

9 The LQR Problem Revisited

Problem: Compute a state feedback controller

$$u(t) = Kx(t)$$

that stabilizes the closed loop system and minimizes

$$J := \lim_{t \rightarrow \infty} E [z(t)^T z(t)]$$

for the LTI system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_u u(t) + B_w w(t), \quad x(0) = 0 \\ z(t) &= C_z x(t) + D_{zu} u(t). \end{aligned}$$

Assumptions:

- a) $D_{zu}^T D_{zu} \succ 0$;
- b) (A, B_u) stabilizable;
- c) $w(t)$ is a Gaussian white noise vector with zero mean and covariance $W \succ 0$.

Solution: As before, restate the cost function with the help of the Gramian

$$J = \text{trace} (P B_w W B_w^T)$$

where

$$(A + B_u K)^T P + P(A + B_u K) + (C_z + D_{zu} K)^T (C_z + D_{zu} K) = 0.$$

Use the Comparison Lemma to conclude that for any K such that $(A + B_u K)$ is Hurwitz then

$$Y \succeq P \quad \implies \quad \text{trace} (Y B_w W B_w^T) \geq J$$

where Y satisfies the inequality

$$(A + B_u K)^T Y + Y(A + B_u K) + (C_z + D_{zu} K)^T (C_z + D_{zu} K) \preceq 0.$$

9.1 LQR revisited: solution by congruence + change-of-variables

Recipe I (congruence + change-of-variables):

First apply Schur complement

$$\begin{aligned} (A + B_u K)^T Y + Y(A + B_u K) + (C_z + D_{zu} K)^T (C_z + D_{zu} K) &\preceq 0 \\ \Leftrightarrow \\ \begin{bmatrix} (A + B_u K)^T Y + Y(A + B_u K) & C_z^T + K^T D_{zu}^T \\ C_z + D_{zu} K & -I \end{bmatrix} &\preceq 0 \end{aligned}$$

Under the assumption that (A, B_u) is stabilizable and $D_{zu}^T D_{zu} \succ 0$ we have that $Y \succeq P \succ 0$ so that we can apply the congruence transformation

$$\begin{aligned} \begin{bmatrix} Y^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} (A + B_u K)^T Y + Y(A + B_u K) & C_z^T + K^T D_{zu}^T \\ C_z + D_{zu} K & -I \end{bmatrix} \begin{bmatrix} Y^{-1} & 0 \\ 0 & I \end{bmatrix} &\preceq 0 \\ \Leftrightarrow \\ \begin{bmatrix} (A + B_u K)Y^{-1} + Y^{-1}(A + B_u K)^T & Y^{-1}C_z^T + Y^{-1}K^T D_{zu}^T \\ C_z Y^{-1} + D_{zu} K Y^{-1} & -I \end{bmatrix} &\preceq 0 \end{aligned}$$

Apply change-of-variables

$$X := Y^{-1}, \quad L := K Y^{-1}$$

to obtain the LMI

$$\begin{bmatrix} AX + XA^T + B_u L + L^T B_u^T & XC_z^T + L^T D_{zu}^T \\ C_z X + D_{zu} L & -I \end{bmatrix} \preceq 0$$

As for the cost function

$$\text{trace}(X^{-1} B_w W B_w^T) = \text{trace}(Y B_w W B_w^T) \geq J$$

Now introduce an extra variable Z such that

$$Z \succeq B_w^T X^{-1} B_w$$

Therefore

$$\text{trace}(ZW) = \text{trace}(B_w^T X^{-1} B_w W) \geq J$$

Using Schur Complement

$$Z \succeq B_w^T X^{-1} B_w \quad \Leftrightarrow \quad \begin{bmatrix} Z & B_w^T \\ B_w & X \end{bmatrix} \succeq 0.$$

9.1.1 Summary: LQR revisited (first form)

The state feedback controller

$$u(t) = Kx(t), \quad K = LX^{-1}$$

where $X \in \mathbb{S}^n$, $L \in \mathbb{R}^{m \times n}$ and $Z \in \mathbb{S}^r$ solve the SDP

$$\begin{aligned} \min_{X \in \mathbb{S}^n, L \in \mathbb{R}^{m \times n}, Z \in \mathbb{S}^r} \quad & \text{trace}(ZW) \\ \text{s.t.} \quad & \begin{bmatrix} AX + XA^T + B_u L + L^T B_u^T & XC_z^T + L^T D_{zu}^T \\ C_z X + D_{zu} L & -I \end{bmatrix} \preceq 0, \\ & \begin{bmatrix} Z & B_w^T \\ B_w & X \end{bmatrix} \succeq 0, \quad X \succ 0. \end{aligned}$$

stabilizes the closed loop system and minimizes the cost function

$$J := \lim_{t \rightarrow \infty} E [z(t)^T z(t)]$$

for the LTI system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_u u(t) + B_w w(t), \quad x(0) = 0 \\ z(t) &= C_z x(t) + D_{zu} u(t) \end{aligned}$$

under the assumptions

- a) $D_{zu}^T D_{zu} \succ 0$;
- b) (A, B_u) stabilizable;
- c) $w(t)$ is a Gaussian white noise vector with zero mean and covariance $W \succ 0$.

9.2 Elimination Lemma

Let $x \in \mathbb{R}^n$, $Q \in \mathbb{S}^n$, $B \in \mathbb{R}^{m \times n}$ and $C \in \mathbb{R}^{p \times n}$ such that $\text{rank}(B) < n$ and $\text{rank}(C) < n$. The following statements are equivalent:

- i) $B^{\perp T} Q B^{\perp} \prec 0$ and $C^{\perp T} Q C^{\perp} \prec 0$
- ii) $\exists \mu \in \mathbb{R} : Q - \mu B^T B \prec 0$, and $Q - \mu C^T C \prec 0$.
- iii) $\exists X \in \mathbb{R}^{p \times m} : Q + C^T X B + B^T X^T C \prec 0$.

Remark: Proofs can be found in Skelton, Iwasaki and Grigoriadis (Theorem 2.3.12, p. 29.) or Boyd, El Gahoui, Feron & Balakrishnan (p. 32). None as nice as the next one :).

Proof:

$ii)$ and $iii) \Rightarrow i)$ is as in Finsler's Lemma, just multiply by $B^{\perp}$ and $C^{\perp}$.

$i) \Rightarrow ii)$ use Finsler's Lemma to show that $i)$ implies the existence of scalars μ_1 and μ_2 such that

$$Q - \mu_1 B^T B \prec 0, \quad Q - \mu_2 C^T C \prec 0.$$

Then $\mu = \max\{\mu_1, \mu_2\}$ is such that $ii)$ holds for some μ .

$ii) \Rightarrow iii)$ requires an explicit construction of X .

9.2.1 Construction of $\mathcal{X}$

Assume that *ii*) holds. If $\mu \leq 0$ then $\mathcal{X} = 0$ solves the problem. So assume $\mu > 0$. Now let $\rho^2 = \mu$ so that

$$\mathcal{Q} - \rho^2 \mathcal{B}^T \mathcal{B} \prec 0, \quad \mathcal{Q} - \rho^2 \mathcal{C}^T \mathcal{C} \prec 0.$$

Define

$$\Phi := \mathcal{Q} - \rho^2 \mathcal{B}^T \mathcal{B} - \rho^2 \mathcal{C}^T \mathcal{C}$$

Then

$$\Phi + \rho^2 \mathcal{C}^T \mathcal{C} \prec 0, \quad \Phi + \rho^2 \mathcal{B}^T \mathcal{B} \prec 0.$$

Also notice that $\Phi \prec 0$ because

$$\Phi := \frac{1}{2} [(\mathcal{Q} - 2\rho^2 \mathcal{B}^T \mathcal{B}) + (\mathcal{Q} - 2\rho^2 \mathcal{C}^T \mathcal{C})]$$

and $2\rho^2 > \rho^2 = \mu$. Therefore, using Schur Complement

$$\Phi + \rho^2 \mathcal{C}^T \mathcal{C} \prec 0 \quad \Longleftrightarrow \quad \begin{bmatrix} \Phi & \rho \mathcal{C}^T \\ \rho \mathcal{C} & -I \end{bmatrix} \prec 0.$$

Schur Complement one more time

$$\begin{bmatrix} \Phi & \rho \mathcal{C}^T \\ \rho \mathcal{C} & -I \end{bmatrix} \prec 0 \quad \Longleftrightarrow \quad -I - \rho^2 \mathcal{C} \Phi^{-1} \mathcal{C}^T \prec 0.$$

Likewise

$$\Phi + \rho^2 \mathcal{B}^T \mathcal{B} \prec 0 \quad \Longleftrightarrow \quad -I - \rho^2 \mathcal{B} \Phi^{-1} \mathcal{B}^T \prec 0.$$

We will now prove that

$$\mathcal{X} = \rho^2 \mathcal{Y} \quad \mathcal{Y} = \rho^2 \mathcal{C} \Phi^{-1} \mathcal{B}^T$$

satisfies *iii*).

With that in mind notice that

$$\begin{bmatrix} -I - \rho^2 \mathcal{B} \Phi^{-1} \mathcal{B}^T & \mathcal{Y}^T - \rho^2 \mathcal{B} \Phi^{-1} \mathcal{C}^T \\ \mathcal{Y} - \rho^2 \mathcal{C} \Phi^{-1} \mathcal{B}^T & -I - \rho^2 \mathcal{C} \Phi^{-1} \mathcal{C}^T \end{bmatrix} \prec 0.$$

Using Schur Complement this is equivalent to

$$\begin{bmatrix} \Phi & \rho \mathcal{B}^T & \rho \mathcal{C}^T \\ \rho \mathcal{B} & -I & \mathcal{Y}^T \\ \rho \mathcal{C} & \mathcal{Y} & -I \end{bmatrix} \prec 0$$

Using Schur Complement one more time

$$\begin{bmatrix} \Phi + \rho^2 \mathcal{C}^T \mathcal{C} & \rho \mathcal{B}^T + \rho \mathcal{C}^T \mathcal{Y} \\ \rho \mathcal{B} + \rho \mathcal{Y}^T \mathcal{C} & -I + \mathcal{Y}^T \mathcal{Y} \end{bmatrix} = \begin{bmatrix} \mathcal{Q} - \rho^2 \mathcal{B}^T \mathcal{B} & \rho \mathcal{B}^T + \rho \mathcal{C}^T \mathcal{Y} \\ \rho \mathcal{B} + \rho \mathcal{Y}^T \mathcal{C} & -I + \mathcal{Y}^T \mathcal{Y} \end{bmatrix} \prec 0$$

and once again

$$\mathcal{Q} - \rho^2 \mathcal{B}^T \mathcal{B} + \rho^2 (\mathcal{B}^T + \mathcal{C}^T \mathcal{Y})(I - \mathcal{Y}^T \mathcal{Y})^{-1}(\mathcal{B} + \mathcal{Y}^T \mathcal{C}) \prec 0$$

and $I - \mathcal{Y}^T \mathcal{Y} \succ 0$. Using the Lemma of the Matrix Inverse²

$$(I - \mathcal{Y}^T \mathcal{Y})^{-1} = I + \mathcal{Y}^T (I - \mathcal{Y} \mathcal{Y}^T)^{-1} \mathcal{Y}$$

so that

$$\begin{aligned} 0 &\succ \mathcal{Q} - \rho^2 \mathcal{B}^T \mathcal{B} + \rho^2 (\mathcal{B}^T + \mathcal{C}^T \mathcal{Y})(I - \mathcal{Y}^T \mathcal{Y})^{-1}(\mathcal{B} + \mathcal{Y}^T \mathcal{C}) \\ &= \mathcal{Q} + \rho^2 \mathcal{C}^T \mathcal{Y} \mathcal{B} + \rho^2 \mathcal{B}^T \mathcal{Y}^T \mathcal{C} + \rho^2 \mathcal{C}^T \mathcal{Y} \mathcal{Y}^T \mathcal{C} \\ &\quad + \rho^2 (\mathcal{B} + \mathcal{C}^T \mathcal{Y}) \mathcal{Y}^T (I - \mathcal{Y}^T \mathcal{Y})^{-1} \mathcal{Y} (\mathcal{B}^T + \mathcal{Y}^T \mathcal{C}) \end{aligned}$$

hence *iii*) holds because

$$\rho^2 \mathcal{C}^T \mathcal{Y} \mathcal{Y}^T \mathcal{C} + \rho^2 (\mathcal{B} + \mathcal{C}^T \mathcal{Y}) \mathcal{Y}^T (I - \mathcal{Y}^T \mathcal{Y})^{-1} \mathcal{Y} (\mathcal{B}^T + \mathcal{Y}^T \mathcal{C}) \succ 0.$$

²Verify that

$$\begin{aligned} [I + \mathcal{Y}^T (I - \mathcal{Y} \mathcal{Y}^T)^{-1} \mathcal{Y}] (I - \mathcal{Y}^T \mathcal{Y}) &= I - \mathcal{Y}^T \mathcal{Y} + \mathcal{Y}^T (I - \mathcal{Y} \mathcal{Y}^T)^{-1} \mathcal{Y} (I - \mathcal{Y}^T \mathcal{Y}) \\ &= I - \mathcal{Y}^T \mathcal{Y} + \mathcal{Y}^T (I - \mathcal{Y} \mathcal{Y}^T)^{-1} (I - \mathcal{Y} \mathcal{Y}^T) \mathcal{Y} \\ &= I - \mathcal{Y}^T \mathcal{Y} + \mathcal{Y}^T \mathcal{Y} \\ &= I. \end{aligned}$$

9.3 LQR revisited: solution by Elimination Lemma

Recipe II (existence conditions):

As before we start with

$$Y \succ 0, \quad \begin{bmatrix} (A + B_u K)^T Y + Y(A + B_u K) & C_z^T + K^T D_{zu}^T \\ C_z + D_{zu} K & -I \end{bmatrix} \prec 0$$

Note that we are using strict inequalities! Now rewrite

$$\underbrace{\begin{bmatrix} A^T Y + Y A & C_z^T \\ C_z & -I \end{bmatrix}}_{\mathcal{Q}} + \underbrace{\begin{bmatrix} Y B_u \\ D_{zu} \end{bmatrix}}_{\mathcal{B}^T} \underbrace{K}_{\mathcal{X}^T} \underbrace{\begin{bmatrix} I & 0 \end{bmatrix}}_{\mathcal{C}} + \underbrace{\begin{bmatrix} I \\ 0 \end{bmatrix}}_{\mathcal{C}^T} \underbrace{K^T}_{\mathcal{X}} \underbrace{\begin{bmatrix} B_u^T Y & D_{zu}^T \end{bmatrix}}_{\mathcal{B}} \prec 0.$$

We need a more general version of Finsler's Lemma.

9.3.1 Computing $\mathcal{B}^\perp$ and $\mathcal{C}^\perp$ under no assumptions

First compute

$$\mathcal{C}^\perp = \begin{bmatrix} I & 0 \end{bmatrix}^\perp = \begin{bmatrix} 0 \\ I \end{bmatrix}.$$

The problem of computing a basis for

$$0 = \mathcal{B}x = \begin{bmatrix} B_u^T Y & D_{zu}^T \end{bmatrix} x = \begin{bmatrix} B_u^T & D_{zu}^T \end{bmatrix} \begin{bmatrix} Y & 0 \\ 0 & I \end{bmatrix} x$$

is more involved. As before, because Y is nonsingular

$$\begin{bmatrix} B_u^T & D_{zu}^T \end{bmatrix} y = 0, \quad y = \begin{bmatrix} Y & 0 \\ 0 & I \end{bmatrix} x \neq 0.$$

Let

$$\begin{bmatrix} E_1 \\ E_2 \end{bmatrix} := \begin{bmatrix} B_u^T & D_{zu}^T \end{bmatrix}^\perp$$

so that

$$\mathcal{B}^\perp = \begin{bmatrix} Y^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} B_u^T & D_{zu}^T \end{bmatrix}^\perp = \begin{bmatrix} Y^{-1} E_1 \\ E_2 \end{bmatrix}.$$

This choice produce the inequalities

$$\begin{aligned}
0 &\succ \mathcal{B}^{\perp T} \mathcal{Q} \mathcal{B}^{\perp} \\
&= \begin{bmatrix} E_1^T Y^{-1} & E_2^T \end{bmatrix} \begin{bmatrix} A^T Y + Y A & C_z^T \\ C_z & -I \end{bmatrix} \begin{bmatrix} Y^{-1} E_1 \\ E_2 \end{bmatrix} \\
&= \begin{bmatrix} E_1^T & E_2^T \end{bmatrix} \begin{bmatrix} Y^{-1} A^T + A Y^{-1} & Y^{-1} C_z^T \\ C_z Y^{-1} & -I \end{bmatrix} \begin{bmatrix} E_1 \\ E_2 \end{bmatrix}.
\end{aligned}$$

and

$$\begin{aligned}
0 &\succ \mathcal{C}^{\perp T} \mathcal{Q} \mathcal{C}^{\perp} \\
&= \begin{bmatrix} 0 & I \end{bmatrix} \begin{bmatrix} A^T Y + Y A & C_z^T \\ C_z & -I \end{bmatrix} \begin{bmatrix} 0 \\ I \end{bmatrix} \\
&= -I
\end{aligned}$$

which is trivially satisfied. The first inequality, along with the cost inequality derived previously can be linearized with the change-of-variables $X := Y^{-1}$.

9.3.2 Summary: LQR revisited (second form)

The optimal state feedback controller

$$u(t) = Kx(t)$$

which can be computed from the solution to the SDP in the variables $X \in \mathbb{S}^n$, $Z \in \mathbb{S}^r$ solve the SDP

$$\begin{aligned} \min_{X \in \mathbb{S}^n, Z \in \mathbb{S}^r} \quad & \text{trace}(ZW) \\ \text{s.t.} \quad & \begin{bmatrix} E_1^T & E_2^T \end{bmatrix} \begin{bmatrix} AX + XA^T & XC_z^T \\ C_z X & -I \end{bmatrix} \begin{bmatrix} E_1 \\ E_2 \end{bmatrix} \prec 0, \\ & \begin{bmatrix} Z & B_w^T \\ B_w & X \end{bmatrix} \succeq 0, \quad X \succ 0. \end{aligned}$$

where

$$\begin{bmatrix} E_1 \\ E_2 \end{bmatrix} = [B_u^T \ D_{zu}^T]^\perp$$

stabilizes the closed loop system and minimizes the cost function

$$J := \lim_{t \rightarrow \infty} E [z(t)^T z(t)]$$

for the LTI system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_u u(t) + B_w w(t), \quad x(0) = 0 \\ z(t) &= C_z x(t) + D_{zu} u(t) \end{aligned}$$

under the assumptions

- a) $D_{zu}^T D_{zu} \succ 0$;
- b) (A, B_u) stabilizable;
- c) $w(t)$ is a Gaussian white noise vector with zero mean and covariance $W \succ 0$.

9.3.3 Computing $\mathcal{B}^\perp$ under the assumption $D_{zu}^T D_{zu} \succ 0$

Proceed as before but now notice that if $D_{zu}^T D_{zu} \succ 0$ then

$$\begin{bmatrix} E_1 \\ E_2 \end{bmatrix} = \begin{bmatrix} I & 0 \\ -D_{zu}(D_{zu}^T D_{zu})^{-1} B_u^T & (D_{zu}^T)^\perp \end{bmatrix} = [B_u^T \ D_{zu}^T]^\perp$$

is a possible choice for E_1 and E_2 so that, using $X := Y^{-1}$ we have

$$\begin{aligned} 0 &\succ \mathcal{B}^{\perp T} \mathcal{Q} \mathcal{B}^\perp \\ &= \begin{bmatrix} E_1^T & E_2^T \end{bmatrix} \begin{bmatrix} AY^{-1} + Y^{-1}A^T + Y^{-1}C_z^T & \\ C_z Y^{-1} & -I \end{bmatrix} \begin{bmatrix} E_1 \\ E_2 \end{bmatrix} \\ &= \begin{bmatrix} I & 0 \\ -D_{zu}(D_{zu}^T D_{zu})^{-1} B_u^T & (D_{zu}^T)^\perp \end{bmatrix}^T \\ &\quad \times \begin{bmatrix} AX + XA^T & XC_z^T \\ C_z X & -I \end{bmatrix} \begin{bmatrix} I & 0 \\ -D_{zu}(D_{zu}^T D_{zu})^{-1} B_u^T & (D_{zu}^T)^\perp \end{bmatrix} \\ &= \begin{bmatrix} I & 0 \\ -D_{zu}(D_{zu}^T D_{zu})^{-1} B_u^T & (D_{zu}^T)^\perp \end{bmatrix}^T \\ &\quad \times \begin{bmatrix} AX + X\tilde{A}^T & XC_z^T (D_{zu}^T)^\perp \\ C_z X + D_{zu}(D_{zu}^T D_{zu})^{-1} B_u^T & (D_{zu}^T)^\perp \end{bmatrix} \\ &= \begin{bmatrix} \tilde{A}X + X\tilde{A}^T - B_u(D_{zu}^T D_{zu})^{-1} B_u^T & XC_z^T (D_{zu}^T)^\perp \\ (D_{zu}^T)^\perp{}^T C_z X & (D_{zu}^T)^\perp{}^T (D_{zu}^T)^\perp \end{bmatrix} \end{aligned}$$

where

$$\tilde{A} := A - B_u(D_{zu}^T D_{zu})^{-1} D_{zu}^T C_z.$$

9.3.4 Summary: LQR revisited (third form)

The optimal state feedback controller

$$u(t) = Kx(t)$$

which can be computed from the solution to the SDP in the variables $X \in \mathbb{S}^n$, $Z \in \mathbb{S}^r$ solve the SDP

$$\begin{aligned} \min_{X \in \mathbb{S}^n, Z \in \mathbb{S}^r} \quad & \text{trace}(ZW) \\ \text{s.t.} \quad & \begin{bmatrix} \tilde{A}X + X\tilde{A}^T - B_u(D_{zu}^T D_{zu})^{-1}B_u^T & XC_z^T(D_{zu}^T)^\perp \\ (D_{zu}^T)^\perp C_z X & (D_{zu}^T)^\perp (D_{zu}^T)^\perp \end{bmatrix} \prec 0, \\ & \begin{bmatrix} Z & B_w^T \\ B_w & X \end{bmatrix} \succeq 0, \quad X \succ 0. \end{aligned}$$

where

$$\tilde{A} := A - B_u(D_{zu}^T D_{zu})^{-1}D_{zu}^T C_z,$$

stabilizes the closed loop system and minimizes the cost function

$$J := \lim_{t \rightarrow \infty} E [z(t)^T z(t)]$$

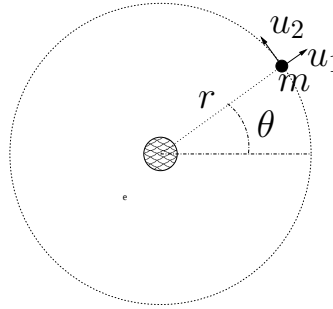
for the LTI system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_u u(t) + B_w w(t), \quad x(0) = 0 \\ z(t) &= C_z x(t) + D_{zu} u(t) \end{aligned}$$

under the assumptions

- a) $D_{zu}^T D_{zu} \succ 0$;
- b) (A, B_u) stabilizable;
- c) $w(t)$ is a Gaussian white noise vector with zero mean and covariance $W \succ 0$.

9.4 Example: satellite in circular orbit



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

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

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 linearizing around

$$\bar{x}(t) = \begin{pmatrix} \bar{r} \\ \bar{\omega}t \\ 0 \\ \bar{\omega} \end{pmatrix}, \quad \bar{u} = \begin{pmatrix} \bar{u}_1 \\ \bar{u}_2 \end{pmatrix} = 0, \quad \bar{w} = \begin{pmatrix} \bar{w}_1 \\ \bar{w}_2 \end{pmatrix} = 0, \quad \bar{\omega} = \sqrt{k/\bar{r}^3}$$

one obtains the linearized system

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

Problem: Given

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

where $G \approx 6.673 \times 10^{-11} \text{ N m}^2/\text{kg}^2$ is the universal gravitational constant, and $M \approx 5.98 \times 10^{24} \text{ kg}$ and $R \approx 6.37 \times 10^3 \text{ km}$ are the mass and radius of the earth. If the variances δ_1 and δ_2 are $0.1N$ find solutions to the LQR control problem where

$$Q = I, \quad R = \rho I,$$

using u_2 only first, then using u_1 and u_2 .

```

% MAE 280 B - Linear Control Design
% Mauricio de Oliveira
%
% LMI 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

```

```

% optimal LQR state feedback control (using u2 only)
n = size(At,1);
m = 1;

% LQR solution
Cz = [eye(n); zeros(m,n)];
Dzu = [zeros(n,m); eye(m)];

[K,X,S] = lqr(At, But(:,2), Cz' * Cz, Dzu' * Dzu);
Klqr = - K
Klqr =
    -1.0055e+00    1.0000e+00   -5.2522e+01   -1.8511e+01

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

% declare LMIs
LMI1 = [At*X+X*At'+But(:,2)*L+L'*But(:,2)', X*Cz'+L'*Dzu';...
        Cz*X+Dzu*L, -eye(m+n)];
LMI2 = [Z, Bwt(:,2)'; Bwt(:,2), X];
LMI3 = X;

LMI = set(LMI1 < 0) + set(LMI2 > 0) + set(LMI3 > 0);
options = sdpsettings('solver','sedumi');
solution = solvesdp(LMI,trace(W(2,2)*Z),options)
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 = 15, order n = 19, dim = 123, blocks = 4
nnz(A) = 65 + 0, nnz(ADA) = 217, nnz(L) = 116
it :      b*y      gap    delta  rate  t/tP*  t/tD*   feas cg cg  prec
 0 :              4.49E+00 0.000
 1 :  -6.05E-02  1.02E+00 0.000 0.2270 0.9000 0.9000   2.18  1  1  1.4E+00
 2 :  -1.66E-02  2.34E-01 0.000 0.2298 0.9000 0.9000   1.64  1  1  6.1E-01
 3 :  -3.52E-03  5.82E-02 0.000 0.2484 0.9000 0.9000   1.25  1  1  4.6E-01
 4 :  -1.59E-03  1.67E-02 0.000 0.2862 0.9000 0.9000   1.14  1  1  4.5E-01
 5 :  -2.29E-03  5.42E-03 0.000 0.3256 0.9000 0.9000   0.86  1  1  5.6E-01
 6 :  -3.45E-03  2.22E-03 0.000 0.4094 0.9000 0.9000   0.51  1  1  2.5E-01
 7 :  -4.71E-03  9.46E-04 0.000 0.4262 0.9000 0.9000   0.45  1  1  1.4E-01
 8 :  -5.65E-03  4.91E-04 0.000 0.5188 0.9000 0.9000   0.32  1  1  1.1E-01
 9 :  -7.25E-03  2.24E-04 0.000 0.4556 0.9000 0.9000   0.43  1  1  5.3E-02
10 :  -8.33E-03  1.05E-04 0.000 0.4707 0.9000 0.9000   0.12  1  1  1.7E-02
11 :  -1.07E-02  3.89E-05 0.000 0.3694 0.9000 0.9000   0.40  1  1  1.7E-04
12 :  -1.23E-02  1.50E-05 0.000 0.3863 0.9000 0.9000   0.26  1  1  1.1E-04
13 :  -1.40E-02  5.31E-06 0.000 0.3537 0.9000 0.9000   0.43  1  1  5.3E-05
14 :  -1.52E-02  2.11E-06 0.000 0.3977 0.9000 0.9000   0.33  1  1  3.2E-05
15 :  -1.63E-02  7.42E-07 0.000 0.3513 0.9000 0.9000   0.46  1  1  1.5E-05
16 :  -1.70E-02  2.94E-07 0.000 0.3962 0.9000 0.9000   0.37  1  1  8.6E-06
17 :  -1.76E-02  1.04E-07 0.000 0.3531 0.9000 0.9000   0.49  1  1  4.0E-06
18 :  -1.79E-02  4.26E-08 0.000 0.4099 0.9000 0.9000   0.46  1  1  2.2E-06
19 :  -1.82E-02  1.41E-08 0.000 0.3322 0.9000 0.9000   0.62  1  1  8.9E-07
20 :  -1.84E-02  5.15E-09 0.000 0.3645 0.9000 0.9000   0.63  1  1  4.0E-07

```

```

21 : -1.84E-02 1.57E-09 0.000 0.3048 0.9000 0.9000 0.77 1 1 1.4E-07
22 : -1.85E-02 4.67E-10 0.000 0.2970 0.9000 0.9000 0.79 1 1 4.7E-08
23 : -1.85E-02 1.58E-10 0.000 0.3389 0.9000 0.9000 0.87 1 1 1.7E-08
24 : -1.85E-02 7.56E-11 0.000 0.4781 0.9000 0.9000 0.80 2 2 9.3E-09
25 : -1.85E-02 2.46E-11 0.000 0.3253 0.9000 0.9000 0.80 2 2 3.4E-09
26 : -1.85E-02 1.20E-11 0.000 0.4868 0.9000 0.9000 0.57 2 2 2.2E-09
27 : -1.85E-02 5.27E-12 0.000 0.4406 0.9000 0.9000 0.48 2 2 1.3E-09
28 : -1.85E-02 2.81E-12 0.000 0.5320 0.9000 0.9000 0.26 2 2 9.6E-10

iter seconds digits c*x b*y
28 0.4 5.3 -1.8504781971e-02 -1.8504688922e-02
|Ax-b| = 8.7e-10, [Ay-c]_+ = 3.0E-10, |x|= 5.0e+03, |y|= 2.4e-01

Detailed timing (sec)
Pre IPM Post
2.786E-01 5.072E-01 3.276E-02
Max-norms: ||b||=1.000000e-01, ||c|| = 1,
Cholesky |add|=0, |skip| = 0, ||L.L|| = 1354.19.
solution =
    yalmiptime: 5.0932e-01
    solvertime: 9.4819e-01
    info: 'Numerical problems (SeDuMi-1.3)'
    problem: 4
    dimacs: [NaN NaN NaN NaN NaN NaN]

% Construct K and check closed loop stability
Klqr
Klqr =
    -1.0055e+00    1.0000e+00   -5.2522e+01   -1.8511e+01
K = double(L) / double(X)
K =
    -1.0035e+00    2.1960e-01   -3.7365e+01   -1.8505e+01
Acl = At + But(:,2)*K;
eig(Acl)
ans =
    -6.7474e-02 + 7.5883e-02i
    -6.7474e-02 - 7.5883e-02i
    -5.0082e-02
    -1.7156e-05

% optimal LQR state feedback control (using u1 and u2)
n = size(At,1);
m = 2;

% LQR solution
Cz = [eye(n); zeros(m,n)];
Dzu = [zeros(n,m); eye(m)];

[K,X,S] = lqr(At, But, Cz' * Cz, Dzu' * Dzu);
Klqr = - K
Klqr =
    -9.8444e-01    1.7795e-01   -1.3843e+01   -2.4919e+00
    -1.7795e-01   -9.8404e-01   -2.4919e+00   -1.4741e+01

```

```

% LMI solution
% 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', X*Cz'+L'*Dzu';...
        Cz*X+Dzu*L, -eye(m+n)];
LMI2 = [Z, Bwt'; Bwt, X];

LMI = set(LMI1 < 0) + set(LMI2 > 0);
options = sdpsettings('solver','sedumi');
solution = solvesdp(LMI,trace(W*Z),options)
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 = 21, order n = 17, dim = 137, blocks = 3
nnz(A) = 65 + 0, nnz(ADA) = 393, nnz(L) = 207
it :      b*y      gap   delta  rate   t/tP*   t/tD*   feas cg cg   prec
 0 :              5.86E+00 0.000
 1 :  -1.21E-01 1.27E+00 0.000 0.2164 0.9000 0.9000   2.32  1  1  1.4E+00
 2 :  -3.20E-02 2.53E-01 0.000 0.1998 0.9000 0.9000   1.54  1  1  7.0E-01
 3 :  -5.95E-03 5.32E-02 0.000 0.2099 0.9000 0.9000   1.23  1  1  5.4E-01
 4 :  -3.86E-03 1.48E-02 0.000 0.2792 0.9000 0.9000   1.03  1  1  6.7E-01
 5 :  -5.46E-03 4.42E-03 0.000 0.2976 0.9000 0.9000   0.53  1  1  2.6E-01
 6 :  -7.75E-03 1.53E-03 0.000 0.3472 0.9000 0.9000   0.14  1  1  1.0E-01
 7 :  -1.09E-02 5.40E-04 0.000 0.3522 0.9000 0.9000   0.17  1  1  8.2E-03
 8 :  -1.38E-02 2.13E-04 0.000 0.3942 0.9000 0.9000   0.12  1  1  5.6E-04
 9 :  -1.74E-02 7.85E-05 0.000 0.3685 0.9000 0.9000   0.22  1  1  3.1E-04
10 :  -2.02E-02 3.39E-05 0.000 0.4321 0.9000 0.9000   0.17  1  1  2.1E-04
11 :  -2.38E-02 1.12E-05 0.000 0.3317 0.9000 0.9000   0.38  1  1  9.4E-05
12 :  -2.60E-02 4.43E-06 0.000 0.3936 0.9000 0.9000   0.45  1  1  5.0E-05
13 :  -2.78E-02 1.09E-06 0.000 0.2468 0.9000 0.9000   0.70  1  1  1.4E-05
14 :  -2.84E-02 2.39E-07 0.000 0.2183 0.9000 0.9000   0.85  1  1  3.4E-06
15 :  -2.86E-02 9.96E-09 0.000 0.0418 0.9900 0.9900   0.96  1  1  1.4E-07
16 :  -2.86E-02 1.00E-10 0.154 0.0101 0.9990 0.9990   0.99  1  1  1.5E-09
17 :  -2.86E-02 4.46E-12 0.067 0.0445 0.9900 0.9900   1.00  1  1  6.5E-11

iter seconds digits      c*x      b*y
 17      0.1   8.0 -2.8583945115e-02 -2.8583944835e-02
|Ax-b| = 4.4e-11, [Ay-c]_+ = 2.0E-11, |x|= 1.6e+02, |y|= 2.9e-01

Detailed timing (sec)
Pre      IPM      Post
4.124E-03 9.976E-02 4.848E-03
Max-norms: ||b||=1.000000e-01, ||c|| = 1,
Cholesky |add|=0, |skip| = 0, ||L.L|| = 218.956.
solution =
  yalmiptime: 3.3812e-02
  solvertime: 1.1207e-01
  info: 'No problems detected (SeDuMi-1.3)'
  problem: 0
  dimacs: [NaN NaN NaN NaN NaN NaN]

```

```

% Construct K and check closed loop stability
Klqr
Klqr =
    -9.8444e-01    1.7795e-01   -1.3843e+01   -2.4919e+00
    -1.7795e-01   -9.8404e-01   -2.4919e+00   -1.4741e+01
K = double(L) / double(X)
K =
    -9.8444e-01    1.7795e-01   -1.3843e+01   -2.4919e+00
    -1.7795e-01   -9.8404e-01   -2.4919e+00   -1.4741e+01
Acl = At + But*K;
eig(Acl)
ans =
    -7.1524e-02 + 8.2854e-02i
    -7.1524e-02 - 8.2854e-02i
    -7.1395e-02 + 5.7005e-02i
    -7.1395e-02 - 5.7005e-02i

diary off

```

Let's solve it again with the second form.

```
% MAE 280 B - Linear Control Design
% Mauricio de Oliveira
%
% LMI 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
```

```

% optimal LQR state feedback control (using u1 and u2)
n = size(At,1);
m = size(But,2);

% LQR solution
Cz = [eye(n); zeros(m,n)];
Dzu = [zeros(n,m); eye(m)];

[K,X,S] = lqr(At, But, Cz' * Cz, Dzu' * Dzu);
Klqr = - K
Klqr =
    -9.8444e-01    1.7795e-01   -1.3843e+01   -2.4919e+00
    -1.7795e-01   -9.8404e-01   -2.4919e+00   -1.4741e+01

% LMI solution (second form)
E = null([But' Dzu']);

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

% declare LMIs
LMI1 = E'*[At*X+X*At', X*Cz';...
          Cz*X, -eye(m+n)]*E;
LMI2 = [Z, Bwt'; Bwt, X];

LMI = set(LMI1 < 0) + set(LMI2 > 0);
options = sdpsettings('solver','sedumi');
solution = solvesdp(LMI,trace(W*Z),options)
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 = 13, order n = 15, dim = 101, blocks = 3
nnz(A) = 104 + 0, nnz(ADA) = 169, nnz(L) = 91
it :      b*y      gap    delta  rate  t/tP*  t/tD*   feas cg cg  prec
 0 :              4.81E+00 0.000
 1 :   -1.24E-01  1.05E+00 0.000 0.2188 0.9000 0.9000   2.16 1 1  1.5E+00
 2 :   -2.72E-02  2.21E-01 0.000 0.2096 0.9000 0.9000   1.72 1 1  6.9E-01
 3 :   -4.83E-03  4.56E-02 0.000 0.2065 0.9000 0.9000   1.28 1 1  5.2E-01
 4 :   -3.67E-03  1.25E-02 0.000 0.2735 0.9000 0.9000   1.02 1 1  5.0E-01
 5 :   -5.44E-03  3.61E-03 0.000 0.2894 0.9000 0.9000   0.42 1 1  2.0E-01
 6 :   -7.84E-03  1.16E-03 0.000 0.3227 0.9000 0.9000   0.10 1 1  5.9E-02
 7 :   -1.09E-02  4.06E-04 0.000 0.3486 0.9000 0.9000   0.14 1 1  8.7E-04
 8 :   -1.39E-02  1.57E-04 0.000 0.3876 0.9000 0.9000   0.09 1 1  5.7E-04
 9 :   -1.76E-02  5.67E-05 0.000 0.3607 0.9000 0.9000   0.21 1 1  3.1E-04
10 :   -2.05E-02  2.34E-05 0.000 0.4130 0.9000 0.9000   0.16 1 1  2.1E-04
11 :   -2.42E-02  7.55E-06 0.000 0.3222 0.9000 0.9000   0.38 1 1  9.0E-05
12 :   -2.63E-02  2.83E-06 0.000 0.3752 0.9000 0.9000   0.47 1 1  4.5E-05
13 :   -2.79E-02  6.59E-07 0.000 0.2328 0.9000 0.9000   0.72 1 1  1.2E-05
14 :   -2.85E-02  1.25E-07 0.000 0.1891 0.9000 0.9000   0.87 1 1  2.4E-06
15 :   -2.86E-02  4.64E-09 0.000 0.0372 0.9900 0.9900   0.97 1 1  9.2E-08
16 :   -2.86E-02  4.82E-11 0.167 0.0104 0.9990 0.9990   0.99 1 1  9.6E-10

iter seconds digits      c*x      b*y

```

```

16      0.1    6.7 -2.8583895204e-02 -2.8583889484e-02
|Ax-b| = 6.5e-10, [Ay-c]_+ = 3.1E-10, |x|= 1.6e+02, |y|= 2.9e-01

Detailed timing (sec)
      Pre      IPM      Post
3.138E-03    8.627E-02    1.016E-03
Max-norms: ||b||=1.000000e-01, ||c|| = 1,
Cholesky |add|=0, |skip| = 0, ||L.L|| = 235.364.
solution =
      yalmiptime: 5.7879e-02
      solvertime: 9.0980e-02
      info: 'No problems detected (SeDuMi-1.3)'
      problem: 0
      dimacs: [NaN NaN NaN NaN NaN NaN]

% Construct K and check closed loop stability
Y = inv(double(X));
QQ = [Y*At+At'*Y, Cz';...
      Cz, -eye(m+n)];
BB = [But'*Y Dzu'];
CC = [eye(n) zeros(n,m+n)];

mu = 1e8
mu =
    1000000000
max(eig(QQ - mu*BB'*BB))
ans =
    2.7429e-06
max(eig(QQ - mu*CC'*CC))
ans =
   -1.0000e+00

Phi = QQ - mu*BB'*BB - mu*CC'*CC;
XX = mu^2 * CC * inv(Phi) * BB';

% Check closed loop stability
Klqr
Klqr =
   -9.8444e-01    1.7795e-01   -1.3843e+01   -2.4919e+00
   -1.7795e-01   -9.8404e-01   -2.4919e+00   -1.4741e+01
Klmi = XX'
Klmi =
   -9.8444e-01    1.7795e-01   -1.3843e+01   -2.4919e+00
   -1.7795e-01   -9.8404e-01   -2.4919e+00   -1.4741e+01
Acl = At + But*Klmi;
eig(Acl)
ans =
   -7.1524e-02 + 8.2854e-02i
   -7.1524e-02 - 8.2854e-02i
   -7.1395e-02 + 5.7005e-02i
   -7.1395e-02 - 5.7005e-02i

diary off

```