Chapter 3: Finite Markov Decision Processes (MDPs)

February 9, 2025

1 Learning Objectives

- Define MDPs and their core components: **states**, **actions**, **rewards**, **dynamics function**, and **policies**.
- Understand the **Bellman equations** for value functions and optimality.
- Differentiate between **episodic** and **continuing tasks** and calculate **returns** with/without discounting.
- Explain how **optimal policies** and **value functions** are derived.

2 Agent-Environment Interface

MDPs formalize sequential decision-making where actions affect both immediate rewards and future states.

2.1 Key Components

- **Agent**: Learner/decision-maker.
- **Environment**: Everything outside the agent.
- **State** (S_t): Representation of the environment at time t.
- **Action** (A_t): Choice made by the agent.
- **Reward** (R_{t+1}): Immediate feedback from the environment.
- **Dynamics Function** (p):

$$p(s', r \mid s, a) \doteq \Pr\{S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a\} \tag{1}$$

  Describes the probability of transitioning to state s' with reward r after taking action a in state s.

2.2 Example: Recycling Robot

- **States**: high (battery level), low.
- **Actions**: search, wait, recharge.
- **Rewards**:
  - Positive for collecting cans.
  - -3 if battery depletes.
- **Dynamics**: Transition probabilities depend on current state and action (e.g., searching with high battery has probability α to stay high).

3 Goals and Rewards

Reward Hypothesis: *All goals can be framed as maximizing cumulative reward.*

- **Reward Signal**: Immediate feedback (R_{t+1}).
- **Value Function**: Long-term expected return ($v_\pi(s)$ or $q_\pi(s,a)$).

3.1 Example

- Chess: $+1$ for win, -1 for loss, 0 otherwise.
- Pole-balancing: -1 on failure, 0 otherwise.

4 Returns and Episodes

Returns

- **Episodic Tasks**: Finite time steps (e.g., a game).

$$G_t = R_{t+1} + R_{t+2} + \cdots + R_T \tag{2}$$

- **Continuing Tasks**: Infinite horizon (e.g., robot operation).

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{3}$$

  where $\gamma \in [0,1]$ is the **discount factor**.

5 Policies and Value Functions

- **Policy** (π): Mapping from states to action probabilities.

$$\pi(a \mid s) = \Pr\{A_t = a \mid S_t = s\} \tag{4}$$

- **State-Value Function** (v_π): Expected return from state s under π:

$$v_\pi(s) = \mathbb{E}_\pi[G_t \mid S_t = s] \tag{5}$$

- **Action-Value Function** (q_π): Expected return from taking a in s:

$$q_\pi(s, a) = \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a] \tag{6}$$

6 Optimal Policies and Value Functions

- **Optimal State-Value Function**:

$$v_*(s) = \max_\pi v_\pi(s) \tag{7}$$

- **Optimal Action-Value Function**:

$$q_*(s, a) = \max_\pi q_\pi(s, a) \tag{8}$$

7 Exercises

1. Design three MDP tasks (e.g., robot navigation, stock trading).
2. Compute returns for $\gamma = 0.5$ and reward sequence $[-1, 2, 6, 3, 2]$.
3. Verify Bellman equation for Gridworld's center state.
4. Determine optimal policies for different γ in a simple MDP.