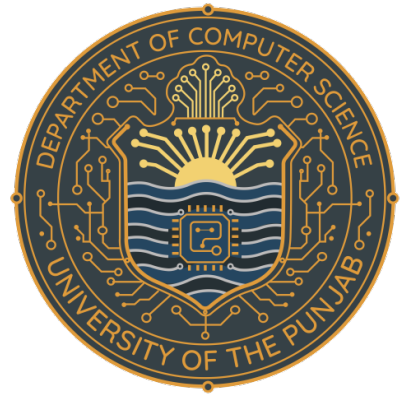
CS-566 Deep Reinforcement Learning

MDP Terminology - II



Nazar Khan
Department of Computer Science
University of the Punjab

Return R: What Are We Optimizing?

- Goal of sequential decision-making: **Find the best policy**.
- To evaluate a policy, we need a measure of *long-term success*.
- This measure is the **return**.

> Return is the sum of rewards collected along a trace.

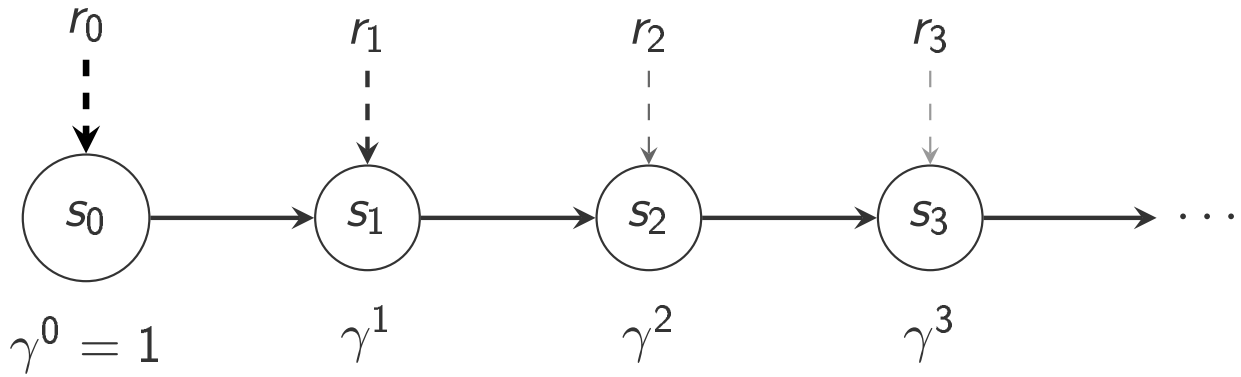
Definition of Return R

- For a trace $\tau_t = (s_t, a_t, r_t, s_{t+1}, a_{t+1}, r_{t+1}, \ldots)$
- The *return starting at time t* can be computed as

$$R(\tau_t) = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots$$

- Multiplication by $0 \leq \gamma \leq 1$ corresponds to *discounted future rewards*.
- Compact form:

$$R(\tau_t) = r_t + \sum_{i=1}^{\infty} \gamma^i r_{t+i}$$



Discount Factor γ

- $\gamma \in [0, 1]$: balances short-term vs. long-term rewards.
- Two extremes
  - $\gamma = 0$: **Myopic agent** (only immediate reward matters)
  - $\gamma = 1$: **Far-sighted agent** (all rewards equally important)
- In *infinite-horizon* tasks
  - $\gamma = 1$ can lead to unbounded returns.
  - Typically use $\gamma \approx 0.99$ to keep returns finite.

Example: Computing Return

- Assume $\gamma = 0.9$
- Rewards along trace

$$r_0 = -1$$
$$r_1 = -1$$
$$r_2 = 20$$

- Return is

$$\begin{aligned} R(\tau_0^2) &= -1 + 0.9(-1) + 0.9^2(20) \\ &= -1 - 0.9 + 16.2 \\ &= 14.3 \end{aligned}$$

Intuition: Why Discounting?

- Ensures **mathematical stability**: infinite sum remains bounded.
- Captures the idea of **time preference**:
  - Immediate rewards are more certain/valuable.
  - Future rewards are less predictable.
- γ lets us trade off:
  - Short-term exploitation
  - Long-term exploration

From Traces to Expectations

- Assume agent is in state s. How good or bad is state s?
- In stochastic environments, different states are possible for a single action.
- With stochastic policy, actions may vary as well.
- Return of a single trace from state s is not a good estimate of how good or bad state s is.
- It makes more sense to consider **expected cumulative reward** over all traces from state s.

Expectation Example in RL

- In RL, rewards are random because:
  - The policy π may be stochastic.
  - The environment transitions T may be stochastic.
- Just like the number on the die, cumulative reward is a random variable.
- Therefore, the value of state s is an **expectation over returns**:

$$V^{\pi}(s) = \mathbb{E}\Big[R(\tau) \,|\, s_0 = s, \pi\Big]$$

  where $V^{\pi}(s)$ is called the *state value function* or simply *value function*.
- Just like the unfair die rolls average to 4.5, the value function averages over many possible traces.

State Value: Definition

- The **state value function** $V^{\pi}(s)$

$$V^{\pi}(s) = \mathbb{E}_{\tau_t \sim p(\tau_t)}\left[\sum_{i=0}^{\infty} \gamma^i r_{t+i} \;\middle|\; s_t = s\right]$$

- Meaning
  - Start in state s.
  - Follow policy π.
  - Value = expected return.

> $V^{\pi}(s)$ is the *expected return* when you *start from state s* and *follow policy π*.

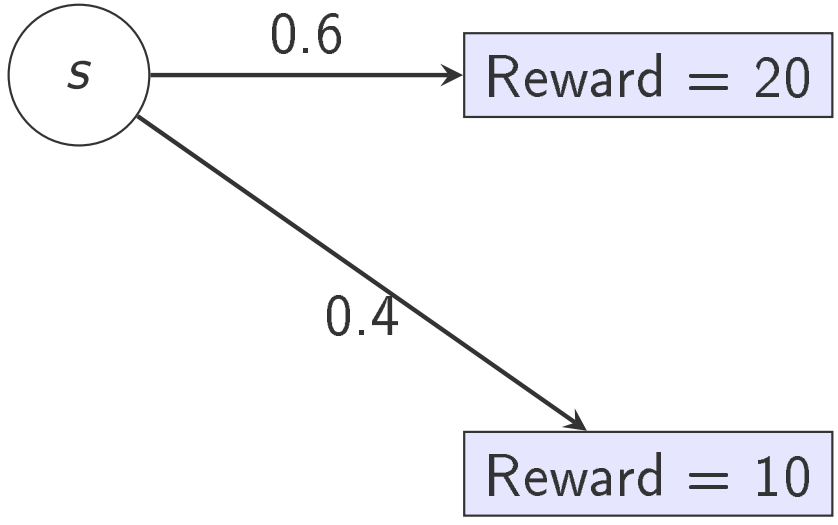
Example: Two Possible Traces

Imagine from state s under policy π:

- With probability 0.6 cumulative reward is 20
- With probability 0.4 cumulative reward is 10

Then:

$$V^{\pi}(s) = 0.6 \cdot 20 + 0.4 \cdot 10 = 16$$



State Value Function

- Every policy π defines a unique value function $V^{\pi}(s)$.
- Often we drop π and just write $V(s)$.
- $V(s)$ assigns a real number (expected return) to each state.

Example: Tabular state values (discrete state space)

State s	$V^{\pi}(s)$
1	2.0
2	4.0
3	1.0
etc.	...

Terminal States

- By definition:

$$s = \text{terminal} \quad \Rightarrow \quad V(s) = 0$$

- Once the episode ends, no further rewards are possible.

From $V(s)$ to $Q(s, a)$

- State value $V^{\pi}(s)$ is expected return starting from state s and following π.
- But sometimes we want to know the value of a *specific action* in a *specific state*.
- So we can define the *state-action value* $Q^{\pi}(s, a)$:

$$Q^{\pi}(s, a) = \mathbb{E}_{\tau_t \sim p(\tau_t)}\Big[\sum_{i=0}^{\infty} \gamma^i \cdot r_{t+i} \;\Big|\; s_t = s, a_t = a\Big]$$

- It is the expected return if we take action a in state s, then follow π.

Formal Definition

- Every policy π has exactly one associated Q-function.
- Domain and codomain:

$$Q : \underbrace{S \times A}_{\text{domain}} \to \underbrace{\mathbb{R}}_{\text{codomain}}$$

- Each state-action pair (s, a) is mapped to the expected return.
- Terminal states have no return by definition

$$s = \text{terminal} \quad \Rightarrow \quad Q(s, a) := 0, \quad \forall a$$

Intuition: Why $Q(s, a)$?

- $V(s)$ tells us how good a state is, on average, under π.
- $Q(s, a)$ tells us how good it is to *take action a* in state s.
- If we know Q, we can easily pick the best action:

$$\pi^*(s) = \arg\max_a Q^*(s, a)$$

- This is why Q is central in reinforcement learning.

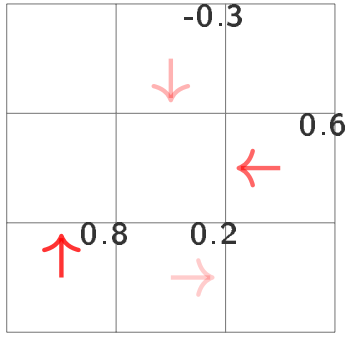
Example: Tabular Representation of $Q(s, a)$

For discrete S and A, Q can be stored in a table of size $|S| \times |A|$.

	a=up	a=down	a=left	a=right
s=1	4.0	3.0	7.0	1.0
s=2	2.0	-4.0	0.3	1.0
s=3	3.5	0.8	3.6	6.2
etc.	.	.	.	.

Visual Intuition: Grid World

- In a grid world:
  - Each state s = cell in the grid.
  - Each action a = arrow direction.
  - $Q(s, a)$ = expected value of moving in that direction.



Numerical Example: Computing $Q^\pi(s, a)$

Setup. From state s we evaluate action a_1.

- Discount: $\gamma = 0.9$
- After taking a_1 in s:
  - With prob. 0.6: get $r_0 = 2$, go to s_1; under π, next step gives $r_1 = 5$, then terminal.
  - With prob. 0.4: get $r_0 = -1$, go to s_1'; under π, next step gives $r_1 = 10$, then terminal.

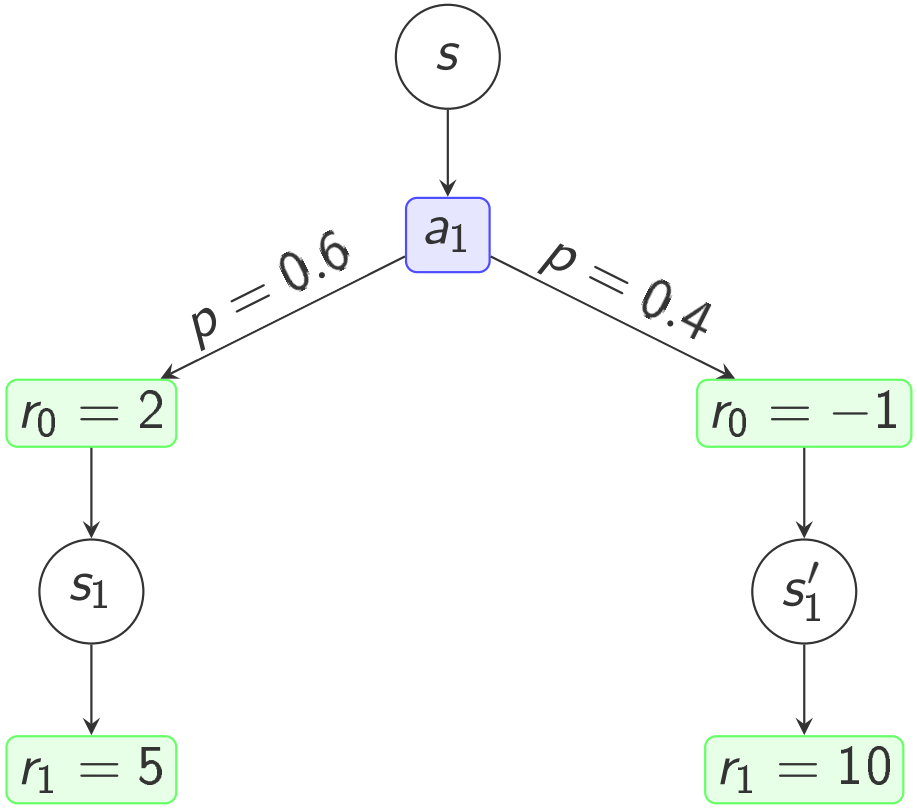
Branch returns (2-step horizon for clarity):

$$G_1 = r_0 + \gamma r_1 = 2 + 0.9 \times 5 = 6.5, \qquad G_2 = r_0 + \gamma r_1 = -1 + 0.9 \times 10 = 8.0$$

Expected return (definition of Q^π):

$$Q^\pi(s, a_1) = 0.6 \times G_1 \; + \; 0.4 \times G_2 \; = \; 0.6 \times 6.5 \; + \; 0.4 \times 8 \; = \; 7.1$$

Visualization of the Example (Branches & Probabilities)



$G_1 = 2 + 0.9 \times 5 = 6.5$ $G_2 = -1 + 0.9 \times 10 = 8.0$

$$\boxed{Q^\pi(s, a_1) = 0.6 \times 6.5 + 0.4 \times 8 = \mathbf{7.1}}$$

Compare Two Actions via Q: Choose $\arg\max_a Q(s,a)$

Action a_1 (from previous slide): $Q^\pi(s,a_1) = 7.1$.

Alternative action a_2:

- With prob. 0.7: $r_0 = 3$, next $r_1 = 2$ (then terminal)
- With prob. 0.3: $r_0 = 3$, next $r_1 = 0$ (then terminal)

$$G_1^{(a_2)} = 3 + 0.9 \times 2 = 4.8, \qquad G_2^{(a_2)} = 3 + 0.9 \times 0 = 3.0$$

$$Q^\pi(s,a_2) = 0.7 \times 4.8 + 0.3 \times 3.0 = 3.36 + 0.9 = 4.26$$

$$\pi^*(s) = \arg\max_{a \in \{a_1, a_2\}} Q^\pi(s,a)$$

$$= a_1 \text{ (since } 7.1 > 4.26)$$

Trace View and Factorization of Probabilities

Each branch is a partial trace (here: 2 time steps after taking a in s).
For longer horizons,

$$Q^{\pi}(s,a) = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i r_{t+i} \;\middle|\; s_t = s, a_t = a\right]$$

If transitions and policy are stochastic, each *full* trace probability factors as:

$$p(\tau_t) = \pi(a_t|s_t) \cdot T_{a_t}(s_t, s_{t+1}) \cdot \pi(a_{t+1}|s_{t+1}) \cdot T_{a_{t+1}}(s_{t+1}, s_{t+2}) \cdots$$

Then $Q^{\pi}(s,a)$ is the expectation of discounted returns over all such traces conditioned on $(s_t = s, a_t = a)$.