

Lecture 2: Markov Decision Processes

CS 639: Introduction to Reinforcement Learning

Tengyang Xie

Tuesday, September 8, 2026

Warm-up: five successes, two questions

	What it did	Which one?	Which challenge?
Deep Blue, 1997	Beat Kasparov by searching 200 million chess positions a second with a hand-crafted evaluation.		
AlphaGo, stage 1	Learned to predict the expert's move from 30 million positions in human games.		
AlphaGo, stage 2	Millions of games against itself, scored only at the end; learned to evaluate positions.		
Four-legged robot	Learns to walk by trial on real ground; falls cost time and hardware; no one writes down its physics.		
Chatbot	Writes a reply; people pick the better of two replies; it is trained to write the kind people pick.		

The four: supervised learning, contextual bandit, planning, reinforcement learning.

The three challenges: credit assignment, exploration, generalization.

Housekeeping

- HW0 is due Fri 09/11 on Gradescope. Everyone submits; it is not graded.
- Prerequisite waiver: send your HW0 answers and your transcript to tx@cs.wisc.edu by tonight.
- Office hours have started
 - Tengyang Xie: Wed 1–2pm, MH 5530
 - Sarina Xi: Mon 9–11am, MH 5683
 - Yurun Yuan: Fri 9–11am, MH 5683
- Questions: Piazza first; private posts are fine.
- **Quiz on 9/15: First 15 minutes of class; multiple-choice and fill-in-the-blank questions only.**

A small shop

- One product. The shelf holds at most 3 units. Each week:
 1. Observe: on Monday morning, count the units on the shelf.
 2. Order: choose how many more units to order; they arrive at once. Stock after delivery ≤ 3 .
 3. Sell: customers ask for D units, independent of the past and of your order; you sell what you can:
$$\Pr(D = 0) = \frac{1}{4}, \quad \Pr(D = 1) = \frac{1}{2}, \quad \Pr(D = 2) = \frac{1}{4}.$$
 4. Carry over: unmet demand is lost; unsold units wait on the shelf for next Monday.
- You order before you see the week's demand. One time step = one week.

Example week: 1 on Monday $\rightarrow$ order 2 $\rightarrow$ 3 on the shelf $\rightarrow$ demand 1, sell 1 $\rightarrow$ 2 next Monday

The stock you see is the state, the order is the action, the demand is chance that happens after you act.

The shop as an MDP

s = Monday's stock before ordering; a = the number of units ordered.

The reward for a week is its profit:

- Sales revenue: 8 per unit actually sold.
- Ordering cost: 0 if $a = 0$; otherwise $4 + 2a$.
- Holding cost: 1 per unit on the shelf right after delivery, charged before the week's sales.

1. What are the states S ? Which orders $A(s)$ are allowed in state s ?
2. Monday's stock is $s = 1$ and you order $a = 1$. Complete the table:

Demand D	Probability	Units sold	Stock next Monday, s'	Profit this week, R
0	1/4	?	?	?
1	1/2	?	?	?
2	1/4	?	?	?

3. From the table: $P(s'|1,1)$ for each $s' \in \{0,1,2,3\}$, and the expected profit $r(1,1)$.

Debrief: the table first

$S = \{0,1,2,3\}$. $A(s) = \{0,1, \dots, 3 - s\}$: no ordering past the shelf.

$s = 1, a = 1$: 2 units on the shelf after delivery. Ordering cost $4 + 2 \cdot 1 = 6$, holding cost 2: the week costs 8 whatever D is.

Demand D	Probability	Units sold	Stock next Monday, s'	Profit this week, R
0	1/4	0	2	-8
1	1/2	1	1	0
2	1/4	2	0	8

$P(\cdot | 1,1) = \left(\frac{1}{4}, \frac{1}{2}, \frac{1}{4}, 0\right)$ over $s' = 0,1,2,3$.

$$r(1,1) = \frac{1}{4}(-8) + \frac{1}{2}(0) + \frac{1}{4}(8) = 0.$$

The profit you see is -8, 0, or 8. $r(1,1) = 0$ is their average, weighted by probability:

$$r(s,a) = \mathbb{E}[\text{profit} | s, a]$$

The whole model in two tables

Every (s, a) works like the row you filled in: list the three demands, average the profit.

Expected units sold: $\mathbb{E}[\min(s + a, D)] = 0, \frac{3}{4}, 1, 1$ for $s + a = 0, 1, 2, 3$.

$r(s, a)$ expected profit	$a = 0$	$a = 1$	$a = 2$	$a = 3$
$s = 0$	0	-1	-2	-5
$s = 1$	5	0	-3	-
$s = 2$	6	-1	-	-
$s = 3$	5	-	-	-

$P(s' s, a)$ rows $s + a$	$s' = 0$	$s' = 1$	$s' = 2$	$s' = 3$
$s + a = 0$	1	0	0	0
$s + a = 1$	3/4	1/4	0	0
$s + a = 2$	1/4	1/2	1/4	0
$s + a = 3$	0	1/4	1/2	1/4

P depends on (s, a) only through $s + a$: ten state–action pairs, four distinct rows. $P(\cdot | 1, 1)$ is the row $s + a = 2$.

Each week's profit depends on the demand, hence on s' ; $r(s, a)$ averages it out, as $r(1, 1) = 0$ did. Same tables whatever the objective.

From the table to the general rules

- $S_{t+1} = \max(S_t + A_t - D_{t+1}, 0)$
- $R_{t+1} = 8 \cdot \min(S_t + A_t, D_{t+1}) - c(A_t) - (S_t + A_t)$
 - $S_t + A_t$ is the stock after delivery; $c(0) = 0$ and $c(a) = 4 + 2a$ for $a \geq 1$ is the ordering cost.
- D_{t+1} is the demand between decision t and decision $t + 1$: the index says when it becomes known.
- Check: $S_t = 1, A_t = 1, D_{t+1} = 1$ gives $S_{t+1} = 1$ and $R_{t+1} = 8 - 6 - 2 = 0$.
- $r(s, a) = \mathbb{E}[R_{t+1} \mid S_t = s, A_t = a] = 8 \cdot \mathbb{E}[\min(s + a, D)] - c(a) - (s + a)$: the average over the demand, as in the table; $r(1,1) = 0$.

Same shop, different objectives

- (a) The shop closes for good after three weeks; leftover stock is worth nothing.
 - Maximize $\mathbb{E}[R_1 + R_2 + R_3]$.
- (b) The shop stays open indefinitely.
 - Maximize $\mathbb{E}[\sum_{t \geq 0} \gamma^t R_{t+1}]$ with $\gamma = 0.9$.
- Same S , $A(s)$, P , and r . How do the two objectives differ? What must a good ordering rule know in each case?

Is the proposed state Markov?

- (a) Chess against a fixed computer opponent: the board position and whose turn it is.
- (b) A robot arm: the current camera image only.
- (c) The shop: the stock level, when demand depends on the month.
- (d) A language model writing a reply: the prompt plus every token generated so far.
- Hands: Markov, not Markov, or it depends. What would you add?

Before you leave, and next time

- Next time, Thu 09/10: value functions. How good is a fixed policy?
- Reading: Sutton & Barto, Sections 3.4–3.5.
- HW0 is due Fri 09/11.
- Quiz 1: Tue 09/15, first 15 minutes of class, closed book.
 - Best 5 of 6 quizzes count; no make-ups.