Advanced Topics in Reinforcement Learning

Lecture 3: Markov Decision Processes

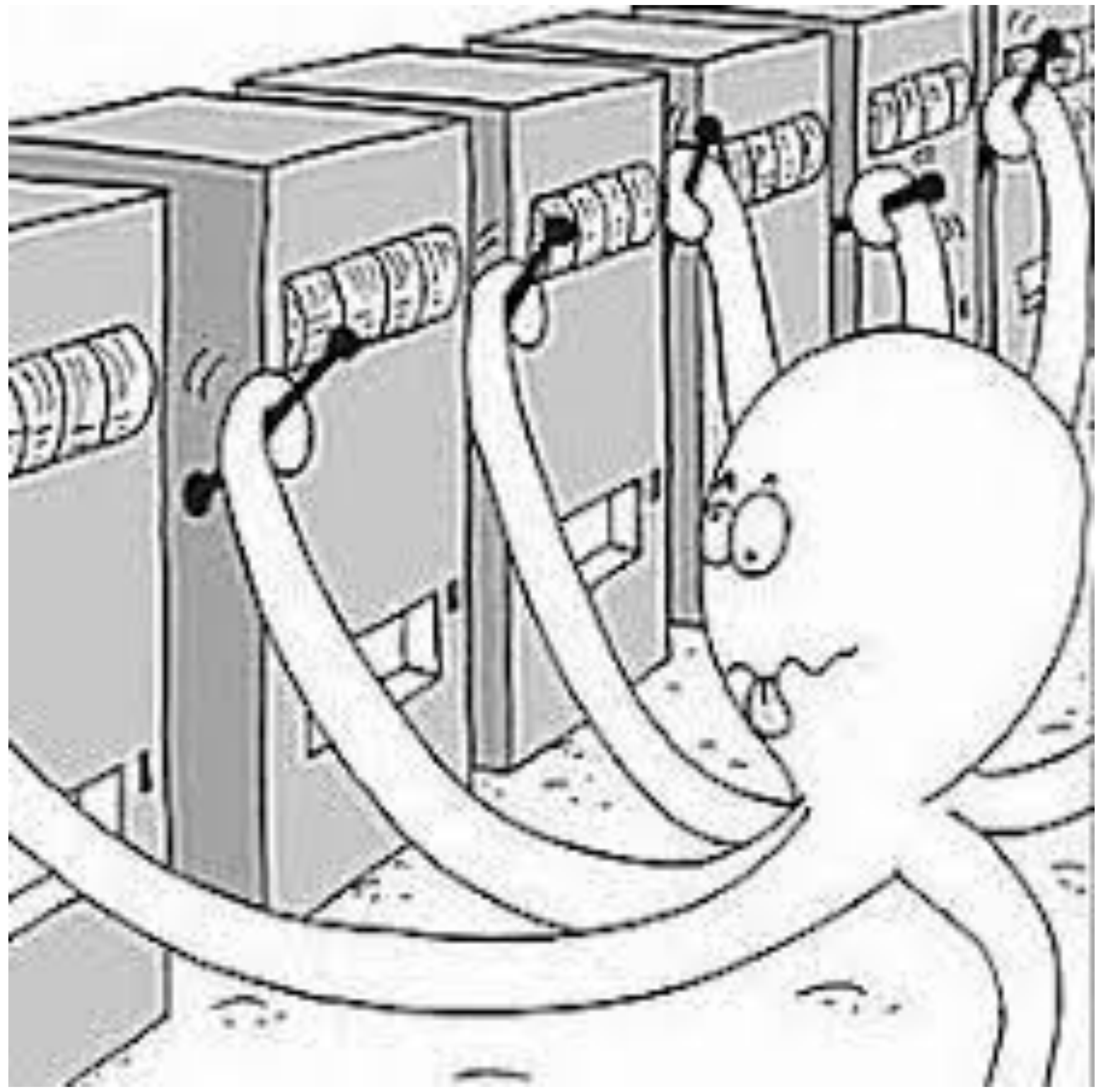
Josiah Hanna
University of Wisconsin — Madison

Announcements

- Homework 1 released on canvas; due Thursday, September 29.
- Reading Sign-Ups: https://docs.google.com/spreadsheets/d/1-dce7-qzt8EVM4gYOLIl5WzYEGpioWM4x0VyA6QimzY/edit#gid=0
- **A note on notation:** you may have noticed the book sometimes uses lower case letters and sometimes upper case.
  - Ex: equation above 3.14 uses a mixture.
  - Textbook author is following a widely used convention for denoting random variables vs fixed values. See page xix in book.

Finishing Bandits

- What if we have an infinite number of arms?
- Examples of non-stationarity?
- Vary reward distributions? Vary k?
- Why does small epsilon eventually perform best?
- Why apply optimistic initial values?



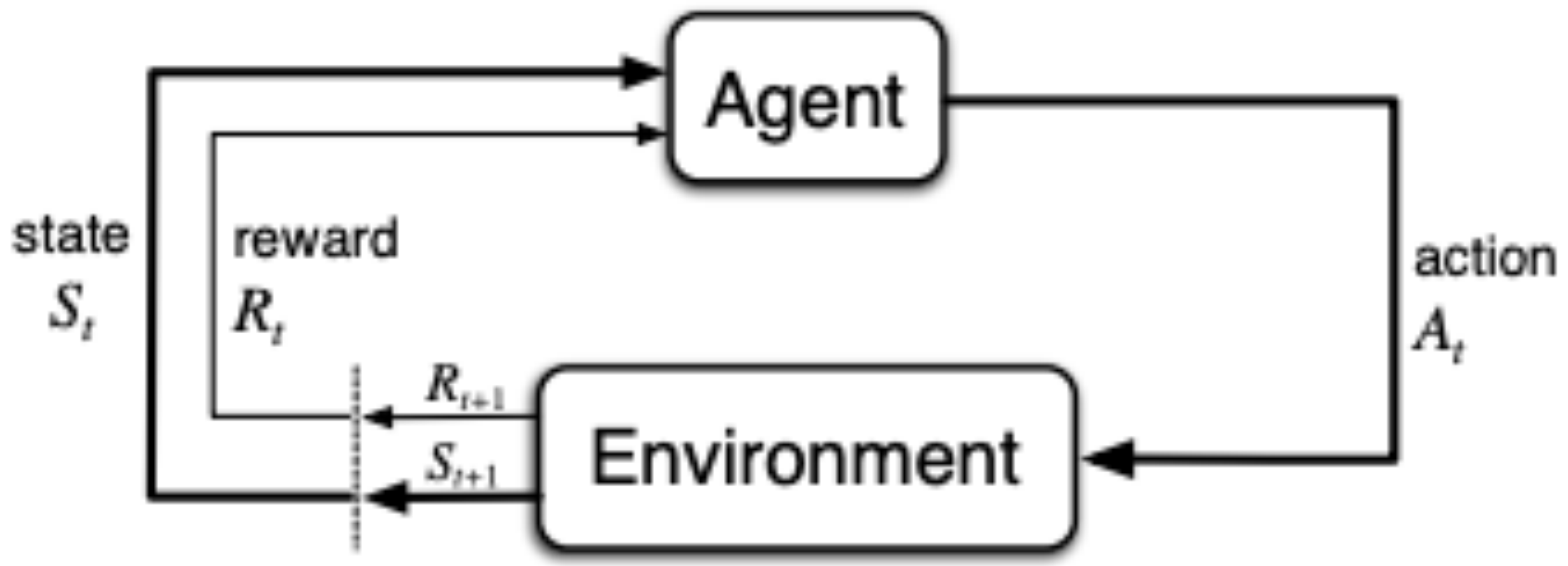
Today's Outline

- **Goal:** how to formulate RL problems.
- Formulating the RL environment.
- Formulating the RL objective.
- Value functions and policies.
- Approximations.

General Reinforcement Learning

- States: $s \in \mathcal{S}$
- Actions: $a \in \mathcal{A}$
- Rewards: $R \sim r(s, a)$
- State transitions: $S \sim p(\cdot \mid s, a)$
- Goal: Find a policy, $\pi : \mathcal{S} \rightarrow \mathcal{A}$, that maximizes cumulative reward.

General Reinforcement Learning



$$\ldots S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1}, \ldots$$

$$S_{t+1}, R_{t+1} \sim p(\cdot \mid S_t, A_t)$$

$$A_{t+1} \leftarrow \pi(S_{t+1})$$

Defining State

- Informally, state is the information available to the agent to base its decision on.
- Formally, an element of the state space, i.e., $s \in \mathcal{S}$.
- Must include information about all aspects of the past that affect the future.
- **Markov property:** future is conditionally independent of the past given current state.

$$\Pr(S_{t+1} = s, R_{t+1} = r \,|\, s_t, a_t) = \Pr(S_{t+1} = s, R_{t+1} = r \,|\, s_t, a_t, s_{t-1}, a_{t-1}, \ldots)$$

Thinking about State

- State as a collection of variables that describe the world at that moment in time.
  - For example, an autonomous vehicle's state includes the vehicle's location, where other vehicles are, road conditions, etc.
- States as elements of a finite set.
  - Simpler model to analyze.

State Examples

- Recommendation agent for a social media timeline.
- A robot with a camera and a laser range finder.
- Home thermostat system.
- Recommending medical treatment.

Limitations

- We will assume the state is given. In practice, must be manually designed or learned.
  - Possible approaches: kalman filters, recurrent neural networks, predictive state representations, frame-stacking.
- Assume reward is given. In practice, must be manually designed or learned.
  - Often the reward is tuned by practitioners.
  - Inverse Reinforcement Learning is the problem of determining a reward that makes an observed behavior optimal.
- Chapter 17 discusses both problems in more detail.

Scaling to Real World Problems

- For the next few weeks we will assume a finite MDP environment.
- MDP formalism generalizes to infinite state and action spaces.
    - Mathematically, $\mathscr{S} \subseteq \mathbb{R}^n$ and $\mathscr{A} \subseteq \mathbb{R}^k$ for some integers n and k.
    - Replace probability mass functions with probability density functions.
    - Replace summations with integrals.
- Policies, value functions, episodes, and returns will still be well-defined.
- Can also consider continuous time processes, however, rarely done in RL practice on digital computers.

Defining Reward

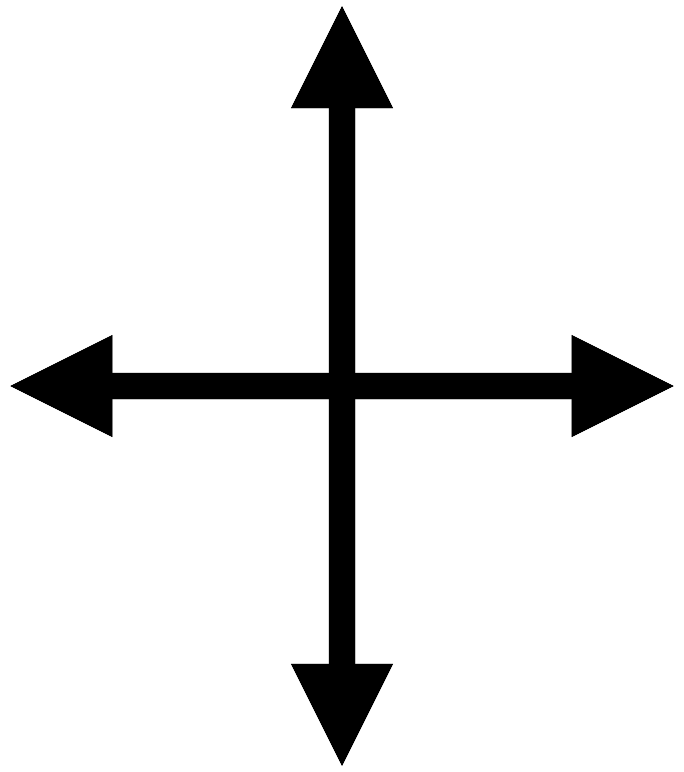
- The agent's objective is to maximize its cumulative reward.
- Expected reward, $r(s, a)$, gives immediate benefit or cost of taking action a in state s.
- What to achieve not how to achieve it.
- In practice, reward often used to guide learning agent ("shaping" reward).

Returns and Episodes

- The **return** is the sum of future rewards: $G_t := R_{t+1} + R_{t+2} + \ldots + R_T$.
- Episodes are subsequences of interaction that begin in some initial state and end in a special terminal state.
- **The initial state of one episode is independent of interaction in the preceding episode.**
- If no termination, the return is: $G_t := R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+2} + \ldots$
- Recursive definition: $G_t = R_{t+1} + \gamma G_{t+1}$.

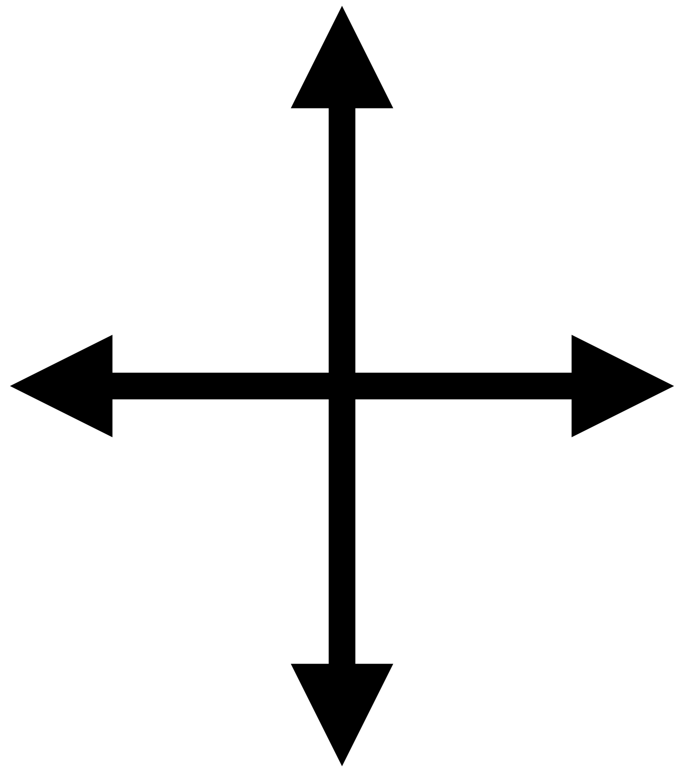
Reward Examples

0	0	0	Start
0	0	0	0
0	0	0	0
0			
0	0	0	+1



Reward Examples

0	0	0	Start
0	0	0	0.1
0.5	0.4	0.3	0.2
0.6			
0.7	0.8	0.9	+1



Reward Examples

- Recommendation agent for a social media timeline.
- An autonomous vehicle learning to drive.
- Home thermostat system.
- Recommending medical treatment.

Policies

- The agent's decision making rule.
- Formally, a function outputting the conditional probability of selecting an action in a particular state: $\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0,1]$.
- A deterministic policy is a function mapping states to actions: $\pi : \mathcal{S} \rightarrow \mathcal{A}$.
- Specifically, these are **Markovian policies;** action selection only depends on current state.

Value functions

- State transitions and rewards are stochastic so we must maximize **expected return.**

- Expected return is only well-defined with respect to a particular policy. (Why?)

- State-value and action-value functions are always defined in terms of some policy.

$$v_\pi(s) = \mathbb{E}_\pi[G_t \,|\, S_t = s] = \mathbb{E}_\pi[\sum_{k=0} \gamma^k R_{t+k+1} \,|\, S_t = s]$$

$$q_\pi(s, a) = \mathbb{E}_\pi[G_t \,|\, S_t = s, A_t = a] = \mathbb{E}_\pi[\sum_{k=0} \gamma^k R_{t+k+1} \,|\, S_t = s, A_t = a]$$

Recursive Relationship of State Values

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

Definition of return

$$= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} \mid S_t = s]$$

Definition of expectation

$$= \sum_a \pi(a \mid s) \sum_{s'} \sum_r p(s', r \mid s, a)[r + \gamma \mathbb{E}_\pi[G_{t+1} \mid S_{t+1} = s']]$$

Definition of state-value

$$= \sum_a \pi(a \mid s) \sum_{s'} \sum_r p(s', r \mid s, a)[r + \gamma v_\pi(s')]$$

Page 59 of the course textbook.

Action Values

Write action-values in terms of environment dynamics and state-values:

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

Definition of return $$= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} \,|\, S_t = s, A_t = a]$$

Definition of expectation $$= \sum_{s'} \sum_{r} p(s', r \,|\, s, a)[r + \gamma \mathbb{E}_\pi[G_{t+1} \,|\, S_{t+1} = s']]$$

Definition of state-value $$= \sum_{s'} \sum_{r} p(s', r \,|\, s, a)[r + \gamma v_\pi(s')]$$

Exercise 3.13, page 58.

Action Values

Write state-values in terms of action-values:

From previous slide

$$q_\pi(s,a) = \sum_{s'} \sum_{r} p(s', r \mid s, a)[r + \gamma v_\pi(s')]$$

From two slides back.

$$v_\pi(s) = \sum_{a} \pi(a \mid s) \underbrace{\sum_{s'} \sum_{r} p(s', r \mid s, a)[r + \gamma v_\pi(s')]}_{q_\pi(s,a)}$$

$$v_\pi(s) = \sum_{a} \pi(a \mid s) q_\pi(s, a)$$

Exercise 3.12, page 58.

Golf Example

- State is ball location. Actions are putt (short distance, accurate) or drive ball (long distance, less accurate).
- Reward is -1 until the ball goes in the hole.
- What is value of policy that always putts?

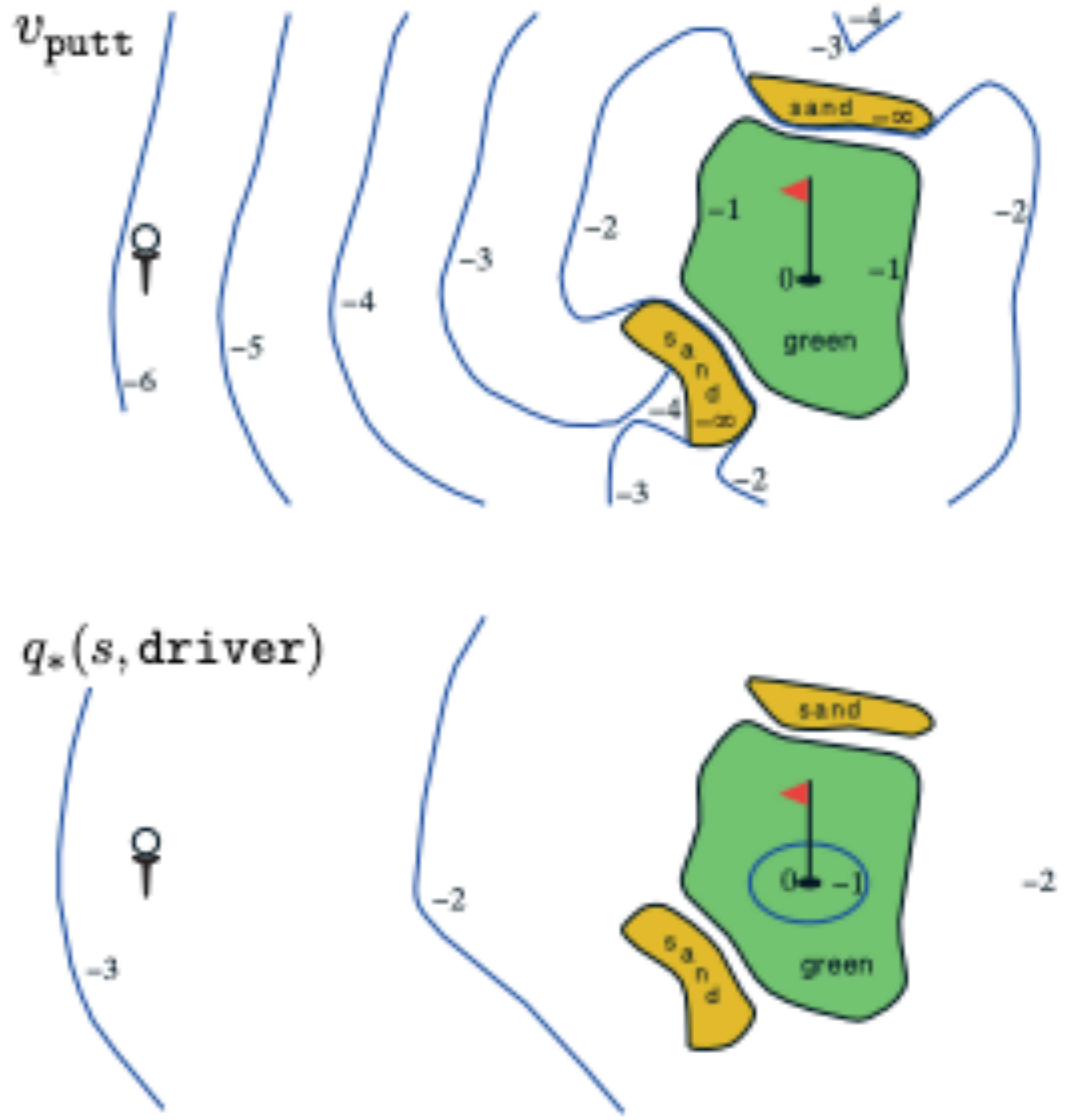


Figure 3.3: A golf example: the state-value function for putting (upper) and the optimal action-value function for using the driver (lower). ■

Optimality

- Agent's objective: find policy that maximizes $v_\pi(s)$ for all s.
- The optimal policy — policy that has maximal value in all states. $\pi^\star \geq \pi$ if $v_{\pi^\star} \geq v_\pi(s)$ for all states and possible policies.
- Possibly multiple, always at least one, deterministic, Markovian optimal policies in a finite MDP.
- $\pi^\star(s) = \arg\max_a q_{\pi^\star}(s, a)$ $\quad q_{\pi^\star}(s, a) = \mathbb{E}[R_{t+1} + \gamma v_{\pi^\star}(S_{t+1}) \,|\, S_t = s, A_t = a]$

Optimality

$$
\begin{aligned}
v_*(s) &= \max_{a\in\mathcal{A}(s)} q_{\pi_*}(s,a) \\
&= \max_a \mathbb{E}_{\pi_*}[G_t \mid S_t = s, A_t = a] \\
&= \max_a \mathbb{E}_{\pi_*}[R_{t+1} + \gamma G_{t+1} \mid S_t = s, A_t = a] \\
&= \max_a \mathbb{E}[R_{t+1} + \gamma v_*(S_{t+1}) \mid S_t = s, A_t = a] \\
&= \max_a \sum_{s',r} p(s', r \mid s, a)\big[r + \gamma v_*(s')\big].
\end{aligned}
$$

Optimal policy must choose action with highest expected return.

Definition of action-value function.

Recursive definition of return.

Definition of optimal state value function.

Definition of expectation.

Golf Example

- State is ball location. Actions are putt (short distance, accurate) or drive ball (long distance, less accurate).
- Reward is -1 until the ball goes in the hole.
- What is action-value of using driver and then following the optimal policy?

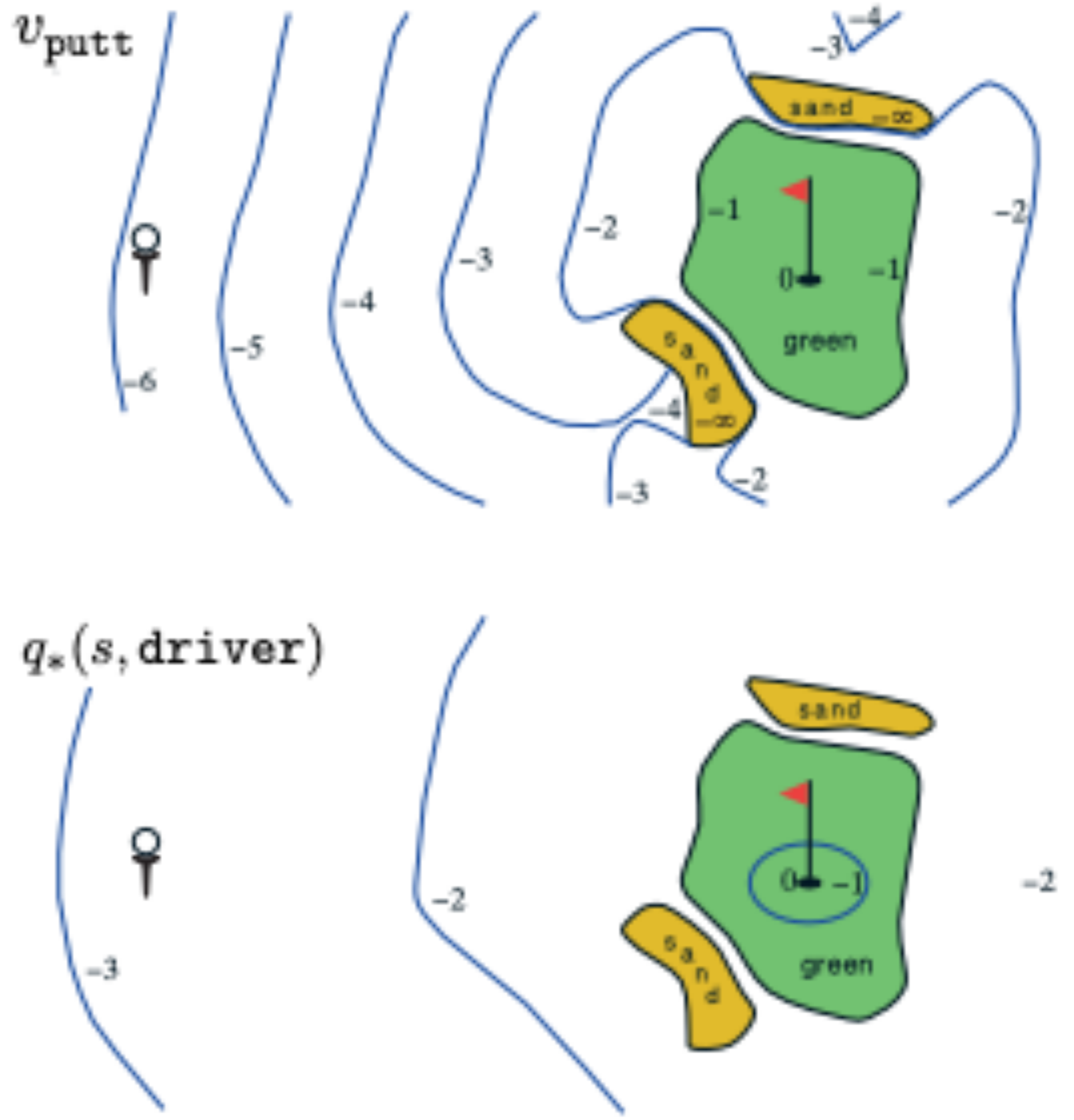


Figure 3.3: A golf example: the state-value function for putting (upper) and the optimal action-value function for using the driver (lower). ■

Approximation

- The optimal policy exists but, in practice, it may not be possible to compute.
- In real world problems, we must settle for approximate optimality.
- This is an opportunity — no need to waste time finding optimal actions in states the agent rarely visits.
- Need to generalize knowledge across states — more on this in October!

Summary

- Agent's state must include all information from past that is needed to predict the future — Markov property.
- The agent's objective is to maximize the cumulative discounted sum of a given reward function.
- Agent's behavior is a policy that maps states to actions.
- The value of a policy in a given state is the expected return from that state.
- The optimal policy maximizes the value function in all states.

Action Items

- Homework 1 now released. Due September 29 @ 9:29 am.
- Read chapter 4 and send responses for next week.
- Presentation sign-ups (posted on Piazza) if you haven’t already.