Linear Systems

Linear Quadratic Optimal Control

Xu Chen

Motivation

state feedback control:

- allows to arbitrarily assign the closed-loop eigenvalues for a controllable system
- the eigenvalue assignment has been manual thus far
- performance is implicit: we assign eigenvalues to induce proper error convergence

linear quadratic (LQ) optimal regulation control, aka, LQ regulator (or LQR):

- no need to specify closed-loop poles
- performance is explicit: a performance index is defined ahead of time

1. Problem formulation

2. Solution to the infinite-horizon/stationary LQ problem

3. Solution to the finite-horizon LQ problem

4. From finite-horizon LQ to stationary LQ

Goal

Consider an n-dimensional state-space system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \;\; x(t_0) = x_0 \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^r$, and $y \in \mathbb{R}^m$.
LQ optimal control aims at minimizing the performance index

$$J = \frac{1}{2}x^T(t_f)Sx(t_f) + \frac{1}{2}\int_{t_0}^{t_f} \left(x^T(t)Qx(t) + u^T(t)Ru(t)\right) dt$$

- $S \succeq 0$, $Q \succeq 0$, $R \succ 0$: for a nonnegative cost and well-posed problem
- $\frac{1}{2}x^T(t_f)Sx(t_f)$ penalizes the deviation of x from the origin at t_f
- $x^T(t)Qx(t)$ $t \in (t_0, t_f)$ penalizes the transient
- often, $Q = C^TC \Rightarrow x^T(t)Qx(t) = y(t)^T y(t)$
- $u^T(t)Ru(t)$ penalizes large control efforts

Observations

$$J = \frac{1}{2}x^T(t_f)Sx(t_f) + \frac{1}{2}\int_{t_0}^{t_f} \left(x^T(t)Qx(t) + u^T(t)Ru(t)\right) dt$$

- when the control horizon is made to be infinitely long, i.e., $t_f \to \infty$, the problem reduces to the infinite-horizon LQ problem

$$J = \frac{1}{2}\int_{t_0}^{\infty} \left(x^T(t)Qx(t) + u^T(t)Ru(t)\right) dt$$

  - terminal cost is not needed, as it will turn out, that the control will have to drive x to the origin. Otherwise J will go unbounded.
  - often, we have

$$J = \frac{1}{2}\int_{0}^{\infty} \left(x^T(t)Qx(t) + u^T(t)Ru(t)\right) dt$$

Solution concept: infinite-horizon/stationary LQ

Consider the performance index

$$J = \frac{1}{2}\int_{t_0}^{\infty}\left(x(t)^T Qx(t) + u(t)^T Ru(t)\right)dt,\ \ Q = C^T C$$

with $\dot{x}(t) = Ax(t) + Bu(t)\,,\ x(t_0) = x_0$ and $R \succ 0$.

- recall when addressing $\overline{J} = \frac{1}{2}\int_0^{\infty} x^T(t)\,Qx(t)\,dt,\ \ \dot{x} = Ax$
- we defined $V(t) = \frac{1}{2}x^T(t)\,Px(t)\,,\ \ P = P^T$, such that

$$\begin{aligned}\overline{J} + V(\infty) - V(0) &= \frac{1}{2}\int_0^{\infty} x^T(t)\,Qx(t)\,dt + \int_0^{\infty}\dot{V}(t)\,dt \\ &= \frac{1}{2}\int_0^{\infty} x^T(t)\left(Q + A^T P + PA\right)x(t)\,dt\end{aligned}$$

- yielding $\overline{J}^0 = \frac{1}{2}x^T(0)\,P_+x(0)$ where P_+ comes from $A^T P + PA + Q = 0$, when the origin is asymptotically stable.

Solution of the infinite-horizon LQ

It turns out that for

$$J = \frac{1}{2}\int_{t_0}^{\infty} \left(x(t)^T Qx(t) + u(t)^T Ru(t) \right) dt, \;\; Q = C^T C$$

with $\dot{x}(t) = Ax(t) + Bu(t), \;\; x(t_0) = x_0$ and $R \succ 0$:

- if (A, B) is **controllable** (stabilizable) & (A, C) is **observable** (detectable)
- then the optimal control input is given by

$$u(t) = -R^{-1}B^T P_+ x(t)$$

- where $P_+ \left(= P_+^T\right)$ is the positive (semi)definite solution of the **algebraic Riccati equation** (ARE)

$$A^T P + PA - PBR^{-1}B^T P + Q = 0$$

- and the closed-loop system is **asymptotically stable**, with

$$J_{\min} = J^0 = \frac{1}{2}x(t_0)^T P_+ x(t_0)$$

Observations

- the control $u(t) = -R^{-1}B^TPx(t)$ is a state feedback law
- under the optimal control, the closed loop is given by $\dot{x} = Ax - BR^{-1}B^TPx = \underbrace{\left(A - BR^{-1}B^TP\right)}_{A_c} x$ and $J = \frac{1}{2}\int_{t_0}^{\infty}\left(x^TQx + u^TRu\right)dt = \frac{1}{2}\int_{t_0}^{\infty} x^T \underbrace{\left(Q + PBR^{-1}B^TP\right)}_{Q_c} xdt$
- for the above closed-loop system, the Lyapunov Eq. is

$$
\begin{aligned}
A_c^TP + PA_c &= -Q_c \\
\Leftrightarrow \left(A - BR^{-1}B^TP\right)^T P + P\left(A - BR^{-1}B^TP\right) &= -Q - PBR^{-1}B^TP \\
\Leftrightarrow A^TP + PA - PBR^{-1}B^TP &= -Q \Leftarrow \text{the ARE!}
\end{aligned}
$$

- when the ARE solution P_+ is positive definite, $\frac{1}{2}x^TP_+x$ is a Lyapunov function for the closed-loop system

Observations

- Lyapunov Eq. and the ARE:

Cost	$\overline{J} = \frac{1}{2}\int_0^\infty x^T Q x dt$	$J = \frac{1}{2}\int_{t_0}^\infty \left(x^T Q x + u^T R u\right) dt$
Syst. dynamics	$\dot{x} = Ax$	$\dot{x} = Ax + Bu$ (A, B) controllable/stabilizable (A, C) observable/detectable
Key Eq.	$A^T P + PA + Q = 0$	$A^T P + PA - PBR^{-1}B^T P + Q = 0$
Optimal control	N/A	$u(t) = -R^{-1}B^T P_+ x(t)$
Opt. cost	$\overline{J}^0 = \frac{1}{2}x^T(0)\, P_+ x(0)$	$J^0 = \frac{1}{2}x(t_0)^T\, P_+ x(t_0)$

- the guaranteed closed-loop stability is an attractive feature
- more nice properties will show up later

Example

Item 4: From the algebraic Riccati equation, we have

$$A^T P_+ + P_+ A - P_+ B R^{-1} B^T P_+ + C^T C + \lambda P_+ - \lambda P_+ = 0,$$
$$\Rightarrow (\lambda I_n + A^T) P_+ - P_+ (\lambda I_n - A) - P_+ B R^{-1} B^T P_+ + C^T C = 0,$$
$$\Rightarrow - P_+ \left[\lambda_n - (A - BR^{-1}B^T P_+)\right] + (\lambda I_n + A^T) P_+ + C^T C = 0$$

Let f_i be the eigenvector of $A - BR^{-1}B^T P_+$ associated with the eigenvalue λ_i. Set $\lambda = \lambda_i$ in the last equality and multiply f_i from the right. Then, the first term vanishes after multiplication:

$$(\lambda_i I_n + A^T) P_+ f_i + C^T C f_i = 0.$$

Then,

$$H \begin{bmatrix} f_i \\ P_+ f_i \end{bmatrix} = \begin{bmatrix} A f_i - BR^{-1}B^T P_+ f_i \\ -C^T C f_1 - A^T P_+ f_i \end{bmatrix} = \lambda_i \begin{bmatrix} f_i \\ P_+ f_i \end{bmatrix}.$$

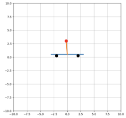
This implies that $\begin{bmatrix} f_i \\ P_+ f_i \end{bmatrix}$ is the eigenvector of H associated with a stable eigenvalue λ_i. (iv) follows from this fact. □

13.4.4 Example: Inverted Pendulum on a Cart

The inverted pendulum on a cart model is widely used and applied to many systems we see regularly. It is a classical problem in dynamics and is used extensively in control theory for designing controllers. Applications include rocket balancing, segway and hoverboards, vertical robots, to name a few.

The derivation of the equations of motion is available at this link: https://ctms.engin.umich.edu/CTMS/index.php?example=InvertedPendulum&section=SystemModeling.

An animated version of the example in Python is provided at https://github.com/macs-lab/Python-Controls-Visualization/tree/main



The system has two equations of motion:

$$(M+m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta = F, \qquad (13.20)$$

$$(I+ml^2)\ddot{\theta} + mgl\sin\theta = -ml\ddot{x}\cos\theta, \qquad (13.21)$$

where I is the moment of inertia of the pendulum, m is the mass of the pendulum, M is the mass of the cart, l is the length between the pendulum center of mass to the mounting joint, and b is the damping of the cart in the horizontal movement direction. Substituting for $\ddot{\theta}$ in 13.20 from 13.21 gives:

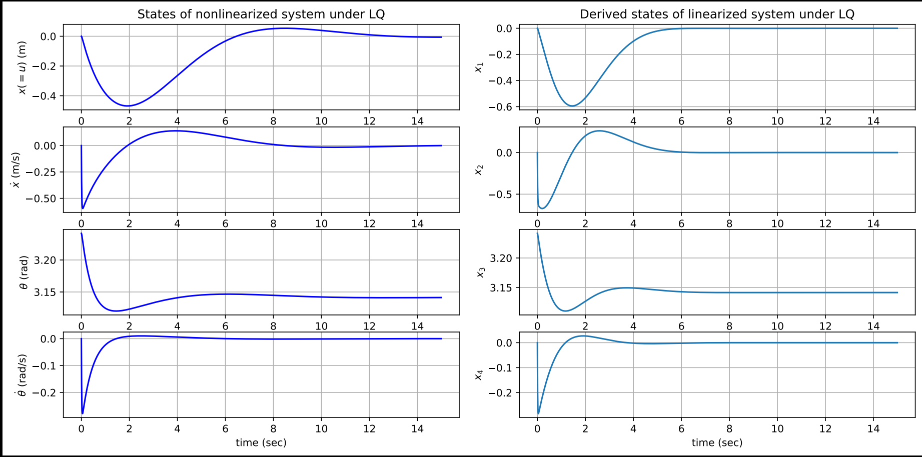
$$\ddot{x} = \frac{F(I+ml^2) - b\dot{x}(I+ml^2) - m^2l^2 g\sin\theta\cos\theta + ml\dot{\theta}^2\sin\theta(I+ml^2)}{(I+ml^2)(M+m) - m^2l^2cos^2(\theta)}, \qquad (13.22)$$

$$\ddot{\theta} = \frac{(M+m)mgl\sin\theta + mlb\dot{x}\cos\theta - m^2l^2\dot{\theta}^2\cos\theta\sin\theta - mlF\cos\theta}{(M+m)(I+ml^2) - m^2l^2\cos^2\theta}. \qquad (13.23)$$

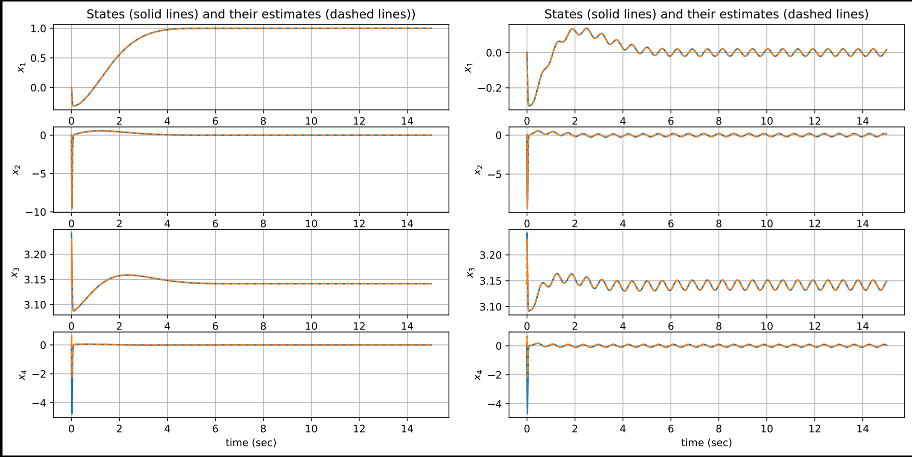
The system model has four states, which give the state vector:

$$X = \begin{bmatrix} x \\ \dot{x} \\ \theta \end{bmatrix}.$$

Example


States of nonlinearized system under LQ
Derived states of linearized system under LQ
$x(=u)$ (m)
$\dot{x}$ (m/s)
θ (rad)
$\dot{\theta}$ (rad/s)
x_1
x_2
x_3
x_4
time (sec)
time (sec)

LQ with State Feedback



Example: Stationary LQR of a pure inertia system

- Consider

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \; J = \frac{1}{2} \int_0^\infty \left(x^T \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} x + Ru^2 \right) dt, \; R > 0$$

- the ARE is

$$0 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} P + P \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} - P \begin{bmatrix} 0 \\ 1 \end{bmatrix} \frac{1}{R} \begin{bmatrix} 0 & 1 \end{bmatrix} P \Rightarrow P_+ = \begin{bmatrix} \sqrt{2} R^{1/4} & R^{1/2} \\ R^{1/2} & \sqrt{2} R^{3/4} \end{bmatrix}$$

- the closed-loop A matrix can be computed to be

$$A_c = A - BR^{-1} B^T P_+ = \begin{bmatrix} 0 & 1 \\ -R^{-1/2} & -\sqrt{2} R^{-1/4} \end{bmatrix}$$

- $\Rightarrow$ closed-loop eigenvalues:

$$\lambda_{1,2} = -\frac{1}{\sqrt{2} R^{1/4}} \pm \frac{1}{\sqrt{2} R^{1/4}} j$$

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \quad J = \frac{1}{2}\int_0^{\infty} \left(x^T \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} x + Ru^2 \right) dt$$

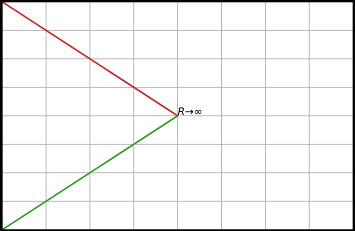


Figure: Eigenvalue $\lambda_{1,2} = -\frac{1}{\sqrt{2}R^{1/4}} \pm \frac{1}{\sqrt{2}R^{1/4}} j$ evolution (root locus)

- $R \uparrow$ (more penalty on the control input) $\Rightarrow \lambda_{1,2}$ move closer to the origin $\Rightarrow$ slower state convergence to zero
- $R \downarrow$ (allow for large control efforts) $\Rightarrow \lambda_{1,2}$ move further to the left of the complex plane $\Rightarrow$ faster speed of closed-loop dynamics

MATLAB commands

- *care*: solves the ARE for a continuous-time system:

$$[P, \Lambda, K] = \text{care}\left(A, B, C^T C, R\right)$$

where $K = R^{-1}B^T P$ and Λ is a diagonal matrix with the closed-loop eigenvalues, i.e., the eigenvalues of $A - BK$, in the diagonal entries.

- *lqr* and *lqry*: provide the LQ regulator with

$$[K, P, \Lambda] = \text{lqr}\left(A, B, C^T C, R\right)$$
$$[K, P, \Lambda] = \text{lqry}\left(\text{sys}, Q_y, R\right)$$

where *sys* is defined by $\dot{x} = Ax + Bu, \;\; y = Cx + Du$, and

$$J = \frac{1}{2}\int_0^{\infty} \left(y^T Q_y y + u^T Ru\right) dt$$

1. Problem formulation

2. Solution to the infinite-horizon/stationary LQ problem

3. Solution to the finite-horizon LQ problem

4. From finite-horizon LQ to stationary LQ

Solution to the finite-horizon LQ

Consider the performance index

$$J = \frac{1}{2}x^T(t_f)Sx(t_f) + \frac{1}{2}\int_{t_0}^{t_f} \left(x^T(t)Qx(t) + u^T(t)Ru(t)\right) dt$$

with $\dot{x} = Ax + Bu,\ x(t_0) = x_0,\ S \succeq 0,\ R \succ 0$, and $Q = C^T C$.

- do a similar Lyapunov construction: $V(t) \triangleq \frac{1}{2}x^T(t)P(t)x(t)$
- then

$$\begin{aligned}\frac{d}{dt}V(t) &= \frac{1}{2}\dot{x}^T(t)P(t)x(t) + \frac{1}{2}x^T(t)\dot{P}(t)x(t) + \frac{1}{2}x^T(t)P(t)\dot{x}(t)\\ &= \frac{1}{2}(Ax+Bu)^T Px + \frac{1}{2}x^T\frac{dP}{dt}x + \frac{1}{2}x^T P(Ax+Bu)\\ &= \frac{1}{2}\left\{x^T(t)\left(A^TP + \frac{dP}{dt} + PA\right)x(t) + u^TB^TPx + x^TPBu\right\}\end{aligned}$$

Solution to the finite-horizon LQ

with $\frac{d}{dt} V(t)$ from the last slide, we have

$$V(t_f) - V(t_0) = \int_{t_0}^{t_f} \dot{V} \, dt$$
$$= \frac{1}{2} \int_{t_0}^{t_f} \left(x^T \left(A^T P + PA + \frac{dP}{dt} \right) x + u^T B^T P x + x^T P B u \right) dt$$

- adding
$$J = \frac{1}{2} x^T(t_f) S x(t_f) + \frac{1}{2} \int_{t_0}^{t_f} \left(x^T(t) Q x(t) + u^T(t) R u(t) \right) dt$$
to both sides yields

$$J + V(t_f) - V(t_0) = \frac{1}{2} x^T(t_f) S x(t_f) +$$
$$\frac{1}{2} \int_{t_0}^{t_f} \left(x^T \left(A^T P + PA + Q + \frac{dP}{dt} \right) x + \underbrace{u^T B^T P x + x^T P B u}_{\text{products of } x \text{ and } u} + \underbrace{u^T R u}_{\text{quadratic}} \right) dt$$

Solution to the finite-horizon LQ

- "complete the squares" in $\underbrace{u^T B^T P x + x^T P B u}_{\text{products of } x \text{ and } u} + \underbrace{u^T R u}_{\text{quadratic}}$:

$$u^T B^T P x + x^T P B u + u^T R u \overset{\text{scalar case}}{=} R u^2 + 2 u B P x$$

$$= R u^2 + 2 \left(x P B R^{-1/2} \right) \underbrace{R^{1/2} u}_{\sqrt{R u^2}} + \left(R^{-1/2} B P x \right)^2 - \left(R^{-1/2} B P x \right)^2$$

$$= \left(R^{1/2} u + R^{-1/2} B P x \right)^2 - \left(R^{-1/2} B P x \right)^2$$

- extending the concept to the general vector case:

$$u^T B^T P x + x^T P B u + u^T R u = \| R^{\frac{1}{2}} u + R^{\frac{-1}{2}} B^T P x \|_2^2 - x^T P B R^{-1} B^T P x$$

Solution to the finite-horizon LQ

$$J + V(t_f) - V(t_0) = \frac{1}{2}x^T(t_f)Sx(t_f) +$$

$$\frac{1}{2}\int_{t_0}^{t_f}\left(x^T\left(A^TP + PA + Q + \frac{dP}{dt}\right)x + u^TB^TPx + x^TPBu + u^TRu\right)dt$$

⇓"completing the squares"

$$J + \underline{\frac{1}{2}x^T(t_f)P(t_f)x(t_f)} - \frac{1}{2}x^T(t_0)P(t_0)x(t_0) = \underline{\frac{1}{2}x^T(t_f)Sx(t_f)} +$$

$$\frac{1}{2}\int_{t_0}^{t_f}\left(x^T\underline{\underline{\left(\frac{dP}{dt} + A^TP + PA + Q - PBR^{-1}B^TP\right)}}x + \|\underline{\underline{\underline{R^{\frac{1}{2}}u + R^{\frac{-1}{2}}B^TPx}}}\|_2^2\right)dt$$

- the best that the control can do in minimizing the cost is to have

$$u(t) \underline{\underline{\underline{= -K(t)x(t) = -R^{-1}B^TP(t)x(t)}}}$$

$$-\frac{dP}{dt} \underline{\underline{= A^TP + PA - PBR^{-1}B^TP + Q}}, \quad \underline{P(t_f) = S}$$

to yield the optimal cost $J^0 = \frac{1}{2}x_0^TP(t_0)x_0$

Observations

$$u(t) = -K(t)\,x(t) = -R^{-1}B^T P(t)x(t) \qquad \text{optimal state feedback control}$$

$$-\frac{dP}{dt} = A^T P + PA - PBR^{-1}B^T P + Q,\ \ P(t_f) = S \qquad \text{the Riccati differential equation}$$

- boundary condition of the Riccati equation is given at the final time t_f $\Rightarrow$ the equation must be integrated backward in time
- *backward* integration of

$$-\frac{dP}{dt} = A^T P + PA + Q - PBR^{-1}B^T P,\ \ P(t_f) = S$$

  is equivalent to the *forward* integration of

$$\frac{dP^*}{dt} = A^T P^* + P^* A + Q - P^* BR^{-1}B^T P^*,\ \ P^*(0) = S \qquad (2)$$

  by letting $P(t) = P^*(t_f - t)$
- Eq. (2) can be solved by numerical integration

Observations

$$J = \frac{1}{2}x^T(t_f)Sx(t_f) + \frac{1}{2}\int_{t_0}^{t_f} \left(x^T(t)Qx(t) + u^T(t)Ru(t)\right) dt$$

$$J^0 = \frac{1}{2}x_0^T P(t_0)x_0$$

- the minimum value J^0 is a function of the initial state $x(t_0)$
- J (and hence J^0) is nonnegative $\Rightarrow P(t_0)$ is at least positive semidefinite
- t_0 can be taken anywhere in $(0, t_f) \Rightarrow P(t)$ is at least positive semidefinite for any t
- the state feedback law is time varying because of $P(t)$

Example: LQR of a pure inertia system

Consider

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \;\; J = \frac{1}{2} x^T(t_f) S x(t_f) + \frac{1}{2} \int_0^{t_f} \left(x^T Q x + R u^2 \right) dt$$

where $S = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $Q = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$, $R > 0$

- we let $P(t) = P^*(t_f - t)$ and solve

$$\frac{dP^*}{dt} = A^T P^* + P^* A + Q - P^* B R^{-1} B^T P^*, \;\; P^*(0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\Leftrightarrow \frac{dP^*}{dt} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} P^* + P^* \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} - P^* \begin{bmatrix} 0 \\ 1 \end{bmatrix} \frac{1}{R} \begin{bmatrix} 0 & 1 \end{bmatrix} P^*$$

- letting

$$P^* = \begin{bmatrix} p_{11}^* & p_{12}^* \\ p_{12}^* & p_{22}^* \end{bmatrix} \Rightarrow \begin{cases} \frac{d}{dt} p_{11}^* = 1 - \frac{1}{R} \left(p_{12}^*\right)^2 & p_{11}^*(0) = 1 \\ \frac{d}{dt} p_{12}^* = p_{11}^* - \frac{1}{R} p_{12}^* p_{22}^* & p_{12}^*(0) = 0 \\ \frac{d}{dt} p_{22}^* = 2 p_{12}^* - \frac{1}{R} \left(p_{22}^*\right)^2 & p_{22}^*(0) = 1 \end{cases}$$

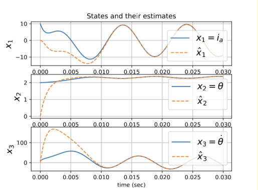
Example: LQR of a pure inertia system

Example 12.2.1 (Motor Control) To see an example performance of the observer design, we apply the algorithm to a motor system and program Equation 12.10. Here, the three states are the current in the motor electronics, the angular position and angular velocity of the motor. Only the angular position is directly measured on the output side. We compute first the plant eigenvalues, then check observability and place the observer eigenvalues.

```
% observer/motorobs.m
% State observer design for motion control in MATLAB
%% Continuous-time system model
% motor parameters
L = 1e-3; R = 1; J = 5e-5; B = 1e-4; K = 0.1;

% state-space model
A = [-R/L, 0, -K/L; 0, 0, 1; K/J, 0, -B/J];
B = [1/L; 0; 0];
C = [0, 1, 0];
D = [0];

% check original eigenvalues
eig(A)
%% Observer design
% check observability
O = obsv(A,C);
rank(O)

% desired poles for the observer
pole_des = [-500+250j, -500-250j, -1000];

% design observer by placing poles of A-LC
Lt = place(A.',C.',pole_des);
L  = Lt.'

% check poles of estimator-error dynamics
est_poles = eig(A - L*C)

%% Simulation
% define augmented system to run the simulation
Aaug = [A, zeros(3,3); L*C, A-L*C];
Baug = [B;B];
Caug = [C, zeros(1,3)];
Daug = 0;
sys = ss(Aaug,Baug,Caug,Daug);

% define intial conditions
x0 = [10, 2, 10]'; xhat0 = [0, 0, 0]'; X0 = [x0; xhat0];

% define simulink parameters
Tend = 0.03; % simulation end time
amplitude = 10; % sin wave input amplitude
initpha = 0; % initial phase
freq = 600; % sin wave freq (rad/s)
t = 0:1e-4:Tend;

u = amplitude*sin(freq*t+initpha);

[Y,T,X] = lsim(sys,u,t,X0);
```

```
x0 = np.array([10, 2, 10]); xhat0 = np.array([0, 0, 0]); X0 =
↪  np.array([x0, xhat0]).reshape((6, 1))

Tend = 0.03; amplitude = 10; initpha = 0; freq = 600
t = np.arange(0, Tend, 1e-4)

u = amplitude * np.sin(freq * t + initpha)

[Y, T, X] = ct.lsim(sys, u, t, X0)

plt.figure()
plt.subplot(3, 1, 1)
plt.plot(t, X[:, 0], t, X[:, 3], '--', linewidth=1.5)
plt.xlabel('time (sec)')
plt.legend(['$x_1 = i_a$', '$\hat x_1$'], fontsize=16)
plt.grid()
plt.ylabel('$x_1$', fontsize=16)
plt.title('States and their estimates')
plt.subplot(3, 1, 2)
plt.plot(t, X[:, 1], t, X[:, 4], '--', linewidth=1.5)
plt.xlabel('time (sec)')
plt.legend(['$x_2 = \\theta$', '$\hat x_2$'], fontsize=16)
plt.grid()
plt.ylabel('$x_2$', fontsize=16)
plt.subplot(3, 1, 3)
plt.plot(t, X[:, 2], t, X[:, 5], '--', linewidth=1.5)
plt.xlabel('time (sec)')
plt.legend(['$x_3 = \dot{\\theta}$', '$\hat x_3$'], fontsize=16)
plt.grid()
plt.ylabel('$x_3$', fontsize=16)
plt.show()
```

From the generated result below, we see that despite the initial error between the true states and the estimated states, the estimation errors quickly converge to zero for all the three states after about 0.01 second. Try modify the observer eigenvalues and see how they affect the convergence.



Example: LQR of a pure inertia system: analysis

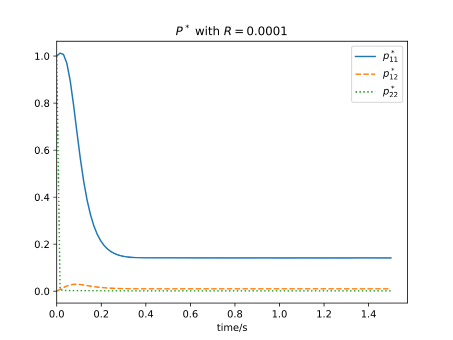


Figure: LQ example: $P^*(0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $P(t) = P^*(t_f - t)$

- if the final time t_f is large, $P^*(t)$ forward converges to a stationary value
- i.e., $P(t)$ backward converges to a stationary value at $P(0)$

Example: LQR of a pure inertia system: analysis

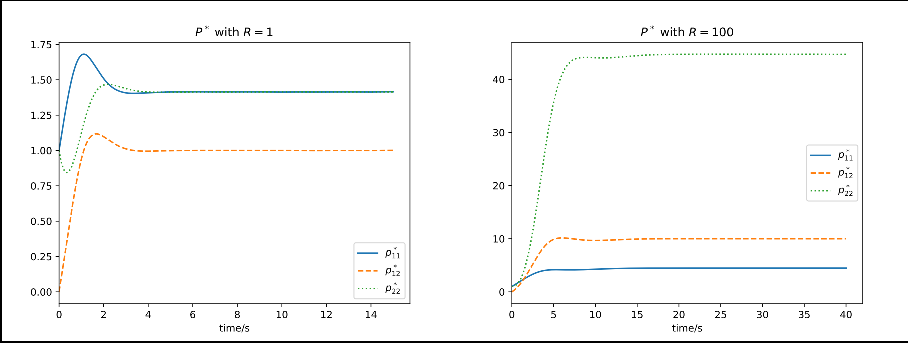
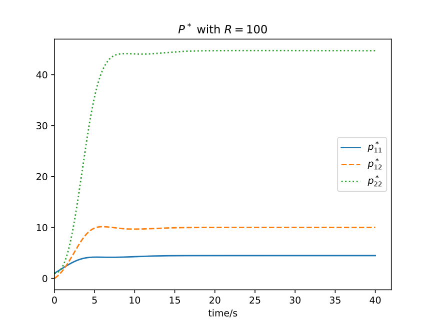


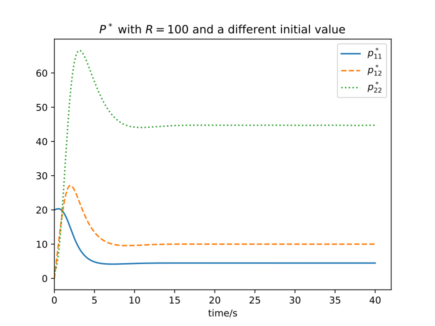
Figure: LQ example with different penalties on control. $P^*(0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

- a larger R results in a longer transient
- i.e., a larger penalty on the control input yields a longer time to settle

Example: LQR of a pure inertia system: analysis



(a) $P^*(0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$



(b) $P^*(0) = \begin{bmatrix} 20 & 0 \\ 0 & 2 \end{bmatrix}$

Figure: LQ with different boundary values in Riccati difference Eq.

- for the same R, the initial value $P(t_f) = S$ becomes irrelevant

1. Problem formulation

2. Solution to the infinite-horizon/stationary LQ problem

3. Solution to the finite-horizon LQ problem

4. From finite-horizon LQ to stationary LQ

From LQ to stationary LQ

- the ARE and the Riccati differential Eq.:

	Stationary LQ	LQ
Cost	$J = \frac{1}{2}\int_{t_0}^{\infty} (x^T Q x + u^T R u)\, dt$	$J = \frac{1}{2}x^T(t_f)Sx(t_f) + \frac{1}{2}\int_{t_0}^{t_f} (x^T(t)Qx(t) + u^T(t)Ru(t))\, dt$
Syst.	$\dot{x} = Ax + Bu$ (A, B) controllable/stabilizable (A, C) observable/detectable	$\dot{x} = Ax + Bu$
Key Eq.	$A^T P + PA - PBR^{-1}B^T P + Q = 0$	$-\frac{dP}{dt} = A^T P + PA - PBR^{-1}B^T P + Q$ $P(t_f) = S$
Opt. control	$u(t) = -R^{-1}B^T P_+ x(t)$	$u(t) = -R^{-1}B^T P(t)x(t)$
Opt. cost	$J^0 = \frac{1}{2}x_0^T P_+ x_0$	$J^0 = \frac{1}{2}x_0^T P(t_0)x_0$

- in the example, we see that P in the Riccati differential Eq. converges to a stationary value given sufficient time
- when $t_f \to \infty$, the Riccati differential Eq. converges to ARE and the LQ becomes the stationary LQ, under two conditions that we now discuss in details:
  - (A, B) is controllable/stabilizable
  - (A, C) is observable/detectable

Need for controllability/stabilizability

if (A, B) is controllable or stabilizable, then $P(t)$ is guaranteed to converge to a bounded and stationary value

- for uncontrollable or unstabilizable systems, there can be unstable uncontrollable modes that cause J to be unbounded
- then if $J^0 = \frac{1}{2} x_0^T P(0) x_0$ is unbounded, we will have $||P(0)|| = \infty$
- e.g.: $\dot{x} = x + 0 \cdot u$, $x(0) = 1$, $Q = 1$ and R be any positive value
  - system is uncontrollable and the uncontrollable mode is unstable
  - $x(t)$ will keep increasing to infinity
  - $\Rightarrow J = \frac{1}{2} \int_0^\infty \left(x^T Q x + u^T R u \right) dt$ unbounded regardless of $u(t)$
  - in this case, the Riccati equation is

$$-\frac{dP}{dt} = P + P + 1 = 2P + 1 \Leftrightarrow \frac{dP^*}{dt} = 2P^* + 1$$

forward integration of P^* (backward integration of P), will drive $P^*(\infty)$ and $P(0)$ to infinity

Need for observability/detectability

if (A, C) is observable or detectable, the optimal state feedback control system will be asymptotically stable

- *intuition*: if the system is observable, $y = Cx$ will relate to all states $\Rightarrow$ regulating $x^T Q x = x^T C^T C x$ will regulate all states
- *formally*: if (A, C) is observable (detectable), the solution of the Riccati equation will converge to a positive (semi)definite value P_+ (proof in course notes)

Additional excellent properties of stationary LQ

- we know stationary LQR yields guaranteed closed-loop stability for controllable (stabilizable) and observable (detectable) systems

It turns out that LQ regulators with full state feedback has excellent additional properties of:

- at least a 60 degree phase margin
- infinite gain margin
- stability is guaranteed up to a 50% reduction in the gain

Applications and practice

choosing R and Q:

- if there is not a good idea for the structure for Q and R, start with diagonal matrices;
- gain an idea of the magnitude of each state variable and input variable
- call them $x_{i,\max}$ $(i = 1, \dots, n)$ and $u_{i,\max}$ $(i = 1, \dots, r)$
- make the diagonal elements of Q and R inversely proportional to $||x_{i,\max}||^2$ and $||u_{i,\max}||^2$, respectively.