will have to integrate terms like  $te^{at}$  and  $t^2e^{at}$ . You can find these integrals in a math handbook or an integral table on the internet.)

Problem 8.B – Consider the system described in Problem 8.1(a). Suppose it is known that  $u_0 = 1$ ,  $u_1 = -1$ ,  $y_0 = 2$ , and  $y_1 = -2$ . Compute  $x_0$ . Using your computed  $x_0$ , begin at  $k = 0$  and use the system equations to compute  $y_0$ ,  $x_1$ , and  $y_1$ .

### Matlab Exercise

Problem 8.1 – Use Matlab's *ctrb*, *obsv*, and *rank* functions to verify your hand-calculated answers to Problem 8.1

Problem 8.A – Use the controllability Grammian that you found in Written Exercise 8.A above to compute the control that drives the initial state  $[1; -2]$  to the final state  $[-2; 1]$  in 10 seconds. (Note that the controllability Grammian is equal to 0 at  $t = 0$ , so the control at  $t = 0$  is undefined). Simulate the system with your computed control. Hand in your Matlab program, a plot of the control, and a plot of the two states.

## Chapter 9 Homework

### *Written Exercises*

Problem 9.4

Problem 9.6

Problem 9.A – Consider the transfer function shown in Problem 9.1 in the text. Use the methods that we discussed in class to find the controllable form realization and the observable form realization.

Problem 9.B – Consider the system  $x' = Ax + Bu$  and  $y = Cx$ , where  $A = [-1 \ 4; 4 \ -1]$ ,  $B = [1; 1]$ , and  $C = [1 \ 1]$ . Find the controllable decomposition of this system. What controllable first order system has the same transfer function as this system?

Problem 9.C – Consider the system  $x' = Ax + Bu$  and  $y = Cx$ , where  $A = [0 \ 1; -1 \ 2]$ ,  $B = [1; 0]$ , and  $C = [1 \ -1]$ . Find the observable decomposition of this system. What observable first order system has the same transfer function as this system?

### *Matlab Exercises*

Problem 9.A – Use Matlab's *tf2ss*, *ss*, and *canon* functions to obtain the observable form realization of the system given in Written Exercise 9.A above. Notice that your written answer will differ from your Matlab answer because there is more than one possible observable form realization.

Problem 9.B – Use Matlab's *ctrbf* function to find the controllable decomposition of the system given in Written Exercise 9.B above. Notice that your written answer will differ from your Matlab answer. According to Matlab, what controllable first order system has the same transfer function as the original system?

Problem 9.C – Use Matlab's *obsvf* function to find the observable decomposition of the system given in Written Exercise 9.C above. Notice that your written answer will differ from your Matlab answer. According to Matlab, what observable first order system has the same transfer function as the original system?

## Chapter 10 Homework

### *Written Exercises*

Problem 10.4(a) and (b)

Problem 10.11

Problem 10.16

### *Matlab Exercise*

Problem 10.A – Use Matlab's *place* and *acker* functions to verify your answer to Written Exercise 10.11.

Problem 10.B – Simulate the nonlinear inverted pendulum system that we discussed in Chapter 1 with the control  $u = -Kx$  where the state feedback matrix  $K$  is determined

using Matlab's place function as applied to the linearized system. You can use [Ex03.m](#) (available on the course web site) as a template for the simulation. Use the system parameters (cart mass, pendulum length, etc.) given in Ex03.m for the simulation. Use the initial conditions  $y(0) = 0.1$  and  $\theta(0) = 0.1$  rad. Run the simulation for two cases:

(1) Closed loop eigenvalues at  $-1.5+0.5j$ ,  $-1.5-0.5j$ ,  $-1+j$ ,  $-1-j$

(2) Closed loop eigenvalues at double the values given in part (1)

Run the simulations for 10 seconds. Hand in your Matlab program. Hand in plots of the cart position, the pendulum angle, and the control as a function of time for both simulations. How does the change in closed-loop eigenvalues between case (1) and case (2) affect the system response and the control magnitude? Why?