12 LQG with Multiple Objectives

Problem:
Given the LTI system

$$\begin{aligned}
\dot{x} &= Ax + B_u u + B_w w, \quad x(0) = 0, \quad x \in \mathbb{R}^n \\
y &= C_y x + D_{yw} w, \\
z_1 &= C_{z1} x + D_{zw1} w + D_{zu1} u, \\
z_2 &= C_{z2} x + D_{zw2} w + D_{zu2} u, \\
&\vdots \\
z_N &= C_{zN} x + D_{zwN} w + D_{zuN} u,
\end{aligned}$$

compute an output feedback controller

$$\begin{aligned}
\dot{x}_c &= A_c x_c + B_c y, \qquad x_c(0) = 0, \quad x_c \in \mathbb{R}^{n_c} \\
u &= C_c x_c + D_c y,
\end{aligned}$$

so that the closed loop system is stable and all of the following constraints

$$J_1 \leq \mu_1, \qquad J_2 \leq \mu_2, \qquad \cdots \qquad J_N \leq \mu_N,$$

be satisfied, where μ_i are given and

$$J_i := \lim_{t \to \infty} E\left[z_i(t)^T z_i(t) \right],$$

for $i = 1, \ldots, N$.
Assumption:

1. $w(t)$ is a Gaussian zero mean white noise with variance $W \succ 0$,

Remark:
This is one of the few multiobjective problems that we know how to solve.

12.1 The Closed Loop System

The closed loop system is

$$\begin{aligned}
\begin{pmatrix} \dot{x} \\ \dot{x}_c \end{pmatrix} &= \begin{bmatrix} A + B_u D_c C_y & B_u C_c \\ B_c C_y & A_c \end{bmatrix} \begin{pmatrix} x \\ x_c \end{pmatrix} + \begin{bmatrix} B_w + B_u D_c D_{yw} \\ B_c D_{yw} \end{bmatrix} w, \\
z_1 &= \begin{bmatrix} C_{z1} + D_{zu1} D_c C_y & D_{zu1} C_c \end{bmatrix} \begin{pmatrix} x \\ x_c \end{pmatrix} + \begin{bmatrix} D_{zw1} + D_{zu1} D_c D_{yw} \end{bmatrix} w, \\
z_2 &= \begin{bmatrix} C_{z2} + D_{zu2} D_c C_y & D_{zu2} C_c \end{bmatrix} \begin{pmatrix} x \\ x_c \end{pmatrix} + \begin{bmatrix} D_{zw2} + D_{zu2} D_c D_{yw} \end{bmatrix} w, \\
&\vdots \\
z_N &= \begin{bmatrix} C_{zN} + D_{zuN} D_c C_y & D_{zuN} C_c \end{bmatrix} \begin{pmatrix} x \\ x_c \end{pmatrix} + \begin{bmatrix} D_{zwN} + D_{zuN} D_c D_{yw} \end{bmatrix} w
\end{aligned}$$

or in compact notation

$$\tilde{x} := \begin{pmatrix} x \\ x_c \end{pmatrix}$$

so that the closed loop is of the form

$$\begin{pmatrix} \dot{\tilde{x}} \\ z_1 \\ z_2 \\ \vdots \\ z_N \end{pmatrix} = \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C}_1 & \tilde{D}_1 \\ \tilde{C}_2 & \tilde{D}_2 \\ \vdots & \vdots \\ \tilde{C}_N & \tilde{D}_N \end{array}\right] \begin{pmatrix} \tilde{x} \\ w \end{pmatrix}$$

where $\tilde{A}$ and $\tilde{B}$ are as before and $\tilde{C}_i$ and $\tilde{D}_i$ are obtained from the above formulas.

12.2 The Closed Loop Constraints

The key to solving this problem is writing the constraints

$$J_i = \lim_{t\to\infty} E\left[z_i(t)^T z_i(t)\right].$$

in terms of the Controllability Gramian. It does not work with the Observability Gramian. That is,

$$J_i \le \operatorname{trace}\left(\tilde{C}_i \tilde{Y} \tilde{C}_i^T\right),$$

where $\tilde{Y}$ satisfies the Lyapunov inequality

$$\tilde{A}\tilde{Y} + \tilde{Y}\tilde{A}^T + \tilde{B}_w W \tilde{B}_w^T \prec 0.$$

Recall that $\tilde{D}_i$ must be zero! Introducing the matrices Z_i, $i = 1, \ldots, N$ as in

$$Z_i \succ \tilde{C}_i \tilde{Y} \tilde{C}_i^T$$

so that

$$\operatorname{trace}(Z_i) > \operatorname{trace}\left(\tilde{C}_i \tilde{Y} \tilde{C}_i^T\right) \ge J_i.$$

we can proceed with Method I as with the standard LQG problem.

12.3 Method I (congruence + change-of-variables)

Apply Schur complement to transform the analysis conditions into the more convenient form

$$\begin{bmatrix} \tilde{A}\tilde{Y} + \tilde{Y}\tilde{A}^T & \tilde{B} \\ \tilde{B}^T & -W^{-1} \end{bmatrix} \prec 0, \qquad \begin{bmatrix} Z_i & \tilde{C}_i\tilde{Y} \\ \tilde{Y}\tilde{C}_i^T & \tilde{Y} \end{bmatrix} \succ 0, \qquad \tilde{D}_i = 0$$

and apply congruence transformations of the form

$$\begin{bmatrix} \tilde{T}^T & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \tilde{A}\tilde{Y} + \tilde{Y}\tilde{A}^T & \tilde{B} \\ \tilde{B}^T & -W^{-1} \end{bmatrix} \begin{bmatrix} \tilde{T} & 0 \\ 0 & I \end{bmatrix} = \begin{bmatrix} \tilde{T}^T\tilde{A}\tilde{Y}\tilde{T} + \tilde{T}^T\tilde{Y}\tilde{A}^T\tilde{T} & \tilde{T}^T\tilde{B} \\ \tilde{B}\tilde{T} & -W^{-1} \end{bmatrix} \prec 0,$$

$$\begin{bmatrix} I & 0 \\ 0 & \tilde{T}^T \end{bmatrix} \begin{bmatrix} Z_i & \tilde{C}_i\tilde{Y} \\ \tilde{Y}\tilde{C}_i^T & \tilde{Y} \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & \tilde{T} \end{bmatrix} = \begin{bmatrix} Z_i & \tilde{C}_i\tilde{Y}\tilde{T} \\ \tilde{T}^T\tilde{Y}\tilde{C}_i^T & \tilde{T}^T\tilde{Y}\tilde{T} \end{bmatrix} \succ 0,$$

where $\tilde{T}$ is a matrix yet to be determined. As before, LMIs are obtained if we are able to determine $\tilde{T}$ such that the matrices

$$\tilde{T}^T\tilde{A}\tilde{Y}\tilde{T}, \qquad \tilde{C}_i\tilde{Y}\tilde{T}, \qquad \tilde{T}^T\tilde{B},$$

are affine functions of the design variables after a change of variables.

12.4 The Congruence Transformation

Define the partitions associated with $\tilde{Y}$ and its inverse

$$\tilde{Y} := \begin{bmatrix} X & U^T \\ U & \hat{X} \end{bmatrix}, \qquad \tilde{Y}^{-1} := \begin{bmatrix} Y & V \\ V^T & \hat{Y} \end{bmatrix}.$$

Now define the transformation matrix $\tilde{T}$ as

$$\tilde{T} := \begin{bmatrix} I & Y \\ 0 & V^T \end{bmatrix}.$$

12.5 The Change-of-Variables

The change-of-variables

$$\begin{aligned}
&R := D_c, \\
&L := C_c U + D_c C_y X, \\
&F := V B_c + Y B_u D_c, \\
&Q := V A_c U + Y A X + V B_c C_y X + Y B_u C_c U + Y B_u D_c C_y X = (\star)
\end{aligned}$$

transform

$$\begin{aligned}
\tilde{T}^T \tilde{A} \tilde{Y} \tilde{T} &= \begin{bmatrix} AX + B_u (C_c U + D_c C_y X) & A + B_u D_c C_y \\ (\star) & YA + (V B_c + Y B_u D_c) C_y \end{bmatrix}, \\
\tilde{C}_i \tilde{Y} \tilde{T} &= \begin{bmatrix} C_{zi} X + D_{zui} (C_c U + D_c C_y X) & C_{zi} + D_{zui} D_c C_y \end{bmatrix}, \\
\tilde{T}^T \tilde{B} &= \begin{bmatrix} B_w + B_u D_c D_{yw} \\ Y B_w + (V B_c + Y B_u D_c) D_{yw} \end{bmatrix}, \\
\tilde{T}^T \tilde{Y} \tilde{T} &= \begin{bmatrix} X & I \\ I & Y \end{bmatrix},
\end{aligned}$$

into

$$\begin{aligned}
\tilde{T}^T \tilde{A} \tilde{X} \tilde{T} &= \begin{bmatrix} AX + B_u L & A + B_u R C_y \\ Q & YA + F C_y \end{bmatrix}, \\
\tilde{C}_i \tilde{X} \tilde{T} &= \begin{bmatrix} C_{zi} X + D_{zui} L & C_{zi} + D_{zui} R C_y \end{bmatrix}, \\
\tilde{T}^T \tilde{B} &= \begin{bmatrix} B_w + B_u R D_{yw} \\ Y B_w + F D_{yw} \end{bmatrix}, \\
\tilde{T}^T \tilde{X} \tilde{T} &= \begin{bmatrix} X & I \\ I & Y \end{bmatrix},
\end{aligned}$$

which are affine in the *synthesis variables* Q, L, F, R.

12.6 Summary: LQG with Multiple Objectives

The full order dynamic output feedback controller

$$\begin{aligned} \dot{x}_c &= A_c x_c + B_c y, \qquad x_c(0) = 0, \quad x_c \in \mathbb{R}^n \\ u &= C_c x_c + D_c y, \end{aligned}$$

computed as

$$\begin{bmatrix} A_c & B_c \\ C_c & D_c \end{bmatrix} = \begin{bmatrix} V^{-1} & -V^{-1} Y B_u \\ 0 & I \end{bmatrix} \begin{bmatrix} Q - YAX & F \\ L & R \end{bmatrix} \begin{bmatrix} U^{-1} & 0 \\ -C_y X U^{-1} & I \end{bmatrix},$$

where $X, Y \in \mathbb{S}^n$, $L \in \mathbb{R}^{m \times n}$, $F \in \mathbb{R}^{n \times q}$, $Q \in \mathbb{R}^{n \times n}$, $R \in \mathbb{R}^{m \times q}$ and $Z_i \in \mathbb{S}^r$ satisfying the LMI

$$\begin{gathered} \operatorname{trace}(Z_i) \leq \mu_i, \qquad D_{zwi} + D_{zui} R D_{yw} = 0, \\ \begin{bmatrix} Z_i & C_{zi} X + D_{zui} L & C_{zi} + D_{zui} R C_y \\ (\bullet)^T & X & I \\ (\bullet)^T & (\bullet)^T & Y \end{bmatrix} \succ 0, \qquad i = 1, \ldots, N, \\ \begin{bmatrix} \left\{ \begin{array}{c} AX + XA^T + \\ B_u L + L^T B_u^T \end{array} \right\} & A + B_u R C_y + Q^T & B_w + B_u R D_{yw} \\ (\bullet)^T & \left\{ \begin{array}{c} A^T Y + YA + \\ F C_y + C_y^T F^T \end{array} \right\} & Y B_w + F D_{yw} \\ (\bullet)^T & (\bullet)^T & -W^{-1} \end{bmatrix} \prec 0 \end{gathered}$$

and $U, V \in \mathbb{R}^{n \times n}$ are such that

$$YX + VU = I$$

if and only if the closed loop system is stable and all constraints

$$J_i = \lim_{t \to \infty} E\left[z_i(t)^T z_i(t) \right] \leq \mu_i,$$

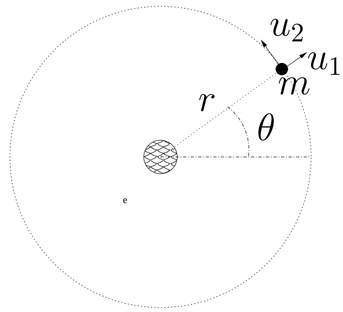
are satisfied for the LTI system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_u u(t) + B_w w(t), \quad x(0) = 0, \quad x \in \mathbb{R}^n \\ z_1 &= C_{z1} x + D_{zw1} w + D_{zu1} u, \\ z_2 &= C_{z2} x + D_{zw2} w + D_{zu2} u, \\ &\vdots \\ z_N &= C_{zN} x + D_{zwN} w + D_{zuN} u, \end{aligned}$$

under the assumption

1. $w(t)$ is a Gaussian zero mean white noise with variance $W \succ 0$.

12.7 Example: controlling a satellite in circular orbit



Satellite of mass m with thrust in the radial direction u_1 and in the tangential direction u_2. Continuing...

$$m(\ddot{r} - r\dot{\theta}^2) = u_1 - \frac{km}{r^2} + w_1,$$
$$m(2\dot{r}\dot{\theta} + r\ddot{\theta}) = u_2 + w_2,$$

where w_1 and w_2 are independent white noise disturbances with variances δ_1 and δ_2.

As before, putting in state space and linearize

$$\dot{x} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 3\,\bar{\omega}^2 & 0 & 0 & 2\,\bar{r}\bar{\omega} \\ 0 & 0 & -2\,\bar{\omega}/\bar{r} & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1/m & 0 \\ 0 & 1/(m\bar{r}) \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1/m & 0 \\ 0 & 1/(m\bar{r}) \end{bmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

Problem: Given

$$m = 100 \text{ kg}, \qquad \bar{r} = R + 300 \text{ km}, \qquad \bar{k} = GM$$

where $G \approx 6.673 \times 10^{-11}$ N m^2/kg^2 is the universal gravitational constant, and $M \approx 5.98 \times 10^{24}$ kg and $R \approx 6.37 \times 10^3$ km are the mass and radius of the earth. If the variances $\delta_1 = \delta_2 = 0.1N$ find a solution to the LQR control problem using u_1 and u_2 where you spend only 20 % of the minimum possible control energy.

The problem we worked out in the notes is the more complicated output feedback problem. What we do here is first determine the minimum possible control energy then solve a multiobjective control where the control energy enters as a constraint. Pay attention to the details...

```
% MAE 280 B - Linear Control Design
% Mauricio de Oliveira
%
% Multiobjective LQR Control
%
m = 100;                % 100 kg
r = 300E3;              % 300 km
R = 6.37E6;             % 6.37 10^3 km
G = 6.673E-11;          % 6.673 N m^2/kg^2
M = 5.98E24;            % 5.98 10^24 kg
k = G * M;              % gravitational force constant
w = sqrt(k/((R+r)^3)); % angular velocity (rad/s)
v = w * (R + r);        % "ground" velocity (m/s)

% linearized system matrices
A = [0 0 1 0; 0 0 0 1; 3*w^2 0 0 2*(r+R)*w; 0 0 -2*w/(r+R) 0];
Bu = [0 0; 0 0; 1/m 0; 0 1/(m*r)];
Bw = [0 0; 0 0; 1/m 0; 0 1/(m*r)];

% noise variances
W = 0.1 * eye(2)
W =
   1.0000e-01            0
            0   1.0000e-01

% scale
T = diag([1 r 1 r])
T =
           1           0           0           0
           0      300000           0           0
           0           0           1           0
           0           0           0      300000

% similarity transformation
At = T * A / T
At =
            0            0   1.0000e+00            0
            0            0            0   1.0000e+00
   4.0343e-06            0            0   5.1565e-02
            0            0  -1.0432e-04            0
But = T * Bu
But =
            0            0
            0            0
   1.0000e-02            0
            0   1.0000e-02
Bwt = T * Bw
```

```
Bwt =
             0            0
             0            0
   1.0000e-02             0
             0   1.0000e-02

% minimum energy state feedback control (using u1 and u2)
n = size(At,1);
m = size(But,2);
Dzu = eye(m);

% declare variables
X = sdpvar(n,n,'symmetric');
Z = sdpvar(m,m,'symmetric');
L = sdpvar(m,n);

% declare LMIs
LMI1 = At*X+X*At'+But*L+L'*But'+Bwt*W*Bwt';
LMI2 = [Z Dzu*L; L'*Dzu' X];

LMI = set(LMI1 < 0) + set(LMI2 > 0);
options = sdpsettings('solver','sedumi');
solution =solvesdp(LMI,trace(Z),options)
SeDuMi 1.1R3 by AdvOL, 2006 and Jos F. Sturm, 1998-2003.
Alg = 2: xz-corrector, theta = 0.250, beta = 0.500
eqs m = 21, order n = 11, dim = 53, blocks = 3
nnz(A) = 49 + 0, nnz(ADA) = 441, nnz(L) = 231
 it :     b*y       gap    delta  rate   t/tP*  t/tD*   feas cg cg  prec
  0 :            2.91E+00 0.000
  1 :  -4.77E-02 2.13E-01 0.000 0.0731 0.9900 0.9900   1.67  1  1  9.9E-01
  2 :  -1.67E-03 1.18E-02 0.000 0.0554 0.9900 0.9900   1.07  1  1  6.7E-01
  3 :  -1.65E-04 7.85E-04 0.000 0.0665 0.9900 0.9900   1.01  1  1  7.7E-02
  4 :  -7.43E-05 1.57E-04 0.089 0.1997 0.9000 0.9000   0.94  1  1  1.0E-02
  5 :  -1.02E-04 5.40E-05 0.054 0.3444 0.9000 0.9000   0.45  1  1  1.2E-04
  6 :  -2.33E-04 2.07E-05 0.000 0.3837 0.9000 0.9000  -0.12  1  1  1.0E-04
  7 :  -7.40E-04 4.70E-06 0.000 0.2271 0.9000 0.9000  -0.38  1  1  7.3E-05
  8 :  -1.72E-03 1.39E-06 0.000 0.2957 0.9000 0.9000  -0.39  1  1  4.7E-05
  9 :  -1.85E-03 4.95E-07 0.000 0.3562 0.9000 0.9000   0.48  1  1  1.8E-05
 10 :  -1.69E-03 1.61E-07 0.000 0.3246 0.9000 0.9000   0.56  1  1  8.4E-06
 11 :  -1.55E-03 5.36E-08 0.000 0.3333 0.9000 0.9000   0.46  2  2  4.1E-06
 12 :  -1.41E-03 1.93E-08 0.000 0.3609 0.9000 0.9000   0.44  3  3  2.1E-06
 13 :  -1.29E-03 8.39E-09 0.000 0.4339 0.9000 0.9000   0.32  3  3  1.5E-06
 14 :  -1.18E-03 3.22E-09 0.000 0.3841 0.9000 0.9000   0.51  4  4  7.2E-07
 15 :  -1.07E-03 1.26E-09 0.000 0.3905 0.9000 0.9000   0.25  4  4  5.0E-07
 16 :  -9.73E-04 4.44E-10 0.000 0.3526 0.9000 0.9000   0.44  5  3  2.4E-07
 17 :  -8.79E-04 1.54E-10 0.000 0.3466 0.9000 0.9000   0.23  5  5  1.5E-07
 18 :  -7.98E-04 5.40E-11 0.000 0.3510 0.9000 0.9000   0.38  5  5  7.6E-08
 19 :  -7.21E-04 1.89E-11 0.000 0.3493 0.9000 0.9000   0.18  5  5  5.0E-08
 20 :  -6.52E-04 6.57E-12 0.000 0.3480 0.9000 0.9000   0.38  5  5  2.5E-08
 21 :  -5.86E-04 2.26E-12 0.000 0.3439 0.9000 0.9000   0.20  5  5  1.6E-08
 22 :  -5.29E-04 7.94E-13 0.000 0.3518 0.9000 0.9000   0.38  5  5  7.9E-09
 23 :  -4.75E-04 2.79E-13 0.000 0.3508 0.9000 0.9000   0.18  5  5  5.2E-09
 24 :  -4.26E-04 9.77E-14 0.000 0.3506 0.9000 0.9000   0.39  5  5  2.5E-09
 25 :  -3.80E-04 3.36E-14 0.000 0.3439 0.9000 0.9000   0.19  5  5  1.6E-09
```

```
 26 :  -3.39E-04 1.18E-14 0.000 0.3513 0.9000 0.9000   0.38  5  5  8.1E-10

iter seconds digits       c*x               b*y
 26      0.2   Inf -3.4941546207e-04 -3.3904194131e-04
|Ax-b| =   9.6e-10, [Ay-c]_+ =   0.0E+00, |x|=  3.5e+01, |y|=  3.1e+05

Detailed timing (sec)
   Pre          IPM          Post
0.000E+00    2.300E-01    0.000E+00
Max-norms: ||b||=1, ||c|| = 1.000000e-05,
Cholesky |add|=3, |skip| = 0, ||L.L|| = 500000.
solution =
    yalmiptime: 6.1165e-02
    solvertime: 2.3275e-01
          info: 'Numerical problems (SeDuMi-1.1)'
       problem: 4
        dimacs: [4.8030e-10 0 0 0 -1.0366e-05 8.9383e-05]

% compute gain
K = double(L) / double(X)
K =
  -2.5116e-06   1.5472e-06  -8.7698e-04  -1.2133e-02
  -4.8114e-05   1.2277e-05  -1.4052e-02  -3.6166e-01
Acl = At + But * K;
eig(Acl)
ans =
  -8.2994e-04 + 2.1087e-03i
  -8.2994e-04 - 2.1087e-03i
  -1.9147e-03
  -5.0818e-05

% look at cost
minenergy = trace(double(Z))
minenergy =
   3.3904e-04

% try to solve Riccati
% K = -lqr(At, But, zeros(n), Dzu'*Dzu)
% solve optimal performance
Cz = eye(n);

X = sdpvar(n,n,'symmetric');
L = sdpvar(m,n);
Z = sdpvar(n,n,'symmetric');

LMI1 = At*X+X*At'+But*L+L'*But'+Bwt*W*Bwt';
LMI2 = [Z Cz*X; X*Cz' X];

LMI = set(LMI1 < 0) + set(LMI2 > 0);
options = sdpsettings('solver','sedumi');
solution =solvesdp(LMI,trace(Z),options)
SeDuMi 1.1R3 by AdvOL, 2006 and Jos F. Sturm, 1998-2003.
Alg = 2: xz-corrector, theta = 0.250, beta = 0.500
eqs m = 28, order n = 13, dim = 81, blocks = 3
```

```
nnz(A) = 64 + 0, nnz(ADA) = 608, nnz(L) = 318
 it :     b*y       gap    delta  rate   t/tP*  t/tD*   feas cg cg  prec
  0 :            2.46E+00 0.000
  1 :  -8.49E-02 5.11E-01 0.000 0.2076 0.9000 0.9000   2.00  1  1  1.3E+00
  2 :  -1.20E-04 2.30E-02 0.000 0.0450 0.9900 0.9900   1.59  1  1  7.9E-02
  3 :   2.33E-07 1.64E-06 0.481 0.0001 1.0000 1.0000   1.02  1  1  1.8E-06
  4 :  -1.94E-09 9.17E-09 0.000 0.0056 0.9990 0.9990   0.99  1  1  8.7E-07
  5 :   9.80E-16 8.28E-15 0.012 0.0000 1.0000 1.0000   1.00  1  1  6.7E-13

iter seconds digits       c*x               b*y
  5      0.0   Inf -4.2933868246e-20  9.7954340739e-16
|Ax-b| =   8.9e-15, [Ay-c]_+ =   1.5E-15, |x|=  2.8e+00, |y|=  3.1e+01

Detailed timing (sec)
   Pre          IPM          Post
0.000E+00    3.000E-02    0.000E+00
Max-norms: ||b||=1, ||c|| = 1.000000e-05,
Cholesky |add|=0, |skip| = 1, ||L.L|| = 100.
solution =
    yalmiptime: 3.3745e-02
    solvertime: 3.6940e-02
          info: 'No problems detected (SeDuMi-1.1)'
       problem: 0
        dimacs: [4.4421e-15 0 0 1.4638e-15 -9.7959e-16 9.2496e-16]

% compute gain
K = double(L) / double(X)
K =
   7.4130e+15  -1.5678e+14  -2.7972e+16   1.3350e+15
  -7.2866e+14   1.2807e+16   1.3350e+15  -3.1282e+16
Acl = At + But * K;
eig(Acl)
ans =
  -2.7501e+14
  -3.1753e+14
   2.7180e-01
   4.1383e-01

% look at cost
minperf = trace(double(Z))
minperf =
  -9.7954e-16

% solve optimal performance
Cz = eye(n);

X = sdpvar(n,n,'symmetric');
L = sdpvar(m,n);
Z = sdpvar(n,n,'symmetric');

LMI1 = At*X+X*At'+But*L+L'*But'+Bwt*W*Bwt'+1e-5*eye(n);
LMI2 = [Z Cz*X; X*Cz' X];

LMI = set(LMI1 < 0) + set(LMI2 > 0);
```

```
options = sdpsettings('solver','sedumi');
solution =solvesdp(LMI,trace(Z),options)
SeDuMi 1.1R3 by AdvOL, 2006 and Jos F. Sturm, 1998-2003.
Alg = 2: xz-corrector, theta = 0.250, beta = 0.500
eqs m = 28, order n = 13, dim = 81, blocks = 3
nnz(A) = 64 + 0, nnz(ADA) = 608, nnz(L) = 318
 it :     b*y       gap    delta  rate   t/tP*  t/tD*   feas cg cg  prec
  0 :            2.46E+00 0.000
  1 :  -8.50E-02 5.11E-01 0.000 0.2076 0.9000 0.9000   2.00  1  1  1.3E+00
  2 :  -1.28E-04 2.30E-02 0.000 0.0450 0.9900 0.9900   1.59  1  1  7.9E-02
  3 :  -7.31E-06 1.42E-05 0.000 0.0006 0.9999 0.9999   1.02  1  1  1.5E-05
  4 :  -1.80E-05 3.46E-06 0.000 0.2433 0.9000 0.9000   0.67  1  1  4.6E-06
  5 :  -2.00E-05 2.36E-08 0.000 0.0068 0.9990 0.9990   0.95  1  1  3.2E-08
  6 :  -2.00E-05 5.46E-09 0.126 0.2309 0.9000 0.9000   1.00  1  1  7.4E-09
  7 :  -2.00E-05 1.34E-09 0.000 0.2465 0.9000 0.9000   1.00  1  1  1.8E-09
  8 :  -2.00E-05 7.34E-11 0.000 0.0546 0.9900 0.9900   1.00  1  1  8.4E-09
  9 :  -2.00E-05 4.05E-12 0.325 0.0552 0.9900 0.9900   1.00  1  1  4.6E-10

iter seconds digits       c*x               b*y
  9      0.1   7.7 -1.9999996479e-05 -1.9999997309e-05
|Ax-b| =   6.6e-12, [Ay-c]_+ =   7.7E-13, |x|=  3.8e+00, |y|=  3.9e+01

Detailed timing (sec)
   Pre          IPM          Post
0.000E+00    5.000E-02    0.000E+00
Max-norms: ||b||=1, ||c|| = 2.000000e-05,
Cholesky |add|=0, |skip| = 1, ||L.L|| = 100.
solution =
    yalmiptime: 3.3302e-02
    solvertime: 5.6067e-02
          info: 'No problems detected (SeDuMi-1.1)'
       problem: 0
        dimacs: [3.2897e-12 0 0 7.7185e-13 8.3014e-13 2.8518e-12]

% compute gain
K = double(L) / double(X)
K =
  -4.9976e+14   4.5305e+10  -4.9976e+14   4.5305e+10
   4.4502e+10  -4.9784e+14   4.4501e+10  -4.9784e+14
Acl = At + But * K;
eig(Acl)
ans =
  -4.9976e+12
  -4.9784e+12
  -1.0000e+00
  -1.0000e+00

% look at cost
minperf = trace(double(Z))
minperf =
   2.0000e-05

% solve multiobjective performance
X = sdpvar(n,n,'symmetric');
```

```
L = sdpvar(m,n);
Zp = sdpvar(n,n,'symmetric');
Zu = sdpvar(m,m,'symmetric');

LMI1 = At*X+X*At'+But*L+L'*But'+Bwt*W*Bwt';
LMI2 = [Zp Cz*X; X*Cz X];
LMI3 = [Zu Dzu*L; L'*Dzu X];

LMI = set(LMI1 < 0) + set(LMI2 > 0) + set(LMI3 > 0) ...
    + set(trace(Zu) <= 1.6*minenergy);
options = sdpsettings('solver','sedumi');
solution =solvesdp(LMI,trace(Zp),options)
SeDuMi 1.1R3 by AdvOL, 2006 and Jos F. Sturm, 1998-2003.
Alg = 2: xz-corrector, theta = 0.250, beta = 0.500
eqs m = 31, order n = 20, dim = 118, blocks = 4
nnz(A) = 87 + 0, nnz(ADA) = 741, nnz(L) = 386
 it :     b*y       gap    delta  rate   t/tP*  t/tD*   feas cg cg  prec
  0 :            1.60E+00 0.000
  1 :  -1.34E-01 3.59E-01 0.000 0.2242 0.9000 0.9000   2.00  1  1  1.4E+00
  2 :   6.57E-05 1.88E-02 0.000 0.0524 0.9900 0.9900   1.82  1  1  1.6E-01
  3 :   2.56E-04 5.00E-04 0.000 0.0266 0.9900 0.9900   1.06  1  1  4.0E-02
  4 :   3.79E-05 9.72E-05 0.000 0.1942 0.9000 0.9000   1.39  1  1  5.7E-03
  5 :   2.95E-06 3.83E-05 0.000 0.3940 0.9000 0.9000   0.60  1  1  1.7E-03
  6 :  -1.04E-04 1.59E-05 0.000 0.4166 0.9000 0.9000  -0.70  1  1  8.3E-05
  7 :  -6.45E-04 4.58E-06 0.000 0.2872 0.9000 0.9000  -0.78  1  1  8.1E-05
  8 :  -3.71E-03 9.47E-07 0.000 0.2068 0.9000 0.9000  -0.94  1  1  7.8E-05
  9 :  -2.01E-02 1.82E-07 0.000 0.1918 0.9000 0.9000  -0.93  1  1  7.0E-05
 10 :  -8.19E-02 4.50E-08 0.000 0.2475 0.9000 0.9000  -0.89  1  1  6.5E-05
 11 :  -4.41E-01 8.61E-09 0.000 0.1915 0.9000 0.9000  -0.92  1  1  6.5E-05
 12 :  -2.07E+00 1.79E-09 0.000 0.2073 0.9000 0.9000  -0.96  1  1  6.4E-05
 13 :  -7.60E+00 4.76E-10 0.000 0.2667 0.9000 0.9000  -0.96  1  1  6.3E-05
 14 :  -2.67E+01 1.29E-10 0.000 0.2711 0.9000 0.9000  -0.93  1  1  5.9E-05
 15 :  -8.05E+01 3.95E-11 0.000 0.3062 0.9000 0.9000  -0.91  1  1  5.6E-05
 16 :  -2.37E+02 1.21E-11 0.000 0.3064 0.9000 0.9000  -0.89  1  1  5.2E-05
 17 :  -5.77E+02 4.33E-12 0.000 0.3574 0.9000 0.9000  -0.84  3  3  4.7E-05
 18 :  -1.17E+03 1.76E-12 0.000 0.4071 0.9000 0.9000  -0.75  5  5  4.0E-05
 19 :  -2.18E+03 7.52E-13 0.000 0.4267 0.9000 0.9000  -0.67  5  5  3.5E-05
 20 :  -3.65E+03 3.33E-13 0.000 0.4432 0.9000 0.9000  -0.54  5  5  2.8E-05
 21 :  -6.11E+03 1.40E-13 0.000 0.4202 0.9000 0.9000  -0.46  7  5  2.4E-05
 22 :  -9.16E+03 5.53E-14 0.000 0.3948 0.9000 0.9000  -0.25  5  5  1.5E-05
 23 :  -1.27E+04 2.30E-14 0.000 0.4163 0.9000 0.9000  -0.11  6  6  1.1E-05
 24 :  -1.60E+04 7.89E-15 0.000 0.3427 0.9000 0.9000   0.22  6  7  5.2E-06
 25 :  -1.83E+04 3.04E-15 0.000 0.3849 0.9000 0.9000   0.37  6  7  2.7E-06
 26 :  -1.99E+04 7.28E-16 0.000 0.2397 0.9000 0.9000   0.69  7  6  7.5E-07
 27 :  -2.05E+04 1.39E-16 0.000 0.1905 0.9000 0.9000   0.85  8  8  1.5E-07
 28 :  -2.06E+04 5.62E-18 0.000 0.0406 0.9900 0.9900   0.96  8  8  6.4E-09
 29 :  -2.06E+04 1.25E-18 0.000 0.2232 0.9000 0.9000   0.99  8  9  1.4E-09
 30 :  -2.07E+04 3.39E-19 0.000 0.2700 0.9000 0.9000   1.00  9 11  3.9E-10

iter seconds digits       c*x               b*y
 30      0.3   Inf -2.0650096289e+04 -2.0650054917e+04
|Ax-b| =   9.2e-07, [Ay-c]_+ =   6.1E-11, |x|=  1.2e+10, |y|=  2.8e+04

Detailed timing (sec)
```

```
   Pre          IPM          Post
0.000E+00    3.000E-01    0.000E+00
Max-norms: ||b||=1, ||c|| = 5.424671e-04,
Cholesky |add|=3, |skip| = 0, ||L.L|| = 500000.
solution =
    yalmiptime: 3.4471e-02
    solvertime: 3.0020e-01
          info: 'No problems detected (SeDuMi-1.1)'
       problem: 0
        dimacs: [4.6892e-07 0 0 6.0677e-11 -1.0017e-06 -1.0016e-06]

% compute gain
K = double(L) / double(X)
K =
  -1.3965e-05   2.3818e-05  -4.8089e-03  -4.3240e-02
  -1.2957e-04   7.1770e-05  -4.3234e-02  -6.5548e-01
Acl = At + But * K;
eig(Acl)
ans =
  -1.6341e-03 + 3.0522e-03i
  -1.6341e-03 - 3.0522e-03i
  -3.2551e-03
  -7.9551e-05

% look at cost
1.2*minenergy
ans =
   4.0685e-04
trace(double(Zu))
ans =
   5.4247e-04
trace(double(Zp))
ans =
   2.0650e+04

return
Your MATLAB session has timed out.  All license keys have been returned.


quit
```