3/12: iLQR

Announcements:
- HW 3 out, due 3/26
- Mid-semester survey out, due 3/17
- Project milestone due 3/19
  - 2-6 pg.
  - intro/related work, problem def., timeline

Last time:
- dynamic programming

Today's q: How do we apply to nonlinear sys?

Recall:

$$\min_{x_{0:T},\, u_{0:T}} \; c_T(x_T) + \sum_{t=0}^{T-1} c_t(x_t, u_t)$$

s.t.

$$x_0 = x(0)$$
$$x_{t+1} = f(x_t, u_t)$$
$$x_t \in \mathcal{X}$$
$$u_t \in \mathcal{U}$$

1. Trajectory opt. $x^*_{0:T}, u^*_{0:T-1}$

2. Dynamic programming

$V_t^*(x)$: "value function" "cost-to-go"

$$V_t^*(x) = \min_{u \in \mathcal{U}} \{ c_t(x,u) + V^*_{t+1}(f(x,u)) \}$$

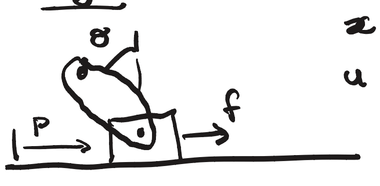
$$\downarrow$$

$$\pi_t^*(x_t) = \arg\min_{u \in \mathcal{U}} \{ \cdots \}$$

Special case: LQR (two views coincide)

Q: How do we extend these results?

e.g.) (cart pole)



$x = [p, \theta, \dot{p}, \dot{\theta}]$

$u = [f]$

goal: $x_t \to 0$

cost: $Q_T = \begin{bmatrix} 10 & & & \\ & 10 & & \\ & & 1 & \\ & & & 1 \end{bmatrix}$ $\quad Q_t = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix}$

$R_t = [0.1]$

problem: nonlinear dynamics!

How can we do this?
1. Linearize system about $\bar{x}$, use LQR
2. Linearize about many points, look LQR gains
3. iLQR

suppose $\theta \approx 0$

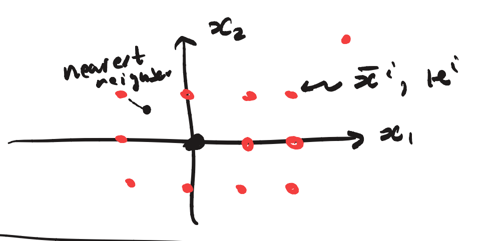
$$A = \frac{\partial f}{\partial x}\Big|_{\bar{x}=0} \qquad B = \frac{\partial f}{\partial u}\Big|_{\bar{x}=0}$$

apply infinite-horizon LQR gain w/ these matrices

Pros: - dead simple
- for $x \approx 0$ it tends to work well

Cons: for $x \neq 0$, it doesn't

Idea 2: Gain scheduling



Idea 3: Iterative LQR

suppose we have initial guess $\bar{x}_{0:T}, \bar{u}_{0:T-1}$

Key idea: $A_t = \frac{\partial f}{\partial x}\Big|_{\bar{x}_t, \bar{u}_t}$ $\quad B_t = \frac{\partial f}{\partial u}\Big|_{\bar{x}_t, \bar{u}_t}$

linearize $\xrightarrow{K, k}$ LQR rollout

LQR rollout $\xrightarrow{\bar{x}, \bar{u}}$ linearize

To derive this, we use Taylor expansions of the cost/dynamics

$$\delta x_t = x_t - \bar{x}_t \qquad \delta u_t = u_t - \bar{u}_t$$

$$c_t(\delta x_t, \delta u_t) = \text{const} + \begin{bmatrix} \delta x_t \\ \delta u_t \end{bmatrix}^T \begin{bmatrix} q_t \\ r_t \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \delta x_t \\ \delta u_t \end{bmatrix} \begin{bmatrix} Q_t & S_t \\ S_t^T & R_t \end{bmatrix} \begin{bmatrix} \delta x_t \\ \delta u_t \end{bmatrix} + \cdots$$

$q_t = \nabla_x c_t \quad Q_t = \nabla^2_{xx} c_t \quad S_t = \nabla^2_{xu} c_t$

$r_t = \nabla_u c_t \quad R_t = \nabla^2_{uu} c_t$

$$\delta x_{t+1} = f(\delta x_t, \delta u_t) = A_t \delta x_t + B_t \delta u_t$$

$$A_t = \frac{\partial f}{\partial x}\Big|_{\bar{x}_t, \bar{u}_t} \qquad B_t = \frac{\partial f}{\partial u}\Big|_{\bar{x}_t, \bar{u}_t}$$

For this variation of LQR,

$$u_t^* = -\bar{K}_t x_t - \underline{k}_t \qquad \bar{K}_t \in \mathbb{R}^{m \times n} \quad \underline{k}_t \in \mathbb{R}^m$$

feedback ($\bar{K}_t x_t$), feedforward ($\underline{k}_t$)

$$V_t^*(\delta x_t) = \underline{p}_t^T \delta x_t + \frac{1}{2} \delta x_t^T \bar{P}_t \delta x_t$$

We can define "the Q-function"

$$Q_t(\delta x_t, \delta u_t) = c_t(\delta x_t, \delta u_t) + V^*_{t+1}(f(\delta x_t, \delta u_t))$$

↑ apply δu_t @ time t, ↑ then act optimally $\tau > t$

$$V_t^*(\delta x_t, \delta u_t) = \min_{\delta u_t} \{ Q_t(\delta x_t, \delta u_t) \}$$

$$Q_t(\delta x_t, \delta u_t) = \begin{bmatrix} h_x \\ h_u \end{bmatrix}^T \begin{bmatrix} \delta x_t \\ \delta u_t \end{bmatrix} + \frac{1}{2} \begin{bmatrix} \delta x_t \\ \delta u_t \end{bmatrix}^T \begin{bmatrix} H_{xx} & H_{xu} \\ H_{ux} & H_{uu} \end{bmatrix} \begin{bmatrix} \delta x_t \\ \delta u_t \end{bmatrix}$$

$h_x = q_t + A_t^T \underline{p}_{t+1} \qquad H_{xx} = Q_t + A_t^T \bar{P}_{t+1} A_t$

$h_u = r_t + B_t^T \underline{p}_{t+1} \qquad H_{uu} = R_t + B_t^T \bar{P}_{t+1} B_t$

$H_{xu} = S_t + A_t^T \bar{P}_{t+1} B_t$

$$\nabla_{\delta u_t} Q(\delta x_t, \delta u_t^*) = 0$$

$$h_u + H_{uu} \delta u_t^* + H_{xu} \delta x_t = 0$$

$$\delta u_t^* = -(H_{uu})^{-1}(H_{xu} \delta x + h_u)$$

Get an LQR-style recursion

$$\bar{P}_t = H_{xx} - H_{xu} \bar{K}_t \qquad \underline{p}_t = h_x - H_{xu} \underline{k}_t$$

Alg] (iLQR)

input: $\bar{x}_{0:T}, \bar{u}_{0:T-1}$ // initial guess

repeat until convergence

$\quad \bar{K}_t, \underline{k}_t \leftarrow \text{backward}(\bar{x}_{0:T}, \bar{u}_{0:T-1})$

$\quad$ for $t = 0, \ldots, T-1$ // forward

$\quad\quad u_t = \bar{u}_t - \bar{K}_t(x_t - \bar{x}_t) - \underline{k}_t$ $\quad (-\alpha \underline{k}_t)$

$\quad\quad x_{t+1} = f(x_t, u_t)$

$\quad$ end for

$\quad \bar{x}_{0:T} \leftarrow x_{0:T}$

$\quad \bar{u}_{0:T-1} \leftarrow u_{0:T-1}$

Discussion:
- no convergence guarantees :(
- *very* sensitive to initialization
- provides you w/ feedback
- need to add reg. to H_{uu}
  $(H_{uu} + \lambda I)^{-1} \quad \lambda > 0$
- often linesearch over $\underline{k}_t$