1.1 Cost-to-go approximations in Dynamic Programming

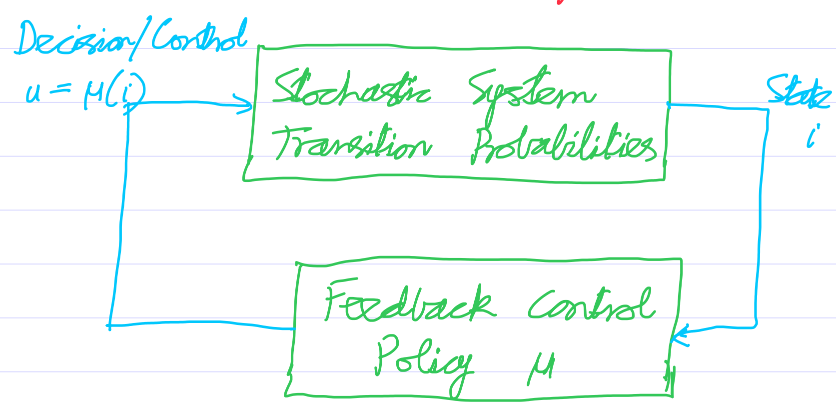
We are interested in decision-making policies that minimize the total cost over a number of stages.

$P_{ij}(u)$ – Probability of moving from state 'i' to state 'j' using control 'u'.

"policy" is defined as the rule for selection of controls.

Optimal cost for a state 'i' –

$$\boxed{J^*(i) = \min_u E\big[\underbrace{g(i,u,j)}_{\text{instantaneous cost of 'i'}} + \underbrace{J^*(j)}_{\text{cost-to-go}} \,\big|\, i, u\big]} \quad \forall i$$



However, large state and control spaces force us to settle for a sub-optimal solution.

$\tilde{J}(j, r)$ – Approximation of $J^*(j)$ with 'r' as a vector of parameters.

$$\underbrace{\tilde{\mu}(i)}_{\text{sub-optimal control}} = \arg\min_u E\big[g(i,u,j) + \underbrace{\tilde{J}(j,r)}_{\text{scoring/approximate cost-to-go function}} \,\big|\, i, u\big]$$

state 'j' $\longrightarrow$ $\boxed{\text{Approximator } (r)}$ $\longrightarrow$ $\tilde{J}(j,r)$

Compact representations - Scoring functions involving fewer parameters.

Lookup Table representation - Tabular description of J^*.

We wish to evaluate $E[g(i,u,j) + \tilde{J}(j,r) \mid i,u]$. The expression is non-trivial evaluate for each 'u' in terms of its computation time.

Alternatively, we can compute -

$$\underbrace{Q^*(i,u)}_{\text{Q-factor}} = E[g(i,u,j) + J^*(j) \mid i,u].$$

Approximate $Q^*(i,u)$ using $\tilde{Q}(i,u,r)$ and minimize its value-

$$\tilde{\mu}(i) = \arg\min_{u} \tilde{Q}(i,u,r).$$
