Problemè 1

Consider the following state-space model:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5 & -6 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$
$$y = [1 \quad 0 \quad 0] x$$

Design an observer to place the observer poles at -10, -10, -15

Solution

The characteristic equation of the observer is given by:

$$det(sI - (A - LC)) = (s + 10)(s + 10)(s + 15) = s^3 + 35s^2 + 350s + 1500$$

From the coefficients of the characteristic equation we get

$$det(sI - (A - LC)) = s^3 + (l_1 - 6)s^2 + (l_2 - 5 - 6l_1)s + (l_3 - 5l_2)$$

Solving for the coefficients we get the observer gain matrix:

$$L = \begin{bmatrix} 4505 \\ 601 \\ 41 \end{bmatrix}$$

Thus the observer dynamics are given by:

$$\dot{\hat{x}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5 & -6 & 0 \end{bmatrix} \hat{x} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 4505 \\ 601 \\ 41 \end{bmatrix} (y - \hat{y})$$

Problemè 2

Given the plant

$$\dot{x} = \begin{bmatrix} -1 & 1 \\ 0 & 2 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$
$$y = [1 \quad 1] x$$

Design an integral controller to yield a 10% overshoot, 0.5 second settling time and zero steady-state error for a step input.

Solution

The code for the integral controller is given by p2.py.

Add an integrator to the plant to ensure zero steady-state error for a step input. The augmented state-space model becomes:

$$\dot{x}_a = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 2 & 0 \\ -1 & -1 & 0 \end{bmatrix} x_a + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} x_a$$

where $x_a = \begin{bmatrix} x \\ \int e\,dt \end{bmatrix}$ and $e = r - y$ is the tracking error.

Then, design the state feedback gains $K = \begin{bmatrix} k_1 & k_2 & k_i \end{bmatrix}$ such that the closed-loop system meets the transient response specifications. The characteristic equation of the closed-loop system is:

$$|sI - (A_a - B_a K)| = 0$$

Expanding this yields:

$$(s + k_1)(s^2 + (1 - k_2)s + k_i) = 0$$

The control law then given by:

$$u = -Kx_a = -k_1 x_1 - k_2 x_2 - k_i \int e\,dt$$

The code yields:

```
zeta: 0.5911550337988974 omega_n: 13.53282902556064
Desired poles: [  -8.          +10.91501083j   -8.          -10.91501083j
 -135.32829026 +0.j        ]
Plant model:
<LinearIOSystem>: sys[2]
Inputs (1): ['u[0]']
Outputs (1): ['y[0]']
States (2): ['x[0]', 'x[1]']

A = [[-1.  1.]
     [ 0.  2.]]

B = [[0.]
     [1.]]
```

```
C = [[1. 1.]]

D = [[0.]]


Augmented plant model:
<LinearIOSystem>: sys[3]
Inputs (1): ['u[0]']
Outputs (1): ['y[0]']
States (3): ['x[0]', 'x[1]', 'x[2]']

A = [[-1.  1.  0.]
     [ 0.  2.  0.]
     [-1. -1.  0.]]

B = [[0.]
     [1.]
     [0.]]

C = [[1. 1. 0.]]

D = [[0.]]


State feedback gains:
K = [[-10193.77795361    152.32829026 -12391.83976888]]

Integral controller transfer function:

-1.239e+04
----------
    s


Open-loop transfer function:
<LinearICSystem>: sys[6]
Inputs (1): ['u[0]']
Outputs (1): ['y[0]']
States (3): ['sys[4]_x[0]', 'sys[2]_x[0]', 'sys[2]_x[1]']

A = [[-0.00000000e+00  0.00000000e+00  0.00000000e+00]
     [ 0.00000000e+00 -1.00000000e+00  1.00000000e+00]
     [-1.23918398e+04  0.00000000e+00  2.00000000e+00]]

B = [[1.]
     [0.]
     [0.]]

C = [[0. 1. 1.]]

D = [[0.]]


Closed-loop transfer function:
<LinearICSystem>: sys[9]
Inputs (1): ['u[0]']
Outputs (1): ['y[0]']
```

```
States (3): ['sys[6]_sys[4]_x[0]', 'sys[6]_sys[2]_x[0]',
'sys[6]_sys[2]_x[1]']

A = [[ 0.00000000e+00 -1.00000000e+00 -1.00000000e+00]
     [ 0.00000000e+00 -1.00000000e+00  1.00000000e+00]
     [-1.23918398e+04  0.00000000e+00  2.00000000e+00]]

B = [[1.]
     [0.]
     [0.]]

C = [[0. 1. 1.]]

D = [[0.]]
```