EE365: Dynamic Programming
Markov decision problem

find policy $\mu = (\mu_0, \ldots, \mu_{T-1})$ that minimizes

$$J = \mathbb{E} \left( \sum_{t=0}^{T-1} g_t(x_t, u_t) + g_T(x_T) \right)$$

Given

- functions $f_0, \ldots, f_{T-1}$
- stage cost functions $g_0, \ldots, g_{T-1}$ and terminal cost $g_T$
- distributions of independent random variables $x_0, w_0, \ldots, w_{T-1}$

Here

- system obeys dynamics $x_{t+1} = f_t(x_t, u_t, w_t)$.
- we seek a state feedback policy: $u_t = \mu_t(x_t)$
- we consider deterministic costs for simplicity
Optimal value function

Define the optimal value function

$$V^*_t(x) = \min_{\mu_t, \mu_{t+1}, \ldots, \mu_{T-1}} \mathbb{E}\left( \sum_{\tau=t}^{T-1} g_\tau(x_\tau, u_\tau) + g_T(x_T) \mid x_t = x \right)$$

- minimization is over policies $\mu_t, \ldots, \mu_{T-1}$
- $x_t$ is known, so we can minimize over action $u_t$ and policies $\mu_{t+1}, \ldots, \mu_{T-1}$
- $V^*_t(x)$ is expected cost-to-go, using an optimal policy, if $x_t = x$
- $J^* = \sum_x \pi_0(x)V^*_0(x) = \pi_0 V^*_0$
- $V^*_t$ also called Bellman value function, optimal cost-to-go function
Optimal policy

- the policy

\[ \mu_t^*(x) \in \arg\min_u (g_t(x, u) + \mathbb{E} V_{t+1}^*(f_t(x, u, w_t))) \]

is optimal

- expectation is over \( w_t \)

- can choose any minimizer when minimizer is not unique

- there can be optimal policies not of the form above

- *looks* circular and useless: need to know optimal policy to find \( V_t^* \)
Interpretation

\[ \mu_t^*(x) \in \arg\min_u \left( g_t(x, u) + \mathbb{E} V_{t+1}^*(f_t(x, u, w_t)) \right) \]

assuming you are in state \( x \) at time \( t \), and take action \( u \)

- \( g_t(x, u) \) (a number) is the current stage cost you pay
- \( V_{t+1}^*(f_t(x, u, w_t)) \) (a random variable) is the cost-to-go from where you land, if you follow an optimal policy for \( t + 1, \ldots, T - 1 \)
- \( \mathbb{E} V_{t+1}^*(f_t(x, u, w_t)) \) (a number) is the expected cost-to-go from where you land

optimal action is to minimize sum of current stage cost and expected cost-to-go from where you land
Greedy policy

- greedy policy is $\mu_t^{gr}(x) \in \arg\min_u g_t(x, u)$
- at any state, minimizes current stage cost without regard for effect of current action on future states
- in optimal policy

$$\mu_t^*(x) \in \arg\min_u (g_t(x, u) + E V_{t+1}^* (f_t(x, u, w_t)))$$

second term summarizes effect of current action on future states
Dynamic programming recursion

- define $V_T^*(x) := g_T(x)$
- for $t = T - 1, \ldots, 0$, 
  - find optimal policy for time $t$ in terms of $V_{t+1}^*$:
    $$\mu_t^*(x) \in \arg\min_u (g_t(x, u) + \mathbf{E} V_{t+1}^*(f_t(x, u, w_t)))$$
  - find $V_t^*$ using $\mu_t^*$:
    $$V_t^*(x) = \min_u (g_t(x, u) + \mathbf{E} V_{t+1}^*(f_t(x, u, w_t)))$$
- a recursion that runs backward in time
- complexity is $T|\mathcal{X}||\mathcal{U}||\mathcal{W}|$ operations (fewer when $P$ is sparse)
Variations

random costs:

\[ \mu_t^*(x) \in \arg\min_u E\left(g_t(x, u, w_t) + V_{t+1}^*(f_t(x, u, w_t))\right) \]
\[ V_t^*(x) := E\left(g_t(x, \mu_t^*(x), w_t) + V_{t+1}^*(f_t(x, \mu_t^*(x), w_t))\right) \]

state-action separable cost \( g_t(x, u) = q_t(x) + r_t(u) \):

\[ \mu_t^*(x) \in \arg\min_u (r_t(u) + E V_{t+1}^*(f_t(x, u, w_t))) \]
\[ V_t^*(x) := q_t(x) + r_t(\mu_t^*(x)) + E V_{t+1}^*(f_t(x, \mu_t^*(x), w_t)) \]

deterministic system:

\[ \mu_t^*(x) \in \arg\min_u (g_t(x, u) + V_{t+1}^*(f_t(x, u))) \]
\[ V_t^*(x) := g_t(x, \mu_t^*(x)) + V_{t+1}^*(f_t(x, \mu_t^*(x))) \]