Linear Quadratic Regulator

Shy

Reinforcement Learning

Model-free

- Emphasize heavily on sampling
- To fit the policy or value function

Model-based

- Develop a model
- To optimize controls

Modern Control ???

RL vs LQR

Difference

	Reinforcement Learning	Linear Quadratic Regulator
Trajectory	$s_1, a_1, s_2, r_1, a_2, s_3, r_2 \ldots$	$x_1, u_1, x_2, u_2, x_3, u_3 \ldots$
Transition	$s_{t+1} = f(s_t, a_t)$	$x_{t+1} = f(x_t, u_t) = F_t^T [x_t, u_t] + f_t$ **Linear**
Target	$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots$	$c(x_t, u_t) = \frac{1}{2}[x_t, u_t] C_t [x_t, u_t]^T + [x_t, u_t] c_t$
Solution	$a = \pi(s)$	$u = Kx + k$
Max or Min ?	$max\ Return$	$min\ Cost$
Need to Design?	Reward	Cost

Overview of LQR

Generate Trajectory

$$x_1, u_1, x_2, u_2, x_3, u_3 \ldots, x_{t-2}, u_{t-2}, x_{t-1}, u_{t-1}$$

Backward Pass

$$u_{t-1} = K_{t-1} x_{t-1} + k_{t-1}$$

…… …

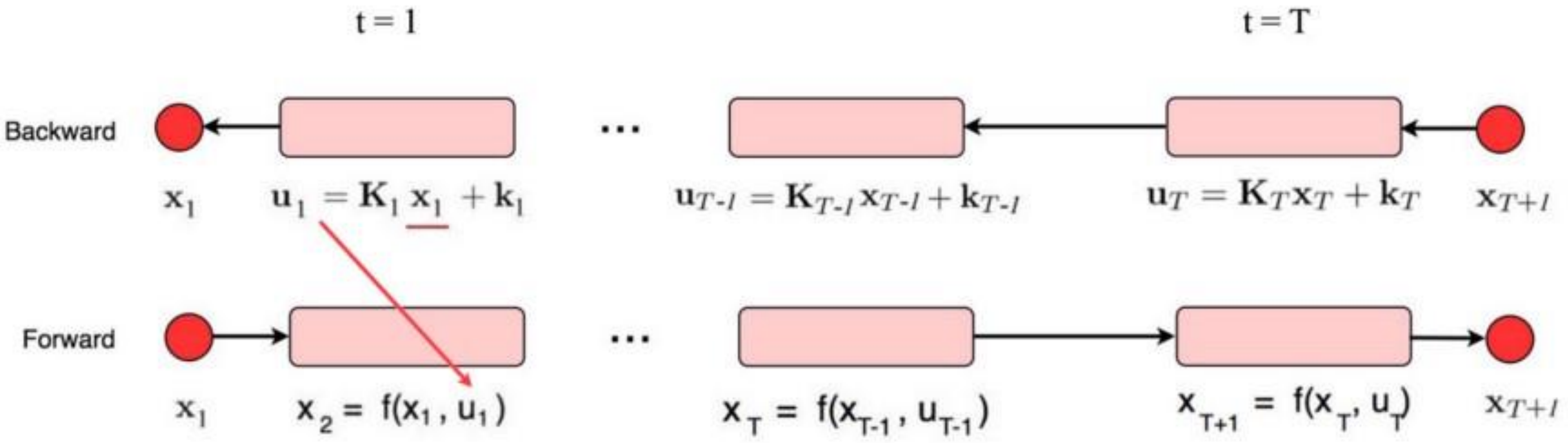
$$u_1 = K_1 x_1 + k_1$$

Forward Pass

$$x_2 = f(x_1, u_1)$$

$$u_2 = K_2 x_2 + k_2$$

…… …



Dynamic Programming

Bellman Equation

$r \rightarrow maxG_t$

$$Q(s_{t-1}, a_{t-1}) = \boxed{r} + \gamma \sum P(s_t \mid s_{t-1}, a_{t-1}) V(s_t)$$

- Policy iteration
- Value iteration
- Find the approximate solution by iteration

Linear Quadratic Regulator

$c \rightarrow minc(x_t, u_t)$

$$Q(x_{t-1}, u_{t-1}) = const + \boxed{\frac{1}{2}[x_{t-1}, u_{t-1}]C_{t-1}[x_{t-1}, u_{t-1}]^T + [x_{t-1}, u_{t-1}]c_{t-1}} + V(x_t)$$

- Quadratic forms is derivable
- Find the optimal solution

$$\begin{aligned} Q(x_t, u_t) &= const + \frac{1}{2}[x_t, u_t]C_t[x_t, u_t]^T + [x_t, u_t]c_t \\ &= const + \frac{1}{2}[x_t, u_t]\begin{bmatrix} C_{x_t,x_t} & C_{x_t,u_t} \\ C_{u_t,x_t} & C_{u_t,u_t} \end{bmatrix}[x_t, u_t]^T + [x_t, u_t]\begin{bmatrix} c_{x_t} \\ c_{u_t} \end{bmatrix} \end{aligned}$$

$$\frac{dQ(x_t, u_t)}{du_t} = C_{u_t,x_t}x_t + C_{u_t,u_t}u_t + c_{u_t}^T = 0$$

$$u_t = -C_{u_t,u_t}^{-1}(C_{u_t,x_t}x_t + c_{u_t}^T)$$

$$u_t = K_t x_t + k_t$$

Q value and V value(1/2)

Action-state value →State value

$$Q(x_t, u_t) = const + \frac{1}{2}[x_t, K_t x_t + k_t] C_t [x_t, K_t x_t + k_t]^T + [x_t, K_t x_t + k_t] c_t$$
$$u_t = K_t x_t + k_t$$

$$\Rightarrow \quad V(x_t) = const + \frac{1}{2}[x_t, K_t x_t + k_t] C_t [x_t, K_t x_t + k_t]^T + [x_t, K_t x_t + k_t] c_t$$

- Q value depends on x_t, u_t
- V value only depends on x_t
- Now t is the terminal step, so $V(x_t) = 0$

$$if \quad V_t = C_{x_t,x_t} + C_{x_t,u_t} K_t + K_t^T C_{u_t,x_t} + K_t^T C_{u_t,u_t} K_t$$
$$v_t = c_{x_t} + C_{x_t,u_t} k_t + K_t^T c_{u_t} + K_t^T C_{u_t,u_t} k_t$$

$$then \quad V(x_t) = const + \frac{1}{2} x_t^T V_t x_t + x_t^T v_t$$

Q value and V value(2/2)

State value →Action-state value

$$V(x_t) = const + \frac{1}{2}x_t^T V_t x_t + x_t^T v_t$$

$$x_t = f(x_{t-1}, u_{t-1}) = F_{t-1}[x_{t-1}, u_{t-1}]^T + f_{t-1}$$

$$\boxed{V(x_t)} = const + \frac{1}{2}[x_{t-1}, u_{t-1}]F_{t-1}^T V_t F_{t-1}[x_{t-1}, u_{t-1}]^T + [x_{t-1}, u_{t-1}]F_{t-1}^T V_t f_{t-1} + [x_{t-1}, u_{t-1}]F_{t-1}^T v_t$$

$$Q(x_{t-1}, u_{t-1}) = const + \frac{1}{2}[x_{t-1}, u_{t-1}]C_{t-1}[x_{t-1}, u_{t-1}]^T + [x_{t-1}, u_{t-1}]c_{t-1} + \boxed{V(x_t)}$$

- Q value depends on x_t, u_t
- V value only depends on x_t
- Now t is not the terminal step, so $V(x_t) \neq 0$

$$if \quad Q_{t-1} = C_{t-1} + F_{t-1}^T V_t F_{t-1}$$

$$q_{t-1} = c_{t-1} + F_{t-1}^T V_t f_{t-1} + F_{t-1}^T V_t$$

$$then \quad Q(x_{t-1}, u_{t-1}) = const + \frac{1}{2}[x_{t-1}, u_{t-1}]Q_{t-1}[x_{t-1}, u_{t-1}]^T + [x_{t-1}, u_{t-1}]q_{t-1}$$

Optimal Solution

Terminal State: $V(x_t) = 0$

$$Q(x_{t-1}, u_{t-1}) = const + \frac{1}{2}[x_{t-1}, u_{t-1}]C_{t-1}[x_{t-1}, u_{t-1}]^T + [x_{t-1}, u_{t-1}]c_{t-1} + V(x_t)$$

$$\begin{aligned} Q(x_t, u_t) &= const + \frac{1}{2}[x_t, u_t]C_t[x_t, u_t]^T + [x_t, u_t]c_t \\ &= const + \frac{1}{2}[x_t, u_t]\begin{bmatrix} C_{x_t,x_t} & C_{x_t,u_t} \\ C_{u_t,x_t} & C_{u_t,u_t} \end{bmatrix}[x_t, u_t]^T + [x_t, u_t]\begin{bmatrix} c_{x_t} \\ c_{u_t} \end{bmatrix} \end{aligned}$$

$$\frac{dQ(x_t, u_t)}{du_t} = C_{u_t,x_t}x_t + C_{u_t,u_t}u_t + c_{u_t}^T = 0$$

$$u_t = -C_{u_t,u_t}^{-1}(C_{u_t,x_t}x_t + c_{u_t}^T)$$

$$u_t = K_t x_t + k_t$$

Not Terminal State: $V(x_t) \neq 0$

Terminal State: $V(x_t) = 0$

$$u_t = -Q_{u_t,u_t}^{-1}(Q_{u_t,x_t}x_t + q_{u_t}^T)$$

$$u_t = K_t x_t + k_t$$

$$\frac{dQ(x, u_t)}{du_t} = Q_{u_t,x_t}x_t + Q_{u_t,u_t}u_t + q_{u_t}^T = 0$$

LQR Algorithm

Generate Trajectory

$$x_1, u_1, x_2, u_2, x_3, u_3 \ldots, x_{t-2}, u_{t-2}, x_{t-1}, u_{t-1}$$

Backward Recursion

$for\ t = T\ to\ 1$:

$$Q_t = C_t + F_t^T V_{t+1} F_t$$

$$q_t = c_t + F_t^T V_{t+1} f_t + F_t^T v_{t+1}$$

$$Q(x_t, u_t) = const + \frac{1}{2}[x_t, u_t] Q_t [x_t, u_t]^T + [x_t, u_t] q_t$$

$$u_t \leftarrow argmin Q(x_t, u_t) = K_t x_t + k_t$$

$$K_t = -Q_{u_t,u_t}^{-1} Q_{u_t,x_t}$$

$$k_t = -Q_{u_t,u_t}^{-1} q_{u_t}^T$$

$$V_t = Q_{x_t,x_t} + Q_{x_t,u_t} K_t + K_t^T Q_{u_t,x_t} + K_t^T Q_{u_t,u_t} K_t$$

$$v_t = q_{x_t} + Q_{x_t,u_t} k_t + K_t^T q_{u_t} + K_t^T Q_{u_t,u_t} k_t$$

$$V(x_t) = const + \frac{1}{2} x_t^T V_t x_t + x_t^T v_t$$

Forward Recursion

$for\ t = 1\ to\ T$:

$$u_t = K_t + k$$

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