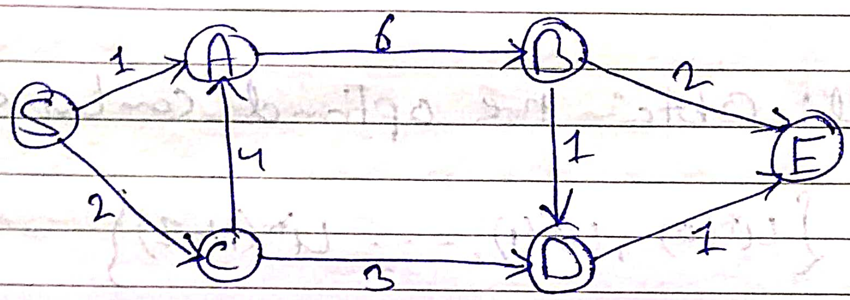
Dynamic Programming

⇒ Developed by Richard Bellman in the 1950s.

Dynamic Programming → Planning (has nothing to do with computers)

★ Essentials of dynamic programming



Bellman's principle of optimality

→ From any point on an optimal trajectory, the remaining trajectory is optimal for the corresponding problem initiated at that point.

★ General optimal control problem

⇒ General discrete-time plant:

$$X(K+1) = f(X(K), u(K), K)$$

State constraint: $X(K) \in X \subset R^n$

Input Contraint: $u(K) \in U \subset R^m$

Page No.
Date: / /

⇒ Performance index:

$$J = S(x(N)) + \sum_{K=0}^{N-1} L(x(K), u(K), K)$$

Where,

- S & L are real scalar-valued functions.
- N — final time (optimization horizon)

⇒ Goal: Obtain the optimal control sequence

$$\{u^*(0), u^*(1), \ldots u^*(N-1)\}$$

{Minimizing J}

⇒ Discrete time LQ problem is a special case of above with LTI System with Quadratic performance index.

★ <u>Dynamic programming for optimal control</u>

Let $U_K = \{u(K), u(K+1) \ldots u(N-1)\}$

Cost to go

$$J_K^0(x(K)) = \min_{U_K} \left\{ S(x(N)) + \sum_{j=K}^{N-1} L(x(j), u(j), j) \right\}$$

Page No.
Date: / /

$$\Rightarrow \min_{u_K} \left\{ L(x(K), u(K), K) + S(x(N)) + \sum_{j=K+1}^{N-1} L(x(j), u(j), j) \right\}$$

$$\Rightarrow \min_{u(K)} \left\{ L(x(K), u(K), K) + \min_{u_{K+1}} \left(S(x(N)) + \sum_{j=K+1}^{N-1} L(x(j), u(j), j) \right) \right\}$$

$$\Rightarrow J_K^o(x(K)) = \min_{u(K)} \left\{ L(x(K), u(K), K) + J_{K+1}^o(x(K+1)) \right\}$$

boundary Condition: $J_N^o(x(N)) = S(x(N))$

$\Rightarrow$ The problem can now be solved by solving a sequence of problems

$$J_{N-1}^o \; J_{N-2}^o \; \cdots \; J_1^o \cdots J^o$$

★ Solving discrete-time finite horizon LQ via DP

⇒ **System dynamics:**

$$X(K+1) = A(K)X(K) + B(K)u(K) \quad , X(K_0) = X_0$$

⇒ **Performance Index:**

$$J = \frac{1}{2} X^T(N)\, S\, X(N) + \frac{1}{2} \sum_{K=K_0}^{N-1} \left\{ x^T(K) Q(K) X(K) + u^T(K) R(K) u(K) \right\}$$

where

$$Q(K) = Q^T(K) \geq 0$$

$$S = S^T \geq 0$$

$$R(K) = R^T(K) > 0$$

⇒ Optimal cost to go:

$$J_K^o(X(K)) = \min_{u(K)} \left\{ \frac{1}{2} X^T(K) Q(K) X(K) + \frac{1}{2} u^T(K) R(K) u(K) + J_{K+1}^o(x(K+1)) \right\}$$

with boundary condition

$$J_N^{\circ}(x(N)) = \frac{1}{2} x^T(N) S x(N)$$

★ <u>Facts of quadratic functions</u>

⇒ Consider

$$f(u) = \frac{1}{2} u^T M u + p^T u + q \qquad M > 0$$

$$\frac{\delta f}{\delta u} = 0 \quad \{\text{for minimum}\}$$

$$\frac{\delta p^T u}{\delta u} = p$$

$$\frac{\delta u^T M u}{\delta u} = (M + M^T) u = 2 M u \quad \{M = M^T > 0\}$$

$$\frac{\delta f}{\delta u^{\circ}} = M u^{\circ} + p = 0$$

$$u^{\circ} = -M^{-1} p$$

⇒ and optimal cost is

$$f^{\circ} = f^{\circ}(u^{\circ}) = -\frac{1}{2} p^T M^{-1} p + q$$

Page No.
Date: / /

* From J_N^0 to J_{N-1}^0 in discrete-time LQ

$$\Rightarrow J_{N-1}^0(x(N-1)) = \min_{u(N-1)} \Big\{ \frac{1}{2} x^T(N) S x(N)$$

$$+ \frac{1}{2}\Big[x^T(N-1) Q(N-1) x(N-1)$$

$$+ u^T(N-1) R(N-1) u(N-1)\Big]\Big\}$$

$$\Rightarrow x(N) = A(N-1) x(N-1) + B(N-1) u(N-1)$$

$$\Rightarrow J_{N-1}^0(x(N-1)) = \frac{1}{2} \min_{u(N-1)} \Big\{$$

$$x^T(N-1) Q(N-1) x(N-1)$$

$$+ u^T(N-1) R(N-1) u(N-1)$$

$$+ \Big[A(N-1) x(N-1) + B(N-1) u(N-1)\Big]^T$$

$$S \Big[A(N-1) x(N-1) + B(N-1) u(N-1)\Big]\Big\}$$

$\Rightarrow$ Optimal control by letting

$$\frac{\delta J_{N-1}}{\delta u(N-1)} = 0$$

$$u^{o}(N-1) =$$

$$-\left[R(N-1) + B^{T}(N-1) S B(N-1)\right]^{-1} B^{T}(N-1) S A(N-1) X(N-1)$$

$\Rightarrow$ At time N: Optimal cost is

$$J_N^{o}(X(N)) = \frac{1}{2} X^{T}(N) S X(N) \triangleq \frac{1}{2} X^{T}(N) P(N) X(N)$$

$$\Rightarrow J^{o}_{N-1}(X(N-1))$$

$$= +\frac{1}{2} X^{T}(N-1) \Big\{ Q(N-1) + A^{T}(N-1) S A(N-1)$$

$$- \left(B^{T}(N-1) S A(N-1)\right)^{T} \left(R(N-1) + B^{T}(N-1) S B(N-1)\right)^{-1}$$

$$\left(B^{T}(N-1) S A(N-1)\right) \Big\} X(N-1)$$

$$J^{o}_{N-1}(X(N-1)) \triangleq \frac{1}{2} X^{T}(N-1) P(N-1) X(N-1)$$

Where,

$$P(N-1) = Q(N-1) + A^{T}(N-1) S A(N-1) - \left(B^{T}(N-1) S A(N-1)\right)^{T} \left(R(N-1) + B^{T}(N-1) S B(N-1)\right)^{-1} \left(B^{T}(N-1) S A(N-1)\right)$$

$$\boxed{P(N-1) \Rightarrow Q(N-1) + A^{T}(N-1) P(N) A(N-1) - \left(B^{T}(N-1) P(N) A(N-1)\right)^{T} \left(R(N-1) + B^{T}(N-1) P(N) B(N-1)\right)^{-1} \left(B^{T}(N-1) P(N) A(N-1)\right)}$$

Riccati Equation

⇒ State feedback law:

$$\boxed{u^o(N-1) = -\left[R(N-1) + B^T(N-1)\,P(N)\,B(N-1)\right]^{-1} B^T(N-1)\,P(N)\,A(N-1)\,X(N-1)}$$

⇒ Same thing applies for $K+1$ to K by induction

$$\boxed{J^o_{K+1}\left(X(K+1)\right) = \frac{1}{2} x^T(K+1)\,P(K+1)\,X(K+1)}$$

⇒ Analogous as the case from N to $N-1$, we can get at K:

$$\boxed{J^o_K\left(X(K)\right) = \frac{1}{2} x^T(K)\,P(K)\,X(K)}$$

⇒ With Riccati equation

$$\boxed{\begin{aligned} P(K) = {} & Q(K) + A^T(K)\,P(K+1)\,A(K) \\ & - \left(B^T(K)\,P(K+1)\,A(K)\right)^T \left(R(K) + B^T(K)\,P(K+1)\,B(K)\right)^{-1} \\ & \left(B^T(K)\,P(K+1)\,A(K)\right) \end{aligned}}$$

⇒ and State-feedback law

$$\boxed{u^o(k) = -\left[R(k) + B^T(k)\,P(k+1)\,B(k)\right]^{-1} B^T(k)\,P(k+1)\,A(k)\,x(k)}$$

⇒ Iterating gives:

→ $u(0)$ depends on $\{A(K), B(K), R(K), Q(K+1)\}_{K=0}^{N-1}$.

→ In practice $P(K)$ can be computed offline since they do not require information of $x(K)$.

★ Observations:

⇒ $P(K)$ is indeed always symmetric

⇒ Regardless of the boundary condition $P(N)$ (≥ 0), the Solution of the Riccati equation converges to the same steady state P_s.

⇒ The Control law thus converges (backwards) to

$$u^{\circ}(k) = \underbrace{-[R + B^T P_s B]^{-1} B^T P_s A}_{K_s} x(k)$$