Contents

Marion Anderson ECE 4550 Fall 2018 HW 5

```
clear; clc; close all

M1 = 1; M2 = 1;

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

5.1: Controllability

```
scriptC = B;  % Initial controllability matrix
for n = 1:length(A)-1  % adding to controllability matrix
    scriptC = [scriptC, (A^n) * B];
end

fprintf('\n'); disp('scriptC ='); fprintf('\n')
disp(scriptC)
fprintf('\n'); disp('det(scriptC) ='); fprintf('\n')
disp(det(scriptC))


% The controllability matrix has non-zero determinant, and so must be
% full rank. This system is controllable.
```

```
scriptC =

     0     1     0    -2
     0     0     0     2
     1     0    -2     0
     0     0     2     0


det(scriptC) =

    -4
```

5.2: Regulator Gains

```
% regulator gain matrix
syms K1 K2 K3 K4 s
```

```
K = [K1 K2 K3 K4];

sR = -10;  % desired pole location

% Symbolic (K) char poly coeffs
KPoly = det(eye(length(A))*s - (A - B*K))
KCoeffs = coeffs(KPoly, s)

% Numeric char poly coeffs
RegPoly = expand((s-sR)^length(A))
RegCoeffs = coeffs(RegPoly, s)

% Solve for K from coeffs
KValsStruct = solve(RegCoeffs == KCoeffs);

% Extract K values
KNames = fieldnames(KValsStruct);
K = [];
for i = 1:length(KNames)
    K = [K double(KValsStruct.(KNames{i}))];
end
disp('------------------------------------------------------------')
K
disp('------------------------------------------------------------')

% Check
AminusBKeig = eig(A - B*K)
```

```
KPoly =

2*K1 + 2*K2 + 2*K3*s + 2*K4*s + K1*s^2 + K3*s^3 + 4*s^2 + s^4


KCoeffs =

[ 2*K1 + 2*K2, 2*K3 + 2*K4, K1 + 4, K3, 1]


RegPoly =

s^4 + 40*s^3 + 600*s^2 + 4000*s + 10000


RegCoeffs =

[ 10000, 4000, 600, 40, 1]

------------------------------------------------------------

K =

         596        4404          40        1960

------------------------------------------------------------

AminusBKeig =

 -10.0022 + 0.0000i
```

```
  -10.0000 + 0.0022i
  -10.0000 - 0.0022i
   -9.9978 + 0.0000i
```

5.3: Observability

```
scriptO = C;  % Initial observability matrix
for n = 1:length(A)-1  % adding to observability matrix
    scriptO = [scriptO; C * (A^n)];
end

fprintf('\n'); disp('scriptO ='); fprintf('\n')
disp(scriptO)
fprintf('\n'); disp('det(scriptO) ='); fprintf('\n')
disp(det(scriptO))


% The observability matrix has non-zero determinant, and so must be
% full rank. This system is observabile.
```

```
scriptO =

     0     1     0     0
     0     0     0     1
     2    -2     0     0
     0     0     2    -2


det(scriptO) =

    -4
```

5.4: Estimator Gains

```
% estimator gain matrix
syms L1 L2 L3 L4
L = [L1; L2; L3; L4];

sL = -10;  % desired pole location

% Symbolic (L) char poly coeffs
LPoly = det(eye(length(A))*s - (A - L*C))
LCoeffs = coeffs(LPoly, s)

% Numeric char poly coeffs
EstPoly = expand((s-sR)^length(A))
EstCoeffs = coeffs(EstPoly, s)

% Solve for K from coeffs
LValsStruct = solve(EstCoeffs == LCoeffs);

% Extract K values
LNames = fieldnames(LValsStruct);
L = [];
for i = 1:length(LNames)
```

```
    L = [L; double(LValsStruct.(LNames{i}))];
end
disp('------------------------------------------------------')
L
disp('------------------------------------------------------')

% Check
AminusLCeig = eig(A - L*C)
```

```
LPoly =

2*L3 + 2*L4 + 2*L1*s + 2*L2*s + L2*s^3 + L4*s^2 + 4*s^2 + s^4


LCoeffs =

[ 2*L3 + 2*L4, 2*L1 + 2*L2, L4 + 4, L2, 1]


EstPoly =

s^4 + 40*s^3 + 600*s^2 + 4000*s + 10000


EstCoeffs =

[ 10000, 4000, 600, 40, 1]

------------------------------------------------------

L =

        1960
          40
        4404
         596

------------------------------------------------------

AminusLCeig =

 -10.0022 + 0.0000i
 -10.0000 + 0.0022i
 -10.0000 - 0.0022i
  -9.9978 + 0.0000i
```

Published with MATLAB® R2017a