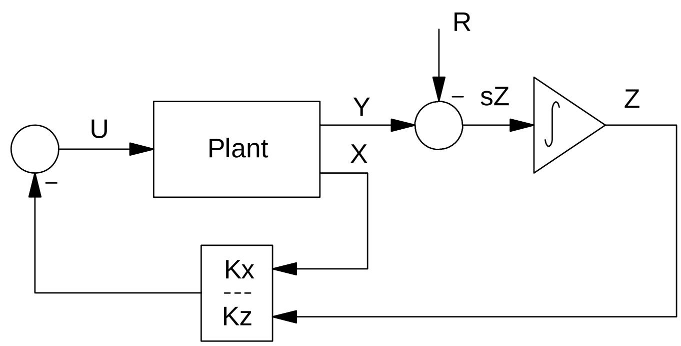
Servo-Compensators: AC Setpoints

Problem:

- Track a set point with spectra $(\lambda_1, \lambda_2, \lambda_3)$ and
- Reject disturbances with spectra sinusoidal set point, R. How to you force Y→R?

Solution: Recall that for a constant set point, you add an integrator



Servo-Compensator design for a constant set-point (R)

One way to think of how this works is as follows. Since R is a constant (s=0), at steady-state all states will be a constant. Since the servo-compensator has an infinte gain at s=0, the only way you can reach steady-state is for the input to the servo-compensator to be zero. This forces

Y-R = 0

Now, suppose that R is a sinusoid with a frequency of $s = j\omega$ rather than a constant. At steady state, all signals will be sinsoids at frequency ω. If you could build a servo-compensator with an infinte gain at this frequency, then choose gains Kx and Kz to stabilize the closed-loop system, then the only way the system could reach steady-state is for the error to be zero.

Hence, design a servo-compensator with poles at $s = \pm j\omega$.

Example: Let the plant be

$$sX = AX + BU$$

$$Y = CX$$

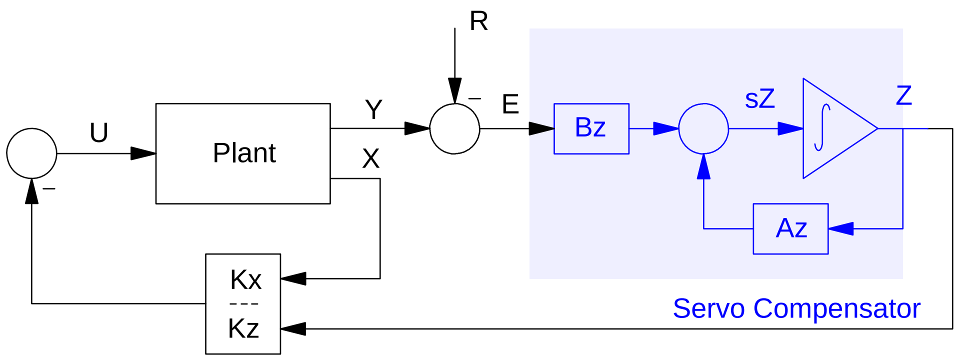
Define a servo-compensator

$$sZ = A_z Z + B_z$$

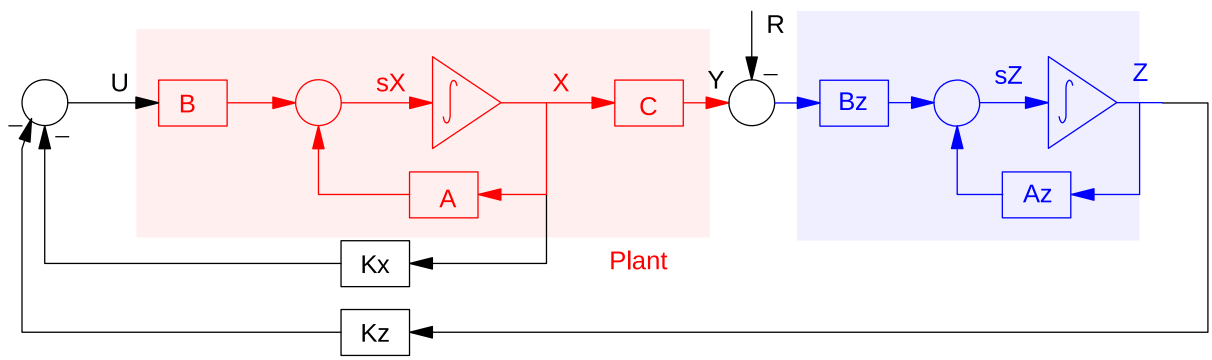
so that the eigenvalue of Az are

$$spec(A_z) = \pm j\omega$$

Feed the servo-compensator with the difference between Y and the set point R



In state-space, the plant plus servo-compensator looks like the following:



$$s\begin{bmatrix} X \\ Z \end{bmatrix} = \begin{bmatrix} A & 0 \\ B_zC & A_z \end{bmatrix}\begin{bmatrix} X \\ Z \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix}U + \begin{bmatrix} 0 \\ -B_z \end{bmatrix}R$$

$$U = -\begin{bmatrix} K_x & K_z \end{bmatrix}\begin{bmatrix} X \\ Z \end{bmatrix}$$

or you can write this as

$$s\begin{bmatrix} X \\ Z \end{bmatrix} = \begin{bmatrix} A - BK_x & -BK_z \\ A_z & B_zC \end{bmatrix}\begin{bmatrix} X \\ Z \end{bmatrix} + \begin{bmatrix} 0 \\ -B_z \end{bmatrix}R$$

Example: Force a 4th-Order RC Filter to Track a Sinusoidal Set Point

Let

$$R(t) = \sin(2t)$$

Design a controller which

Design a feedback control law for the following system so that

- The 2% settling time is 4 seconds,
- There is no overshoot for a step input, and
- $Y \rightarrow R$

for

$$R(t) = \sin(2t)$$

Solution: First, design a system with poles at $s = \pm j2$. There are multiple solutions, one being:

$$sZ = \begin{bmatrix} 0 & 2 \\ -2 & 0 \end{bmatrix} Z + \begin{bmatrix} 1 \\ 1 \end{bmatrix} U_z$$

Create an augmented system: plant plus servo

$$s\begin{bmatrix} X \\ Z \end{bmatrix} = \begin{bmatrix} A & 0 \\ B_z C & A_z \end{bmatrix} \begin{bmatrix} X \\ Z \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} U + \begin{bmatrix} 0 \\ -B_z \end{bmatrix} R$$

$$s\begin{bmatrix} X \\ \cdots \\ Z \end{bmatrix} = \left[\begin{array}{cccc:cc} -2 & 1 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 & -2 & 0 \end{array}\right] \begin{bmatrix} X \\ \cdots \\ Z \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \cdots \\ 0 \\ 0 \end{bmatrix} U + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \cdots \\ -1 \\ -1 \end{bmatrix} R$$

Design a full-state feedback control law to meet the design specs. Somwhat arbitrarilly, place the closed-loop poles at {-1, -2, -3, -4, -5, -6} using Bass Gura

In Matlab:

```
-->A

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

-->B

    1.
    0.
    0.
    0.

-->C

    0.    0.    0.    1.
```

Add in the servo-compensator (any system with poles at $\pm j2$)

```
-->Az = [0,2;-2,0]

    0.    2.
  - 2.    0.

-->Bz = [1;1]

    1.
    1.
```

Augment the plant plus servo compensator

```
-->A6 = [A,zeros(4,2);Bz*C,Az]
 A6  =

  - 2.    1.    0.    0.  :  0.    0.
    1.  - 2.    1.    0.  :  0.    0.
    0.    1.  - 2.    1.  :  0.    0.
    0.    0.    1.  - 1.  :  0.    0.
    - - - - - - - - - - - - - - - - -
    0.    0.    0.    1.  :  0.    2.
    0.    0.    0.    1.  : -2.    0.


-->B6 = [B;0;0]
 B6  =

    1.
    0.
    0.
    0.
    0.
    0.
```

Use Bass Gura, find the transformation matrix to take you to controller canonical form:

```
-->K6 = ppl(A6, B6, [-1,-2,-3,-4,-5,-6])

    14.     86.     299.     540.   - 1180.     340.
```

Check that the closed-loop poles of (A - BK) are where they should be:

```
>> eig(A6 - B6*K6)

   -6.0000
   -5.0000
   -4.0000
   -3.0000
   -2.0000
   -1.0000
```

Validation:

This gets a bit tricky: you have to create a sinusoidal set point for Ref. One way is to define a system with an initial condition:

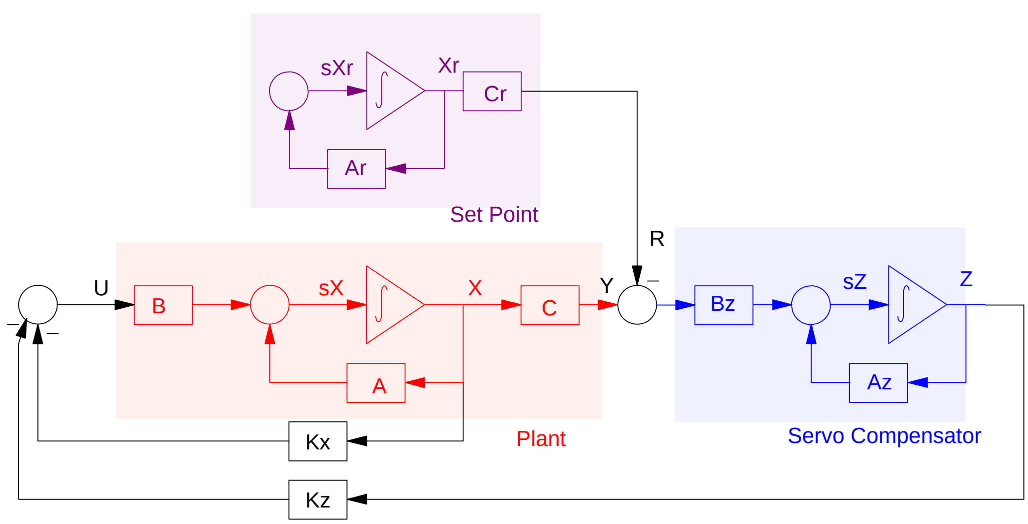
$$sX_r = \begin{bmatrix} 0 & 2 \\ -2 & 0 \end{bmatrix} X_r = A_r X_r$$

$$R = \begin{bmatrix} 1 & 0 \end{bmatrix} X_r = C_r X_r$$

Augement the 6th-order system with this system

$$s\begin{bmatrix} X_6 \\ X_r \end{bmatrix} = \begin{bmatrix} A_6 - B_6 K_6 & B_r C_r \\ 0 & A_r \end{bmatrix} \begin{bmatrix} X_6 \\ X_r \end{bmatrix}$$

Take the impulse response with an initial condition on Xr



Plant plut Servo Compensator plus Set Point model

$$s\begin{bmatrix} X \\ Z \\ X_r \end{bmatrix} = \begin{bmatrix} A - BK_x & -BK_r & 0 \\ B_z C & A_z & -B_z C_r \\ 0 & 0 & A_r \end{bmatrix}\begin{bmatrix} X \\ Z \\ X_r \end{bmatrix}$$

In Matlab: Input the system:

The Servo Compensator (same dynamics as the disturbance: Cz only applies to the disturbance)

```
>> Az = [0,2;-2,0]

     0     2
    -2     0

>> Bz = [1;1]

     1
     1

>> Cz = [1,0]

     1     0

>> A6 = [A,zeros(4,2);Bz*C,Az]

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

>> B6 = [B;0;0]

     1
     0
     0
     0
     0
     0
```

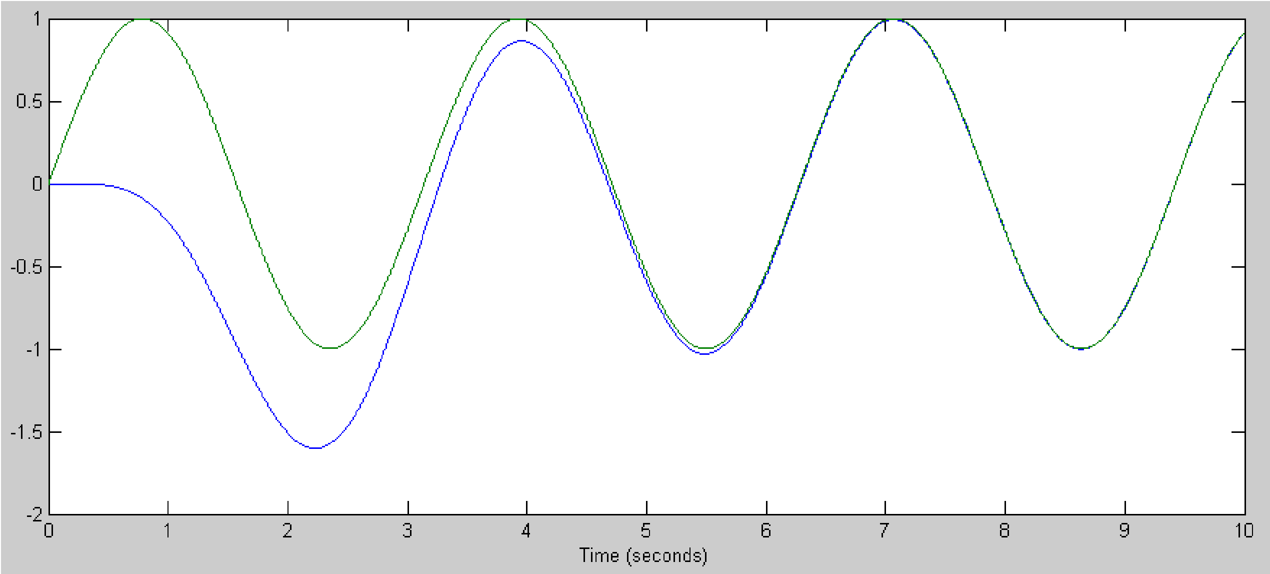
From before, using Bass Gura, the feedback gains were:

```
>> K6 = [14.    86.    299.    540.  - 1180.    340.]

    14    86   299  -640   340

>> eig(A6 - B6*K6)
```

```
   -6.0000
   -5.0000
   -4.0000
   -3.0000
   -2.0000
   -1.0000

>> Br = [0;0;0;0;Bz]

Br =

     0
     0
     0
     0
     1
     1
```

Create the augmented system (plant plus servo compensator plus disturbance)

```
>> A8 = [A6-B6*K6, -Br*Cz; zeros(2,6), Az]

                Plant                      Servo Comp          Dist
     -16      -85     -299     -540   :   1180     -340   :    0       0
       1       -2        1        0   :      0        0   :    0       0
       0        1       -2        1   :      0        0   :    0       0
       0        0        1       -1   :      0        0   :    0       0
       - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
       0        0        0        1   :      0        2   :   -1       0
       0        0        0        1   :     -2        0   :   -1       0
       - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
       0        0        0        0   :      0        0   :    0       2
       0        0        0        0   :      0        0   :   -2       0


>> X0 = [0;0;0;0;0;0;0;1];
>> C8

     0     0     0     1     0     0     0     0
     0     0     0     0     0     0     1     0

>> D8 = [0;0]

     0
     0

>> G8 = ss(A8, X0, C8, D8);
>> y = impulse(G8,t);
>> plot(t,y);
>> xlabel('Time (seconds)');
>>
```



2 rad/sec set point (green) and system output (blue) with poles placed at {-1, -2, -3, -4, -5, -6 }