EECS 127 Apr 16, 2020.

Applications of Optimization.

Today: Control

$Q, R \succeq 0$.

LQR: Linear Quadratic Regulator.

$$\min_{\vec{x}_t, \vec{u}_t} \sum_{t=0}^{N-1} \frac{1}{2}\Big(\underbrace{\vec{x}_t^T Q \vec{x}_t}_{\text{Penalty on state}} + \underbrace{\vec{u}_t^T R \vec{u}_t}_{\text{penalty on my control}}\Big) + \underbrace{\frac{1}{2}\vec{x}_N^T Q_f \vec{x}_N}_{\text{Terminal cost}}$$

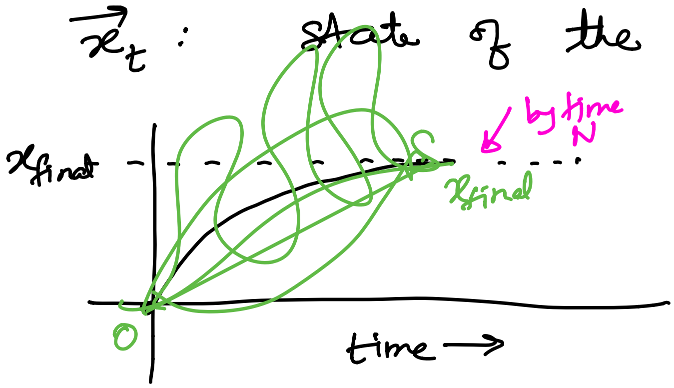
state, control.

s.t. $\boxed{\vec{x}_{t+1} = A\vec{x}_t + B\vec{u}_t}$ $t = 0, \cdots N-1$ → System dynamics.

$\vec{x}_0 = \vec{x}_{init}$.

N: Terminal time

$\vec{x}_t$: state of the system at time t.



Controls: u_t

x_t: deviation from desired trajectory.

Simple: but powerful.

Quadratic program $\vec{x}_t, \vec{u}_t \to \vec{x}_{t+1}$

$\to$ Can be reformulated as LS type problem on $\vec{u}_1 \dots \vec{u}_N$. But this is computationally unwieldy. $\Rightarrow$

Dynamic Programming. Bellman equations.

Adjoint method

$\vec{\lambda}$: inequality

ν: equality.

$$L(\vec{x}_0 \dots \vec{x}_N, \vec{u}_0, \dots \vec{u}_N, \vec{\lambda}_1 \cdots \vec{\lambda}_N)$$

$$= \sum_{t=0}^{N-1} \frac{1}{2}(\vec{x}_t Q \vec{x}_t + \vec{u}_t R \vec{u}_t) + \frac{1}{2} \vec{x}_N^T Q_f \vec{x}_N$$

$$+ \sum_{t=0}^{N-1} \vec{\lambda}_{t+1}^T (A\vec{x}_t + B \cdot \vec{u}_t - \vec{x}_{t+1})$$

Constraint

$\vec{x}_{t+1} = A\vec{x}_t + B\vec{u}_t$

$t = 0, \dots N-1.$

$A\vec{x}_t + B\vec{u}_t = \vec{x}_{t+1}$

$A\vec{x}_{t-1} + B\vec{u}_{t-1} = \vec{x}_t$

KKT conditions

→ No inequality related KKT conditions.

The only interesting one is the "first order" condition.

Derivative w.r.t. variables should be zero.

$$\nabla_{\vec{u}_t} L = R \cdot \vec{u}_t + B^T \cdot \vec{\lambda}_{t+1} = 0 \quad (1)$$

$$\nabla_{\vec{x}_t} L = Q \cdot \vec{x}_t + A^T \vec{\lambda}_{t+1} - \vec{\lambda}_t = 0 \quad (2)$$

$t = 0, \cdots N-1.$

$$\nabla_{\vec{x}_N} L = Q_f \cdot \vec{x}_N - \vec{\lambda}_N = 0 \quad (3)$$

---

(2) Rewrite (2):

$$(4) \quad \vec{\lambda}_t = A^T \vec{\lambda}_{t+1} + Q \cdot \vec{x}_t$$

$$\vec{\lambda}_N = Q_f \cdot \vec{x}_N \leftarrow (5)$$

"dynamics"

"Adjoint system".

λ: Co-state
adjoint

Rewrite (1):

$$(6) \quad \vec{u}_t = -R^{-1} \cdot B^T \cdot \vec{\lambda}_{t+1}$$

Original Dynamics:

$$\boxed{\begin{aligned}\vec{x}_{t+1} &= A\vec{x}_t + B\vec{u}_t \\ \vec{x}_0 &= \vec{x}_{init}\end{aligned}}$$

"Backward induction to solve"

<u>Reminder:</u> We want optimal $\vec{u}_t$'s $\vec{x}_t$'s

<u>Assumption</u> $\vec{\lambda}_t = P_t \cdot \vec{x}_t$ (Induction hypothesis)

Linear relationship

$t = N$: $\vec{\lambda}_N = Q_f \cdot \vec{x}_N$ $\boxed{P_N = Q_f.}$

Indu. Hyp. ✓ $t = N$

<u>Assume:</u> $\vec{\lambda}_{t+1} = P_{t+1} \cdot \vec{x}_{t+1}$

we now need to show,

that:
$$\vec{\lambda}_t = P_t \cdot \vec{x}_t$$

~~given:~~ $\vec{\lambda}_{t+1} = P_{t+1} \cdot \vec{x}_{t+1}$

use system dynamics:
$$= P_{t+1}\left(A\vec{x}_t + B\vec{u}_t\right)$$
$$= P_{t+1}\left(A\vec{x}_t + B\left(-R^{-1}B^T\vec{\lambda}_{t+1}\right)\right)$$
use eq. 6.

$$= P_{t+1}\left(A\vec{x}_t - BR^{-1}B^T\vec{\lambda}_{t+1}\right) = \vec{\lambda}_{t+1}$$
I

Rearrange:
$$\left(I + P_{t+1}BR^{-1}B^T\right)\vec{\lambda}_{t+1} = P_{t+1}A\cdot\vec{x}_t$$

$$\boxed{\vec{\lambda}_{t+1} = \left(I + P_{t+1}BR^{-1}B^T\right)^{-1} \cdot P_{t+1} \cdot A \cdot \vec{x}_t .}$$

Eq 4 $\Rightarrow$ $\vec{\lambda}_t = A^T\vec{\lambda}_{t+1} + Q\cdot\vec{x}_t$

$$= A^T \left(I + P_{t+1} \cdot B \cdot R^{-1} \cdot B^T\right)^{-1} P_{t+1} \cdot A \cdot \underline{\underline{\vec{x}_t}}$$

$$+ Q \cdot \underline{\underline{\vec{x}_t}}$$

$$\Rightarrow \vec{\lambda}_t = \underbrace{\left(A^T \left(I + P_{t+1} B R^{-1} B^T\right)^{-1} P_{t+1} \cdot A + Q\right)}_{P_t} \vec{x}_t$$

$$\vec{\lambda}_t = P_t \cdot \vec{x}_t$$

$$P_t := A^T \left(I + P_{t+1} B R^{-1} B^T\right)^{-1} \cdot P_{t+1} \cdot A + Q$$

$P_N = Q_f$.

Run backwards from N to get all $P_0 \dots P_N$

$P_{t+1} \to P_t$

Ricatti Equation.

Solutions:

① Solve for all P's using the Ricatti equation.

② $\vec{u}_t = -R^{-1} B^T \vec{\lambda}_{t+1}$

$$= -R^{-1}B^T(I + P_{t+1} B R^{-1} B^T)^{-1} P_{t+1} \cdot A \cdot \vec{x}_t$$

Relationship between current state + current control.

③ Now, use system dynamics to get all controls (u) and all states.

Start at $\vec{x}_0 \rightarrow \vec{u}_0$

$\vec{x}_0, \vec{u}_0 \longrightarrow \vec{x}_1$

$$\vec{x}_1 \longrightarrow \vec{u}_1$$

$$\vec{x}_1, \vec{u}_1 \longrightarrow \vec{u}_2$$

---

① Magically generated extra equations

② Recursion (backwards) on P_t

Knew final $P_N = Q_f$.

$P_{N-1}, P_{N-2}, \dots \quad P_0$

③ $\vec{x}_0, P_0 \longrightarrow \vec{u}_0$

④ Forward dynamics to get rest of $\vec{x}_t, \vec{u}_t$ upto N.

Cool things:

① $\vec{u}_t$ is a LINEAR of $\vec{x}_t$

x_0

Time 0:

$$\vec{x}_0 Q \vec{x}_0 + \vec{u}_0 R \vec{u}_0$$

$$\vec{x}_1 Q \vec{x}_1 + \vec{u}_1 R \vec{u}_1$$

"cost at ~~t~~time k to go"

= cost incurred @ time k

+ cost to go @ state x_{k+1} @ time x_{k+1}