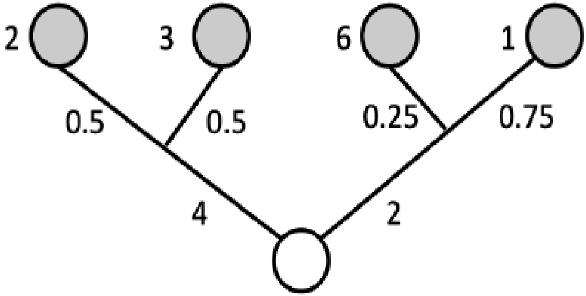
CS 7.603
Reinforcement Learning

Tejas Bodas

Assistant Professor, IIIT Hyderabad

Stochastic Dynamic Programming

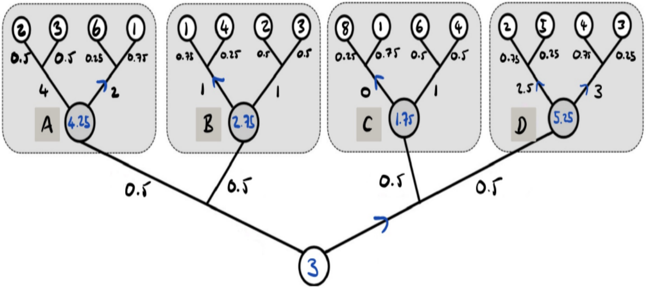
A stochastic shortest path example



- We want to find the least costly path from R to any leaf. [1]
- Leaf nodes additionally have terminal costs.
- The leaf node that you reach after taking an action is a random variable.
- Hence we calculate expected costs.

[1] Example from Neil Walton's notes on Stochastic Control

An Example



- As earlier, the original problem can be broken down into sequence of simpler problems (shaded boxes)
- Key difference is the need to take expectations.

MDP's: State, Action, Reward, State

- Lets consider discrete set of times $t = 0, 1, \ldots, T$
- Let $\mathcal{S}$ denote the state space and $\mathcal{A}$ denote the action space.
- Unless specified, we assume that $\mathcal{S}$ and $\mathcal{A}$ are countable sets.
- $S_t \in \mathcal{S}$ denotes the random state of the system at time t.
- Let A_t denote the action (possibly random) at time t
- $S_{t+1} = f_t(S_t, A_t, W_t)$ is the dynamics where W_t is i.i.d noise.
- As seen earlier, this implies state transitions are Markovian with transition probabilities $\mathbb{P}(S_{t+1} = s' | S_t = s, A_t = a) = P(s'|s, a)$
- $r(S_t, A_t)$ is the random reward from action A_t in state S_t.
- $r_T(S_T)$ denotes the reward for terminating in S_T at time T.

Types of policies

- A policy $\pi = (\pi_t : t = 0, 1, \ldots, T-1)$ specifies action $\pi_t \in \mathcal{A}$ to be taken at time t.
- If $\pi_t : t \to \mathcal{A}$, its a state independent, deterministic policy.
- If $\pi_t : t \to \mathcal{P}(\mathcal{A})$, its a state independent, randomized policy. $\mathcal{P}(\mathcal{A})$ denotes the set of probability distributions over $\mathcal{A}$.
- Let $\mathcal{H}_t = (S_{1:t}, A_{1:t-1})$. Then $\pi_t : \mathcal{H}_t \to \mathcal{P}(\mathcal{A})$, its a history dependent, randomized policy.
- If $\pi_t : (s_t, t) \to \mathcal{A}$, its a Markovian, non-stationary and deterministic policy.
- If $\pi_t : s \to \mathcal{A}$, its a Markovian, stationary and deterministic policy.
- We let Π^K denote class of policies with property K where $K \in \{HR, HD, MR, MD, DS\}$. Π denotes space of all policies.

MDP: Cumulative reward, value function

- How good is any policy $\pi = (\pi_1, \ldots, \pi_{T-1})$? Measured using expected cumulative reward (ak.a value of a policy)
- $V^\pi(s_0) := \mathbb{E}_{s_0}[r(s_0, \pi_0) + r(S_1, \pi_1) + \ldots + r_T(S_T)]$.
- $\mathbb{E}_{s_0}$ denotes expectation conditioned on starting in s_0. Sometimes, the notation $\mathbb{E}^\pi_{s_0}$ is used to denote dependence on π. We will however supress this notation throughout.
- How do we get the best policy π^*?

Problem P3: $V(s_0) := \sup_{\pi \in \Pi} V^\pi(s_0)$

- Note: the optimal policy depends on the starting state.

Optimality of Deterministic Markovian policies

Theorem

The cost incurred by the best Markovian strategy, is same as the cost incurred by the best history dependent strategy, i.e.,
$V(s_0) = \sup_{\pi \in \Pi^{HR}} V^{\pi}(s_0) = \sup_{\pi \in \Pi^{MD}} V^{\pi}(s_0)$

- Proof is outside scope of the course and uses Balckwell's result.
- See Putterman Chapter 4, Thm 4.4.2 or Aditya Mahajan notes.
- Note that the policy need not be stationary.
- We will only focus on deterministic Markovian policies henceforth.
- HW: Why supremum and not maximum ? When can you replace supremum by maximum?

MDP: Discounted criteria

- Now what if the value of money changes with time? How do you account for that?
- Let α denote a discount factor ($0 \leq \alpha \leq 1$).
- $V_{\alpha}^{\pi}(s_0) := \mathbb{E}_{s_0}\left[\sum_{t=0}^{T-1} \alpha^t r(S_t, \pi_t) + \alpha^T r_T(S_T)\right]$.
- For finite time horizon problems, the theory of with and without discounting is the same.
- We will henceforth assume $\alpha = 1$.

Subproblems

- Recall $V^{\pi}(s_0) := \mathbb{E}_{s_0}\left[\sum_{t=0}^{T-1} r(S_t, \pi_t) + r_T(S_T)\right]$
- $V(s_0) := \sup_{\pi \in \Pi^{MD}} V^{\pi}(s_0)$ and $\pi^* := \underset{\pi}{argmax}\, V^{\pi}(s_0)$
- We will now consider notations for sub-problems starting at t.
- Let $\boldsymbol{\pi}_t := (\pi_t, \ldots, \pi_{T-1})$.
- Define $V_t^{\boldsymbol{\pi}_t}(s_t) := \mathbb{E}\left[\sum_{u=t}^{T-1} r(S_u, \pi_u) + r_T(S_T)\right]$;
- $V_T^{\boldsymbol{\pi}_t}(s) = r_T(s)$ and $V_0^{\pi}(s) = V^{\pi}(s)$

Towards evaluating V^π

- Recall $V^\pi(s_0) := \mathbb{E}_{s_0}\left[\sum_{t=0}^{T-1} r(S_t, \pi_t) + r_T(S_T)\right]$
- $V_t^{\pi_t}(s_t) := \mathbb{E}\left[\sum_{u=t}^{T-1} r(S_u, \pi_u) + r_T(S_T)\right]$;
- Is there an alternative expression for $V_t^{\pi_t}(s_t)$?
- As in case of the Markov Reward process, it can be shown that

$$V_t^{\pi_t}(s_t) = r(s_t, \pi_t) + \mathbb{E}_{s,\pi_t}\left[V_{t+1}^{\pi_{t+1}}(S')\right]$$

  where $\mathbb{E}_{s,\pi_t}\left[V_{t+1}^{\pi_{t+1}}(S')\right] = \sum_j P(j|s, \pi_t) V_{t+1}^{\pi_{t+1}}(j)$
- This can be used for evaluating $V^\pi(s)$.

Policy Evaluation Algorithm

- How do we evaluate $V^\pi(s)$ for any $\pi \in \Pi^{MD}$?

Start with

$$V_T(S_T) = r_T(S_T) \quad \text{for all possible } S_T \tag{1}$$

and for $t = T - 1 \ldots 0$, set

$$V_t^{\boldsymbol{\pi}_t}(s_t) = r(s_t, \pi_t) + \mathbb{E}_{s,\pi_t}\left[V_{t+1}^{\boldsymbol{\pi}_{t+1}}(S')\right] \quad \text{for all } s_t. \tag{2}$$

Set $V^\pi(s) = V_0^\pi(s)$ for all s.

- Recall that $\mathbb{E}_{s,\pi_t}$ represents conditional expectation conditioned on $S_t = s$ and $A_t = \pi_t(s)$.

Towards Bellman optimality equations

- Recall the definition $V_t^{\boldsymbol{\pi}_t}(s_t) := \mathbb{E}\left[\sum_{u=t}^{T-1} r(S_u, \pi_u) + r_T(S_T)\right]$ which can be written as

$$V_t^{\boldsymbol{\pi}_t}(s_t) = r(s_t, \pi_t) + \mathbb{E}_{s,\pi_t}\left[V_{t+1}^{\boldsymbol{\pi}_{t+1}}(S')\right]$$

- Now define $V_t(s_t) := \max_{\boldsymbol{\pi}_t} V_t^{\boldsymbol{\pi}_t}(s_t)$

- Bellman equations related V_t with V_{t+1} which we use recursively to obtain $V_0 = V$.

Bellman Optimality Equations for the MDP

Theorem

For $t = 0, \ldots, T-1$ and all $s \in \mathcal{S}$, the following is true:

$$V_t(s) = \max_{a \in \mathcal{A}} \{r(s,a) + E_{s,a}[V_{t+1}(S')]\}$$

where $S' \in \mathcal{S}$ is the random state in the next time instant.

- Note that $E_{s,a}[V_{t+1}(S')] = \sum_j P(j|s,a) V_{t+1}(j)$
- It is this expectation term on the RHS that necessitates going backwards in time.
- Proof is HW. We have seen the necessary ingredients in the Markov reward process and deterministic dynamic programming.
- As in the deterministic case, all results hold when the transition probabilities, and reward function depend on time.

The finite horizon MDP algorithm

Start with

$$V_T(s_T) = r_T(s_T) \quad \text{for all possible } S_T \tag{3}$$

and for $t = T - 1 \ldots 0$, find

$$V_t(s_t) = \max_{a \in \mathcal{A}_{s_t}} \{r_t(s_t, a) + \mathbb{E}_{s_t,a}\left[V_{t+1}(f_t(s_t, a, W_t))\right]\} \quad \text{for all } s_t. \tag{4}$$

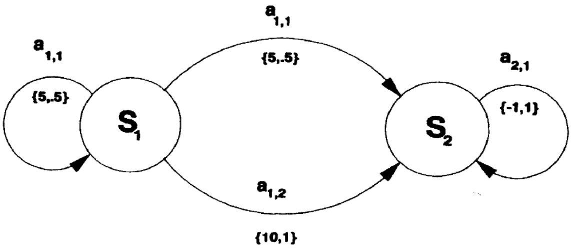
Now construct optimal policy $\pi^* = (\pi_0^*, \ldots, \pi_{T-1}^*)$ as follows by sequentially going forward for $t = 0, \ldots, T - 1$ to set

$$\pi_t^* = \underset{a \in \mathcal{A}_{s_t}}{argmax} \{r_t(s, a) + \mathbb{E}_{s_t,a}\left[V_{t+1}(f_t(s_t, a, W_t))\right]\} \tag{5}$$

Q function formulation

- One can obtain an equivalent algorithm in terms of $Q_t(s, a)$ which defines the best possible cumlative reward from starting in (s, a) at time t.
- $Q_t(s, a) = r_t(s, a) + \mathbb{E}_{s,a}\left[V_{t+1}(f_t(s, a, W_t)\right]$
- What is the relation between $V_t(s)$ and $Q_t(s, a)$?
- $V_t(s) = \max_{a \in \mathcal{A}} Q_t(s, a)$
- $\pi_t^* = \underset{a \in \mathcal{A}_{s_t}}{argmax}\, Q_t(s, a)$

Example: A Two state MDP



- $T = 1, 2, \ldots N$, $\mathcal{S} = \{s_1, s_2\}$, $\mathcal{A}_{s_1} = \{a_{1,1}, a_{1,2}\}$ and $\mathcal{A}_{s_2} = \{a_{2,1}\}$ [2]
- Rewards and transition probabilities on arrow.
- Assume that the terminal rewards are zero.
- HW: Write the Bellman Optimality Equation and find the optimal policy.

[2] Example from Puterman's MDP book