The Separation Principle

NDSU ECE 463/663

Lecture #20

Inst: Jake Glower

Please visit Bison Academy for corresponding
lecture notes, homework sets, and solutions

Recap

If a system is controllable, you can stabilze it using full-state feedback

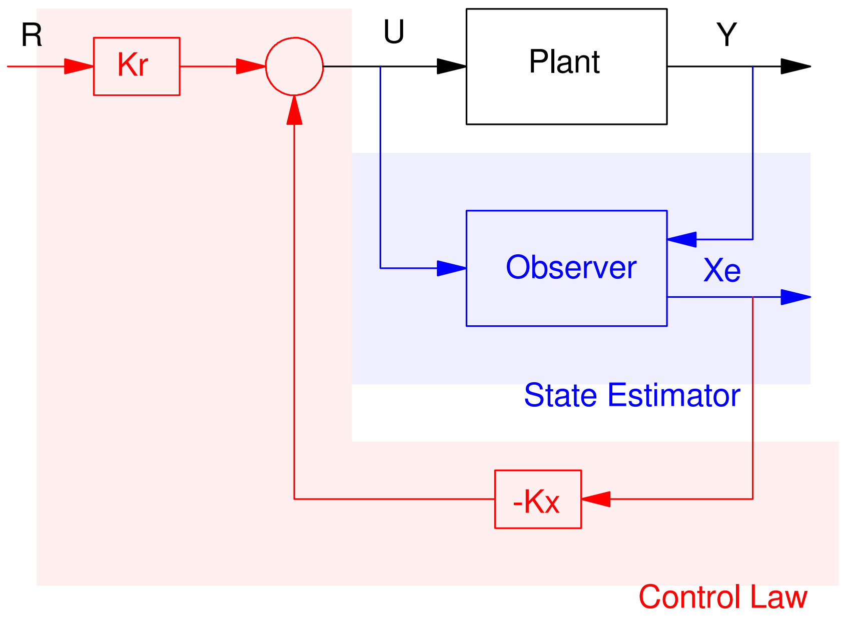
- This requires measurements of *all* of the system's states

If a system is observable, you can estimate the systems states

- Full-order observer

If a system is both controllable and observable, you can

- Estimate the states using a full-order observer, and
- Stabilize the system using these estimates

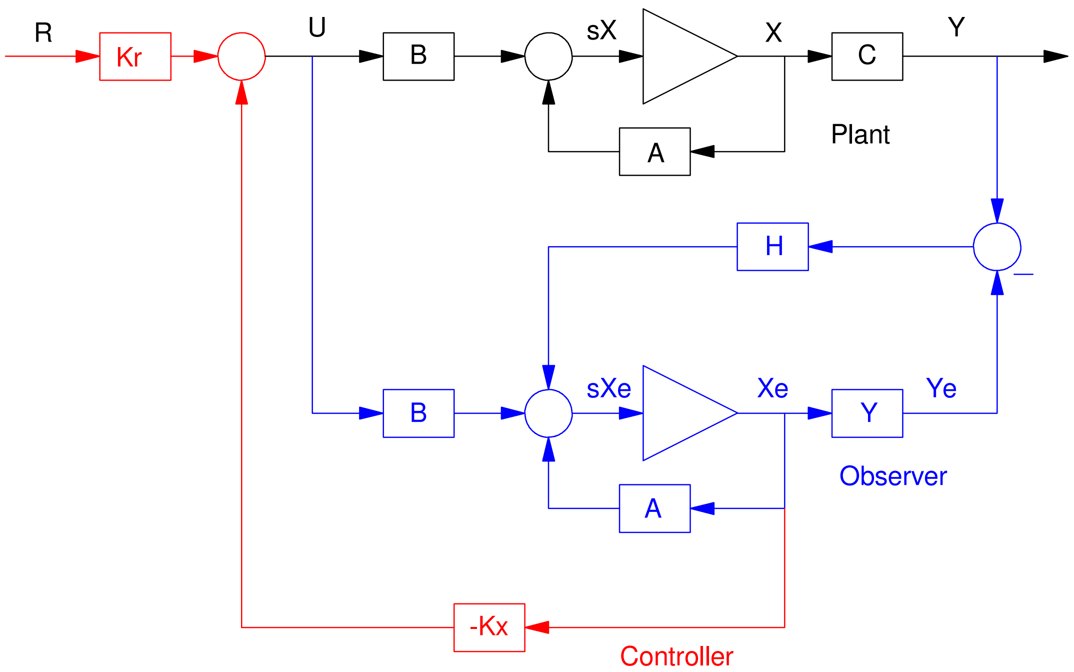


Problem:

How do you determine *both*

- The feedback gains, Kx, and
- The observer gains, H?

Do they interract?



The Separation Principle:

Assume you have a plant

$$sX = AX + BU$$

$$Y = CX$$

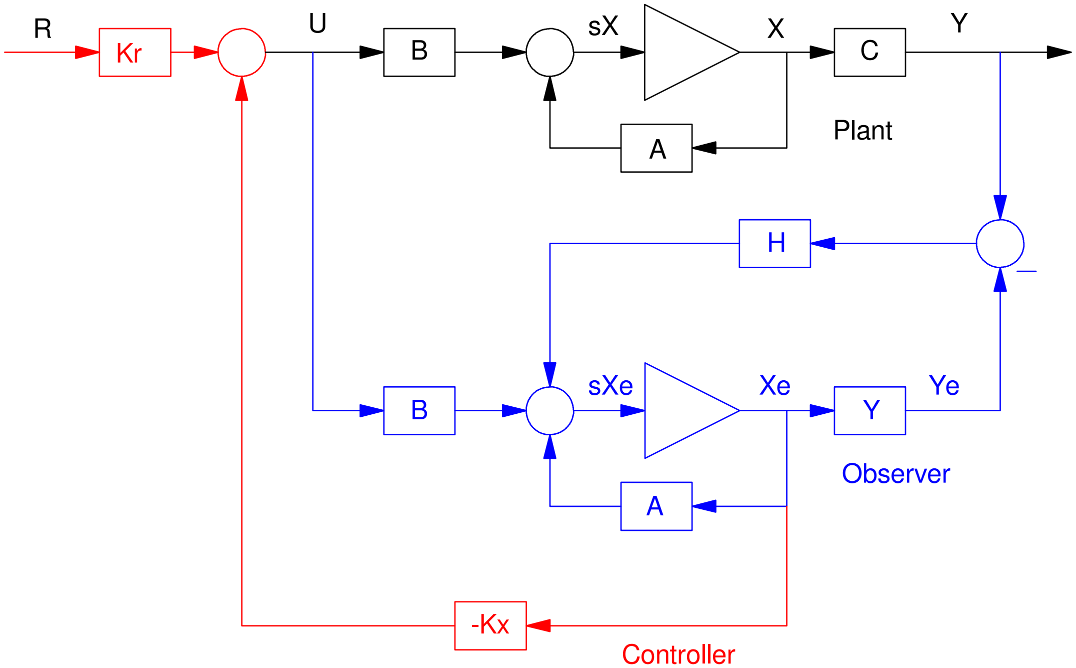
along with a state-estimator

$$sX_e = AX_e + BU + H(Y - Y_e)$$

$$Y_e = CX_e$$

along with the control law

$$U = K_r R - K_x X_e$$



The augmented system becomes

$$s\begin{bmatrix} X \\ X_e \end{bmatrix} = \begin{bmatrix} A & 0 \\ HC & A - HC \end{bmatrix}\begin{bmatrix} X \\ X_e \end{bmatrix} + \begin{bmatrix} B \\ B \end{bmatrix} U$$

or, substituting for U

$$s\begin{bmatrix} X \\ X_e \end{bmatrix} = \begin{bmatrix} A & -BK_x \\ HC & \hat{A} - HC - BK_x \end{bmatrix}\begin{bmatrix} X \\ X_e \end{bmatrix} + \begin{bmatrix} B \\ B \end{bmatrix} K_r R$$

Do a change of variable:

$$\begin{bmatrix} X \\ E \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} X \\ X_e \end{bmatrix}$$

Do a similarity transform

$$s\begin{bmatrix} X \\ E \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & -1 \end{bmatrix}^{-1} \begin{bmatrix} A & -BK_x \\ HC & A-HC-BK_x \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} X \\ E \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 1 & -1 \end{bmatrix}^{-1} \begin{bmatrix} B \\ B \end{bmatrix} K_r R$$

Simplify

$$s\begin{bmatrix} X \\ E \end{bmatrix} = \begin{bmatrix} A-BK_x & BK_x \\ 0 & A-HC \end{bmatrix} \begin{bmatrix} X \\ E \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} K_r R$$

Note that the augmented system's A matrix is diagonal. The eigenvalues are...

- The eigenvalues of (A - BKx) and
- The eigenvalues of (A - HC)

You can design the full-state feedback gains without any regard of the observer gains and visa versa.

- This is the separation principle

Selection of Observer Gains

Kx: This is determined by the system requirements

- Settling time, Overshoot, ...

H: How do you determine H?

- Thought #1: Make the observer 3-10x faster than the plant

  The observer states determine the plant's input.

  You want the observer states to be correct before applying U

  There is always sensor noise, so don't go overboard.

  3-10x faster than the closed-loop system is a reasonable compromise

- Thought #2: It depends...

  If there is a lot of sensor noise, make H small

  If there is a lot of input disturbances, make H large

  If there is both sensor noise and input disturbances, it's a trade-off

The second approach leads to a Kalman filter (optimal observer)

Coming soon....

Example 1: Heat Equation

Design a fedback controller for the following system using only the output measurement, Y, so that the closed-loop system has

- No overshoot for a step input,
- A 2% settling time of 4 seconds, and
- A DC gain of one.

$$sX = \begin{bmatrix} -2 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} X + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} U$$

$$Y = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} X$$

Controller Design

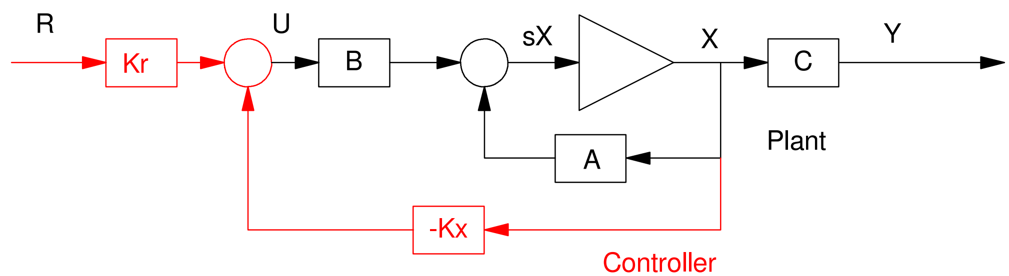
Use the separation principle to design the controller and observer separately.

Pick your favorite controller: Assume full-state feedback:

$$U = K_r R - K_x X$$

In Matlab:

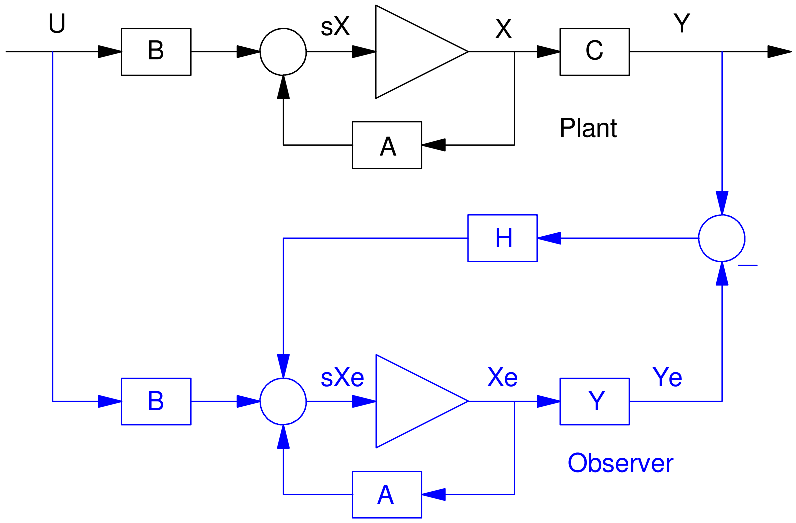
```
A = [-2,1,0,0;1,-2,1,0;0,1,-2,1;0,0,1,-1];
B = [1;0;0;0];
Kx = ppl(A, B, [-1,-2,-3,-4])

Kx =   3    5    7    8

DC = -C*inv(A-B*Kx)*B;
Kr = 1/DC

Kr = 24
```



Observer Design

Since the closed-loop system has a dominant pole at s = -1, place the observer poles 3x faster

```
H = ppl(A', C', [-3, -4, -5, -6])'

H =

    61.0000
    70.0000
    38.0000
    11.0000
```



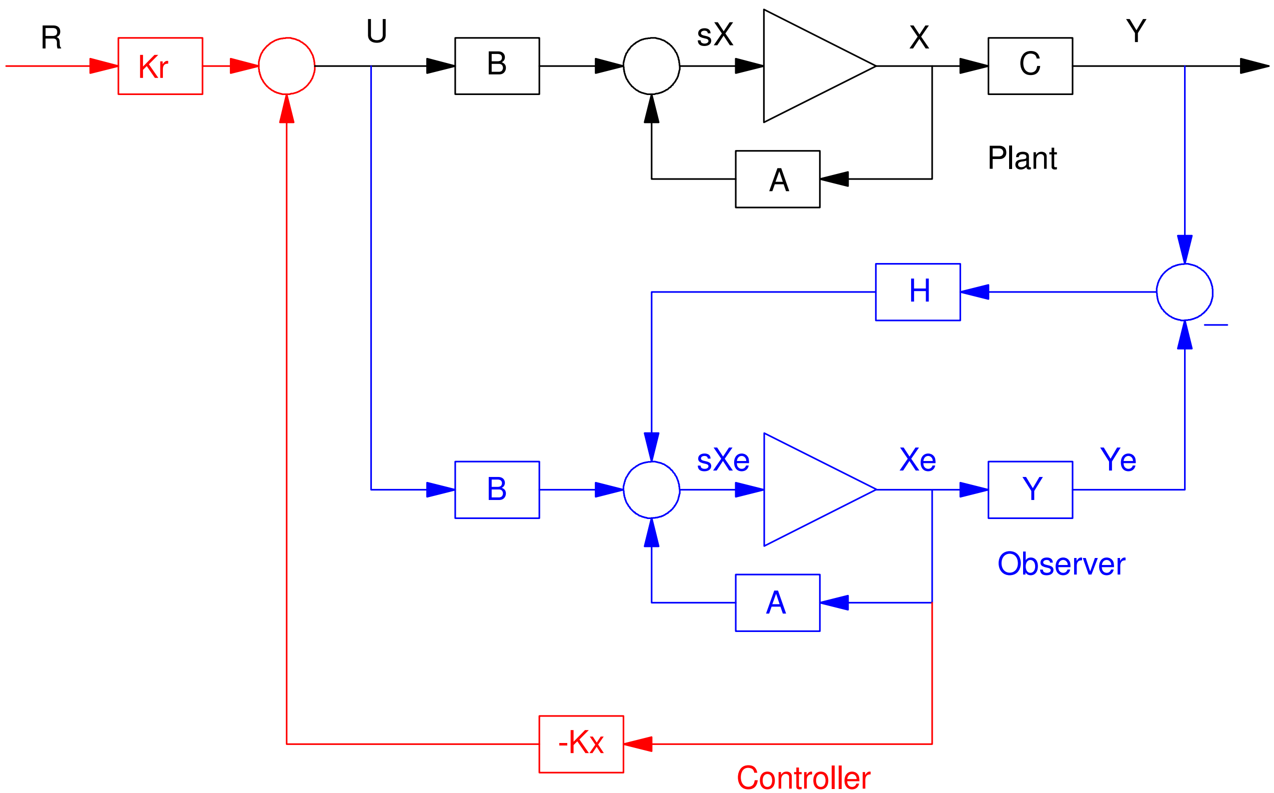
Note:

- You really don't care how you generate U.
- The control law doesn't affect the observer design

Overall Design

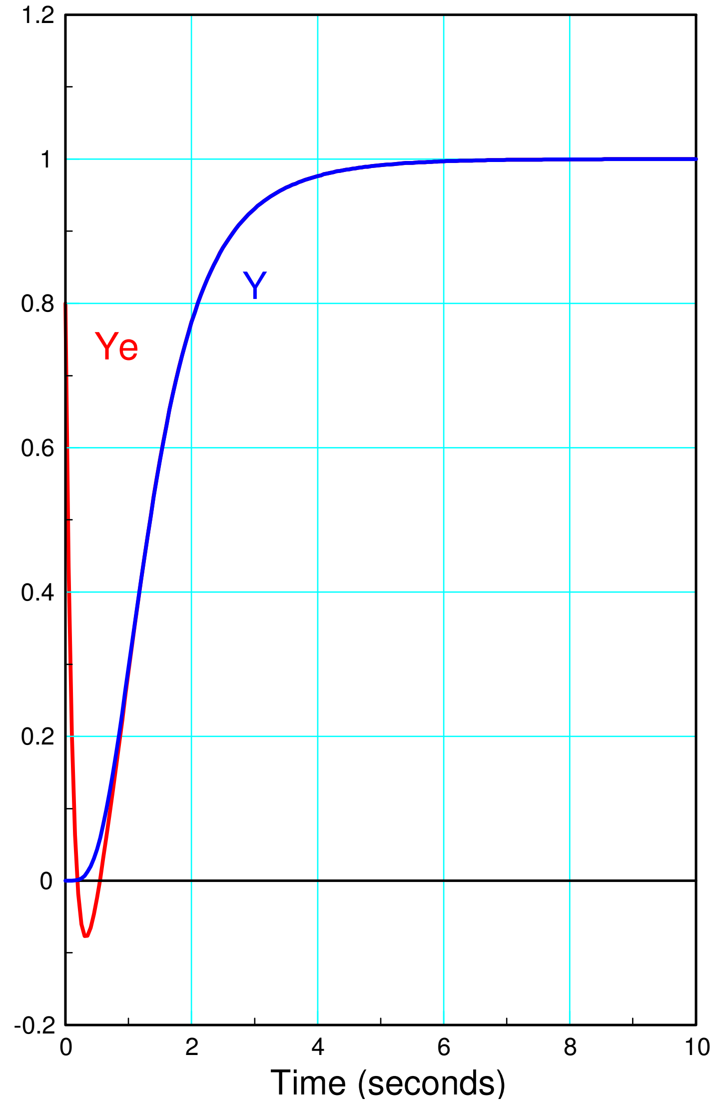
- Instead of feeding back the plant states, feed back the state estimates (Xe)

$$s\begin{bmatrix} X \\ X_e \end{bmatrix} = \begin{bmatrix} A & -BK_x \\ HC & A-HC-BK_x \end{bmatrix}\begin{bmatrix} X \\ X_e \end{bmatrix} + \begin{bmatrix} B \\ B \end{bmatrix} K_r R$$



In Matlab:

```
A8 = [A,-B*Kx ; H*C, A-H*C-B*Kx];
B8 = [B;B];
B8 = [B;B]*Kr;
C8 = [C,0*C ; 0*C,C];
D8 = [0;0];

t = [0:0.01:10]';
X0 = [0;0;0;0;  0.2;0.4;0.6;0.8];

R = 0*t+1;
y = step3(A8,B8,C8,D8,t,X0,R);
plot(t,y);
```



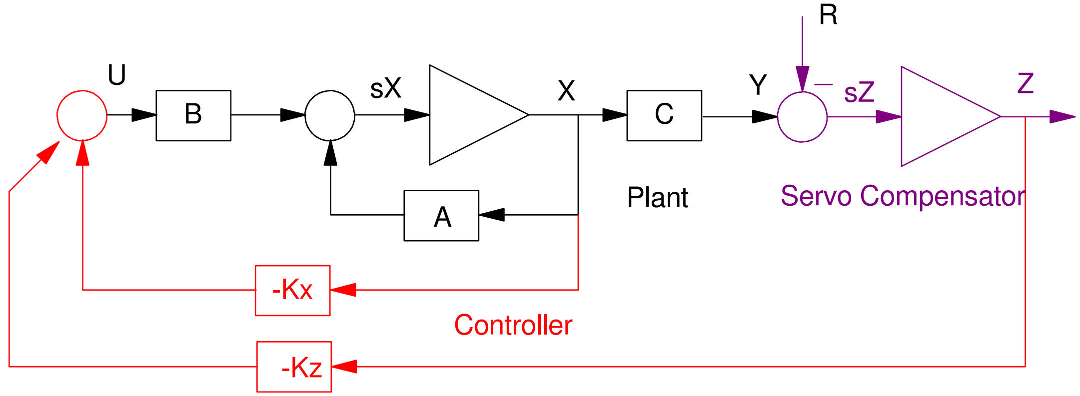
Example 2: Servo Compensator Design

- Note: The separation principle states that the controller design and the observer design are completely separate. To illustrate this, design the controller using a servo compensator.

Let

$$sZ = R - Y$$

$$U = -K_x X - K_z Z$$

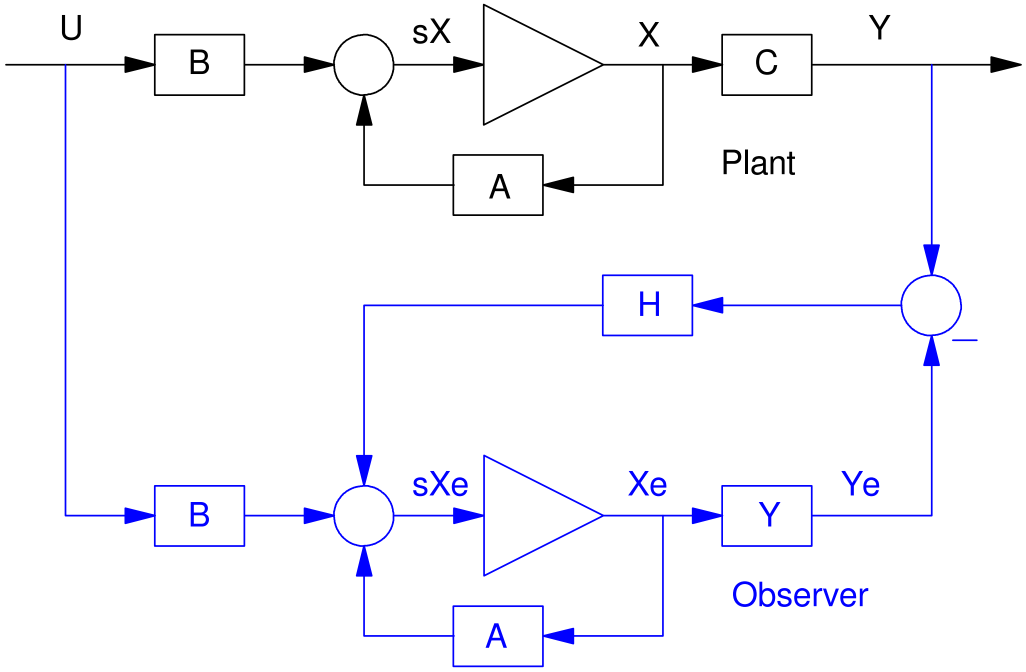



```
A5 = [A , 0*B ; C, 0];
B5 = [B;0];
K5 = ppl(A5,B5,[-1,-2,-3,-4,-5])

                          Kx                                        Kz
K5 =     8.0000    30.0000    77.0000   158.0000  120.0000
```

Observer Design

- No change
- Place the observer poles 3x faster

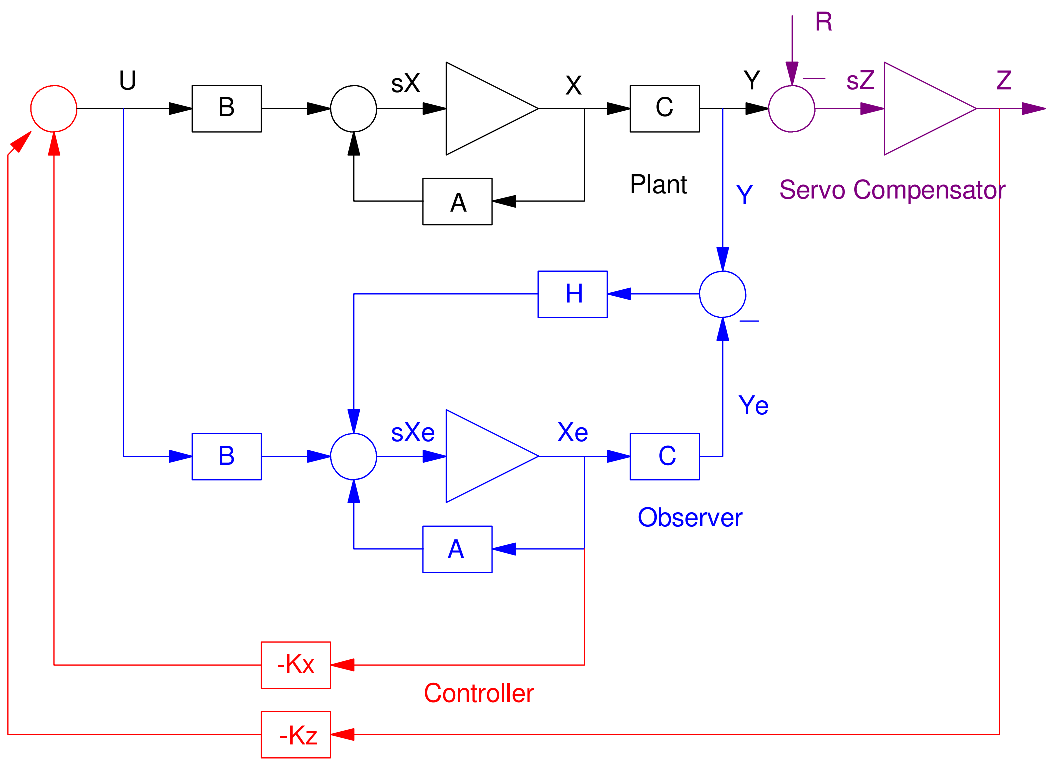
```
H = ppl(A', C', [-3, -4, -5, -6])'

H =

    61.0000
    70.0000
    38.0000
    11.0000
```



Overall System:

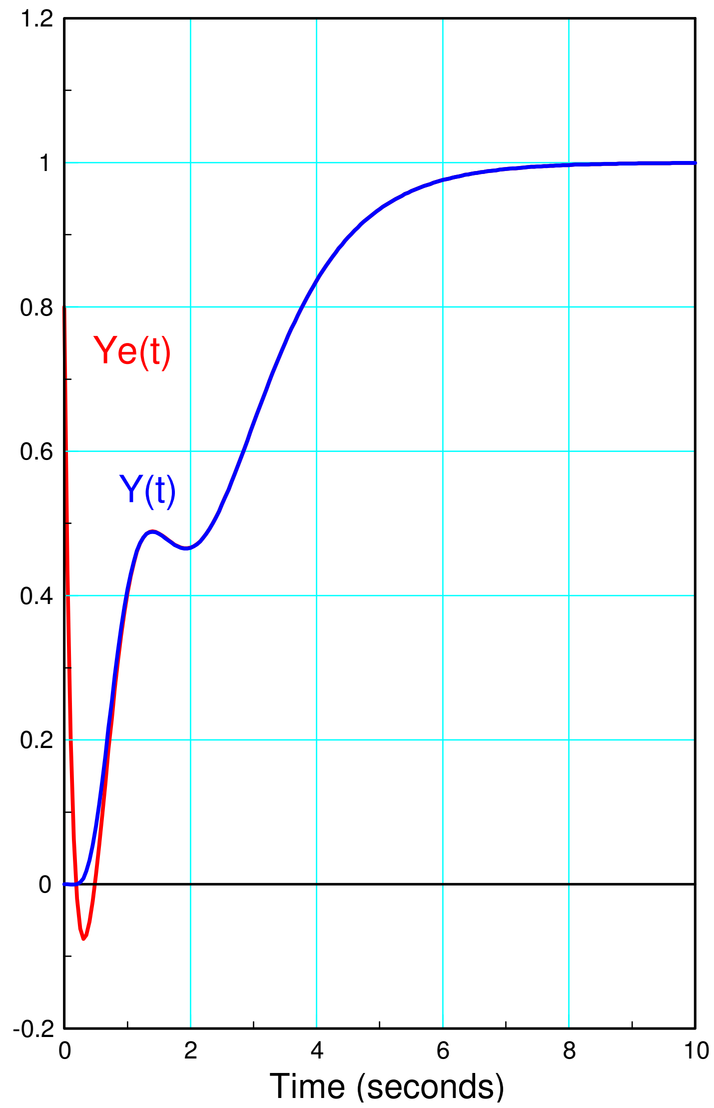
- Plant & Servo Compensator & Observer

$$\begin{bmatrix} sX \\ sX_e \\ sZ \end{bmatrix} = \begin{bmatrix} A & -BK_x & -BK_z \\ HC & A-HC-BK_x & -BK_z \\ C & 0 & 0 \end{bmatrix} \begin{bmatrix} X \\ X_e \\ Z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} R$$



In Matlab:

```
Kx = K5(1:4);
Kz = K5(5);

A9 = [A,      -B*Kx,      -B*Kz;
      H*C, A-H*C-B*Kx, -B*Kz;
      C,        0*C,          0];
B9 = [0*B ; 0*B; -1];
C9 = [C,0*C,0 ; 0*C, C, 0];
D9 = [0;0];

X0 = [0;0;0;0;0.2;0.4;0.6;0.8;0];

t = [0:0.01:10]';
R = 0*t+1;
y = step3(A9, B9, C9, D9, t, X0, R);
plot(t,y);
```



Resulting A9:

$$\begin{bmatrix} sX \\ sX_e \\ sZ \end{bmatrix} = \begin{bmatrix} A & -BK_x & -BK_z \\ HC & A-HC-BK_x & -BK_z \\ C & 0 & 0 \end{bmatrix} \begin{bmatrix} X \\ X_e \\ Z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} R$$

```
           X                        Xe                   Z
  -2     1     0     0 :   -8   -30   -77  -158 :  -120
   1    -2     1     0 :    0     0     0     0 :     0
   0     1    -2     1 :    0     0     0     0 :     0
   0     0     1    -1 :    0     0     0     0 :     0
----------------------------------------------------------
   0     0     0    61 :  -10   -29   -77  -219 :  -120
   0     0     0    70 :    1    -2     1   -70 :     0
   0     0     0    38 :    0     1    -2   -37 :     0
   0     0     0    11 :    0     0     1   -12 :     0
----------------------------------------------------------
   0     0     0     1 :    0     0     0     0 :     0
```

Example 3: Cart and Pendulum

Assume you *only* measure position.

- Can you control the system?

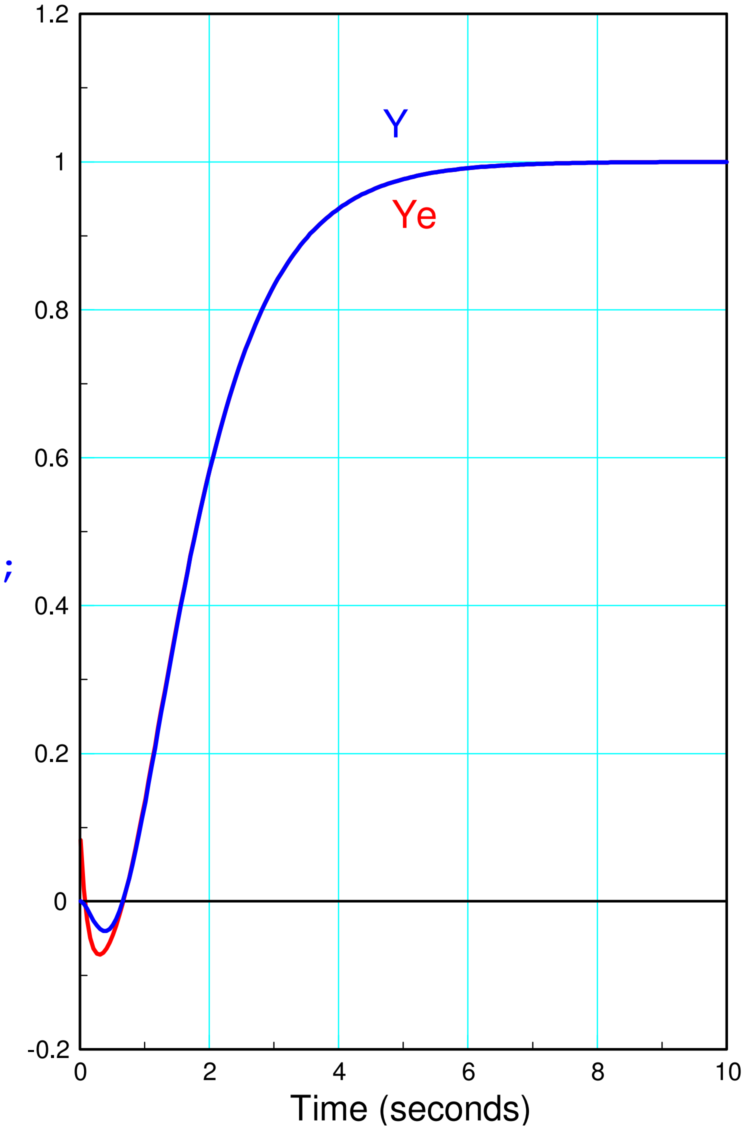
Yes - the system is observable from position.

```
A = [0,0,1,0;0,0,0,1;0,-4.9,0,0;0,14.7,0,0];
B = [0;0;0.5;-0.5];
C = [1,0,0,0];
Kx = ppl(A,B,[-1,-2,-3,-4])
     -4.8980 -104.2980  -10.2041  -30.2041

DC = -C*inv(A-B*Kx)*B;
Kr = 1/DC
     -4.8980

H = ppl(A', C', [-3, -4, -5, -6])'
   18.0000
 -123.7959
  133.7000
 -474.5694
```

Nonlinear Simulation Code

- Kx, Kr, H found using previous methods

```
X   = zeros(4,1);
Xe = zeros(4,1) + 0.01*randn(4,1);
Ref = 1;
dt = 0.01;
t = 0;
Kx = [-4.8980 -104.2980  -10.2041  -30.2041];
Kr = -4.8980;
% Observer
A = [0,0,1,0;0,0,0,1;0,-4.9,0,0;0,14.7,0,0];
B = [0;0;0.5;-0.5];
C = [1,0,0,0];
H = [ 18.0000 -123.7959  133.7000 -474.5694]';
y = [];
while(t < 10)
  U = Kr * Ref -Kx*Xe;
  dX = CartDynamics(X, U);
  dXe = A*Xe + B*U + H*(C*X - C*Xe);
  X   = X   + dX   * dt;
  Xe = Xe + dXe * dt;
  t = t + dt;
  CartDisplay(X, Ref);
  end
```



Example 3:

For the first 4/3 second, the state estimates are poor

- Observer poles: { -3, -4, -5, -6}
- Poor estimates results in poor response

How do you adjust the feedback gains, Kx, so that they only kick in once the observer converges?

This is a *very* difficult problem and has yet to be solved.

- Time-varying gains (time doman) imply convolution in the frequency domain
- This is a nonlinear system

  Eigenvalues no longer apply

  Many of our definitions of stability no longer apply

Summary

Observers and controllers do not affect each other

- Design the controller as if all states were measured
- Design the observer without any concern about the controller

The two should work together without any impact on each other's stability