EE572 - Solution to HW#12

```
ahat =

    0.2500   -0.2500    0.5000         0
    0.2500    0.7500   -0.5000         0
   -0.2500   -0.2500    1.0000         0
    0.2500    0.7500    0.5000    1.0000

»
» bhat = [ 1 2;-1 1;2 -1;1 1]

bhat =

     1     2
    -1     1
     2    -1
     1     1

» chat=[1 -1 2 0;2 1 0 1]

chat =

     1    -1     2     0
     2     1     0     1

» % Find the controllable and uncontrollable eigenvalues of the open-loop
system.
» M = [bhat ahat*bhat ahat^2*bhat ahat^3*bhat]

M =

  Columns 1 through 7

    1.0000    2.0000    1.5000   -0.2500    1.7500   -1.3750    1.8750
   -1.0000    1.0000   -1.5000    1.7500   -1.7500    2.1250   -1.8750
    2.0000   -1.0000    2.0000   -1.7500    2.0000   -2.1250    2.0000
    1.0000    1.0000    1.5000    1.7500    1.7500    2.1250    1.8750

  Column 8

   -1.9375
    2.3125
   -2.3125
    2.3125

» rank(M)

ans =

     4

» % The rank of M is full, therefore, all eigenvalues are controllable
» % The eigenvalues are:
» eig(ahat)

ans =

    1.0000
    1.0000
    0.5000
    0.5000

» % 1b) Is the system stablizable?
» % Yes!  It is completely controllable!
» % c)     If yes, design a feedback regulator, wk = -kxk such that the
closed-loop controllable eigenvalues are {1/8}.
» % First, we have repeated eigenvalues.  Must use Feedback to jumble them:
» Kf=[1 0 1 0;0 -1 0 1]

Kf =

     1     0     1     0
     0    -1     0     1

» eig(ahat-bhat*Kf)

ans =

  -2.7216
   1.3459 + 1.1248i
   1.3459 - 1.1248i
   0.0299

» % Great! Eigenvalues are distinct!
» anew=ahat-bhat*Kf

anew =

   -0.7500    1.7500   -0.5000   -2.0000
    1.2500    1.7500    0.5000   -1.0000
   -2.2500   -1.2500   -1.0000    1.0000
   -0.7500    1.7500   -0.5000         0

» % Next, reduce down to a single input with random Ks:
» Ks = [1;-1]

Ks =

     1
    -1

» Bs = bhat*Ks

Bs =

    -1
    -2
     3
     0

» Ms = [Bs anew*Bs anew^2*Bs anew^3*Bs]

Ms =

   -1.0000   -4.2500    5.1250  -11.1875
   -2.0000   -3.2500   -5.8750    3.3125
    3.0000    1.7500    7.6250  -15.1875
         0   -4.2500   -3.3750  -17.9375

» rank(Ms)

ans =

     4

» % Super! The single-input system, x_k+1 = anew*x_k + Bs*ws, is still
controllable!
» % Put in phase-variable form:
» poly(anew)

ans =

    1.0000    0.0000   -4.2500    8.5000   -0.2500

» apv=[0 1 0 0; 0 0 1 0; 0 0 0 1; -ans(5) -ans(4) -ans(3) -ans(2)]

apv =

         0    1.0000         0         0
         0         0    1.0000         0
         0         0         0    1.0000
    0.2500   -8.5000    4.2500   -0.0000

» bpv = [0;0;0;1]

bpv =

     0
     0
     0
     1

» poly([1/8 1/8 1/8 1/8])

ans =

    1.0000   -0.5000    0.0938   -0.0078    0.0002

» kpv = [ans(5)+apv(4,1) ans(4)+apv(4,2) ans(3)+apv(4,3) ans(2)+apv(4,4)]

kpv =

    0.2502   -8.5078    4.3437   -0.5000

» % Interim test: Check the eig(apv-bpv*kpv)
» eig(apv-bpv*kpv)

ans =

   0.1250 + 0.0000i
   0.1250 - 0.0000i
   0.1250 + 0.0000i
   0.1250 - 0.0000i

» % Good! They are all {1/8}.  Now find regulator, w_k = -
[Kf+Ks*Kpv*inv(Tpv)]*x_k
» Mpv=[bpv apv*bpv apv^2*bpv apv^3*bpv]

Mpv =

         0         0         0    1.0000
         0         0    1.0000   -0.0000
         0    1.0000   -0.0000    4.2500
    1.0000   -0.0000    4.2500   -8.5000

» Tpv=Ms*inv(Mpv)

Tpv =

   -1.6250    9.3750   -4.2500   -1.0000
    0.1250    2.6250   -3.2500   -2.0000
    2.8750   -5.1250    1.7500    3.0000
    0.1250   -3.3750   -4.2500         0

» K = Kf+Ks*kpv*inv(Tpv)

K =

   -0.2294   -0.0437    0.3944   -0.0086
    1.2294   -0.9563    0.6056    1.0086

» eig(ahat-bhat*K)

ans =

   0.1247
   0.1250 + 0.0003i
   0.1250 - 0.0003i
   0.1253

» % It works!  All eigenvalues are {1/8}
» % 1d)    What is the settling-time of the closed-loop system if Ts = 10
msec.
» Ts = 0.01

Ts =

    0.0100

» z_max = 1/8

z_max =

    0.1250

» ts = log(z_max)/Ts
» ts = -4*Ts/log(z_max)

ts =

    0.0192

» % Therefore, the settling time is 19.2 msec
» % 1e)    Now design a controller such that yk goes to [ 3 -8 ]T in steady-
state
» y_ref = [3 8]'

y_ref =

     3
     8

» I_pxp = eye(2)

I_pxp =

     1     0
     0     1

» inv([chat zeros(2,2);ahat-eye(4) bhat])*[I_pxp;zeros(4,2)]

ans =

    0.2500         0
   -0.2500    0.0000
    0.2500         0
   -0.2500    1.0000
         0         0
         0    0.0000

» Nx = ans(1:4,:)

Nx =

    0.2500         0
   -0.2500    0.0000
    0.2500         0
   -0.2500    1.0000

» Nu = ans(5:6,:)

Nu =
```

```
  1.0e-017 *

         0         0
         0    0.3469

» % Note that Nu is zero because the original system was unstable
» % 1f)    Find the observable eigenvalues of the discrete-time system in
problem 1.
» % Find Ao and Bo and treat as a control problem
» Ao = ahat'

Ao =

    0.2500    0.2500   -0.2500    0.2500
   -0.2500    0.7500   -0.2500    0.7500
    0.5000   -0.5000    1.0000    0.5000
         0         0         0    1.0000

» Bo = chat'

Bo =

     1     2
    -1     1
     2     0
     0     1

» O = [Bo Ao*Bo Ao^2*Bo Ao^3*Bo]'

O =

    1.0000   -1.0000    2.0000         0
    2.0000    1.0000         0    1.0000
   -0.5000   -1.5000    3.0000         0
    1.0000    1.0000    1.0000    1.0000
   -1.2500   -1.7500    3.5000         0
    0.5000    1.0000    1.5000    1.0000
   -1.6250   -1.8750    3.7500         0
    0.2500    1.0000    1.7500    1.0000

» rank(O)

ans =

     4

» % All eigenvalues are observable!!
» % 1g)    If detectable is the dual of stabilizable, define detectable and
determine if the given system is detectable
» % A system is detectable if we can find Ko to make the observer eigenvalues
stable (i.e., eig(A-KoC))
» % Therefore, a system is detectable iff all unobservable eigenvalues are
stable
» % Since our system is completely observable, it must be detectable
» % 1g)    Design an observer such that all the observable eigenvalues are
deadbeat.
» % We must jumble the repeated eigenvalues
» Kfo=[1 0 1 0;0 -1 0 1]

Kfo =

     1     0     1     0
     0    -1     0     1

» anewo=Ao-Bo*Kf

anewo =

   -0.7500    2.2500   -1.2500   -1.7500
    0.7500    1.7500    0.7500   -0.2500
   -1.5000   -0.5000   -1.0000    0.5000
         0    1.0000         0         0

» eig(anewo)

ans =

   -2.6237
    1.8073
    0.3163
    0.5000

» % Good! We have jumbled the eigenvalues so that they are all distinct
» % Reduce to single "input" system

» Kso = [1;0]

Kso =

     1
     0

» Bso = Bo*Kso

Bso =

     1
    -1
     2
     0

» Mso = [Bso anewo*Bso anewo^2*Bso anewo^3*Bso]

Mso =

    1.0000   -5.5000   10.7500  -33.8750
   -1.0000    0.5000   -5.2500    6.6250
    2.0000   -3.0000   10.5000  -23.7500
         0   -1.0000    0.5000   -5.2500

» rank(Mso)

ans =

     4

» % Good!  The single "input" system is still controllable!
» % Now, let's put it in Phase Variable form:
» poly(anewo)

ans =

    1.0000    0.0000   -5.2500    4.0000   -0.7500

» apvo=[0 1 0 0; 0 0 1 0; 0 0 0 1; -ans(5) -ans(4) -ans(3) -ans(2)]

apvo =

         0    1.0000         0         0
         0         0    1.0000         0
         0         0         0    1.0000
    0.7500   -4.0000    5.2500   -0.0000

» bpvo = [0;0;0;1]

bpvo =

     0
     0
     0
     1


» % Check if eig(apvo-bpvo*kpvo) are deadbeat (zero)
» eig(apvo-bpvo*kpvo)


» kpvo = [apvo(4,1) apvo(4,2) apvo(4,3) apvo(4,4)]

kpvo =

    0.7500   -4.0000    5.2500   -0.0000

» eig(apvo-bpvo*kpvo)

ans =

     0
     0
     0
     0

» % Great!  Now lets find the complete "feedback" gain, Kc:
» Mpvo=[bpvo apvo*bpvo apvo^2*bpvo apvo^3*bpvo]

Mpvo =

         0         0         0    1.0000
         0         0    1.0000   -0.0000
         0    1.0000   -0.0000    5.2500
    1.0000   -0.0000    5.2500   -4.0000

» Tpvo=Mso*inv(Mpvo)

Tpvo =

   -1.0000    5.5000   -5.5000    1.0000
         0   -0.0000    0.5000   -1.0000
   -0.0000    0.0000   -3.0000    2.0000
   -0.0000    0.5000   -1.0000         0


» Kc = Kfo+Kso*kpvo*inv(Tpvo)

Kc =

    0.2500   -2.5000    0.1250    0.2500
         0   -1.0000         0    1.0000

» Ko = Kc'

Ko =

    0.2500         0
   -2.5000   -1.0000
    0.1250         0
    0.2500    1.0000

» % To check the final observer gain, check eig(ahat-Ko*chat):
» eig(ahat-Ko*chat)

ans =

  1.0e-003 *

  -0.3760
   0.3760
  -0.0000 + 0.3760i
  -0.0000 - 0.3760i

» % Close enough to zero (rounding error)!
» % 1h)    How fast will your observer error go to zero if Ts = 10 msec?
» % Since the observer is deadbeat, will will go to zero in a maximum of n=4
steps
» % Thus, the observer error will be zero by 4*Ts or 40 msec
```

Block Diagram: