

Lecture 2: Markov Decision Processes

How to use these notes. Board content in board order: figures, equations, tables, one-line statements. The slides carry the housekeeping and the pair exercise. Reading: [Sutton and Barto \(2018, Sections 3.1–3.4\)](#); Section 3.5 prepares Lecture 3. Optional: [Puterman \(1994, Chapter 2 and Section 3.2\)](#) for the inventory model in full generality. Throughout the course, the notes draw on the public materials of Stanford CS 234 ([Brunskill, 2026](#)), UIUC CS 443 ([Jiang, 2026](#)), and Cornell CS 4789 ([Sun, 2025](#)); some examples are adapted from them.

The Object Every Algorithm Runs On

- Lecture 1: a five-node map with an unreliable B-road. We priced the uncertainty by an expectation and called the result an MDP without defining it.
- Today: define the object every method in this course runs on; say what a “state” must satisfy; write the objective under the three horizon conventions.

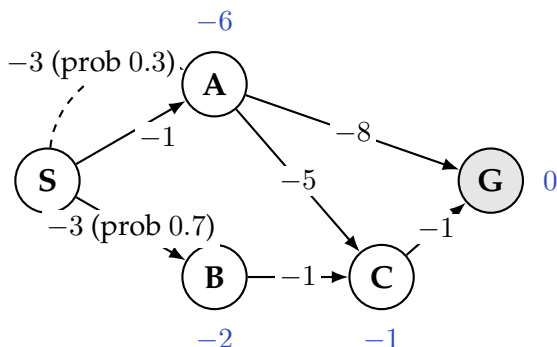
What has to be written down before any algorithm can run, and what must a state satisfy for the one-step model to be honest?

Today’s targets.

1. Write a decision problem as $(\mathcal{S}, \mathcal{A}, P, r, \gamma, d_0)$; check that every row of P sums to one.
2. State the Markov property; test a proposed state with two histories; repair it.
3. Write the probability of a trajectory and its return under a given γ ; say what a policy is and count the deterministic ones.
4. Choose among the episodic, discounted, and fixed-horizon conventions; say when $\gamma = 1$ is safe; tell termination from truncation.
5. Formulate Gridworld and inventory control completely: one transition row and one expected reward each.

From the Map to a Definition

Rewards from today on: the reward of a road is minus its travel time.



$$\begin{aligned}
 V^*(S) &= \max \left\{ -1 + V^*(A), \right. \\
 &\quad \left. -3 + 0.7 V^*(B) + 0.3 V^*(A) \right\} \\
 &= \max \{-7, -6.2\} = -6.2.
 \end{aligned}$$

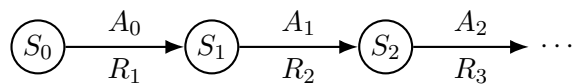
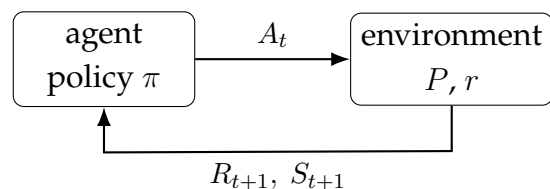
- Same numbers, opposite sign, max for min. Rewards and max for the rest of the course.
- The B-road made the map an MDP: an action that fixes a *distribution* over next nodes.

Finite Markov decision process $M = (\mathcal{S}, \mathcal{A}, P, r, \gamma, d_0)$

- $\mathcal{S}$: finite set of *states*; $\mathcal{A}$: finite set of *actions*; a state-dependent $\mathcal{A}(s)$ is allowed.
- $P(s' | s, a)$: probability of s' after action a in state s ; $P(\cdot | s, a)$ is a distribution over $\mathcal{S}$ for every (s, a) .
- $r(s, a) := \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$: *expected* immediate reward.
- $\gamma \in [0, 1]$: discount factor; $\gamma = 1$ only in episodic or fixed-horizon tasks.
- d_0 : distribution of the start state S_0 .

Interaction protocol

$$\begin{aligned}
 S_0 &\sim d_0; & \text{then for } t = 0, 1, 2, \dots : \\
 A_t &\sim \pi(\cdot | S_t), & S_{t+1} \sim P(\cdot | S_t, A_t), \\
 R_{t+1} &\text{ is received, with } \mathbb{E}[R_{t+1} | S_t, A_t] = r(S_t, A_t).
 \end{aligned}$$



Index convention: the reward that follows A_t is R_{t+1} .

One step of the protocol; then $t + 1$.

- The map as a tuple: $\mathcal{S} = \{S, A, B, C, G\}$; $\mathcal{A}(S) = \{\text{A-road, B-road}\}$; $P(B | S, \text{B-road}) = 0.7$, $P(A | S, \text{B-road}) = 0.3$; $r(S, \text{B-road}) = -3$; $\gamma = 1$; $d_0 = \delta_S$; G terminal.
- P is a table: $|\mathcal{S}| \times |\mathcal{A}|$ rows, $|\mathcal{S}|$ columns, every row a distribution. r : one number per row. An MDP is written down when both tables are filled.

- r is an expectation because R_{t+1} may depend on S_{t+1} (Gridworld exits) or on outside noise (this week's demand). Expected returns see only r :

$$\mathbb{E}[R_{t+1}] = \mathbb{E}[\mathbb{E}[R_{t+1} \mid S_t, A_t]] = \mathbb{E}[r(S_t, A_t)].$$

Reward noise matters when learning from samples: more noise, same objective.

- γ and d_0 belong to the objective, not to the physics. Change γ : what “good” means (cash-out versus invest, Lecture 1). Change d_0 : which states matter.

(s, a)	$s' = S$	A	B	C	G	$r(s, a)$
$(S, A\text{-road})$	0	1	0	0	0	-1
$(S, B\text{-road})$	0	0.3	0.7	0	0	-3
$(A, \text{to } C)$	0	0	0	1	0	-5
$(A, \text{to } G)$	0	0	0	0	1	-8
$(B, \text{to } C)$	0	0	0	1	0	-1
$(C, \text{to } G)$	0	0	0	0	1	-1
(G, any)	0	0	0	0	1	0

The whole map as tables: seven rows of P , each summing to one, and the column r . G is absorbing.

Sampled versus expected reward. Sutton and Barto (2018, Section 3.1) start from the joint kernel $p(s', r \mid s, a)$; a transition reward $r(s, a, s')$ is another common form. Both reduce to ours:

$$P(s' \mid s, a) = \sum_r p(s', r \mid s, a), \quad r(s, a) = \sum_{s', r} r p(s', r \mid s, a);$$

$$r(s, a) = \sum_{s'} P(s' \mid s, a) r(s, a, s').$$

We keep P and r .

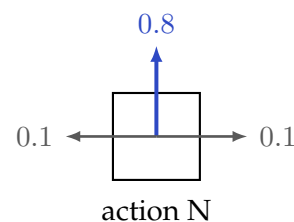
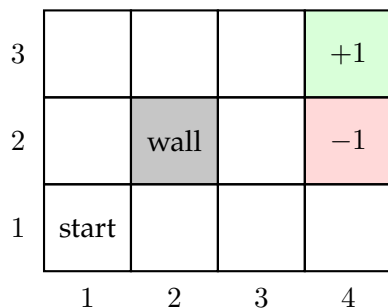
Finite is an assumption. CartPole: $\mathcal{S} \subset \mathbb{R}^4$ (cart position, cart velocity, pole angle, angular velocity); $\mathcal{A} = \{\text{left}, \text{right}\}$; $R_{t+1} = 1$ while the pole is within 12° of upright. Protocol, Markov property, return: unchanged. $P(\cdot \mid s, a)$ is a probability distribution over $\mathbb{R}^4$ (for CartPole's deterministic physics, all of its mass on one point) and $\sum_{\mathcal{S}}$ becomes $\int$. What breaks: anything that fills a table. Function approximation, later.

Notation for the semester.

Symbol	Meaning	Convention
t	time index	$t = 0, 1, 2, \dots; S_0 \sim d_0$
S_t, A_t, R_{t+1}	state, action, reward as random variables	uppercase random, lowercase realized; the reward that follows A_t carries index $t + 1$
$P(s' s, a)$	transition probability	one distribution over $\mathcal{S}$ per (s, a)
$r(s, a)$	expected immediate reward	$r(s, a) = \mathbb{E}[R_{t+1} S_t = s, A_t = a]$
$\pi(a s), \pi(s), \pi_t$	policy	stochastic; deterministic; time-dependent (fixed horizon only)
τ	trajectory	$(S_0, A_0, S_1, A_1, \dots)$, the states and actions; the rewards $R_1, R_2, \dots$ are recorded alongside
T	terminal time of an episode	random: the step at which a terminal state is entered
H	fixed horizon	a number given with the problem; exactly H decisions
G_t	return	$\sum_{k \geq 0} \gamma^k R_{t+k+1}; G_t = R_{t+1} + \gamma G_{t+1}$
$J(\pi)$	objective	$\mathbb{E}_\pi[G_0]$ with $S_0 \sim d_0$

Worked Model I: Gridworld

Adapted from the 3×4 grid of [Russell and Norvig \(2020, Section 17.1\)](#): the cells, slips, and exits are theirs; the rewards below are attached to moves, and $\gamma = 0.9$.



Where one action can lead.

- $\mathcal{S}$: eleven cells (x, y) , $x \in \{1, \dots, 4\}$, $y \in \{1, 2, 3\}$, minus the wall $(2, 2)$. Nine *decision* cells; $(4, 3)$ and $(4, 2)$ *terminal*.
- $\mathcal{A} = \{N, S, E, W\}$ in every cell.
- P : 0.8 to the intended neighbor, 0.1 to each perpendicular neighbor; a move into the wall or off the grid stays put.
- $R_{t+1} = -0.04$ for every move, plus +1 if the move enters $(4, 3)$ and -1 if it enters $(4, 2)$: entering an exit pays 0.96 or -1.04 . From an exit, every reward is 0 (below).

- $\gamma = 0.9$, the course's standard for this grid; the textbook uses $\gamma = 1$ (below).
- d_0 : all mass on $(1, 1)$.

Two rows of P , and one entry of r

$$P(\cdot \mid (1, 1), W) : (1, 1) \mapsto 0.9, (1, 2) \mapsto 0.1$$

$$P(\cdot \mid (3, 3), E) : (4, 3) \mapsto 0.8, (3, 2) \mapsto 0.1, (3, 3) \mapsto 0.1$$

$$r(s, a) = -0.04 + P((4, 3) \mid s, a) - P((4, 2) \mid s, a) \quad \text{for decision cells } s$$

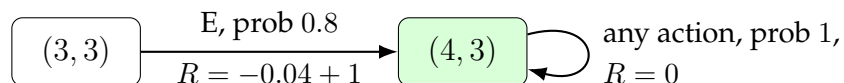
$$r((3, 3), E) = 0.8(0.96) + 0.1(-0.04) + 0.1(-0.04) = 0.76$$

- Row 1: W and the slip S bump; only the slip N moves. Row 2: the slip N bumps; the slip S moves down.
- R_{t+1} depends on where the slip lands; $r(s, a)$ averages it. This is the expectation in the definition.
- Each row: at most three nonzero entries, sum one. $11 \times 4 = 44$ rows in all.

Example. The row $P(\cdot \mid (3, 2), E)$ and the entry $r((3, 2), E)$?

$(4, 2) \mapsto 0.8, (3, 3) \mapsto 0.1, (3, 1) \mapsto 0.1$; $r((3, 2), E) = -0.04 - 0.8 = -0.84$: the row aimed at the -1 exit.

Terminal cells are absorbing.



The exit pays once, on entry. Afterwards every action leads back with reward 0.

- A terminal state is a state like any other: $P(s \mid s, a) = 1$ and $r(s, a) = 0$ for every a . The episode “ends” at the random entry time T ; no special case in the formulas, all later rewards are 0.
- The exit reward is paid on the transition *into* the exit, not for being there. Paid forever, it would be a different problem.

One trajectory, three returns.

t	0	1	2	3	4	5
S_t	(1, 1)	(1, 2)	(1, 3)	(2, 3)	(3, 3)	(4, 3)
A_t	N	N	E	E	E	
R_{t+1}	-0.04	-0.04	-0.04	-0.04	+0.96	

$$G_0 = -0.04(1 + \gamma + \gamma^2 + \gamma^3) + 0.96\gamma^4 = \begin{cases} 0.80 & \gamma = 1, \\ 0.49 & \gamma = 0.9, \\ -0.015 & \gamma = 0.5. \end{cases}$$

- $\gamma = 0.5$: four -0.04 's arrive early, the $+0.96$ arrives times $1/16$; a successful run scores below zero.
- γ is part of the objective. Choose it so that returns rank policies as you would; then evaluate with the same γ .

Policies. A policy is one more table, over the decision cells.

3				+1
2				-1
1				
	1	2	3	4

A deterministic policy π_0 : one arrow per decision cell.

- Deterministic policy: one action per decision cell, a table; $4^9 = 262,144$ of them.
- Stochastic policy: one distribution per cell; uniformly random, $\pi(a | s) = 1/4$; or “mostly N.”
- A policy names an action in *every* state, including cells it never means to visit: slips take you there (Lecture 1’s diverted traveler).
- The same table at every step: a *stationary* policy, the default all semester. A rule that changes with the step, π_t , appears only under a fixed horizon, in the horizon section.
- Whether π_0 is any good: Lecture 3.

An MDP is finished when its tables are filled. P : one distribution per (s, a) . r : one number per (s, a) . d_0 : one distribution. γ or a horizon: one choice. A policy is a table of the same kind, and it is the object we are looking for.

Worked Model II: Inventory Control

Lecture 1, exercise (d): ordering with the demand distribution unknown is reinforcement learning. Demand known: an MDP to plan in. Write it down.

Pair exercise: a store as an MDP. A shop sells one product from a shelf that holds at most 3 units. Every Monday morning the manager sees the stock s , orders a units, which arrive at once, and then meets the week's demand D , which is 0, 1, or 2 units with probabilities $\frac{1}{4}, \frac{1}{2}, \frac{1}{4}$. Demand beyond the stock is lost. Each unit sold brings in 8. An order of $a > 0$ units costs $4 + 2a$. Every unit on the shelf after the delivery costs 1 to hold for the week.

1. Write $\mathcal{S}$ and the actions $\mathcal{A}(s)$ available in state s .
2. Monday's stock is $s = 1$ and you order $a = 1$. For each demand $D = 0, 1, 2$: units sold, next Monday's stock, this week's profit.
3. From those rows: $P(\cdot \mid 1, 1)$ and the expected profit $r(1, 1)$.
4. Does this task end? Which convention fits a shop that closes after a three-week season, and which fits a shop that stays open?

One way to write it.

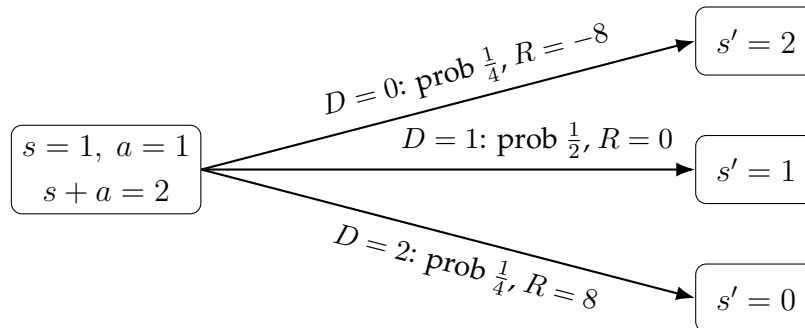
- $\mathcal{S} = \{0, 1, 2, 3\}$, the stock on Monday morning. $\mathcal{A}(s) = \{0, 1, \dots, 3 - s\}$: no ordering past the shelf.
- $S_t + A_t$ is the stock after delivery; sales are $\min(S_t + A_t, D_{t+1})$:

$$S_{t+1} = \max(S_t + A_t - D_{t+1}, 0),$$

$$R_{t+1} = 8 \min(S_t + A_t, D_{t+1}) - (4 + 2A_t) \mathbb{1}\{A_t > 0\} - (S_t + A_t).$$

- $\mathbb{E}[\min(s + a, D)] = 0, \frac{3}{4}, 1, 1$ for $s + a = 0, 1, 2, 3$, so

$$r(s, a) = 8 \mathbb{E}[\min(s + a, D)] - (4 + 2a) \mathbb{1}\{a > 0\} - (s + a), \quad r(1, 1) = 8 \cdot 1 - 6 - 2 = 0.$$



Where one action can lead: three demands, three next states; $r(1, 1) = \frac{1}{4}(-8) + \frac{1}{2}(0) + \frac{1}{4}(8) = 0$.

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

$P(s' s, a)$	$s' = 0$	$s' = 1$	$s' = 2$	$s' = 3$
$s + a = 0$	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 $s + a = 2$.
- R_{t+1} depends on the demand, hence on S_{t+1} ; $r(s, a)$ averages it out, as in the Gridworld.
- Three-week season: $H = 3$ decisions, no discount, and the best order may depend on the week. Open shop: continuing, $\gamma < 1$. Same tables, different objective.
- Is the stock a sufficient state? Independent weekly demand: yes. Demand that depends on the month, or a two-week delivery delay: no. Next section.
- Same problem at scale: Amazon's ordering decision, with stochastic lead times, lost sales, and correlated demand, is trained by deep RL and A/B tested against classical policies (Madeka et al., 2022; Andaz et al., 2023). The extra ingredients are the next section's repairs: orders in transit and recent demand go into the state.

The Markov Property

Markov property. For every t , every history $h_t = (s_0, a_0, \dots, s_{t-1}, a_{t-1})$, and every s, a : the joint distribution of (S_{t+1}, R_{t+1}) given $H_t = h_t, S_t = s, A_t = a$ does not depend on h_t . In particular, for every s' ,

$$\begin{aligned}\Pr(S_{t+1} = s' | H_t = h_t, S_t = s, A_t = a) &= P(s' | s, a), \\ \mathbb{E}[R_{t+1} | H_t = h_t, S_t = s, A_t = a] &= r(s, a),\end{aligned}$$

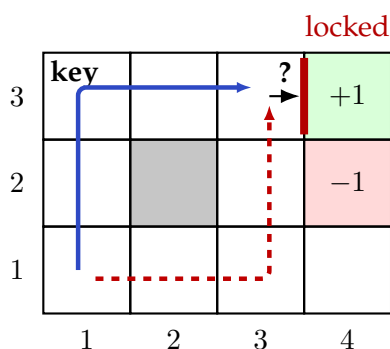
and these two tables are all the expected return needs.

- The state is a sufficient statistic of the history for what happens next. Given the present, the past is irrelevant.
- A property of the *representation* you chose, not of the world: the same system is Markov in one set of variables and not in another.
- Markov is not deterministic: slips and demand are random, and the state is Markov all the same.

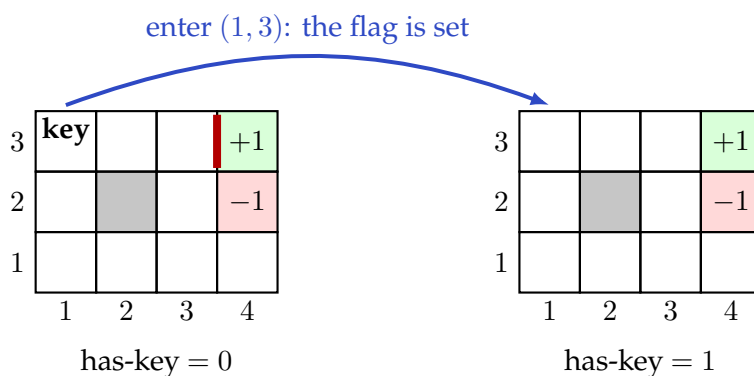
The two-histories test. Not Markov: two histories that end in the same (s, a) but give different distributions of (R_{t+1}, S_{t+1}) . Repair: add to the state whatever tells the two histories apart.

- Demand that depends on the month: the same stock in December and in February, different next-stock distributions; no single table P fits. Add the month.
- Two-week delivery: the same stock with and without an order in transit, different next week. Add the outstanding order.

Failure mode: a state that is not Markov. The +1 exit is locked. The key lies in (1, 3) and is picked up on entry. Without it, a move into (4, 3) bumps like a wall.



- History 1 (solid): up column 1 to (1, 3), along row 3 to (3, 3); now E. Key in hand: (4, 3) w.p. 0.8, reward +0.96.
- History 2 (dashed): along row 1 to (3, 1), up to (3, 3); now E. No key: bump, stay w.p. 0.9, reward -0.04.
- Same cell, same action, different futures: *position is not a Markov state of this world.*
- Repair: state = (cell, has-key), $11 \times 2 = 22$ states; entering (1, 3) sets the flag.



The repaired state (cell, has-key): two copies of the grid, 22 states. The lock exists only in the left copy.

Example. How many deterministic policies does the repaired world have?

Nine decision cells in each copy, 18 in all: $4^{18} \approx 6.9 \times 10^{10}$, against $4^9 = 262,144$ before the key.

If the next state or reward depends on something earlier, put that something into the state. Position without velocity, stock without the order in transit, a cell without the key: one more coordinate each. The price is a larger table.

Property	Says	Holds in	Fails in, and the repair
Markov	the next state and reward depend on the past only through (S_t, A_t)	Gridworld; inventory with independent weekly demand	the key world described by position alone; a rolling ball described by position without velocity. Add the flag, add the velocity.
Deterministic	$P(\cdot s, a)$ puts all its mass on one state	Lecture 1's map without the B-road	Gridworld, inventory. Not a defect: the definition allows randomness, and a deterministic policy is still fine.

Observation versus state. An agent receives observations (an image, a sensor reading), not states. The full history is always Markov but grows without bound; practical states compress it (the last few frames, a running estimate of what is hidden). No compact Markov state: partial observability, outside this course.

What the Markov property buys: the probability of a trajectory.

For $\tau = (s_0, a_0, s_1, a_1, \dots, s_T)$, by the chain rule and then the two one-step rules,

$$\begin{aligned}
 P^\pi(\tau) &= \Pr(s_0) \Pr(a_0 | s_0) \Pr(s_1 | s_0, a_0) \\
 &\quad \times \Pr(a_1 | s_0, a_0, s_1) \Pr(s_2 | s_0, a_0, s_1, a_1) \cdots \\
 &= d_0(s_0) \prod_{t=0}^{T-1} \pi(a_t | s_t) P(s_{t+1} | s_t, a_t).
 \end{aligned}$$

- Two facts: π looks at s_t only (by construction); P looks at (s_t, a_t) only (Markov). Without the second the chain rule still holds, but each factor carries the whole past, and no table P supplies it.
- The table's trajectory under π_0 : $d_0 = 1$, every π factor 1, every P factor 0.8: $0.8^5 \approx 0.33$. Uniformly random: $(\frac{1}{4} \cdot 0.8)^5 = 0.2^5 = 0.00032$.
- $J(\pi) = \sum_{\tau} P^\pi(\tau) \sum_{t < T} \gamma^t r(s_t, a_t)$: a weighted average whose weights we know (planning) or can sample (learning). Rewards enter through their means: one run's R_{t+1} is random, but $\mathbb{E}[R_{t+1} | S_t, A_t] = r(S_t, A_t)$, and linearity of expectation does the rest. Every algorithm in the course manipulates this product.

A fixed policy turns the MDP into a Markov chain with rewards.

$$P^\pi(s' | s) = \sum_a \pi(a | s) P(s' | s, a), \quad r^\pi(s) = \sum_a \pi(a | s) r(s, a).$$

- Uniformly random from $(1, 1)$: average the four rows of P : $(1, 1) \mapsto 0.5$, $(1, 2) \mapsto 0.25$, $(2, 1) \mapsto 0.25$; $r^\pi((1, 1)) = -0.04$.
- Inventory, “order up to 2”: $\pi(s) = \max(2 - s, 0)$. Rows $s + \pi(s) = 2$ for $s \leq 2$ and $s + \pi(s) = 3$ for $s = 3$:

$$P^\pi = \begin{pmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{pmatrix}, \quad r^\pi = (r(0, 2), r(1, 1), r(2, 0), r(3, 0)) = (-2, 0, 6, 5).$$

- P^π : an $|\mathcal{S}| \times |\mathcal{S}|$ matrix, rows sum to one; r^π : a vector. The reward this chain collects from each state is the value of π : Lecture 3.

Horizon and Discounting

One objective, the expected sum of rewards; three ways for the sum to end.

	Episodic (random T)	Discounted (continuing)	Fixed horizon H
Ends when	a terminal state is entered, at a random time T	never	after exactly H decisions
Return	$G_0 = \sum_{t=0}^{T-1} \gamma^t R_{t+1}$, $\gamma \in [0, 1]$	$G_t = \sum_{k \geq 0} \gamma^k R_{t+k+1}$, $\gamma < 1$	$G_0 = \sum_{t=0}^{H-1} R_{t+1}$
Policy	stationary, $\pi(a s)$	stationary, $\pi(a s)$	may depend on time, $\pi_t(a s)$
Finite for sure when	$\gamma < 1$, or $T < \infty$ w.p. one for every policy compared	always: $ R_{t+1} \leq R_{\max}$, so $ G_t \leq R_{\max}/(1 - \gamma)$	always
Example	Gridworld with $\gamma = 1$; a board game	Gridworld with $\gamma = 0.9$; a shop that stays open	the three-week season; a fixed number of moves

- *Episodic tasks fit the discounted formulas*: a terminal state is absorbing with reward 0, so the finite sum equals the infinite one (Sutton and Barto, 2018, Section 3.4).
- $\gamma = 1$ is safe only if every policy you will compare terminates w.p. one. Gridworld, $\gamma = 1$, “always W” from $(1, 1)$: never exits, $R_{t+1} = -0.04$ forever, $G_0 = -\infty$.
- Same policy at $\gamma = 0.9$: $G_0 = -0.04/(1 - 0.9) = -0.4$; no policy leaves $[-10.4, 10.4]$

since $R_{\max} = 1.04$. Hence the course's $\gamma = 0.9$: all 262,144 policies have finite values.

Discounting is a random horizon. Suppose the episode is cut off after each step with probability $1 - \gamma$, independently of everything else. Then $\Pr(T > t) = \gamma^t$, and for any reward sequence

$$\mathbb{E}\left[\sum_{t < T} R_{t+1}\right] = \sum_{t \geq 0} \Pr(T > t) R_{t+1} = \sum_{t \geq 0} \gamma^t R_{t+1}, \quad \mathbb{E}[T] = \sum_{t \geq 0} \gamma^t = \frac{1}{1 - \gamma}.$$

γ^t	$t = 1$	$t = 5$	$t = 10$	$t = 20$	$t = 50$	$t = 100$	$1/(1 - \gamma)$
$\gamma = 0.5$	0.5	0.03	0.001	≈ 0	≈ 0	≈ 0	2
$\gamma = 0.9$	0.9	0.59	0.35	0.12	0.005	≈ 0	10
$\gamma = 0.99$	0.99	0.95	0.90	0.82	0.61	0.37	100

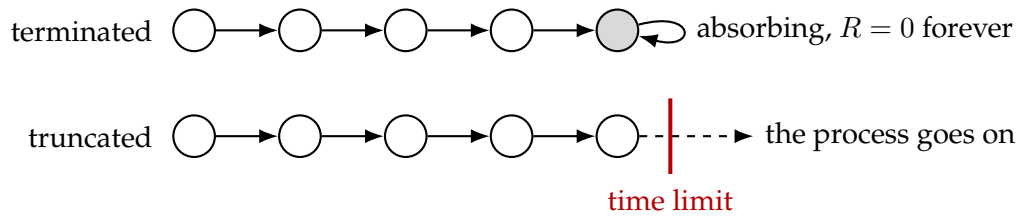
How much a reward t steps ahead counts, and the effective horizon.

- Discounted return = expected undiscounted return of an agent switched off w.p. $1 - \gamma$ after each step. Effective horizon $1/(1 - \gamma)$: ten steps at 0.9, a hundred at 0.99.
- Why discount: every policy gets a finite value; one stationary rule per state suffices (shown with optimal policies); smaller γ , faster planning and learning (later). Interest rates are rarely the reason.
- *Fixed horizon.* With a fixed number of steps left, the best action may depend on how many remain: in the shop, an order in the last week of a season is never worth it, a week earlier it can be. So the rule carries the time, $\pi_t(s)$. Put the time into the state, (s, t) , and one table over states is enough again: the three-week season has 4×3 open states and one closed shop, 13 in all, an episodic task ending at $t = H$.

Example. Two actions in the start state. *Stop*: reward 0, the episode ends. *Go*: reward -1 , then a state whose only move pays $+3$ and ends. Which is optimal?

Go is worth $-1 + 3\gamma$, *Stop* is worth 0: go iff $\gamma > \frac{1}{3}$. At $\gamma = 0.9$ the agent pays the cost for the later reward; at $\gamma = 0.2$ it stops at once. Same MDP, two optimal policies: γ is part of the problem. A cost on every step and a payoff far away: the smaller γ , the sooner the optimal policy stops.

Termination is not truncation. Termination: part of the model; the process enters an absorbing state and nothing follows. Truncation: a cutoff of data collection (time limit, budget); the last recorded state still has a future. Simulators keep two flags ([Farama Foundation, 2026](#)). A language model writing a reply: the end-of-sequence token terminates, the token limit truncates. Treat truncated as terminated and the agent learns that the world ends at the time limit.



Termination enters an absorbing state. Truncation stops the record; the last state still has a future.

- **Reward shift.** Add c to every reward. Continuing and discounted: every policy gains $c/(1 - \gamma)$; fixed horizon: cH ; the ranking is unchanged. Episodic: the gain is $c \mathbb{E}_\pi [\sum_{t < T} \gamma^t]$, and T depends on the policy, so the ranking can change even with $\gamma < 1$. Stop/Go with $c = -2$: Go is worth $-3 + 0.9 \cdot 1 = -2.1$, Stop -2 ; Stop wins. $c < 0$ says hurry; $c > 0$ says stay: Gridworld at $\gamma = 1$ with $+1$ per step never exits (Russell and Norvig, 2020, Section 17.1). A later modeling studio returns to this.

Summary

Lecture 2 on one board

- $M = (\mathcal{S}, \mathcal{A}, P, r, \gamma, d_0)$ with $r(s, a) = \mathbb{E}[R_{t+1} \mid S_t = s, A_t = a]$. Every row of P sums to one. γ and d_0 belong to the objective. Rewards and max from now on.
- Gridworld: eleven cells, four actions, $0.8/0.1/0.1, -0.04$ per move plus ± 1 on entering an exit, then absorbing, $\gamma = 0.9$. One trajectory, three returns $(0.80, 0.49, -0.015)$. Inventory: $S_{t+1} = \max(S_t + A_t - D_{t+1}, 0)$, $r(1, 1) = 0$; a season or an open shop.
- Markov: given the history, (S_{t+1}, R_{t+1}) depends only on (S_t, A_t) ; the tables P and r follow. Test with two histories; repair by adding the missing information to the state. A property of the representation, not of the world.
- $P^\pi(\tau) = d_0(s_0) \prod_t \pi(a_t \mid s_t) P(s_{t+1} \mid s_t, a_t)$ and $J(\pi) = \sum_\tau P^\pi(\tau) \sum_t \gamma^t r(s_t, a_t)$. Fixing π leaves a Markov chain with rewards, (P^π, r^π) .
- Three conventions: random T , discount $\gamma < 1$, fixed H . Terminal = absorbing with reward 0. $\gamma = 1$ only if every policy terminates. γ is a continuation probability with effective horizon $1/(1 - \gamma)$. A fixed horizon allows π_t . Termination is not truncation.
- Next: how good is a fixed policy? Lecture 3 evaluates the chain (P^π, r^π) .

References

Sohrab Andaz, Carson Eisenach, Dhruv Madeka, Kari Torkkola, Randy Jia, Dean Foster, and Sham Kakade. Learning an inventory control policy with general inventory arrival

- dynamics. arXiv preprint arXiv:2310.17168, 2023. URL <https://arxiv.org/abs/2310.17168>.
- Emma Brunskill. CS 234: Reinforcement learning. Stanford University course materials, 2026. URL <https://web.stanford.edu/class/cs234/>. Winter 2026 offering.
- Farama Foundation. Handling time limits. Gymnasium documentation, 2026. URL https://gymnasium.farama.org/main/tutorials/handling_time_limits/. Accessed 2026-07-30.
- Nan Jiang. CS 443: Reinforcement learning. University of Illinois Urbana-Champaign course materials, 2026. URL <https://nanjiang.cs.illinois.edu/cs443/>. Spring 2026 offering.
- Dhruv Madeka, Kari Torkkola, Carson Eisenach, Anna Luo, Dean P. Foster, and Sham M. Kakade. Deep inventory management. arXiv preprint arXiv:2210.03137, 2022. URL <https://arxiv.org/abs/2210.03137>.
- Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.
- Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson, 4 edition, 2020.
- Wen Sun. CS 4/5789: Introduction to reinforcement learning. Cornell University course materials, 2025. URL https://wensun.github.io/CS4789_spring_2025.html. Spring 2025 offering.
- Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2 edition, 2018. URL <http://incompleteideas.net/book/the-book-2nd.html>.