Problem 2

# **CODE:**

% PROBLEM 2

% Clear workspace

close all

clear

clc

% System matrices in affine form

A0 = [0 0 0 1 0 0; 0 0 0 0 1 0; 0 0 0 0 0 1; 0 0 0 0 0 0; 0 0 0 0 0 0; 0 0 0 0 0 0];

A1 = [0 0 0 0 0 0; 0 0 0 0 0 0; 0 0 0 0 0 0; -1 1 0 0 0 0; 1 -1 0 0 0 0; 0 0 0 0 0 0];

A2 = [0 0 0 0 0 0; 0 0 0 0 0 0; 0 0 0 0 0 0; 0 0 0 0 0 0; 0 -1 1 0 0 0; 0 1 -1 0 0 0];

B0 = [0; 0; 0; 1; 0; 0];

C0 = [0 0 1 0 0 0];

D0 = 0;

% Nominal system parameters

k1\_nominal = 1;

k2\_nominal = 1;

% Uncertain LTI system in affine form

S0 = ltisys(A0, B0, C0, D0, 1);

S1 = ltisys(A1, zeros(size(B0)), zeros(size(C0)), zeros(size(D0)), 0);

S2 = ltisys(A2, zeros(size(B0)), zeros(size(C0)), zeros(size(D0)), 0);

% Uncertainty bounds

alpha = 0.1;

LB = [k1\_nominal - alpha, k2\_nominal - alpha];

UB = [k1\_nominal + alpha, k2\_nominal + alpha];

% Parameter vector

P = pvec('box', [LB(1), UB(1); LB(2), UB(2)]);

% Affine system

affsys = psys(P, [S0, S1, S2]);

% Confirm that the open-loop uncertain system is not quadratically stable

result = quadstab(affsys)

if result < 0

disp('The open-loop uncertain system is quadratically stable.');

else

disp('The open-loop uncertain system is NOT quadratically stable.');

end

% Nominal system representation

A = A0 + A1 + A2;

B = B0;

% LQR control with unit weights

K = -lqr(A, B, eye(6), 1) % Q = I, R = 1

% Closed loop system considering state-feedback control law u = Kx

Acl = A + B\*K

% Uncertain closed-loop LTI system in affine form

S0cl = ltisys(Acl, zeros(size(B0)), C0, D0, 1);

S1cl = ltisys(A1, zeros(size(B0)), zeros(size(C0)), zeros(size(D0)), 0);

S2cl = ltisys(A2, zeros(size(B0)), zeros(size(C0)), zeros(size(D0)), 0);

% Uncertainty bounds

alpha = 0.1;

LB = [k1\_nominal - alpha, k2\_nominal - alpha];

UB = [k1\_nominal + alpha, k2\_nominal + alpha];

% Parameter vector

P = pvec('box', [LB(1), UB(1); LB(2), UB(2)]);

% Affine closed loop system

affsys\_cl = psys(P, [S0cl, S1cl, S2cl]);

% Determine if the closed-loop uncertain system is quadratically stable

result = quadstab(affsys\_cl)

if result < 0

disp('The closed-loop uncertain system is quadratically stable.');

else

disp('The closed-loop uncertain system is NOT quadratically stable.');

end

% Determine the maximum region of quadratic stability of the closed-loop uncertain system

expansion\_factor = quadstab(affsys\_cl, [1 0 0]) % Compute expansion factor

% Find the maximum α = α\_max such that the uncertain system is quadratically stable for all stiffness perturbations in the interval [nominal−α\_max, nominal+α\_max]

k1\_side = UB(1)-LB(1);

k2\_side = UB(2)-LB(2);

k1\_side\_scaled = expansion\_factor\*k1\_side;

k2\_side\_scaled = expansion\_factor\*k2\_side;

k1\_side\_diff = k1\_side\_scaled-k1\_side;

k2\_side\_diff = k2\_side\_scaled-k2\_side;

LB\_stable\_1 = LB(1)-(k1\_side\_diff/2);

UB\_stable\_1 = UB(1)+(k1\_side\_diff/2);

LB\_stable\_2 = LB(2)-(k2\_side\_diff/2);

UB\_stable\_2 = UB(2)+(k2\_side\_diff/2);

alpha\_max\_1 = UB\_stable\_1 - k1\_nominal % (or k1\_nominal - LB\_stable\_1) α\_max for k1

alpha\_max\_2 = UB\_stable\_2 - k2\_nominal % (or k2\_nominal - LB\_stable\_2) α\_max for k2

% Uncertainty bounds for which the closed-loop system is stable

LB\_stable = [k1\_nominal - alpha\_max\_1, k2\_nominal - alpha\_max\_2];

UB\_stable = [k1\_nominal + alpha\_max\_1, k2\_nominal + alpha\_max\_2];

% Parameter vector

P\_stable = pvec('box', [LB\_stable(1), UB\_stable(1); LB\_stable(2), UB\_stable(2)]);

% Affine stable closed loop system

affsys\_stable = psys(P\_stable, [S0cl, S1cl, S2cl]);

% Confirm that the closed-loop uncertain system is quadratically stable for all upper and lower bounds of the stiffness values

result = quadstab(affsys\_stable)

if result < 0

disp('The closed-loop uncertain system is quadratically stable for all upper and lower bounds of the stiffness values.');

else

disp('The closed-loop uncertain system is NOT quadratically stable for all upper and lower bounds of the stiffness values.');

end

# **OUTPUT:**

Solver for LMI feasibility problems L(x) < R(x)

This solver minimizes t subject to L(x) < R(x) + t\*I

The best value of t should be negative for feasibility

Iteration : Best value of t so far

1 0.108566

2 0.013617

3 0.012804

4 8.313415e-03

5 8.313415e-03

6 9.585323e-04

7 9.585323e-04

8 3.127972e-04

9 3.127972e-04

10 2.848959e-04

11 2.848959e-04

12 2.341144e-04

13 2.256522e-04

14 2.256522e-04

15 2.111485e-04

\* switching to QR

16 2.111485e-04

17 2.071661e-04

18 2.071661e-04

19 2.067091e-04

20 2.064719e-04

21 2.064719e-04

22 2.064440e-04

23 2.064440e-04

24 2.064329e-04

25 2.064329e-04

26 2.064329e-04

27 2.064329e-04

\*\*\* new lower bound: 2.064286e-04

Result: best value of t: 2.064329e-04

guaranteed absolute accuracy: 4.27e-09

f-radius saturation: 1.383% of R = 1.00e+08

Marginal infeasibility: these LMI constraints may be

feasible but are not strictly feasible

This system is not quadratically stable

result = 2.0643e-04

The open-loop uncertain system is NOT quadratically stable.

K = 1×6

-2.2106 0.9710 -0.4924 -2.3284 -1.3671 -1.3048

Acl = 6×6

0 0 0 1.0000 0 0

0 0 0 0 1.0000 0

0 0 0 0 0 1.0000

-3.2106 1.9710 -0.4924 -2.3284 -1.3671 -1.3048

1.0000 -2.0000 1.0000 0 0 0

0 1.0000 -1.0000 0 0 0

Solver for LMI feasibility problems L(x) < R(x)

This solver minimizes t subject to L(x) < R(x) + t\*I

The best value of t should be negative for feasibility

Iteration : Best value of t so far

1 0.051882

2 -0.127668

Result: best value of t: -0.127668

f-radius saturation: 0.000% of R = 1.00e+08

This system is quadratically stable

result = -0.1277

The closed-loop uncertain system is quadratically stable.

Solver for generalized eigenvalue minimization

Iterations : Best objective value so far

1

2

3

4

5

6

7

8 309.375000

9 146.808105

10 101.297593

11 69.895339

12 48.227784

13 33.277171

14 22.961248

15 2.669203

16 1.841750

17 1.270807

18 0.876857

19 0.491696

20 0.491696

21 0.491696

22 0.486779

23 0.481911

24 0.477092

25 0.472321

26 0.467598

27 0.462922

28 0.458293

29 0.458293

30 0.453710

31 0.449173

\*\*\* new lower bound: 0.313060

32 0.449173

33 0.436412

\*\*\* new lower bound: 0.317039

34 0.436412

35 0.434547

36 0.434547

\*\*\* new lower bound: 0.375793

37 0.434547

\*\*\* new lower bound: 0.405170

38 0.433892

\*\*\* new lower bound: 0.420215

Result: feasible solution

best value of t: 0.433892

guaranteed absolute accuracy: 1.37e-02

f-radius saturation: 0.000% of R = 1.00e+08

Termination due to SLOW PROGRESS:

the gen. eigenvalue t decreased by less than

1.000% during the last 5 iterations.

Quadratic stability established on 230.4722% of the

prescribed parameter box

expansion\_factor = 2.3047

alpha\_max\_1 = 0.2305

alpha\_max\_2 = 0.2305

Solver for LMI feasibility problems L(x) < R(x)

This solver minimizes t subject to L(x) < R(x) + t\*I

The best value of t should be negative for feasibility

Iteration : Best value of t so far

1 0.056929

2 0.017833

3 0.012808

4 0.012808

5 4.667073e-03

6 4.667073e-03

7 -5.181586e-04

Result: best value of t: -5.181586e-04

f-radius saturation: 0.000% of R = 1.00e+08

This system is quadratically stable

result = -5.1816e-04

The closed-loop uncertain system is quadratically stable for all upper and lower bounds of the stiffness values.

# **SCREENSHOT:**

