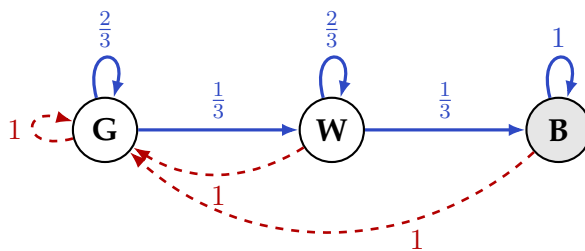


# Lecture 3: Value Functions and Policy Evaluation

**How to use these notes.** Board content in board order: figures, equations, tables, one-line statements. Today's model is a three-state machine, chosen so that every number in these notes can be checked by hand; the definitions and equations are general. The return  $G_t$ , the reward bound  $R_{\max}$ , and the chain  $(P^\pi, r^\pi)$  a fixed policy leaves appear in the Lecture 2 note but were not covered in class; all three are introduced here from scratch. Reading: [Sutton and Barto \(2018, Sections 3.5 and 4.1\)](#); Example 4.1 there is a hand exercise of exactly today's kind, and Section 3.6 prepares Lecture 4. Optional: [Puterman \(1994, Section 6.1\)](#) and [Agarwal et al. \(2022, Section 1.1\)](#) for the same material with proofs. 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 Model for Today: A Machine

A workshop runs one machine. Each week it is good, worn, or broken, and the manager either runs it or has it repaired. A good machine earns 8 for the week, a worn one 4, a broken one nothing. Running wears it down: a good machine stays good with probability  $\frac{2}{3}$  and is worn by next week otherwise; a worn machine stays worn with probability  $\frac{2}{3}$  and breaks otherwise; a broken machine stays broken. A repair costs 10, takes the week, and leaves the machine good, whatever it was.



| $r(s, a)$ | run | repair |
|-----------|-----|--------|
| G         | 8   | -10    |
| W         | 4   | -10    |
| B         | 0   | -10    |

Where each action leads. Solid: run. Dashed: repair, always back to G.

This week's profit, fixed once  $(s, a)$  is.

- $\mathcal{S} = \{G, W, B\}$ ;  $\mathcal{A} = \{\text{run}, \text{repair}\}$  in every state (repairing a good machine is allowed, and pointless).
- $P$ : six rows, at most two nonzero entries each. Run:  $G \mapsto (\frac{2}{3}, \frac{1}{3}, 0)$ ,  $W \mapsto (0, \frac{2}{3}, \frac{1}{3})$ ,  $B \mapsto (0, 0, 1)$ . Repair:  $(1, 0, 0)$  from every state.

- $r$ : the table. Nothing random in it:  $R_{t+1} = r(S_t, A_t)$  exactly, and the only chance is in where the machine goes next.
- $\gamma = 0.9$ ; the machine starts good,  $d_0(G) = 1$ ; the workshop never closes.

Now fix how the manager acts. The Lecture 2 note wrote the next box down, but class did not reach it, so it starts from zero here.

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

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

One row per state in place of one per  $(s, a)$ ; each row still sums to one. Deterministic  $\pi$ : the row of  $(s, \pi(s))$ ,  $P^\pi(\cdot | s) = P(\cdot | s, \pi(s))$  and  $r^\pi(s) = r(s, \pi(s))$ .

- Where the sums come from: from  $s$  the step draws  $a \sim \pi(\cdot | s)$  and then  $s' \sim P(\cdot | s, a)$ , so the chance of landing in  $s'$  adds the two-stage chances over  $a$  (total probability), and the expected reward averages  $r(s, a)$  the same way. Lecture 2's two facts make this legitimate:  $P$  looks at  $(s, a)$  only,  $\pi$  at  $s$  only. With no action left to choose, what remains is a Markov chain: one fixed distribution over next states per state, and a reward attached to each state.
- “Never repair”,  $\pi(s) = \text{run}$  for every  $s$ : the three run rows and the run column,

$$P^\pi = \begin{pmatrix} \frac{2}{3} & \frac{1}{3} & 0 \\ 0 & \frac{2}{3} & \frac{1}{3} \\ 0 & 0 & 1 \end{pmatrix}, \quad r^\pi = (8, 4, 0).$$

Today's policy. B leads only to B and pays 0 there: the chain ends in B.

- A coin flip at W, run or repair with probability  $\frac{1}{2}$  each, averages two rows:  $P^\pi(\cdot | W) = \frac{1}{2}(0, \frac{2}{3}, \frac{1}{3}) + \frac{1}{2}(1, 0, 0) = (\frac{1}{2}, \frac{1}{3}, \frac{1}{6})$  and  $r^\pi(W) = \frac{1}{2} \cdot 4 + \frac{1}{2} \cdot (-10) = -3$ .

## How Good Is a Fixed Policy?

- Lecture 1: two routes out of  $S$  on the map, priced  $-7$  and  $-6.2$ ; we kept the larger. Each price was the value of one fixed rule of behavior; the max over rules was something else.
- Above: fix  $\pi$  and the MDP becomes a Markov chain with rewards,  $(P^\pi, r^\pi)$ . Today: the number each state of that chain deserves, and two ways to compute it.

**Given a policy, what number does each state deserve, and how do we compute it from  $(P^\pi, r^\pi)$ ?**

**Today's targets.**

1. Build the chain  $(P^\pi, r^\pi)$  a fixed policy leaves from the model's two tables; for a deterministic  $\pi$ , take the row of  $(s, \pi(s))$ .
2. Define the return  $G_t$ , then  $V^\pi$  and  $Q^\pi$ ; say what is averaged in each and who chooses which action.
3. Derive the Bellman equation for  $V^\pi$ ; write it as one row per state; state the version for  $Q^\pi$ .
4. Evaluate a policy exactly by a linear solve: three equations for “never repair”, solved by hand from the bottom up; a policy whose chain has a cycle, solved by computer.
5. Evaluate it by iteration: two iterations by hand; say what the  $k$ -th iterate means and how fast it converges.
6. Recognize when the equation has no solution:  $\gamma = 1$  with a policy that never stops.

**Two Numbers per State**

One run pays a sequence of rewards; first, a name for their discounted sum. The Lecture 2 note lists it in its notation table; class did not reach it, so it is defined here.

**The return of one run, from time  $t$** 

$$G_t := R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k \geq 0} \gamma^k R_{t+k+1} = R_{t+1} + \gamma G_{t+1}.$$

A random variable: one trajectory, one number. Under “never repair” the run G, G, W, B, B, ... pays 8, 8, 4, 0, 0, ... and scores  $G_0 = 8 + 0.9 \cdot 8 + 0.81 \cdot 4 = 18.44$ . The objective of Lecture 2, the expected discounted total reward, is its mean:  $J(\pi) = \mathbb{E}_\pi[G_0]$ . Infinitely many terms, always a number: with  $R_{\max}$  the largest one-step reward in absolute value,  $|G_t| \leq R_{\max}(1 + \gamma + \gamma^2 + \cdots) = R_{\max}/(1 - \gamma)$ ; here  $R_{\max} = 10$ , the repair bill, so no run scores outside  $[-100, 100]$ . A broken machine pays 0 forever, so the one sum also covers tasks that end. The last equality factors  $\gamma$  out of the tail; everything today rests on it.

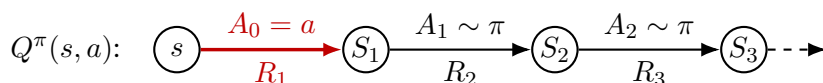
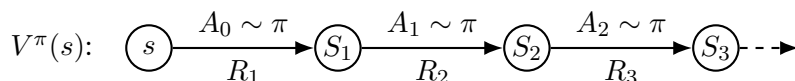
A value is an expected return, so it belongs to a policy; there are two, depending on who chooses the first action.

**State value and action value of a policy  $\pi$** 

$$V^\pi(s) := \mathbb{E}_\pi[G_t \mid S_t = s] = \mathbb{E}_\pi\left[\sum_{k \geq 0} \gamma^k R_{t+k+1} \mid S_t = s\right],$$

$$Q^\pi(s, a) := \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a].$$

$\mathbb{E}_\pi$ : every action after the conditioning is drawn from  $\pi$ . A state that pays 0 and leads only to itself has  $V^\pi = 0$ . Objective:  $J(\pi) = \sum_s d_0(s) V^\pi(s)$ .



Who chooses what.  $V^\pi$ :  $\pi$  picks every action.  $Q^\pi$ : the first action is ours,  $\pi$  picks the rest.

- Both are averages over whole trajectories, each weighted by its probability under  $\pi$ . One trajectory's return is a sample of the value, not the value.
- $\pi$  is stationary, so  $V^\pi$  does not depend on  $t$ : one number per state, a vector  $V^\pi \in \mathbb{R}^{|\mathcal{S}|}$ ;  $Q^\pi$  is a table over  $\mathcal{S} \times \mathcal{A}$ .
- $Q^\pi$  answers a “what if”: deviate once, then hand control back to  $\pi$ . Condition on the first action:

$$V^\pi(s) = \sum_a \pi(a \mid s) Q^\pi(s, a); \quad \text{deterministic } \pi : \quad V^\pi(s) = Q^\pi(s, \pi(s)).$$

**The machine under “never repair”.** Three numbers, one per state: the average of  $G_0$  over every run from there under the policy. Where they come from is the rest of today; first, how to read them.

| $s$ | $V^\pi(s)$ | $Q^\pi(s, \text{run})$ | $Q^\pi(s, \text{repair})$ |
|-----|------------|------------------------|---------------------------|
| G   | 27.5       | <b>27.5</b>            | 14.75                     |
| W   | 10         | <b>10</b>              | 14.75                     |
| B   | 0          | <b>0</b>               | 14.75                     |

“Never repair” at  $\gamma = 0.9$ .  $Q$ : the first action is ours, then the policy resumes. Bold: the policy's own action, equal to  $V^\pi$ .

- B: 0. Broken and never repaired, the machine earns nothing again.
- W: 10. Four a week while it stays worn, three weeks on average: 12 undiscounted, 10 with the discount. Checked by hand after the Bellman equation.

- G:  $27.5 = J(\pi)$ , since the machine starts good. Above the 18.44 of the run above; the box below says why.
- The run column is  $V^\pi$ : the policy's own action.
- The repair column is 14.75 in every row: after a repair the machine is good whatever it was, so  $-10 + 0.9 \cdot 27.5$  from G, W, and B alike. At W and at B one repair beats the policy's own action; at G it does not. What to do with such a comparison is Lecture 4.

**Value is not return.**  $G_0$  is one trajectory's number;  $V^\pi(s)$  averages  $G_0$  over every trajectory from  $s$  under  $\pi$ , each weighted by its probability. From G under "never repair", the run G, G, W, B has probability  $\frac{2}{3} \cdot \frac{1}{3} \cdot \frac{1}{3} = \frac{2}{27}$  and return 18.44; the run G, W, B has probability  $\frac{1}{9}$  and return 11.6; a machine that stays good five weeks and worn two scores 37.25. Average over all runs: 27.5.

## The Bellman Equation for $V^\pi$

The average over whole trajectories collapses to an average over one step, because after one step the future is again a value.

**Derivation.** For every  $s$ , with  $G_t = R_{t+1} + \gamma G_{t+1}$ ,

$$\begin{aligned} V^\pi(s) &= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} \mid S_t = s] \\ &= r^\pi(s) + \gamma \mathbb{E}_\pi[G_{t+1} \mid S_t = s] \end{aligned} \quad (1)$$

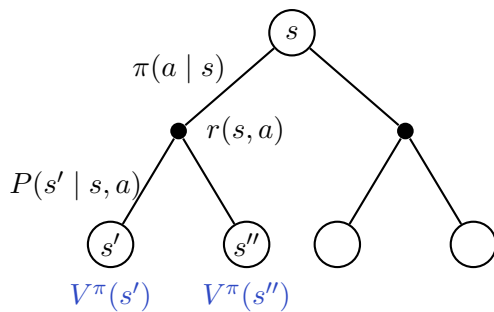
$$= r^\pi(s) + \gamma \sum_{s'} P^\pi(s' \mid s) \mathbb{E}_\pi[G_{t+1} \mid S_t = s, S_{t+1} = s'] \quad (2)$$

$$= r^\pi(s) + \gamma \sum_{s'} P^\pi(s' \mid s) \mathbb{E}_\pi[G_{t+1} \mid S_{t+1} = s'] \quad (3)$$

$$= r^\pi(s) + \gamma \sum_{s'} P^\pi(s' \mid s) V^\pi(s'). \quad (4)$$

Spelled out over actions, for all  $s \in \mathcal{S}$ :

$$V^\pi(s) = \sum_a \pi(a \mid s) \left[ r(s, a) + \gamma \sum_{s'} P(s' \mid s, a) V^\pi(s') \right]$$



Backup diagram: average over  $a$ , then over  $s'$ ; the leaves are values, not more trees.

- (1)  $G_t = R_{t+1} + \gamma G_{t+1}$ , then linearity; the first reward has mean  $r^\pi(s) = \sum_a \pi(a | s) r(s, a)$ .
- (2) tower property: condition on where the step lands, w.p.  $P^\pi(s' | s)$ .
- (3) the two facts of Lecture 2:  $P$  looks at  $(S_t, A_t)$  only,  $\pi$  at  $S_t$  only. Given  $S_{t+1} = s'$ , nothing earlier matters.
- (4) stationarity:  $\pi, P, r$  are the same at every step, so the same expectation started at  $s'$  is  $V^\pi(s')$  by definition.

**Example.** Check the boxed equation at G with the numbers in the table.

“Never repair” says run:  $r(G, \text{run}) = 8$ ; the step stays in G w.p.  $\frac{2}{3}$  and lands in W w.p.  $\frac{1}{3}$ . So  $V^\pi(G) = 8 + 0.9 \left( \frac{2}{3} \cdot 27.5 + \frac{1}{3} \cdot 10 \right) = 8 + 0.9 \cdot \frac{65}{3} = 8 + 19.5 = 27.5$ . At W:  $4 + 0.9 \cdot \frac{2}{3} \cdot 10 = 10$ . At B:  $0 + 0.9 \cdot 0 = 0$ .

### The same equation for $Q^\pi$

$$\begin{aligned} Q^\pi(s, a) &= r(s, a) + \gamma \sum_{s'} P(s' | s, a) V^\pi(s') \\ &= r(s, a) + \gamma \sum_{s'} P(s' | s, a) \sum_{a'} \pi(a' | s') Q^\pi(s', a'). \end{aligned}$$

With  $V^\pi(s) = \sum_a \pi(a | s) Q^\pi(s, a)$ , each of  $V^\pi$  and  $Q^\pi$  determines the other.

- One equation per state, unknowns on both sides: a system to solve, not a formula to evaluate.
- Lecture 4 changes one thing: at  $s'$ , the average over  $a'$  becomes a max.
- Only  $r(s, a)$  enters. In a model whose reward is random, or depends on  $S_{t+1}$ , step (1) replaces it by its mean given  $(s, a)$ ; that mean is what  $r(s, a)$  denotes, and nothing else changes (Jiang, 2024, Section 1.5). Today’s  $r$  is that mean already.

**Example.**  $Q^\pi(W, \text{repair})$  from the table, by the first line of the box.

$r(W, \text{repair}) = -10$ ; the step lands in G w.p. 1:  $Q^\pi(W, \text{repair}) = -10 + 0.9 \cdot 27.5 = 14.75$ , the table’s entry. The same sum from G and from B: the repair column is one number.

**Reward timing.** Sources differ: a reward on the state  $s$ , on  $(s, a)$ , or on  $(s, a, s')$  (Sutton and Barto, 2018). Values change if a reward moves from “on entering  $s''$ ” to “for being in  $s''$ ”. Ours:  $R_{t+1}$  follows  $A_t$ , and  $r(s, a)$  is its mean; for the machine the two coincide.

## Solving It I: A Linear System

$|\mathcal{S}|$  linear equations in  $|\mathcal{S}|$  unknowns; for “never repair” at  $\gamma = 0.9$ , three.

**One equation per state.** Rows  $(\frac{2}{3}, \frac{1}{3}, 0)$ ,  $(0, \frac{2}{3}, \frac{1}{3})$ ,  $(0, 0, 1)$ ;  $r^\pi = (8, 4, 0)$ :

$$V(G) = 8 + 0.9\left(\frac{2}{3}V(G) + \frac{1}{3}V(W)\right),$$

$$V(W) = 4 + 0.9\left(\frac{2}{3}V(W) + \frac{1}{3}V(B)\right),$$

$$V(B) = 0 + 0.9V(B).$$

Unknowns on both sides: a system to solve, not a formula to evaluate.

**Matrix form.**  $V^\pi, r^\pi \in \mathbb{R}^{|\mathcal{S}|}$ ;  $P^\pi \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$  with  $P^\pi[s, s'] = P^\pi(s' | s)$ , rows indexed by the current state, each row summing to one.

$$V^\pi = r^\pi + \gamma P^\pi V^\pi \iff (I - \gamma P^\pi) V^\pi = r^\pi \iff V^\pi = (I - \gamma P^\pi)^{-1} r^\pi \quad (\gamma < 1).$$

$$\text{row } s \quad V^\pi = r^\pi + \gamma \begin{matrix} P^\pi(\cdot | s) \\ P^\pi \end{matrix} V^\pi = \sum_{s'} P^\pi(s' | s) V^\pi(s')$$

Row  $s$  of  $P^\pi$  times  $V^\pi$ : the average next value. One row, one equation.

**Solved by hand, from the bottom up.** This chain never flows back: B leads only to B, W only to W and B. So each equation can be closed in turn.

$$V(B) = 0.9V(B) \implies V(B) = 0,$$

$$V(W) = 4 + 0.6V(W) \implies V(W) = 4/0.4 = 10,$$

$$V(G) = 8 + 0.6V(G) + 0.3 \cdot 10 \implies V(G) = 11/0.4 = 27.5.$$

$V^\pi = (27.5, 10, 0)$ , the table. Three lines because nothing flows back. A policy that repairs sends B back to G and closes a cycle; then no equation can be finished before the others, and that is what the solver is for.

**Example.** A worn machine earns 4 this week. Why is it worth 10?

$10 = 4 + 0.6 \cdot 10$ : this week's 4, plus next week's situation, which is the same one with probability  $\frac{2}{3}$  and worth 0 otherwise, discounted by 0.9. Unrolled,  $4(1 + 0.6 + 0.36 + \dots) = 4/0.4$ . Without the discount the machine stays worn three weeks on average and earns 12; the discount trims it to 10.

- $Q$  from  $V$ :  $Q^\pi(\text{B, repair}) = -10 + 0.9 \cdot 27.5 = 14.75$  (repair once, then never again) against  $V^\pi(\text{B}) = 0$ . Repairing every time the machine breaks is a different policy, evaluated next.

**Two more policies, by computer.** “Repair when broken” sends B back to G: the chain  $G \rightarrow W \rightarrow B \rightarrow G$  is a cycle, and its three equations must be solved together. One line of NumPy, with  $P_{\pi}$  and  $r_{\pi}$  the chain's matrix and vector:

```
V = np.linalg.solve(np.eye(3) - gamma * P_pi, r_pi)
```

| $\gamma = 0.9$                     | $V^\pi(\text{G}) = J(\pi)$ | $V^\pi(\text{W})$ | $V^\pi(\text{B})$ |
|------------------------------------|----------------------------|-------------------|-------------------|
| never repair                       | 27.5                       | 10                | 0                 |
| repair when broken                 | 44.30                      | 32.41             | 29.87             |
| repair when worn (and when broken) | 38.46                      | 24.62             | 24.62             |

*Three of the eight deterministic policies, one linear solve each.*

- Check one entry: under “repair when broken”,  $V(\text{B}) = -10 + 0.9 \cdot 44.30 = 29.87$ .
- Cost: solving an  $n \times n$  system is  $O(n^3)$ ; fine for 3 states, not for  $10^6$ .
- Repairing at all beats never repairing; repairing a worn machine does not pay here. Lecture 4 finds the best of the eight without evaluating them one by one.

**Why  $I - \gamma P^\pi$  is invertible for  $\gamma < 1$  (Jiang, 2024).** Each entry of  $P^\pi x$  is an average of entries of  $x$ , so  $\|P^\pi x\|_\infty \leq \|x\|_\infty$ . Hence for  $x \neq 0$ ,  $\|(I - \gamma P^\pi)x\|_\infty \geq \|x\|_\infty - \gamma\|x\|_\infty = (1 - \gamma)\|x\|_\infty > 0$ : no nonzero  $x$  is sent to 0. The Bellman equation has exactly one solution, so any vector that satisfies it is  $V^\pi$ .

**Shorthand for later.** Write the right-hand side as an operator on vectors,  $(\mathcal{T}^\pi V)(s) := r^\pi(s) + \gamma \sum_{s'} P^\pi(s' | s) V(s')$ : one iteration. The Bellman equation says  $V^\pi = \mathcal{T}^\pi V^\pi$ , a fixed point; the iteration below is  $V_{k+1} = \mathcal{T}^\pi V_k$ . Value iteration uses the same symbol with a max inside.

## Solving It II: Iteration

Use the right-hand side as an update: guess  $V$ , evaluate the right-hand side in every state, repeat.

**Iterative policy evaluation** (Sutton and Barto, 2018, Section 4.1)

- Input:  $\pi$  and the model  $(P, r, \gamma)$ ; a tolerance  $\theta > 0$ .
- $V_0(s) \leftarrow 0$  for all  $s$  (any start works; a state that pays 0 and leads only to itself stays at 0).
- For  $k = 0, 1, 2, \dots$ : for every  $s \in \mathcal{S}$ ,

$$V_{k+1}(s) \leftarrow \sum_a \pi(a | s) \left[ r(s, a) + \gamma \sum_{s'} P(s' | s, a) V_k(s') \right], \quad \text{i.e.} \quad V_{k+1} = r^\pi + \gamma P^\pi V_k.$$

- Stop when  $\max_s |V_{k+1}(s) - V_k(s)| < \theta$ ; return  $V_{k+1}$ .

One iteration is one pass over the table of  $P$ :  $O(|\mathcal{S}|^2)$  for a dense  $P^\pi$ , with two arrays of length  $|\mathcal{S}|$  ( $V_k$  and  $V_{k+1}$ ).

| $k$      | $V_k(\text{G})$ | $V_k(\text{W})$ | $V_k(\text{B})$ | $V^\pi(\text{W}) - V_k(\text{W})$ |
|----------|-----------------|-----------------|-----------------|-----------------------------------|
| 1        | 8               | 4               | 0               | 6                                 |
| 2        | 14              | 6.4             | 0               | 3.6                               |
| 3        | 18.32           | 7.84            | 0               | 2.16                              |
| 4        | 21.34           | 8.70            | 0               | 1.30                              |
| 5        | 23.42           | 9.22            | 0               | 0.78                              |
| 10       | 27.03           | 9.94            | 0               | 0.06                              |
| 20       | 27.50           | 10.00           | 0               | 0.00                              |
| $\infty$ | 27.5            | 10              | 0               | 0                                 |

Iterations from  $V_0 = 0$  under “never repair”,  $\gamma = 0.9$ , each  $V_{k+1}$  from the whole of  $V_k$ . At W the error is exactly  $6 \cdot 0.6^{k-1}$ .

- By hand:  $V_1 = r^\pi = (8, 4, 0)$ . Then  $V_2(\text{G}) = 8 + 0.9(\frac{2}{3} \cdot 8 + \frac{1}{3} \cdot 4) = 8 + 6 = 14$ ,  $V_2(\text{W}) = 4 + 0.9 \cdot \frac{2}{3} \cdot 4 = 6.4$ ,  $V_2(\text{B}) = 0$ . Once more:  $V_3(\text{G}) = 8 + 0.9(\frac{2}{3} \cdot 14 + \frac{1}{3} \cdot 6.4) = 18.32$ ,  $V_3(\text{W}) = 4 + 0.9 \cdot \frac{2}{3} \cdot 6.4 = 7.84$ .
- The numbers climb toward  $(27.5, 10, 0)$  here; from  $V_0 = (40, 40, 40)$  one iteration gives  $(44, 40, 36)$ , then the same limit. Only the numbers move; the policy does not.

**What  $V_k$  is.** From  $V_0 = 0$ ,

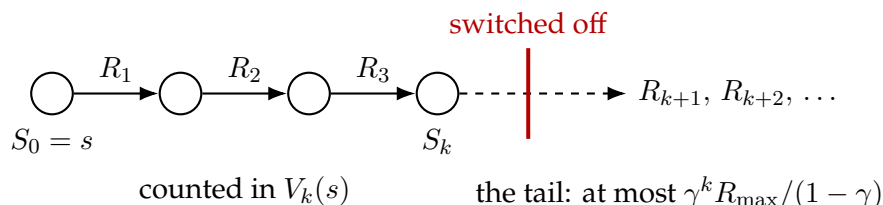
$$V_k(s) = \mathbb{E}_\pi \left[ \sum_{t=0}^{k-1} \gamma^t R_{t+1} \mid S_0 = s \right] :$$

the value of running  $\pi$  for  $k$  steps and then switching off. Induction:  $V_1 = r^\pi$  is one step;  $V_{k+1} = r^\pi + \gamma P^\pi V_k$  is one step and then  $k$  more from  $S_1$ .

- A machine rented for three weeks, no discount, under “never repair”: three iterations at  $\gamma = 1$ ,  $V_3 = (20, 8.44, 0)$ , the expected output of the rental from each starting condition. A fixed horizon is evaluated by iterating, one week per iteration, last week first.
- The error is the discarded tail:

$$V^\pi(s) - V_k(s) = \mathbb{E}_\pi \left[ \sum_{t \geq k} \gamma^t R_{t+1} \mid S_0 = s \right], \quad |V^\pi(s) - V_k(s)| \leq \gamma^k \frac{R_{\max}}{1 - \gamma}.$$

With  $R_{\max} = 10$ : within  $100 \cdot 0.9^k$  after  $k$  iterations, for every policy of the machine. Geometric at rate  $\gamma$ : ten iterations at  $\gamma = 0.9$  cut it by  $0.9^{10} \approx 0.35$ . Here it is faster, 0.6 per iteration at W, because the chain drains into B.



$V_k(s)$  counts the first  $k$  rewards. From a general  $V_0$ , the tail is replaced by  $\gamma^k V_0(S_k)$ .

- General start:  $V_{k+1} - V^\pi = \gamma P^\pi (V_k - V^\pi)$ , and a row of  $P^\pi$  averages, so the largest error shrinks by  $\gamma$  per iteration:  $\|V_k - V^\pi\|_\infty \leq \gamma^k \|V_0 - V^\pi\|_\infty$ . Value iteration repeats this line with a max inside, where it needs one more step.
- “Repair when broken” from zero:  $(8, 4, -10) \rightarrow (14, 3.4, -2.8) \rightarrow (17.42, 5.2, 2.6) \rightarrow \dots \rightarrow (44.30, 32.41, 29.87)$ . A negative iterate, and the dip at W from 4 to 3.4, say nothing about the policy:  $V_k$  counts  $k$  weeks, and the first week from B is a repair bill.

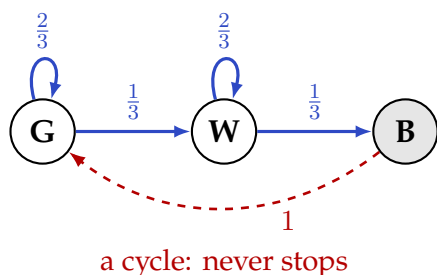
**In place.** Sutton and Barto (2018, Section 4.1) keep one array and overwrite  $V(s)$  as soon as it is computed, so later states in the same iteration already use new values. Same limit, usually fewer iterations: on the machine, updating in the order B, W, G from zero gives  $(9.2, 4, 0)$  instead of  $(8, 4, 0)$ , because G already uses the new 4 at W. The order of the states changes the speed, not the answer.

$V_k$  is **not**  $V^\pi$ . It is the  $k$ -step value of  $\pi$ , and it is not the value of any stationary policy in the discounted problem. Halting at the tolerance  $\theta$  leaves an error; how large, and what  $\theta$  buys, is settled with value iteration. Iterative policy evaluation never touches the policy: the actions stay, the numbers move.

|           | Linear solve                                 | Iteration                                                                                                          |
|-----------|----------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| Answer    | exact $V^\pi$                                | within $\gamma^k R_{\max}/(1-\gamma)$ after $k$ iterations                                                         |
| Cost      | $O( \mathcal{S} ^3)$ , once                  | $O( \mathcal{S} ^2)$ per iteration; accuracy $\varepsilon$ after about $\log(1/\varepsilon)/(1-\gamma)$ iterations |
| Needs     | $P^\pi$ and $r^\pi$ as a matrix and a vector | the same tables, one row at a time                                                                                 |
| Scales to | thousands of states                          | millions; and the update survives when $P$ is replaced by samples (temporal-difference learning, later)            |

## Failure Mode: $\gamma = 1$ and a Policy That Never Stops

Lecture 2:  $\gamma = 1$  is safe only if every policy considered terminates. Here is what goes wrong in the equations.



*“Repair when broken”: whatever the state, the machine is running again within a few weeks, forever.*

- The machine is never scrapped: it earns  $\frac{26}{7} \approx 3.71$  a week on average, forever. Total reward infinite: *no number satisfies the equations.*
- From zero:  $V_k(G) = 8, 14.67, 18.89, 22.44, \dots$ , growing by about 3.7 per iteration without bound.
- Matrix view: the rows of  $P^\pi$  sum to one, so  $(I - P^\pi)\mathbf{1} = 0$ : singular, and the solver fails. All three states recur, so no equation can be dropped.
- At  $\gamma = 0.9$ :  $I - \gamma P^\pi$  is invertible and the values are  $(44.30, 32.41, 29.87)$ .

At  $\gamma = 1$  the three equations of “repair when broken” read

$$\begin{aligned} V(G) &= 8 + \frac{2}{3}V(G) + \frac{1}{3}V(W), \\ V(W) &= 4 + \frac{2}{3}V(W) + \frac{1}{3}V(B), \\ V(B) &= -10 + V(G). \end{aligned}$$

Weight them by  $\frac{3}{7}, \frac{3}{7}, \frac{1}{7}$ , the long-run fractions of weeks the machine spends in G, W, B, and add: every  $V$  has the same total coefficient on both sides and cancels, so  $0 = \frac{26}{7}$  is left. (The weights are the row vector  $\mu$  with  $\mu P^\pi = \mu$ ; that is why the  $V$  terms match.)

- Contrast: “never repair” at  $\gamma = 1$ . B pays 0 and leads only to itself, so  $V(B) = 0$  and two equations remain:  $V(W) = 4 + \frac{2}{3}V(W)$  gives 12, then  $V(G) = 8 + \frac{2}{3}V(G) + \frac{1}{3} \cdot 12$  gives 36. Exactly one solution, and a meaningful one: the expected total output before the machine dies, three weeks at 8 and three at 4.
- Rule restated:  $\gamma = 1$  needs termination with probability one under the policy being evaluated (a *proper* policy). A policy that loops forever with nonzero average reward has no value at  $\gamma = 1$ .

**Example.** Which of the three policies in the table has a value at  $\gamma = 1$ ?

Only “never repair”. “Repair when broken” and “repair when worn” keep the machine running forever and earn  $\frac{26}{7}$  and 3.5 a week on average, so their iterates grow without bound and their systems are singular.

**What to do.** Discount ( $\gamma < 1$ : always exactly one solution); or check that the policy terminates from every state (episodic tasks); or change the objective to reward per step, an average-reward criterion that this course does not use.

## Summary

### Lecture 3 on one board

- Fix  $\pi$ :  $P^\pi(s' | s) = \sum_a \pi(a | s)P(s' | s, a)$ ,  $r^\pi(s) = \sum_a \pi(a | s)r(s, a)$ ; a Markov chain with rewards, one row per state. “Never repair”: rows  $(\frac{2}{3}, \frac{1}{3}, 0)$ ,  $(0, \frac{2}{3}, \frac{1}{3})$ ,  $(0, 0, 1)$  and  $r^\pi = (8, 4, 0)$ . Return of one run:  $G_t = \sum_{k \geq 0} \gamma^k R_{t+k+1} = R_{t+1} + \gamma G_{t+1}$ .
- $V^\pi(s) = \mathbb{E}_\pi[G_t | S_t = s]$ ,  $Q^\pi(s, a) = \mathbb{E}_\pi[G_t | S_t = s, A_t = a]$ ;  $V^\pi(s) = \sum_a \pi(a | s)Q^\pi(s, a)$ ;  $J(\pi) = \sum_s d_0(s)V^\pi(s)$ . A value is an average of returns, not a return: 27.5 against one run’s 18.44.
- Bellman:  $V^\pi(s) = \sum_a \pi(a | s)[r(s, a) + \gamma \sum_{s'} P(s' | s, a)V^\pi(s')]$ , i.e.  $V^\pi = r^\pi + \gamma P^\pi V^\pi$ ;  $Q^\pi(s, a) = r(s, a) + \gamma \sum_{s'} P(s' | s, a)V^\pi(s')$ . One equation per state; the Markov property makes the future a value. Check at G:  $8 + 0.9(\frac{2}{3} \cdot 27.5 + \frac{1}{3} \cdot 10) = 27.5$ .
- Linear solve:  $(I - \gamma P^\pi)V^\pi = r^\pi$ , exactly one solution for  $\gamma < 1$ ,  $O(|S|^3)$ . “Never repair”:  $V^\pi = (27.5, 10, 0)$ , closed from B upward; “repair when broken”: (44.30, 32.41, 29.87), by computer.
- Iteration:  $V_{k+1} = r^\pi + \gamma P^\pi V_k$ . From zero,  $V_k$  is the  $k$ -step value; error  $\leq \gamma^k R_{\max}/(1 - \gamma)$ .  $(8, 4, 0) \rightarrow (14, 6.4, 0) \rightarrow (18.32, 7.84, 0) \rightarrow \dots \rightarrow (27.5, 10, 0)$ .
- $\gamma = 1$  with a policy that never stops: no solution, the iterates diverge (“repair when broken” earns 3.7 a week forever). With a policy that terminates w.p. one: fine (“never repair”: 36, 12, 0).
- Next: the best policy.  $V^*$ ,  $Q^*$ , and a max over actions in place of the average (Lecture 4).

## References

- Alexh Agarwal, Nan Jiang, Sham M. Kakade, and Wen Sun. Reinforcement learning: Theory and algorithms. Book draft, 2022. URL <https://rltheorybook.github.io/>.
- Emma Brunskill. CS 234: Reinforcement learning. Stanford University course materials, 2026. URL <https://web.stanford.edu/class/cs234/>. Winter 2026 offering.
- Nan Jiang. MDP preliminaries. Lecture notes, University of Illinois Urbana-Champaign, 2024. URL <https://nanjiang.cs.illinois.edu/files/cs542f22/note1.pdf>. Dated October 5, 2019; updated September 27, 2024. Accessed 2026-09-06.
- Nan Jiang. CS 443: Reinforcement learning. University of Illinois Urbana-Champaign course materials, 2026. URL <https://nanjiang.cs.illinois.edu/cs443/>. Spring 2026 offering.
- Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.
- Wen Sun. CS 4/5789: Introduction to reinforcement learning. Cornell University course materials, 2025. URL [https://wensun.github.io/CS4789\\_spring\\_2025.html](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>.