EN530.603 Applied Optimal Control
Lecture 6: Linear-Quadratic Regulator Basics
September 30, 2020

Lecturer: Marin Kobilarov

已知, $P(t)$ 未知 (可能有 可能没有)

$P_f = P(t_f)$

1 Continuous Linear Quadratic Regulator (LQR)

1.1 Finite-Time LQR

$\psi = 0$

无 terminal constraint.

terminal cost

Consider a system with dynamics

☆ $\phi = \frac{1}{2} x^T P_f x$

$$f(x) = \quad \dot{x} = Ax + Bu \quad \text{linear control}$$

which must optimally reach the origin, a task specified by the cost function

$L = \frac{1}{2} x^T Q x + \frac{1}{2} u^T R u$

P_f 和 R 是 symm and p.s.d

$$J = \frac{1}{2} x^T(t_f) P_f x(t_f) + \int_{t_0}^{t_f} \frac{1}{2} \left[x(t)^T Q(t) x(t) + u(t)^T R(t) u(t) \right] \mathrm{d}t,$$

p.s.d pd

$x^T Q x$ 与每点到原点距离的 square 有关

where P_f and Q are symmetric positive semi-definite matrices and R is a symmetric positive definite matrix. Applying the optimality conditions using the Hamiltonian

$\nabla H = 0$ $\quad \nabla_u H = 0$ $\quad \nabla_\lambda H = 0$ $\quad \nabla_x H = 0$

free terminal

$$H = \frac{1}{2}\left(x^T Q x + u^T R u\right) + \lambda^T \underbrace{(Ax + Bu)}_{f(x,u)}$$

$$\begin{aligned}
&\dot{x} = f(x, u, t) \\
&\dot{\lambda} = -\nabla_x H = -\nabla_x f \cdot \lambda - \nabla_x L, \\
&\nabla_u H = \nabla_u f \cdot \lambda + \nabla_u L = 0, \\
&\lambda(t_f) = \nabla_x \phi(x(t_f), t_f) + \nabla_x \psi(x(t_f), t_f) \cdot \nu, \\
&\left(\partial_t \phi + \nu^T \partial_t \psi + L + \lambda^T f\right)_{t = t_f} = 0,
\end{aligned}$$

we obtain

$$\dot{\lambda} = -Qx - A^T \lambda,$$

while the control is computed according to

$$Ru + B^T \lambda = 0 \quad \Rightarrow \quad u = -R^{-1} B^T \lambda$$

and transversality conditions become

$$\dot{x} = Ax + Bu = Ax - BR^{-1}B^T\lambda$$

$$= \nabla_x \phi(t_f)$$

$$\lambda(t_f) = P_f x(t_f). \tag{2}$$

The optimal state $(x(t), \lambda(t))$ then evolves according to the EL equations

$$\begin{pmatrix} \dot{x} \\ \dot{\lambda} \end{pmatrix} = \underbrace{\begin{pmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{pmatrix}}_{=M} \begin{pmatrix} x \\ \lambda \end{pmatrix}.$$

with a final boundary condition (??). When A, B, Q, R are constant this can be solved for given $y(0) = (x(0), \lambda(0))$ as $y(t) = e^{tM} y(0)$ for instance by computing the Laplace transform of $(sI - M)^{-1}$. This becomes difficult in high-dimensions, but thankfully there's an easier way.

Kalman showed that the multipliers $\lambda(t)$ are in fact linear function of the states, i.e.

$$\lambda(t) = P(t)x(t). \quad \lambda(t) \text{ 与 } x(t) \text{ 成线性关系}$$

$$u = -R^{-1}B^T\lambda \qquad \lambda(t) = P(t)x(t).$$

Hence the control can be written according to

$$u = -R^{-1}B^T Px \equiv Kx.$$

The matrix P can now be computed by noting that

$$\dot{\lambda} = -Qx - A^T\lambda, \ \dot{\lambda} = \dot{P}x + P\dot{x} \qquad \begin{aligned} \lambda(t) &= P(t)x(t). \\ \dot{x} &= Ax + Bu \end{aligned}$$

which is equivalent to

$$-Qx - A^T Px = \dot{P}x + PAx - PBR^{-1}B^T Px.$$

A solution exists [redacted] if we can find a P which satisfies 若存在，就能确定 $\lambda(t)$ 因为Q.R是p.s.d.和p.d 的τix

$$\dot{P} = -A^T P - PA + PBR^{-1}B^T P - Q, \qquad P(t_f) = P_f$$

This is called the *Riccati ODE* and is integrated from t_f to t_0 backwards in time. After $P(t)$ is found the control is updated according to Care function in Matlab (反向积分)

$$u(t) = -R^{-1}B^T P(t)x(t).$$

Pf已知，可倒推出 P(t) 从而 get u(t)

u=kx → Constant 会自动向origin靠近 不需人为输

Note that it turns out that the optimal control u is in a linear feedback form, i.e. it is a linear function of the state x. This means that we have obtained not only a single optimal control signal from the start state $x(t_0)$ but also an optimal feedback controller from any state $x(t)$ for $t > t_0$. Therefore, we have completely eliminated the need for an additional controller to physically bring the system to the equilibrium state $x = 0$. Furthermore, when the system deviates from the initially computed path e.g. due to disturbances, the same $P(t)$ computed once in the beginning (at time $t = t_0$) can be used from the perturbed state $x(t)$.

U=-R⁻¹BᵀP(t)X. X变了P(t)可沿用 因为P与X无关. u会自动变化

Stability. In this context it is instructive to study the stability of the optimal control regarded as a feedback controller $u = Kx$. The closed-loop matrix is

$$A + BK = A - BR^{-1}B^T P$$

and one should be able to verify that the real parts of its eigenvalues are negative, i.e. that

$$\text{Real}[\texttt{eig}(A + BK)] < 0.$$

Hurtwiz

会越来越接近 origin 但tf可以是 ∞ 通常设成一个定值.

Here we have assumed that the system (A, B) is controllable.

Example 1. Linear-quadratic problem. Consider the system $\dot{x}(t) = x(t) + u(t)$ with initial condition $x(0) = x_0$ and quadratic cost functional

$$J = \frac{1}{2}\int_0^{t_f} x(t)^2 + u(t)^2 \mathrm{d}t$$

We have $A = B = Q = R = 1$ and the Riccati equation becomes

$$\dot{P} = -A^T P - PA + PBR^{-1}B^T P - Q, \qquad \dot{P} = -2P + P^2 - 1,$$

tf不知道只能多试几次

with final condition $P(t_f) = 0$. The solution can be obtained analytically and is

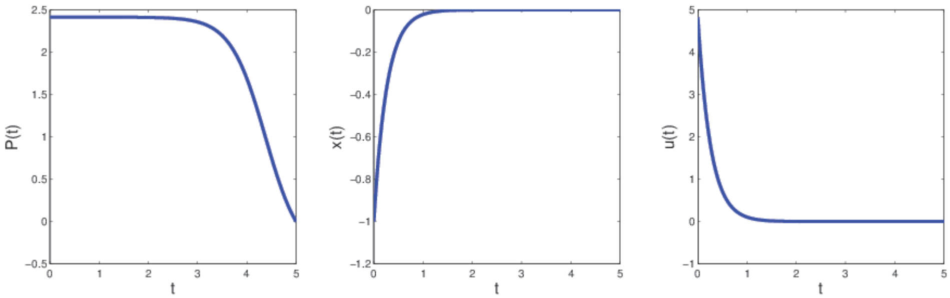
$$P(t) = 1 - \sqrt{2}\tanh(\sqrt{2}(t - tf + (\sqrt{2}\text{atanh}(\sqrt{2}/2))/2))$$

The gain computed with $t_f = 5$ is given below

$\dot{x} = Ax + Bu$

$\uparrow kx$

n



Note: one standard way to get the solution above is to set $P(t) = -\frac{\dot{b}(t)}{b(t)}$ for some $b(t)$:

这样使 $\dot{P} = -\frac{\ddot{b}}{b} + P^2$

$$\dot{P} = -\frac{\ddot{b}}{b} + \frac{\dot{b}^2}{b^2} = -\frac{\ddot{b}}{b} + k^2 \Rightarrow \ddot{b} = -2\dot{b} + b$$

$\dot{P} = -2P + P^2 - 1,$

$-\frac{\ddot{b}}{b} = -2\frac{\dot{b}}{b} - 1$

The solution of the second order linear ODE

t_f任意选的

$$\ddot{b} = -2\dot{b} + b, \qquad t < t_f, \tag{3}$$

$$\dot{b}(t_f) = 0, \;\; b(t_f) = 1 \tag{4}$$

is then computed from which we find $P(t)$.

注意: $u = -R^{-1}B^T P x = kx$

在t_f时 $u=0$, 所以 $P = 0 \Rightarrow \dot{b}(t_f) = 0$, $b(t_f)$可以是任意值

Example 2. 2-dim Linear-quadratic problem. Consider the dynamics

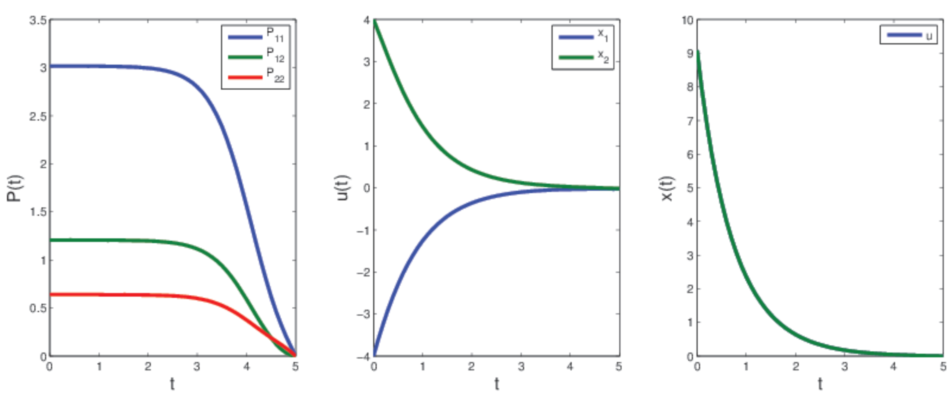
$$\dot{x}_1 = x_2, \tag{5}$$

$$\dot{x}_2 = 2x_1 - x_2 + u \tag{6}$$

and cost function

$$J = \frac{1}{2}\int_0^{t_f} [x_1^2 + \frac{1}{2}x_2^2 + \frac{1}{4}u^2]dt$$

We integrate the Riccati ODE numerically using $t_f = 5$ and start state $x(0) = (-4, 4)$. The following matrix $P(t)$, inputs $u(t)$ and state histories $x(t)$ are obtained:



$$\frac{d}{dt}\left(\frac{1}{2}x^T P x\right) = x^T P \dot{x} + x^T \dot{P} x = x^T P(Ax + BKx) + x^T \dot{P} x$$

$$\dot{P} = -A^T P - PA + PBR^{-1}B^T P - Q, \qquad u = -R^{-1}B^T P x \equiv Kx.$$

Optimal Cost. Assume the system is at state $x(t)$. We can compute the resulting optimal cost from time t to time t_f as follows:

$$\begin{aligned}
J(t) &= \frac{1}{2}x^T(t_f)P_f x(t_f) + \int_t^{t_f} \frac{1}{2}\left[x^T Q x + u^T R u\right] \mathrm{d}t, \\
&= \frac{1}{2}x^T(t_f)P_f x(t_f) + \int_t^{t_f} \frac{1}{2}\left[x(t)^T(Q + K^T R K)x\right] \mathrm{d}t, \\
&= \frac{1}{2}x^T(t_f)P_f x(t_f) - \int_t^{t_f} \frac{d}{dt}\left(\frac{1}{2}x^T P x\right) \mathrm{d}t = \frac{1}{2}x^T(t)P(t)x(t)
\end{aligned}$$

This will be important for several reasons: stability, cost-to-go,etc...

$J(t) = \frac{1}{2}x^T(t)P(t)x(t)$

optimal cost to go

the cost from now to the end.

1.2 Infinite-Time LQR

Consider the state equations $\dot{x} = Ax + Bu$ with cost function

$$J = \int_{t_0}^{\infty} \frac{1}{2}\left[x(t)^T Q(t)x(t) + u(t)^T R(t)u(t)\right] \mathrm{d}t,$$

The Riccati ODE has the same form but at $t = \infty$ reaches the stationary value $\Rightarrow \dot{P} = 0$

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

which called the *algebraic Riccati equation.*

The equation can be solved using Matlab

$$[\mathtt{K}, \mathtt{P}] = \mathtt{lqr}(\mathtt{A}, \mathtt{B}, \mathtt{Q}, \mathtt{R})$$

and $u = Kx$ can then be used as the input from state x.

1.3 Trajectory Tracking

Consider the problem of not stabilizing to the origin, i.e. $x \to 0$ but tracking a given reference trajectory $x_d(t)$, i.e. $x \to x_d$. This is often useful when x_d was an optimized trajectory for a complex nonlinear system with constraints, which we cannot reoptimize in real time but can easily track.

One approach is to formulate the *error state*

$$e = x - x_d,$$

the control error (assuming we have the control u_d which produced x_d)

$$v = u - u_d,$$

and essentially apply LQR to the dynamics of e subject to "virtual" inputs v. In particular, note that in the general nonlinear case we have

$$\dot{e} = \dot{x} - \dot{x}_d = f(x, u) - f(x_d, u_d) = f(x_d + e, u_d + v) - f(x_d, u_d) \equiv F(e, v, x_d(t), u_d(t)),$$

in error dynamic

Sol #1 non-linear dynamics: $\dot e = f(x_d+e, u_d+v) - f(x_d, u_d) \approx \cancel{f(x_d,u_d)} + \underbrace{\partial_x f(x_d,u_d)}_{A(t)}\cdot e + \underbrace{\partial_u f(x_d,u_d)}_{B(t)}\cdot v - \cancel{f(x_d,u_d)}$

A, B are linearization of f taylor expansion. $= A(t)\cdot e + B(t)\cdot v$ linearization around $x_d(t)$, $u_d(t)$

linear dynamics

or in other words we have obtained a new ODE in for e, v, and time-varying parameters. In the linear case the we have

$$e = x - x_d, \quad v = u - u_d, \qquad \dot e = Ae + Bv,$$

and so once the optimal $v = Ke$ is computed using standard LQR the actual control u is recovered by

$$u = K(x - x_d) + u_d.$$

Sol #2

Another approach is to directly obtain necessary conditions. In particlar, let the cost be defined as

$$J = \underbrace{\frac{1}{2}\|x(t_f) - x_d(t_f)\|^2_{P_f}}_{\phi} + \frac{1}{2}\int_{t_0}^{t_f} \underbrace{\left\{\|x(t) - x_d(t)\|^2_{Q(t)} + \|u(t)\|^2_{R(t)}\right\}}_{L} dt \quad \text{无}\psi$$

The Hamiltonian is

dynamics

$$H = \frac{1}{2}\|x(t) - x_d(t)\|^2_{Q(t)} + \frac{1}{2}\|u(t)\|^2_{R(t)} + \lambda^T(Ax + Bu)$$

$U = U_d + V$

$$\dot x = f(x, u, t)$$
$$\dot\lambda = -\nabla_x H = -\nabla_x f \cdot \lambda - \nabla_x L,$$
$$\nabla_u H = \nabla_u f \cdot \lambda + \nabla_u L = 0,$$
$$\lambda(t_f) = \nabla_x \phi(x(t_f), t_f) + \nabla_x \psi(x(t_f), t_f) \cdot \nu,$$
$$\left(\partial_t \phi + \nu^T \partial_t \psi + L + \lambda^T f\right)_{t=t_f} = 0,$$

and the necessary conditions become

这是相乘.

$$u = -R^{-1}B^TPx \equiv Kx. \qquad \dot\lambda = -Q(x - x_d) - A^T\lambda,$$

and

$$u = -R^{-1}B^T\lambda,$$

while the transverasility conditions are

这是相乘.

$$\lambda(t_f) = P_f(x(t_f) - x_d(t_f)) = \overbrace{P_f \cdot x(t_f)}^{Px} - \overbrace{P_f \cdot x_d(t_f)}^{s}$$

In order to derive a control law, we follow the same reasoning as in the regular case and assume that the multiplier is of the form

cannot assume $\lambda = Px$ must assume $\lambda = Px + s$

$$\lambda = Px + s \quad \Rightarrow u = -R^{-1}B^TPx - R^{-1}B^Ts$$

We will now attempt to derive expressions for P and s that satisfy the necessary conditions. Differentiating we have

$$\dot\lambda = \dot Px + P\dot x + \dot s,$$

which is equivalent to

$$-Qx + Qx_d - A^T(Px + s) = \dot Px + (PA - PBR^{-1}B^TP)x - PBR^{-1}B^Ts + \dot s$$

or

$$(-A^TP - PA - Q + PBR^{-1}B^TP - \dot P)x + (\dot s + A^TS - PBR^{-1}B^Ts - Qx_d) = 0.$$

If we set

$$\dot P = -A^TP - PA - Q + PBR^{-1}B^TP, \tag{7}$$
$$\dot s = -A^Ts + PBR^{-1}B^Ts + Qx_d \tag{8}$$

and integrate them backwards starting with

$$P(t_f) = P_f, \tag{9}$$
$$s(t_f) = -P_f x_d \tag{10}$$

then the necessary conditions will be satisfied.

get $P(t)$
和 $s(t)$
从而 get $\lambda(t)$
everything is known now

Example 3. Non-zero signal tracking. Consider the dynamics

$$\dot{x}_1 = x_2, \tag{11}$$
$$\dot{x}_2 = 2x_1 - x_2 + u \tag{12}$$

and cost function

$$J = 4(x_1 - 1)^2 + \frac{1}{2}\int_0^{t_f} [2(x_1 - 1)^2 + 0.005u^2]dt$$

which corresponds to driving *only* the x_1 coordinate to 1 while minimizing control effort. This is a simplified version of the more general condition $x(t) \to x_d(t)$, where $x_{d1} = 1$ is constant. We integrate the Riccati ODE numerically using $t_f = 5$ to obtain the following matrix $P(t)$, inputs $u(t)$ and state histories $x(t)$: