Engr 447

Homework #4

915815724

Farnam Adelkhani

1) Compute and concatenate the controllability and observability matrices yourself to practice how this is done. Use MATLAB to compute sub matrices and assemble them to check the ranks. Later, check your answers by `ctrb` and `obsv` commands to make sure you have been right:

System below has 1 input and 2 outputs (how do you know?), answer the following questions:

a. Is the system controllable?
b. Is it observable?
c. Is it observable from the 1st element of the output vector alone?
d. Is it observable from the 2nd element of the output vector alone?

$$\begin{cases} dx/dt = \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} = \mathbf{Cx} + \mathbf{Du} \end{cases}; \quad \mathbf{A} = \begin{bmatrix} -3 & 1 & 0 \\ 2 & -3 & 2 \\ 0 & 1 & -3 \end{bmatrix}; \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}; \ \mathbf{C} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}; \mathbf{D} = 0$$

ⓐ

```
A = [-3 1 0;2 -3 2;0 1 -3];
B = [1;0;0];
C = [0 0 1;1 0 0];

Co=ctrb(A,B)
unco = length(A) - rank(Co)
```

Save file to synchronize breakpoints.

Command Window

```
Co =

     1    -3    11
     0     2   -12
     0     0     2


unco =

     0
```

⟹ Uncontrollable States are zero, so it is controllable

ⓑ

```
A = [-3 1 0;2 -3 2;0 1 -3];
B = [1;0;0];
C = [0 0 1;1 0 0];

Co=ctrb(A,B)
unco = length(A) - rank(Co)
```

Save file to synchronize breakpoints.

Command Window

```
Co =

     1    -3    11
     0     2   -12
     0     0     2


unco =

     0
```

⟹ Observable states are zero, so it is observable c

ers/farnamadelkhani/Documents/MATLAB/447

```
A = [-3 1 0;2 -3 2;0 1 -3];
B = [1;0;0];
C = [0 0 1];
D = 0;
ob=obsv(A,C)
unob = length(A) - rank(ob)
```

Command Window

```
ob =

     0     0     1
     0     1    -3
     2    -6    11


unob =

     0

fx >>
```

⇒ unobservable states are zero, so it is in the observable

d

sers/farnamadelkhani/Documents/MATLAB/447 H

```
A = [-3 1 0;2 -3 2;0 1 -3];
B = [1;0;0];
C = [1 0 0];
D = 0;
ob=obsv(A,C)
unob = length(A) - rank(ob)
```

Command Window

```
ob =

     1     0     0
    -3     1     0
    11    -6     2


unob =

     0

fx >>
```

⇒ unobservable states are zero, so it is observable

$$\begin{cases} dx/dt = \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} = \mathbf{Cx} + \mathbf{Du} \end{cases}; \quad \mathbf{A} = \begin{bmatrix} -3 & 1 & 0 \\ 2 & -3 & 2 \\ 0 & 1 & -3 \end{bmatrix}; \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}; \ \mathbf{C} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}; \mathbf{D} = 0$$

---

2) For the system in problem 1:

a. Where are the system poles?

b. What are the system time constants? → Inverse of eigen

c. Define the above system in MATLAB by `ss` command and find its unit step responses by `step` command.

d. Use the symbolic method in handout 16 to place the system poles at $-10, -10, -10$. What are the feedback gains for each of the three state variables? Check your answers with that of MATLAB `acker` command to make sure they are the same.

e. Compute the system plus the state feedback new **A**, **B**, **C**, **D** matrices (see equation 16 – 1) and use `step` command to find the unit step responses of the system with the state feedback for 5 seconds, compare to the open loop case (part c).

f. By what factor should the new system input r be scaled (**N** in equation 16 – 1) to have the second output steady state value equal to 1?

ⓐ

```
A = [-3 1 0;2 -3 2;0 1 -3];
B = [1;0;0];
C = [1 0 0];
D = 0;
ob = obsv(A,C)
unob = length(A) - rank(ob)

eig(A)
```

Finding eigen values gives system poles

Command Window

```
unob =

     0


ans =

   -5.0000
   -3.0000
   -1.0000
```

poles are all negative

ⓑ

$e^{-t/\tau}$ → tau is time constant

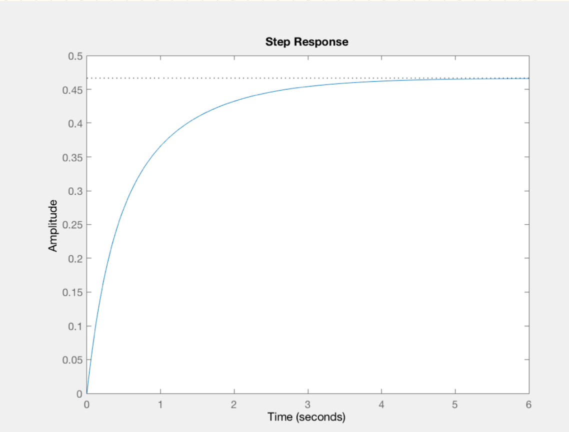
$1/s+5 \Rightarrow \dfrac{1/5}{\frac{s}{5}+1}$

ⓒ

```
A = [-3 1 0;2 -3 2;0 1 -3];
B = [1;0;0];
C = [1 0 0];
D = 0;
ob = obsv(A,C)
unob = length(A) - rank(ob)

eigen = eig(A)
%tc = inv(eigen)

sys = ss(A, B, C, D)
step(sys)
```



$$\begin{cases} dx/dt = \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} = \mathbf{Cx} + \mathbf{Du} \end{cases}; \quad \mathbf{A} = \begin{bmatrix} -3 & 1 & 0 \\ 2 & -3 & 2 \\ 0 & 1 & -3 \end{bmatrix}; \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}; \; \mathbf{C} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}; \mathbf{D} = 0$$

---

2) For the system in problem 1:
   a. Where are the system poles?
   b. What are the system time constants? → inverse of eigen
   c. Define the above system in MATLAB by `ss` command and find its unit step responses by `step` command.
   d. Use the symbolic method in handout 16 to place the system poles at $-10, -10, -10$. What are the feedback gains for each of the three state variables? Check your answers with that of MATLAB `acker` command to make sure they are the same.
   e. Compute the system plus the state feedback new $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ matrices (see equation 16 – 1) and use `step` command to find the unit step responses of the system with the state feedback for 5 seconds, compare to the open loop case (part c).
   f. By what factor should the new system input r be scaled ($\mathbf{N}$ in equation 16 – 1) to have the second output steady state value equal to 1?

(d)

```
A= [-3 1 0; 2 -3 2; 0 1 -3]
B = [1; 0; 0]
c = [0 0 1; 1 0 0]
D= [0]
L_trans = acker (A', c', [-10 -10 -10]);
```

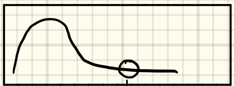
ans →

```
L = L_trans'
L=
  21
  151
  192.5
```

(e)

```
...
sys = ss(A, B, C, D);
step(sys)
AA = (A - B*K)
BB = B
CC = C - DK → 0
DD = 0
sys2 = ss(AA, BB, CC, DD)
figure; step(sys2)
```

(f)



→ invert

```
BB= B/0.00698
sys3 = (AA, BB, ect...
figure; step(sys3)
```

N is a scalar... N is a constant

N=1

$$\begin{cases} d\mathbf{x}/dt = \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} = \mathbf{Cx} + \mathbf{Du} \end{cases}; \quad \mathbf{A} = \begin{bmatrix} -3 & 1 & 0 \\ 2 & -3 & 2 \\ 0 & 1 & -3 \end{bmatrix}; \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}; \ \mathbf{C} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}; \mathbf{D} = 0$$

3) For the system in problem 1: design an observer (find its output feedback gain matrix) using MATLAB `acker` command. Place the observer poles at $-20, -20, -20$.

Find the system plus observer unit step responses by simulation in Simulink: Construct the Simulink block diagram of the feedback system plus the observer as in the figure on page 9 of Handout 16. As seen in the figure, the feedback gains found in problem 2 are now applied to the state variables coming out of the observer to provide the feedback to the plant (plant = system in problem 1). Put a "step" block at the plant input and a scope on the system output $\mathbf{y}$ and simulate for 5 seconds. Compare the responses to those of problem 2. Don't forget to change multiplication rules to "matrix (K*u)" as explained on page 8 of Handout 16.

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \overbrace{\begin{bmatrix} - & - & - \\ - & - & - \end{bmatrix}}^{C} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

rows columns

C(1,:) → returns first row

obsv(A,C) → instead obsv(A, C(1,:)) = o1

rank(o1) = 3

obsv Matrix =

$$\begin{bmatrix} 0 \ 0 \ 1 \\ [0 \ 0 \ 1] A \\ [0 \ 0 \ 1] A^2 \end{bmatrix}$$

Both observable