Student : NINH DO

CS 287 - Homework #3

SID # 25949105

Problem #2 :

linear system : $x_{t+1} = Ax_t + Bu_t$

quadratic cost : $C = \frac{1}{2}\sum_{\tau} x_t^T Q x_t$

let $R_t = R(x_t) = \frac{1}{2}x_t^T Q x_t$

$$C = R_0 + R_1 + \dots + R_H = \frac{1}{2}x_0^T Q x_0 + \frac{1}{2}x_1^T Q x_1 + \frac{1}{2}x_2^T Q x_2 + \dots + \frac{1}{2}x_H^T Q x_H$$

$$\frac{\partial C}{\partial u_0} = \frac{\partial R_0}{\partial u_0} + \frac{\partial R_1}{\partial u_0} + \dots + \frac{\partial R_H}{\partial u_0}$$

$$= \frac{\partial R_0}{\partial x_0}\frac{\partial x_0}{\partial u_0} + \frac{\partial R_1}{\partial x_1}\frac{\partial x_1}{\partial u_0} + \frac{\partial R_2}{\partial x_2}\underbrace{\frac{\partial x_2}{\partial u_0}}_{\frac{\partial x_2}{\partial x_1}\frac{\partial x_1}{\partial u_0}} + \dots + \frac{\partial R_H}{\partial x_H}\frac{\partial x_H}{\partial x_{H-1}}\dots\frac{\partial x_1}{\partial u_0}$$

$$\frac{\partial R_t}{\partial x_t} = Qx_t \qquad \frac{\partial x_t}{\partial x_{t-1}} = A^T \qquad \frac{\partial x_t}{\partial u_{t-1}} = B^T$$

Thus ,
$$\frac{\partial C}{\partial u_0} = \underbrace{\frac{\partial R_0}{\partial x_0}}_{Qx_0}\underbrace{\frac{\partial x_0}{\partial u_0}}_{0} + B^T Q x_1 + (AB)^T Q x_2 + (A^2B)^T Q x_3 + \dots$$

$$= \sum_{t=1}^{H} (A^{t-1-0}B)^T Q x_t$$

Similarly ,
$$\frac{\partial C}{\partial u_1} = \frac{\partial R_0}{\partial x_0}\underbrace{\frac{\partial x_0}{\partial u_1}}_{0} + \frac{\partial R_1}{\partial x_1}\underbrace{\frac{\partial x_1}{\partial u_1}}_{0} + \underbrace{\frac{\partial R_2}{\partial x_2}\frac{\partial x_2}{\partial u_1}}_{B^T Q x_2} + \dots$$

$$= \sum_{t=2}^{H} (A^{t-1-1}B)^T Q x_t$$

Generally, $\frac{\partial C}{\partial u_i} = \sum_{t=i+1}^{H} (A^{t-i-1}B)^T Q x_t$

The gradient update for shooting method:

$$u_i^{(j+1)} = u_i^{(j)} - \gamma \frac{\partial C}{\partial u_i^{(j)}} = u_i^{(j)} - \gamma \sum_{t=i+1}^{H} (A^{t-i-1}B)^T Q x_t \quad (I)$$

where $u_i^{(j)}$ is the action at time step i at iteration j

For collocation method, we should update x_i simultaneously by computing gradient $\frac{\partial C}{\partial x_i}$:

$$\frac{\partial C}{\partial x_0} = \frac{\partial R_0}{\partial x_0} + \frac{\partial R_1}{\partial x_0} + \frac{\partial R_2}{\partial x_0} + \cdots + \frac{\partial R_H}{\partial x_0}$$

$$= \frac{\partial R_0}{\partial x_0} + \frac{\partial R_1}{\partial x_1}\frac{\partial x_1}{\partial x_0} + \frac{\partial R_2}{\partial x_2}\frac{\partial x_2}{\partial x_1}\frac{\partial x_1}{\partial x_0} + \cdots + \frac{\partial R_H}{\partial x_H}\frac{\partial x_H}{\partial x_{H-1}}\cdots\frac{\partial x_1}{\partial x_0}$$

$$\frac{\partial R_t}{\partial x_t} = Q x_t \qquad \frac{\partial x_t}{\partial x_{t-1}} = A^T$$

$$\Rightarrow \frac{\partial C}{\partial x_0} = Q x_0 + A^T Q x_1 + (A^T)^2 Q x_2 + \cdots + (A^T)^H Q x_H$$

$$= \sum_{t=0}^{H} (A^T)^{t-0} Q x_t$$

Similarly: $$\frac{\partial C}{\partial x_1} = \underbrace{\frac{\partial R_0}{\partial x_1}}_{0} + \underbrace{\frac{\partial R_1}{\partial x_1}}_{Q x_1} + \underbrace{\frac{\partial R_2}{\partial x_1}}_{\frac{\partial R_2}{\partial x_2}\frac{\partial x_2}{\partial x_1} = A^T Q x_2} + \cdots + \underbrace{\frac{\partial R_H}{\partial x_1}}_{(A^T)^{H-1} Q x_H}$$

$$= \sum_{t=1}^{H} (A^T)^{t-1} Q x_t$$

Generally, $\frac{\partial C}{\partial x_i} = \sum_{t=i}^{H} (A^T)^{t-i} Q x_t$

The gradient update for collocation method includes:

$$x_i^{(j+1)} = x_i^{(j)} - \gamma \frac{\partial c}{\partial x_i^{(j)}} = x_i^{(j)} - \gamma \sum_{t=i}^{H} (A^T)^{t-i} Q\, x_\tau^{(j)}$$

$$u_i^{(j+1)} = u_i^{(j)} - \gamma \frac{\partial c}{\partial u_i^{(j)}} = u_i^{(j)} - \gamma \sum_{t=i+1}^{H} (A^{t-i-1} B)^T Q\, x_\tau^{(j)} \qquad (II)$$

Comparing the shooting method (I) and the collocation method (II) we see that the collocation method updates the states. In the absence of stochasticity, both have the same performance. In the presence of noise, the shooting method may not adapt well due to the effect of early action so much higher than later action, which may make the pre-calculation of actions become invalid, while the collocation method takes into account the noisy states in each update step. In such a way, the state updates decouple between time steps, making computation stable.