Problem 2

# **CODE:**

% PROBLEM 2

% Clear workspace

close all

clear

clc

% Add parser and solver to path

addpath(genpath('C:\Users\csamak\Downloads\MathWorks\Toolboxes\archives\required\YALMIP'))

addpath(genpath('C:\Users\csamak\Downloads\MathWorks\Toolboxes\archives\required\SeDuMi'))

for g = 3:1:5 % Gamma (γ)

disp('-----------------------------------')

if g == 3

disp('Standard LMI Method | CASE 1: γ = 3')

elseif g == 4

disp('Standard LMI Method | CASE 2: γ = 4')

elseif g == 5

disp('Standard LMI Method | CASE 3: γ = 5')

end

disp('-----------------------------------')

% 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];

% Define the LMI variables

X = sdpvar(2, 2);

Y = sdpvar(2, 2);

% Define the LMI constraints

C1 = null([Bp; By]')' \* [(Ap\*X + X\*Ap' + Dp\*Dp') (X\*Cp' + Dp\*Dy'); (Cp\*X + Dy\*Dp') (Dy\*Dy' - g^2\*eye(size(Dy)))] \* null([Bp; By]') <= 0;

C2 = null([Mp'; Dz']')' \* [(Y\*Ap + Ap'\*Y + Cp'\*Cp) (Y\*Dp + Cp'\*Dy); (Dp'\*Y + Dy'\*Cp) (Dy'\*Dy - g^2\*eye(size(Dy)))] \* null([Mp'; Dz']') <= 0;

C3 = [X g\*eye(size(Dy)); g\*eye(size(Dy)) Y] >= 0;

% Set up the objective

Objective = trace(X + Y);

% Define the solver settings (use an LMI solver like YALMIP with a solver of your choice)

options = sdpsettings('verbose', 1, 'solver', 'sedumi');

% Solve the LMI problem

solution = optimize([C1, C2, C3], Objective, options);

% 1. Obtain X and Y by solving above optimization problem

if solution.problem == 0

% Extract the optimal solutions

X = value(X);

Y = value(Y);

% Display the results

disp('X\*:');

disp(X);

disp('Y\*:');

disp(Y);

else

fprintf('LMI problem could not be solved.\n');

end

% 2. Confirm that rank([X γ\*I; γ\*I Y]) = 3 and compute terms Y12 and Y22

if rank([X g\*eye(size(Dy)); g\*eye(size(Dy)) Y]) == 3

disp('rank([X γ\*I; γ\*I Y]) = 3')

disp('Solving LMI problem resulted in reduced order controller with nc = 1')

elseif rank([X g\*eye(size(Dy)); g\*eye(size(Dy)) Y]) == 4

disp('rank([X γ\*I; γ\*I Y]) = 4')

disp('Solving LMI problem resulted in full order controller with nc = 2')

else

disp('rank([X γ\*I; γ\*I Y]) < 3')

disp('Solving LMI problem resulted in zero order controller with nc = 0')

end

[U, S, V] = svd(Y - g^2\*inv(X));

u1 = U(:,1); % Extract u1 so that u1 = np x nc

S1 = S(1,1); % Extract S1 so that S1 = nc x nc

disp('Extracted u1 & Σ1 so as to obtain reduced order controller with nc = 1')

Y12 = u1\*(S1^0.5)

Y22 = eye(size(Y12', 1), size(Y12, 2))

% 3. Define augmented matrix P & find controller by solving general matrix inequality

P = [Y Y12; Y12' Y22]

A = [Ap [0; 0]; [0 0] 0];

B = [Bp [0; 0]; 0 1];

C = [Cp [0; 0]];

D = [Dp; [0 0]];

E = [Dz; [0 0]];

F = Dy;

H = [By [0; 0]];

M = [Mp 0; [0 0] 1];

Gamma = [P\*B; [0 0]; [0 0]; H] % Γ

Lambda = [M E [0; 0] [0; 0]] % Λ

Q = [P\*A+A'\*P P\*D C'; D'\*P -g^2\*eye(2) F'; C F -eye(2)] % Q

G = basiclmi(Q, Gamma', Lambda); % Solve general matrix inequality

if size(G) ~= 0

disp('Controller matrix (G):')

disp(G)

fprintf('Ac = %f', value(G(2,2)))

fprintf('Bc = %f', value(G(2,1)))

fprintf('Cc = %f', value(G(1,2)))

fprintf('Dc = %f', value(G(1,1)))

end

end

% Clear workspace

close all

clear

clc

% Add parser and solver to path

addpath(genpath('C:\Users\csamak\Downloads\MathWorks\Toolboxes\archives\required\YALMIP'))

addpath(genpath('C:\Users\csamak\Downloads\MathWorks\Toolboxes\archives\required\SeDuMi'))

for g = 3:1:5 % Gamma (γ)

disp('---------------------------------------------')

if g == 3

disp('Alternating Projection Method | CASE 1: γ = 3')

elseif g == 4

disp('Alternating Projection Method | CASE 2: γ = 4')

elseif g == 5

disp('Alternating Projection Method | CASE 3: γ = 5')

end

disp('---------------------------------------------')

% 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];

% Define the LMI variables

X = sdpvar(2, 2);

Y = sdpvar(2, 2);

% Define the LMI constraints

C1 = null([Bp; By]')' \* [(Ap\*X + X\*Ap' + Dp\*Dp') (X\*Cp' + Dp\*Dy'); (Cp\*X + Dy\*Dp') (Dy\*Dy' - g^2\*eye(size(Dy)))] \* null([Bp; By]') <= 0;

C2 = null([Mp'; Dz']')' \* [(Y\*Ap + Ap'\*Y + Cp'\*Cp) (Y\*Dp + Cp'\*Dy); (Dp'\*Y + Dy'\*Cp) (Dy'\*Dy - g^2\*eye(size(Dy)))] \* null([Mp'; Dz']') <= 0;

C3 = [X g\*eye(size(Dy)); g\*eye(size(Dy)) Y] >= 0;

% Set up the objective

Objective = trace(X + Y);

% Define the solver settings (use an LMI solver like YALMIP with a solver of your choice)

options = sdpsettings('verbose', 1, 'solver', 'sedumi');

% Alternating projection method

max\_iterations = 100;

tolerance = 1e-22;

% Working in C1-C3 space

solution = optimize([C1, C2, C3], Objective, options);

if solution.problem == 0

X = value(X);

Y = value(Y);

end

R = [X, g\*eye(size(Dy)); g\*eye(size(Dy)), Y];

for iteration = 1:max\_iterations

[U, S, V] = svd(R);

if not(rank(R) == 3)

S(size(S,1), size(S,2)) = 0;

R\_prime = U\*S\*V';

else

R\_prime = R;

end

% Check for convergence between R and R\_prime

if norm(R - R\_prime) < tolerance

disp(['Converged at iteration ', num2str(iteration)]);

break;

end

% Projection onto C4 space & working in C4 space

R1 = sdpvar(size(R,1), size(R,2)); % Define the LMI variable

g\_sqr = sdpvar(1, 1); % Define the LMI variable

LMI\_R1 = [-g\_sqr\*eye(size(R)) R1-R\_prime; R1-R\_prime -eye(size(R))] <= 0; % Define the LMI constraint

Objective\_R1 = g\_sqr; % Set up the objective

solution\_R1 = optimize(LMI\_R1, Objective\_R1, options); % Solve the LMI problem

if solution\_R1.problem == 0

R1 = value(R1); % Extract the optimal solution

end

% Check for convergence between R1 and R\_prime

if norm(R1 - R\_prime) < tolerance

disp(['Converged at iteration ', num2str(iteration)]);

break;

end

% Projection onto C1-C3 space

R = R1;

end

% Display the results

disp('X\*:');

disp(X);

disp('Y\*:');

disp(Y);

% Display rank information

if rank(R\_prime) == 3

disp('Solving LMI problem resulted in a reduced-order controller with nc = 1');

else

disp('Rank constraint not satisfied.');

return; % Exit if the rank constraint is not satisfied

end

% Extract u1 & Σ1 to obtain reduced order controller with nc = 1

[U, S, V] = svd(Y - g^2\*inv(X));

u1 = U(:, 1);

S1 = S(1, 1);

Y12 = u1 \* (S1^0.5);

Y22 = eye(size(Y12', 1), size(Y12, 2));

% Define augmented matrix P for the controller

P = [Y, Y12; Y12', Y22];

% Obtain the controller matrix G

A = [Ap, [0; 0]; [0 0], 0];

B = [Bp, [0; 0]; 0, 1];

C = [Cp, [0; 0]];

D = [Dp; [0 0]];

E = [Dz; [0 0]];

F = Dy;

H = [By, [0; 0]];

M = [Mp, 0; [0 0], 1];

Gamma = [P \* B; [0 0]; [0 0]; H]; % Γ

Lambda = [M, E, [0; 0], [0; 0]]; % Λ

Q = [P \* A + A' \* P, P \* D, C'; D' \* P, -g^2 \* eye(2), F'; C, F, -eye(2)]; % Q

G = basiclmi(Q, Gamma', Lambda); % Solve general matrix inequality

if size(G) ~= 0

disp('Controller matrix (G):')

disp(G)

fprintf('Ac = %f', value(G(2,2)))

fprintf('Bc = %f', value(G(2,1)))

fprintf('Cc = %f', value(G(1,2)))

fprintf('Dc = %f', value(G(1,1)))

end

end

# **OUTPUT:**

-----------------------------------

Standard LMI Method | CASE 1: γ = 3

-----------------------------------

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 6, order n = 11, dim = 35, blocks = 4

nnz(A) = 20 + 0, nnz(ADA) = 36, nnz(L) = 21

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 2.71E+01 0.000

1 : -5.65E+00 9.11E+00 0.000 0.3364 0.9000 0.9000 1.41 1 1 3.2E+00

2 : -1.25E+01 2.87E+00 0.000 0.3153 0.9000 0.9000 0.91 1 1 9.6E-01

3 : -1.70E+01 6.35E-01 0.000 0.2209 0.9000 0.9000 0.66 1 1 2.5E-01

4 : -1.98E+01 5.13E-02 0.000 0.0808 0.9900 0.9900 0.78 1 1 2.1E-02

5 : -2.01E+01 1.90E-03 0.000 0.0369 0.9900 0.9900 0.99 1 1 8.3E-04

6 : -2.01E+01 1.03E-04 0.407 0.0545 0.9900 0.9900 1.00 1 1 4.5E-05

7 : -2.01E+01 7.99E-06 0.254 0.0773 0.9900 0.9900 1.00 1 1 3.5E-06

8 : -2.01E+01 7.20E-07 0.438 0.0902 0.9900 0.9900 1.00 1 1 3.2E-07

9 : -2.01E+01 1.51E-07 0.000 0.2100 0.9000 0.9000 1.00 2 2 6.7E-08

10 : -2.01E+01 1.86E-08 0.017 0.1233 0.9450 0.9450 1.00 2 2 8.3E-09

11 : -2.01E+01 1.61E-09 0.060 0.0865 0.9900 0.9900 1.00 2 2 7.2E-10

iter seconds digits c\*x b\*y

11 0.2 9.8 -2.0062574492e+01 -2.0062574496e+01

|Ax-b| = 4.8e-10, [Ay-c]\_+ = 9.5E-10, |x|= 1.1e+01, |y|= 1.1e+01

Detailed timing (sec)

Pre IPM Post

3.090E-01 5.140E-01 3.300E-02

Max-norms: ||b||=1, ||c|| = 9,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 6.52152.

X\*:

6.9382 -1.0201

-1.0201 3.9239

Y\*:

4.2969 -2.3744

-2.3744 4.9036

rank([X γ\*I; γ\*I Y]) = 4

Solving LMI problem resulted in full order controller with nc = 2

Extracted u1 & Σ1 so as to obtain reduced order controller with nc = 1

Y12 = 2×1

-1.7170

1.5871

Y22 = 1

P = 3×3

4.2969 -2.3744 -1.7170

-2.3744 4.9036 1.5871

-1.7170 1.5871 1.0000

Gamma = 7×2

-2.3744 -1.7170

4.9036 1.5871

1.5871 1.0000

0 0

0 0

0 0

1.0000 0

Lambda = 2×7

1 0 0 0 1 0 0

0 0 1 0 0 0 0

Q = 7×7

0 4.2969 0 -2.3744 0 0 0

4.2969 -4.7488 -1.7170 4.9036 0 1.0000 0

0 -1.7170 0 1.5871 0 0 0

-2.3744 4.9036 1.5871 -9.0000 0 0 0

0 0 0 0 -9.0000 0 0

0 1.0000 0 0 0 -1.0000 0

0 0 0 0 0 0 -1.0000

Warning in BASICLMI: the solvability conditions are not satisfied

-----------------------------------

Standard LMI Method | CASE 2: γ = 4

-----------------------------------

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 6, order n = 11, dim = 35, blocks = 4

nnz(A) = 20 + 0, nnz(ADA) = 36, nnz(L) = 21

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 4.74E+01 0.000

1 : -7.71E+00 1.53E+01 0.000 0.3239 0.9000 0.9000 1.46 1 1 2.8E+00

2 : -1.67E+01 4.33E+00 0.000 0.2820 0.9000 0.9000 1.04 1 1 8.1E-01

3 : -2.02E+01 8.97E-01 0.000 0.2072 0.9000 0.9000 0.83 1 1 1.8E-01

4 : -2.23E+01 4.34E-02 0.000 0.0484 0.9900 0.9900 0.85 1 1 8.9E-03

5 : -2.25E+01 2.43E-03 0.160 0.0560 0.9675 0.9675 1.00 1 1 5.0E-04

6 : -2.25E+01 2.68E-04 0.198 0.1101 0.9450 0.9450 1.00 1 1 5.5E-05

7 : -2.25E+01 2.09E-05 0.319 0.0779 0.9900 0.9900 1.00 1 1 4.3E-06

8 : -2.25E+01 9.96E-06 0.205 0.4772 0.9000 0.9000 1.00 1 1 2.1E-06

9 : -2.25E+01 2.48E-06 0.000 0.2490 0.9000 0.9000 1.00 1 1 5.2E-07

10 : -2.25E+01 2.29E-07 0.369 0.0922 0.9900 0.9900 1.00 2 2 4.9E-08

11 : -2.25E+01 8.19E-08 0.171 0.3580 0.9000 0.9000 1.00 2 2 1.8E-08

12 : -2.25E+01 6.92E-09 0.149 0.0845 0.9900 0.9900 1.00 2 2 1.5E-09

13 : -2.25E+01 6.74E-10 0.139 0.0974 0.9900 0.9900 1.00 2 2 1.5E-10

iter seconds digits c\*x b\*y

13 0.0 10.2 -2.2456285268e+01 -2.2456285269e+01

|Ax-b| = 8.7e-11, [Ay-c]\_+ = 2.7E-10, |x|= 7.9e+00, |y|= 1.2e+01

Detailed timing (sec)

Pre IPM Post

1.200E-02 3.201E-02 2.997E-03

Max-norms: ||b||=1, ||c|| = 16,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 4.35694.

X\*:

6.9145 -0.8214

-0.8214 4.9001

Y\*:

4.9551 -2.0759

-2.0759 5.6866

rank([X γ\*I; γ\*I Y]) = 4

Solving LMI problem resulted in full order controller with nc = 2

Extracted u1 & Σ1 so as to obtain reduced order controller with nc = 1

Y12 = 2×1

-1.6106

1.5346

Y22 = 1

P = 3×3

4.9551 -2.0759 -1.6106

-2.0759 5.6866 1.5346

-1.6106 1.5346 1.0000

Gamma = 7×2

-2.0759 -1.6106

5.6866 1.5346

1.5346 1.0000

0 0

0 0

0 0

1.0000 0

Lambda = 2×7

1 0 0 0 1 0 0

0 0 1 0 0 0 0

Q = 7×7

0 4.9551 0 -2.0759 0 0 0

4.9551 -4.1518 -1.6106 5.6866 0 1.0000 0

0 -1.6106 0 1.5346 0 0 0

-2.0759 5.6866 1.5346 -16.0000 0 0 0

0 0 0 0 -16.0000 0 0

0 1.0000 0 0 0 -1.0000 0

0 0 0 0 0 0 -1.0000

Controller matrix (G):

-2.6758 1.3083

8.5579 -5.0717

Ac = -5.071654

Bc = 8.557912

Cc = 1.308284

Dc = -2.675828

-----------------------------------

Standard LMI Method | CASE 3: γ = 5

-----------------------------------

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 6, order n = 11, dim = 35, blocks = 4

nnz(A) = 20 + 0, nnz(ADA) = 36, nnz(L) = 21

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 7.35E+01 0.000

1 : -1.03E+01 2.28E+01 0.000 0.3097 0.9000 0.9000 1.48 1 1 2.6E+00

2 : -2.10E+01 5.93E+00 0.000 0.2602 0.9000 0.9000 1.11 1 1 7.3E-01

3 : -2.36E+01 1.27E+00 0.000 0.2149 0.9000 0.9000 0.92 1 1 1.6E-01

4 : -2.57E+01 5.44E-02 0.000 0.0427 0.9900 0.9900 0.87 1 1 7.1E-03

5 : -2.58E+01 2.00E-03 0.024 0.0368 0.9900 0.9900 1.00 1 1 2.6E-04

6 : -2.58E+01 2.24E-04 0.206 0.1119 0.9450 0.9450 1.00 1 1 2.9E-05

7 : -2.58E+01 1.85E-05 0.480 0.0825 0.9900 0.9900 1.00 1 1 2.4E-06

8 : -2.58E+01 8.75E-06 0.164 0.4733 0.9000 0.9000 1.00 1 1 1.2E-06

9 : -2.58E+01 1.63E-06 0.000 0.1859 0.9000 0.9000 1.00 1 1 2.2E-07

10 : -2.58E+01 1.36E-07 0.471 0.0837 0.9900 0.9900 1.00 1 2 1.8E-08

11 : -2.58E+01 2.69E-08 0.000 0.1976 0.9000 0.9000 1.00 2 2 3.6E-09

12 : -2.58E+01 1.17E-09 0.359 0.0434 0.9900 0.9900 1.00 2 2 1.6E-10

iter seconds digits c\*x b\*y

12 0.0 10.1 -2.5756604760e+01 -2.5756604762e+01

|Ax-b| = 8.3e-11, [Ay-c]\_+ = 4.2E-10, |x|= 6.6e+00, |y|= 1.3e+01

Detailed timing (sec)

Pre IPM Post

3.300E-02 2.400E-02 2.002E-03

Max-norms: ||b||=1, ||c|| = 25,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 4.93398.

X\*:

7.4856 -0.7233

-0.7233 5.8495

Y\*:

5.8149 -1.9387

-1.9387 6.6066

rank([X γ\*I; γ\*I Y]) = 4

Solving LMI problem resulted in full order controller with nc = 2

Extracted u1 & Σ1 so as to obtain reduced order controller with nc = 1

Y12 = 2×1

-1.5604

1.5103

Y22 = 1

P = 3×3

5.8149 -1.9387 -1.5604

-1.9387 6.6066 1.5103

-1.5604 1.5103 1.0000

Gamma = 7×2

-1.9387 -1.5604

6.6066 1.5103

1.5103 1.0000

0 0

0 0

0 0

1.0000 0

Lambda = 2×7

1 0 0 0 1 0 0

0 0 1 0 0 0 0

Q = 7×7

0 5.8149 0 -1.9387 0 0 0

5.8149 -3.8774 -1.5604 6.6066 0 1.0000 0

0 -1.5604 0 1.5103 0 0 0

-1.9387 6.6066 1.5103 -25.0000 0 0 0

0 0 0 0 -25.0000 0 0

0 1.0000 0 0 0 -1.0000 0

0 0 0 0 0 0 -1.0000

Controller matrix (G):

-3.1340 1.6176

11.9541 -7.5012

Ac = -7.501234

Bc = 11.954090

Cc = 1.617626

Dc = -3.134028

---------------------------------------------

Alternating Projection Method | CASE 1: γ = 3

---------------------------------------------

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 6, order n = 11, dim = 35, blocks = 4

nnz(A) = 20 + 0, nnz(ADA) = 36, nnz(L) = 21

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 2.71E+01 0.000

1 : -5.65E+00 9.11E+00 0.000 0.3364 0.9000 0.9000 1.41 1 1 3.2E+00

2 : -1.25E+01 2.87E+00 0.000 0.3153 0.9000 0.9000 0.91 1 1 9.6E-01

3 : -1.70E+01 6.35E-01 0.000 0.2209 0.9000 0.9000 0.66 1 1 2.5E-01

4 : -1.98E+01 5.13E-02 0.000 0.0808 0.9900 0.9900 0.78 1 1 2.1E-02

5 : -2.01E+01 1.90E-03 0.000 0.0369 0.9900 0.9900 0.99 1 1 8.3E-04

6 : -2.01E+01 1.03E-04 0.407 0.0545 0.9900 0.9900 1.00 1 1 4.5E-05

7 : -2.01E+01 7.99E-06 0.254 0.0773 0.9900 0.9900 1.00 1 1 3.5E-06

8 : -2.01E+01 7.20E-07 0.438 0.0902 0.9900 0.9900 1.00 1 1 3.2E-07

9 : -2.01E+01 1.51E-07 0.000 0.2100 0.9000 0.9000 1.00 2 2 6.7E-08

10 : -2.01E+01 1.86E-08 0.017 0.1233 0.9450 0.9450 1.00 2 2 8.3E-09

11 : -2.01E+01 1.61E-09 0.060 0.0865 0.9900 0.9900 1.00 2 2 7.2E-10

iter seconds digits c\*x b\*y

11 0.0 9.8 -2.0062574492e+01 -2.0062574496e+01

|Ax-b| = 4.8e-10, [Ay-c]\_+ = 9.5E-10, |x|= 1.1e+01, |y|= 1.1e+01

Detailed timing (sec)

Pre IPM Post

1.100E-02 3.900E-02 2.997E-03

Max-norms: ||b||=1, ||c|| = 9,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 6.52152.

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 11, order n = 9, dim = 65, blocks = 2

nnz(A) = 20 + 0, nnz(ADA) = 121, nnz(L) = 66

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 8.16E+00 0.000

1 : -2.50E+00 3.13E+00 0.000 0.3836 0.9000 0.9000 2.27 1 1 2.7E+00

2 : 1.31E-01 8.96E-01 0.000 0.2861 0.9000 0.9000 3.42 1 1 3.8E-01

3 : 5.07E-03 1.99E-02 0.000 0.0222 0.9900 0.9900 1.25 1 1 1.2E-01

4 : 1.19E-07 3.91E-07 0.000 0.0000 1.0000 1.0000 1.01 1 1 2.3E-05

5 : 5.71E-14 1.85E-13 0.442 0.0000 1.0000 1.0000 1.00 1 1 1.1E-11

iter seconds digits c\*x b\*y

5 0.0 8.9 1.1776667857e-13 5.7097292825e-14

|Ax-b| = 3.0e-14, [Ay-c]\_+ = 5.7E-14, |x|= 5.0e-01, |y|= 1.1e+01

Detailed timing (sec)

Pre IPM Post

5.501E-02 2.299E-02 9.002E-03

Max-norms: ||b||=1, ||c|| = 1.387637e+01,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 1.

Converged at iteration 2

X\*:

6.9382 -1.0201

-1.0201 3.9239

Y\*:

4.2969 -2.3744

-2.3744 4.9036

Solving LMI problem resulted in a reduced-order controller with nc = 1

Warning in BASICLMI: the solvability conditions are not satisfied

---------------------------------------------

Alternating Projection Method | CASE 2: γ = 4

---------------------------------------------

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 6, order n = 11, dim = 35, blocks = 4

nnz(A) = 20 + 0, nnz(ADA) = 36, nnz(L) = 21

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 4.74E+01 0.000

1 : -7.71E+00 1.53E+01 0.000 0.3239 0.9000 0.9000 1.46 1 1 2.8E+00

2 : -1.67E+01 4.33E+00 0.000 0.2820 0.9000 0.9000 1.04 1 1 8.1E-01

3 : -2.02E+01 8.97E-01 0.000 0.2072 0.9000 0.9000 0.83 1 1 1.8E-01

4 : -2.23E+01 4.34E-02 0.000 0.0484 0.9900 0.9900 0.85 1 1 8.9E-03

5 : -2.25E+01 2.43E-03 0.160 0.0560 0.9675 0.9675 1.00 1 1 5.0E-04

6 : -2.25E+01 2.68E-04 0.198 0.1101 0.9450 0.9450 1.00 1 1 5.5E-05

7 : -2.25E+01 2.09E-05 0.319 0.0779 0.9900 0.9900 1.00 1 1 4.3E-06

8 : -2.25E+01 9.96E-06 0.205 0.4772 0.9000 0.9000 1.00 1 1 2.1E-06

9 : -2.25E+01 2.48E-06 0.000 0.2490 0.9000 0.9000 1.00 1 1 5.2E-07

10 : -2.25E+01 2.29E-07 0.369 0.0922 0.9900 0.9900 1.00 2 2 4.9E-08

11 : -2.25E+01 8.19E-08 0.171 0.3580 0.9000 0.9000 1.00 2 2 1.8E-08

12 : -2.25E+01 6.92E-09 0.149 0.0845 0.9900 0.9900 1.00 2 2 1.5E-09

13 : -2.25E+01 6.74E-10 0.139 0.0974 0.9900 0.9900 1.00 2 2 1.5E-10

iter seconds digits c\*x b\*y

13 0.0 10.2 -2.2456285268e+01 -2.2456285269e+01

|Ax-b| = 8.7e-11, [Ay-c]\_+ = 2.7E-10, |x|= 7.9e+00, |y|= 1.2e+01

Detailed timing (sec)

Pre IPM Post

2.100E-02 2.800E-02 2.002E-03

Max-norms: ||b||=1, ||c|| = 16,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 4.35694.

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 11, order n = 9, dim = 65, blocks = 2

nnz(A) = 20 + 0, nnz(ADA) = 121, nnz(L) = 66

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 8.16E+00 0.000

1 : -2.49E+00 3.13E+00 0.000 0.3838 0.9000 0.9000 2.28 1 1 2.7E+00

2 : 1.30E-01 8.94E-01 0.000 0.2857 0.9000 0.9000 3.42 1 1 3.8E-01

3 : 5.01E-03 1.97E-02 0.000 0.0221 0.9900 0.9900 1.25 1 1 1.2E-01

4 : 1.17E-07 3.85E-07 0.000 0.0000 1.0000 1.0000 1.01 1 1 2.3E-05

5 : 3.01E-14 9.08E-14 0.202 0.0000 1.0000 1.0000 1.00 1 1 5.5E-12

iter seconds digits c\*x b\*y

5 0.0 9.4 5.0899527759e-14 3.0144842876e-14

|Ax-b| = 1.7e-14, [Ay-c]\_+ = 3.0E-14, |x|= 5.0e-01, |y|= 1.3e+01

Detailed timing (sec)

Pre IPM Post

2.997E-03 1.001E-02 9.958E-04

Max-norms: ||b||=1, ||c|| = 1.382892e+01,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 1.

Converged at iteration 2

X\*:

6.9145 -0.8214

-0.8214 4.9001

Y\*:

4.9551 -2.0759

-2.0759 5.6866

Solving LMI problem resulted in a reduced-order controller with nc = 1

Controller matrix (G):

-2.6758 1.3083

8.5579 -5.0717

Ac = -5.071654

Bc = 8.557912

Cc = 1.308284

Dc = -2.675828

---------------------------------------------

Alternating Projection Method | CASE 3: γ = 5

---------------------------------------------

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 6, order n = 11, dim = 35, blocks = 4

nnz(A) = 20 + 0, nnz(ADA) = 36, nnz(L) = 21

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 7.35E+01 0.000

1 : -1.03E+01 2.28E+01 0.000 0.3097 0.9000 0.9000 1.48 1 1 2.6E+00

2 : -2.10E+01 5.93E+00 0.000 0.2602 0.9000 0.9000 1.11 1 1 7.3E-01

3 : -2.36E+01 1.27E+00 0.000 0.2149 0.9000 0.9000 0.92 1 1 1.6E-01

4 : -2.57E+01 5.44E-02 0.000 0.0427 0.9900 0.9900 0.87 1 1 7.1E-03

5 : -2.58E+01 2.00E-03 0.024 0.0368 0.9900 0.9900 1.00 1 1 2.6E-04

6 : -2.58E+01 2.24E-04 0.206 0.1119 0.9450 0.9450 1.00 1 1 2.9E-05

7 : -2.58E+01 1.85E-05 0.480 0.0825 0.9900 0.9900 1.00 1 1 2.4E-06

8 : -2.58E+01 8.75E-06 0.164 0.4733 0.9000 0.9000 1.00 1 1 1.2E-06

9 : -2.58E+01 1.63E-06 0.000 0.1859 0.9000 0.9000 1.00 1 1 2.2E-07

10 : -2.58E+01 1.36E-07 0.471 0.0837 0.9900 0.9900 1.00 1 2 1.8E-08

11 : -2.58E+01 2.69E-08 0.000 0.1976 0.9000 0.9000 1.00 2 2 3.6E-09

12 : -2.58E+01 1.17E-09 0.359 0.0434 0.9900 0.9900 1.00 2 2 1.6E-10

iter seconds digits c\*x b\*y

12 0.0 10.1 -2.5756604760e+01 -2.5756604762e+01

|Ax-b| = 8.3e-11, [Ay-c]\_+ = 4.2E-10, |x|= 6.6e+00, |y|= 1.3e+01

Detailed timing (sec)

Pre IPM Post

2.002E-03 2.700E-02 2.002E-03

Max-norms: ||b||=1, ||c|| = 25,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 4.93398.

SeDuMi 1.3 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, theta = 0.250, beta = 0.500

eqs m = 11, order n = 9, dim = 65, blocks = 2

nnz(A) = 20 + 0, nnz(ADA) = 121, nnz(L) = 66

it : b\*y gap delta rate t/tP\* t/tD\* feas cg cg prec

0 : 8.33E+00 0.000

1 : -2.75E+00 3.18E+00 0.000 0.3812 0.9000 0.9000 2.25 1 1 2.7E+00

2 : 1.48E-01 9.32E-01 0.000 0.2935 0.9000 0.9000 3.43 1 1 3.9E-01

3 : 6.51E-03 2.33E-02 0.000 0.0250 0.9900 0.9900 1.28 1 1 1.2E-01

4 : 1.86E-07 5.57E-07 0.000 0.0000 1.0000 1.0000 1.01 1 1 3.1E-05

5 : 4.75E-14 1.27E-13 0.301 0.0000 1.0000 1.0000 1.00 1 1 6.9E-12

iter seconds digits c\*x b\*y

5 0.0 9.2 7.8883843350e-14 4.7484765648e-14

|Ax-b| = 2.1e-14, [Ay-c]\_+ = 4.8E-14, |x|= 5.0e-01, |y|= 1.5e+01

Detailed timing (sec)

Pre IPM Post

2.299E-02 9.998E-03 1.006E-03

Max-norms: ||b||=1, ||c|| = 1.497125e+01,

Cholesky |add|=0, |skip| = 0, ||L.L|| = 1.

Converged at iteration 2

X\*:

7.4856 -0.7233

-0.7233 5.8495

Y\*:

5.8149 -1.9387

-1.9387 6.6066

Solving LMI problem resulted in a reduced-order controller with nc = 1

Controller matrix (G):

-3.1340 1.6176

11.9541 -7.5012

Ac = -7.501234

Bc = 11.954090

Cc = 1.617626

Dc = -3.134028

# **SCREENSHOT:**

