$$\dot{x}_p = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_p + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} w$$

$$y = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_p + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$z = \begin{bmatrix} 1 & 0 \end{bmatrix} x_p + \begin{bmatrix} 0 & 1 \end{bmatrix} w$$

System matrices.

$A_p = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ $B_p = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ $D_p = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$

$C_p = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ $B_y = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ $D_y = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

$M_p = \begin{bmatrix} 1 & 0 \end{bmatrix}$ $D_z = \begin{bmatrix} 0 & 1 \end{bmatrix}$

$r = [n_z, n_u]$

Lumped system matrices

$A = A_p$

$B = [D_p \quad B_p]$

$C = [C_p \quad M_p]$

$D = \begin{bmatrix} D_y & B_y \\ D_z & 0 \end{bmatrix}$

Open-loop system

$S_{ol} = ss(A, B, C, D)$

Inputs to system:

$$w = \begin{cases} 0.1 & ; \ 0 \leq t \leq 1 \\ 0 & ; \ \text{otherwise} \end{cases}$$

$u = 0 \quad \forall t$

w is a 2x1 vector since the disturbance input w(t) consists of plant disturbance and measurement disturbance

Simulate open-loop system using "lsim" for t=0 to t=10 seconds

H_∞ Dynamic Controller Design in MATLAB:

s = ltisys (A, B, C, D)

r = [n_z, n_u]

[g_{opt}, G] = hinflmi (s, r)

$\Gamma_{cc} < \gamma^*$ controller

[A_c, B_c, C_c, D_c] = ltiss (G)

Closed-loop system:

$$A_{cl} = \begin{bmatrix} A_p + B_p D_c M_p & B_p C_c \\ B_c M_p & A_c \end{bmatrix}.$$

$$B_{cl} = \begin{bmatrix} D_p + B_p D_c D_z \\ B_c D_z \end{bmatrix}$$

$$C_{cl} = \begin{bmatrix} C_p + B_y D_c M_p & B_y C_c \end{bmatrix}$$

$$D_{cl} = D_y + B_y D_c D_z$$

Examination/validation of system:

- Stability: Eigenvalues (Poles): $Re(\lambda_i(A_{cl})) < 0 \leftarrow$ stability guaranteed

- Performance: H_∞ norm (energy-to-energy gain Γ_{ee}):

$$\left.\begin{aligned} &S_{cl} = ss(A_{cl}, B_{cl}, C_{cl}, D_{cl}) \\ &hinf\_norm = hinfnorm(S_{cl}) \\ &hinf\_norm \leq \underbrace{g_{opt}}_{\gamma^*} \end{aligned}\right\} \text{MATLAB}$$

$(\Gamma_{ee} < \gamma^*)$

Inputs to closed-loop system:

$$w = \begin{cases} 0.1; & 0 \leq t \leq 1 \\ 0; & \text{otherwise} \end{cases}$$

$$u = 0 \quad \forall t$$

w is a 2×1 vector since the disturbance input $w(t)$ consists of plant disturbance and measurement disturbance

Simulate closed-loop system using "lsim" for $t = 0$ to $t = 10$ seconds

Problem 3

CODE:

```
% PROBLEM 3


% Clear workspace
close all
clear
clc


% Define the system matrices
Ap = [0 1; 0 0];
Bp = [0; 1];
Dp = [0 0; 1 0];
Cp = [0 1; 0 0];
By = [0; 1];
Dy = [0 0; 0 0];
Mp = [1 0];
Dz = [0 1];


% Lumped system matrices
A = Ap;
B = [Dp Bp];
C = [Cp; Mp];
D = [Dy By; Dz 0]


% State-space system
Sol = ss(A, B, C, D)
% sys_openloop = ss(Ap, Dp, Cp, 0)


% Define the lumped (disturbance + control) input
t = 0:0.01:10;
w_amplitude = 0.1;
w_duration = 1;
w_pulse = w_amplitude * (t >= 0 & t <= w_duration);
u = zeros(size(w_pulse));
w = [w_pulse; w_pulse; u];
% w = [w_pulse; w_pulse];


% Simulate the open-loop system response
```

```
[y_ol, t_out, x_ol] = lsim(Sol, w', t);
% [y_ol, t_out, x_ol] = lsim(Sol, w, t);


% Plot the results
figure;
sgtitle('Open-Loop System Response');
subplot(4, 1, 1);
plot(t, w(2, :), t, w(3, :));
legend('W', 'U');
subplot(4, 1, 2);
plot(t, x_ol(:, 1), t, x_ol(:, 2));
legend('X1', 'X2');
subplot(4, 1, 3);
plot(t, y_ol(:, 1), t, y_ol(:, 2));
legend('Y1', 'Y2');
subplot(4, 1, 4);
plot(t, y_ol(:, 3));
legend('Z');
```

```
% H-infinity LMI
S = ltisys(A, B, C, D);
[gopt, G] = hinflmi(S,[1 1])


% Controller matrices
disp('H∞ controller:')
[Ac, Bc, Cc, Dc] = ltiss(G)
```

```
% Closed-loop system matrices
disp('Closed-loop system:')
Acl = [Ap+Bp*Dc*Mp, Bp*Cc; Bc*Mp, Ac]
Bcl = [Dp+Bp*Dc*Dz; Bc*Dz]
Ccl = [Cp+By*Dc*Mp, By*Cc]
Dcl = Dy+By*Dc*Dz


% Closed-loop system
Scl = ss(Acl, Bcl, Ccl, Dcl);
% Scl = slft(S, G);


% Verification
disp('H∞ norm:')
hinf_norm = hinfnorm(Scl)
```

```
% hinf_norm = norminf(Scl)
disp('Closed-loop poles:')
eig_Acl = eig(Acl)
if((hinf_norm < gopt) && all(real(eig_Acl) < 0.0))
    disp('Verification of H∞ norm and pole location constraints successful!')
else
    disp('Verification of H∞ norm and pole location constraints failed!')
end
```

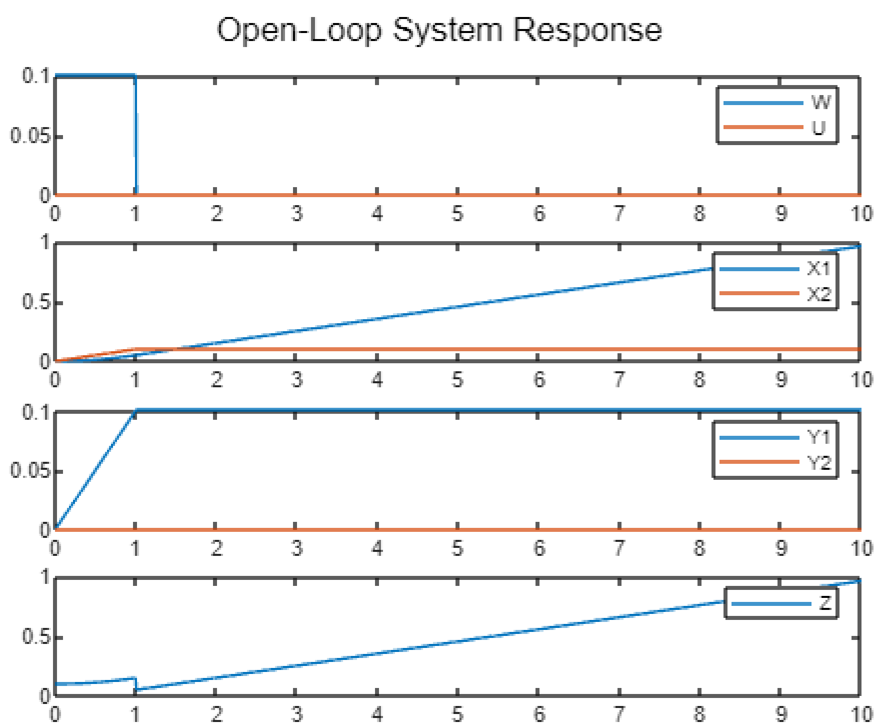
```
% Define the lumped (disturbance + control) input
t = 0:0.01:10;
w_amplitude = 0.1;
w_duration = 1;
w_pulse = w_amplitude * (t >= 0 & t <= w_duration);
u = zeros(size(w_pulse));
w = [w_pulse; w_pulse];


% Simulate the closed-loop system response
[y_cl, t_out, x_cl] = lsim(Scl, w', t);


% Plot the results
figure;
sgtitle('Closed-Loop System Response');
subplot(3, 1, 1);
plot(t, w(2, :));
legend('W');
subplot(3, 1, 2);
plot(t, x_cl(:, 1), t, x_cl(:, 2), t, x_cl(:, 3));
legend('X1', 'X2', 'X3');
subplot(3, 1, 3);
plot(t, y_cl(:, 1), t, y_cl(:, 2));
legend('Y1', 'Y2');
```

OUTPUT:

```
D = 3×3
     0     0     0
     0     0     1
     0     1     0
Sol =

  A =
        x1   x2
   x1    0    1
   x2    0    0
```

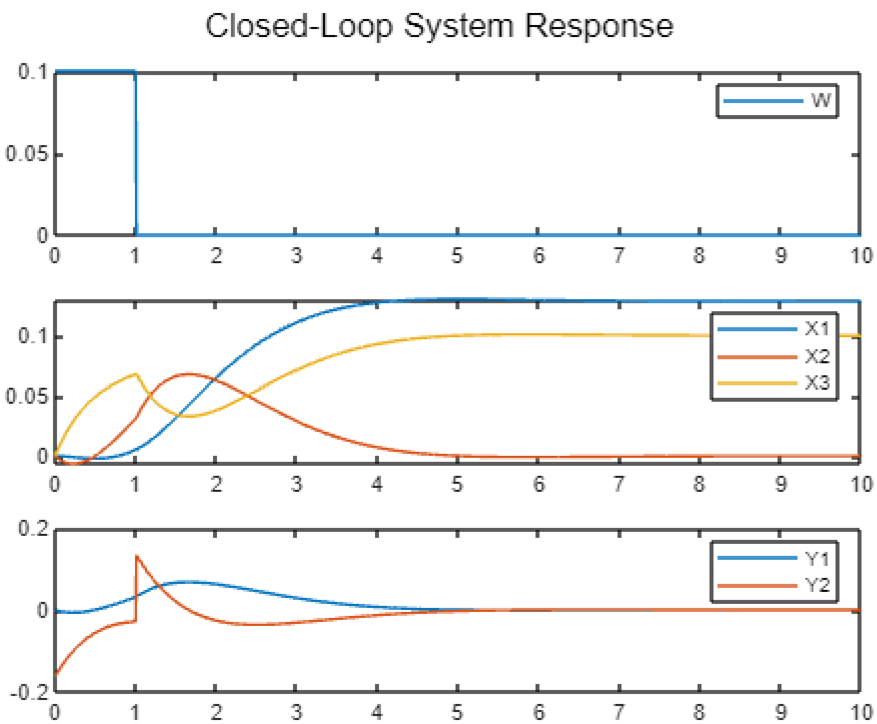
```
B =
      u1  u2  u3
 x1    0   0   0
 x2    1   0   1

C =
      x1  x2
 y1    0   1
 y2    0   0
 y3    1   0

D =
      u1  u2  u3
 y1    0   0   0
 y2    0   0   1
 y3    0   1   0
```

Continuous-time state-space model.
Model Properties

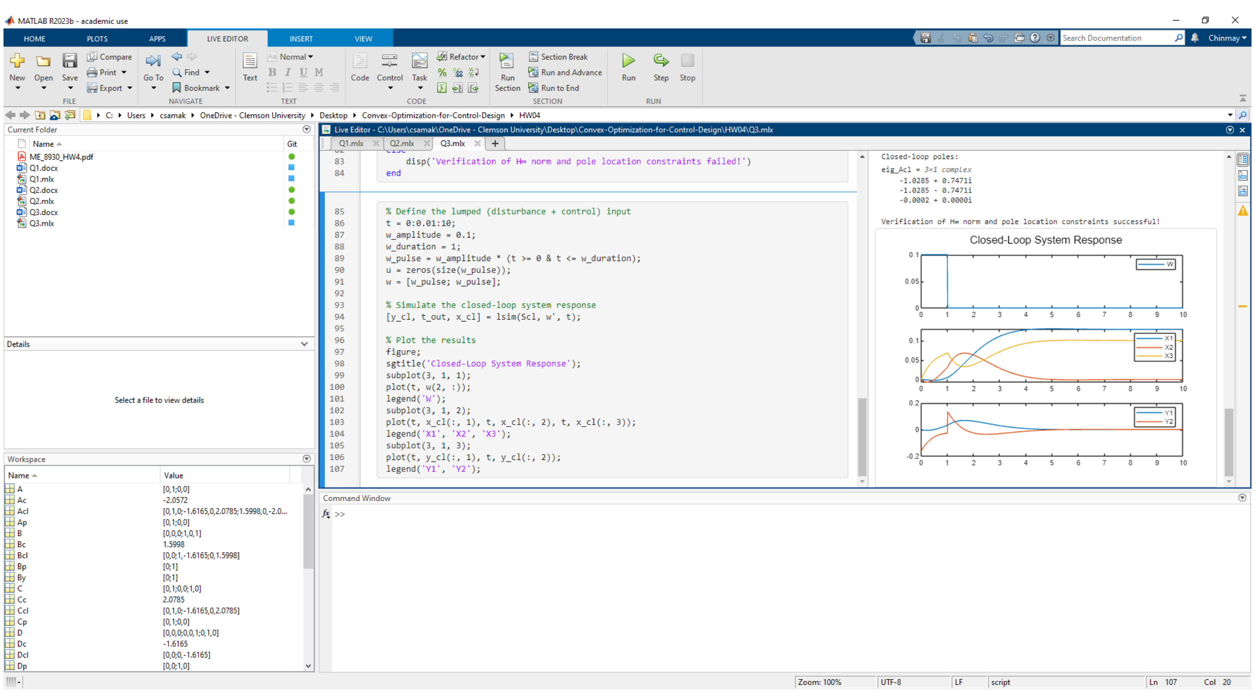



```
Minimization of gamma:

Solver for linear objective minimization under LMI constraints

Iterations   :    Best objective value so far

    1
    2                    4.331487
    3                    2.480378
    4                    2.094585
    5                    1.984537
```

```
     6                   1.984537
     7                   1.891906
     8                   1.891906
***                 new lower bound:     0.208335
     9                   1.693528
    10                   1.693528
***                 new lower bound:     0.773250
    11                   1.636185
***                 new lower bound:     1.240010
    12                   1.636185
***                 new lower bound:     1.567929
    13                   1.622326
    14                   1.620133
***                 new lower bound:     1.604742

 Result:  feasible solution of required accuracy
          best objective value:     1.620133
          guaranteed relative accuracy:  9.50e-03
          f-radius saturation:  0.399% of R =  1.00e+08

 Optimal Hinf performance:  1.620e+00
gopt = 1.6197
G = 3×3
   -2.0572    1.5998    1.0000
    2.0785   -1.6165         0
         0         0      -Inf
H∞ controller:
Ac = -2.0572
Bc = 1.5998
Cc = 2.0785
Dc = -1.6165
Closed-loop system:
Acl = 3×3
         0    1.0000         0
   -1.6165         0    2.0785
    1.5998         0   -2.0572
Bcl = 3×2
         0         0
    1.0000   -1.6165
         0    1.5998
Ccl = 2×3
         0    1.0000         0
   -1.6165         0    2.0785
Dcl = 2×2
         0         0
         0   -1.6165
H∞ norm:
hinf_norm = 1.6185
Closed-loop poles:
eig_Acl = 3×1 complex
  -1.0285 + 0.7471i
  -1.0285 - 0.7471i
  -0.0002 + 0.0000i
Verification of H∞ norm and pole location constraints successful!
```


Closed-Loop System Response
W
X1
X2
X3
Y1
Y2

SCREENSHOT:


MATLAB R2023b - academic use
HOME
PLOTS
APPS
LIVE EDITOR
INSERT
VIEW
Live Editor - C:\Users\csamak\OneDrive - Clemson University\Desktop\Convex-Optimization-for-Control-Design\HW04\Q3.mlx
Q1.mlx
Q2.mlx
Q3.mlx
disp('Verification of H∞ norm and pole location constraints failed!')
end
% Define the lumped (disturbance + control) input
t = 0:0.01:10;
w_amplitude = 0.1;
w_duration = 1;
w_pulse = w_amplitude * (t >= 0 & t <= w_duration);
u = zeros(size(w_pulse));
w = [w_pulse; w_pulse];
% Simulate the closed-loop system response
[y_cl, t_out, x_cl] = lsim(Scl, w', t);
% Plot the results
figure;
sgtitle('Closed-Loop System Response');
subplot(3, 1, 1);
plot(t, w(2, :));
legend('W');
subplot(3, 1, 2);
plot(t, x_cl(:, 1), t, x_cl(:, 2), t, x_cl(:, 3));
legend('X1', 'X2', 'X3');
subplot(3, 1, 3);
plot(t, y_cl(:, 1), t, y_cl(:, 2));
legend('Y1', 'Y2');
Closed-loop poles:
eig_Acl = 3×1 complex
-1.0285 + 0.7471i
-1.0285 - 0.7471i
-0.0002 + 0.0000i
Verification of H∞ norm and pole location constraints successful!
Closed-Loop System Response
Command Window