535514: Reinforcement Learning
Lecture 4 — Value Iteration, Policy Iteration, and Regularized MDPs

Ping-Chun Hsieh

March 4, 2024

This Lecture:
We Discuss 3 Surprising Facts of MDPs!

1. Value iteration (VI) can find an optimal policy

(VI $\rightarrow$ Value-based RL, e.g., Q-learning)

2. Policy iteration (PI) can also find an optimal policy

(PI $\rightarrow$ Policy-based RL, e.g., Policy Gradient, PPO, …)

3. "Existence" of an optimal policy for MDPs

2-Minute Review: What We Learned in Lecture 3

- Optimal value function $V^*(s)$:

- Optimal action-value function $Q^*(s, a)$:

- Optimal policy π^*:

- Existence of an optimal policy (to be proved):

Review: Bellman Optimality Equations

(1) V^* written in Q^*

$$V^*(s) = \max_{a \in \mathcal{A}} Q^*(s, a)$$

(2) Q^* written in V^*

$$Q^*(s, a) = R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^a_{ss'} V^*(s')$$

(3) V^* written in V^*

$$V^*(s) = \max_{a \in \mathcal{A}} R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^a_{ss'} V^*(s')$$

(4) Q^* written in Q^*

$$Q^*(s, a) = R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^a_{ss'} \Big(\max_{a \in \mathcal{A}} Q^*(s, a) \Big)$$

How to Solve the Bellman Optimality Equation?

$$V^*(s) = \max_{a \in \mathcal{A}} \left(R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^a_{ss'} V^*(s') \right)$$

- Question: Solve this by linear algebra?
- The "max" operation makes it non-linear
- We need to resort to iterative methods:
  - Value iteration
  - Policy iteration

1. Value Iteration (VI)

From Bellman Optimality Equation to "*Bellman Optimality Backup Operator*"

- Recall: Bellman optimality equation

$$V^*(s) = \max_{a \in \mathscr{A}} \left(R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^a_{ss'} V^*(s') \right)$$

- Define: Bellman <u>optimality</u> backup operator $T^* : \mathbb{R}^{|\mathcal{S}|} \rightarrow \mathbb{R}^{|\mathcal{S}|}$

$$T^*(V) := \max_{a \in \mathscr{A}} R^a + \gamma P^a V$$

(Comparison: IPE operator $T^\pi(V) := R^\pi + \gamma P^\pi V$)

Value Iteration: Pseudo Code

Step 1. Initialize $k = 0$ and set $V_0(s) = 0$ for all states

Step 2. Repeat the following until convergence:

$$V_{k+1} \leftarrow T^*(V_k)$$

- Equivalently: for each state s

$$V_{k+1}(s) \leftarrow \max_{a \in \mathscr{A}} \left(R_{s,a} + \gamma \sum_{s' \in \mathscr{S}} P^a_{ss'} V_k(s') \right)$$

- Remark: Complexity per iteration is $O(|\mathscr{S}|^2 |\mathscr{A}|)$
- Remark: Intermediate value functions V_k's may not correspond to any policy

Example: Shortest Path

$$V_{k+1}(s) \leftarrow \max_{a \in \mathcal{A}} \left(R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^a_{ss'} V_k(s') \right)$$

Problem

g			

V_1

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

V_2

0	-1	-1	-1
-1	-1	-1	-1
-1	-1	-1	-1
-1	-1	-1	-1

V_3

0	-1	-2	-2
-1	-2	-2	-2
-2	-2	-2	-2
-2	-2	-2	-2

V_4

0	-1	-2	-3
-1	-2	-3	-3
-2	-3	-3	-3
-3	-3	-3	-3

V_5

0	-1	-2	-3
-1	-2	-3	-4
-2	-3	-4	-4
-3	-4	-4	-4

V_6

0	-1	-2	-3
-1	-2	-3	-4
-2	-3	-4	-5
-3	-4	-5	-5

V_7

0	-1	-2	-3
-1	-2	-3	-4
-2	-3	-4	-5
-3	-4	-5	-6

(Suppose $\gamma = 1$ in this example)

Example: Shortest Path (Cont.)

$$V_{k+1}(s) \leftarrow \max_{a \in \mathcal{A}} \left(R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^a_{ss'} V_k(s') \right)$$

g			

Problem

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

V_1

0	-1	-1	-1
-1	-1	-1	-1
-1	-1	-1	-1
-1	-1	-1	-1

V_2

(Suppose $\gamma = 1$ in this example)

Convergence of Value Iteration

- **Theorem (VI converges on V^*):** For any initial $V_0 \in \mathbb{R}^{|\mathcal{S}|}$, Value Iteration achieves that $V_k \to V^*$, as $k \to \infty$.

- Question: How to show this?

Convergence of Value Iteration

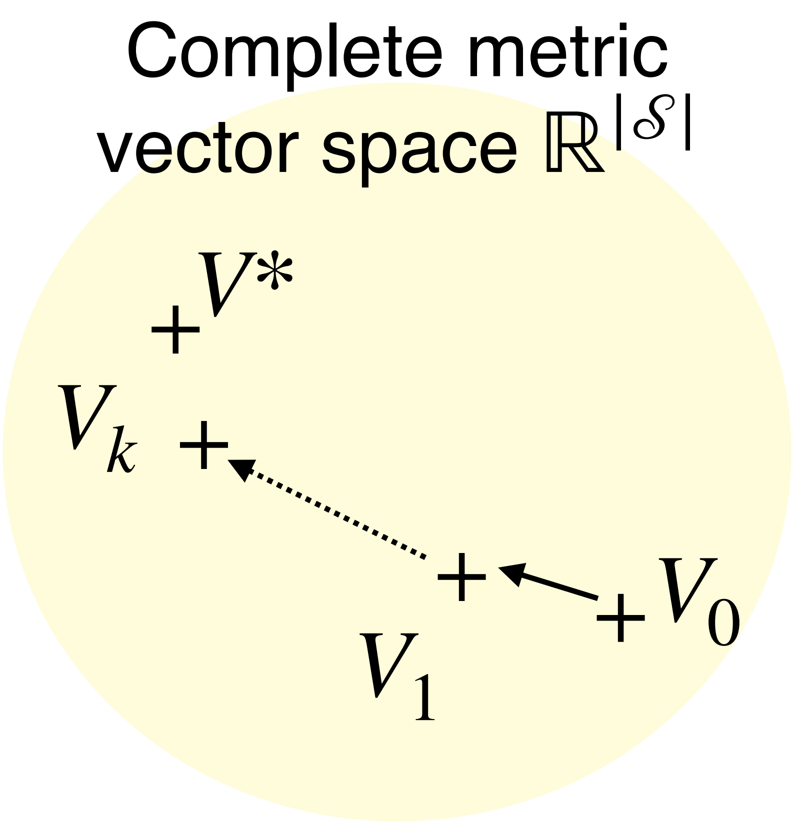
- **Theorem (VI converges on V^*):** For any initial $V_0 \in \mathbb{R}^{|\mathcal{S}|}$, Value Iteration achieves that $V_k \to V^*$, as $k \to \infty$.

- Proof: We prove convergence by the following 3 steps

(B1): Show that T^* is a *contraction operator*

(B2): Under a contraction operator, $\{V_k\}$ shall converge to the unique fixed point (why?)

(B3): Since V^* is a fixed point, then $V_k \to V^*$ due to uniqueness



(B1): T^* is a γ-Contraction Operator on V

- Bellman optimality backup operator: $T^*(V) := \max_{a \in \mathscr{A}} (R^a + \gamma P^a V)$

$$
\begin{aligned}
&||T^*(V) - T^*(\hat{V})||_\infty \\
&= \max_s \left| T^*(V)(s) - T^*(\hat{V})(s) \right| \\
&= \max_s \left| \max_a \left(R_{s,a} + \gamma \sum_{s'} P^a_{ss'} V(s') \right) - \max_{a'} \left(R_{s,a'} + \gamma \sum_{s'} P^{a'}_{ss'} \hat{V}(s') \right) \right| \\
&\le \max_s \max_a \left| \left(R_{s,a} + \gamma \sum_{s'} P^a_{ss'} V(s') \right) - \left(R_{s,a} + \gamma \sum_{s'} P^a_{ss'} \hat{V}(s') \right) \right| \\
&\le \max_s \max_a \left| \gamma \sum_{s'} P^a_{ss'} \big(V(s') - \hat{V}(s') \big) \right| \\
&\le \gamma ||(V - \hat{V})||_\infty
\end{aligned}
$$

Therefore, T^* is a <u>γ-contraction operator ($\gamma < 1$)</u>

(B2): T^* Converges to the Unique Fixed Point

- T^* is a γ-contraction operator in a complete metric space
- By Banach Fixed Point Theorem, T^* converges to the unique fixed point
- Note that V^* is one fixed point of T^* (why?)
- Therefore, V^* is the unique fixed point of T^*

(B3): $V_k \to V^*$ as $k \to \infty$

- Let's put everything together!

Discussion: Issues With Value Iteration

- Question 1: What would happen if T^* has multiple fixed points?
- Question 2: In how many iterations will VI converge?
- Question 3: By applying VI, could we find an optimal policy?
- Question 4: By using VI, could we directly prove the existence of an optimal policy?

Discussion: Asymptotic Convergence vs Convergence Rate

VI enjoys the following types of convergence:

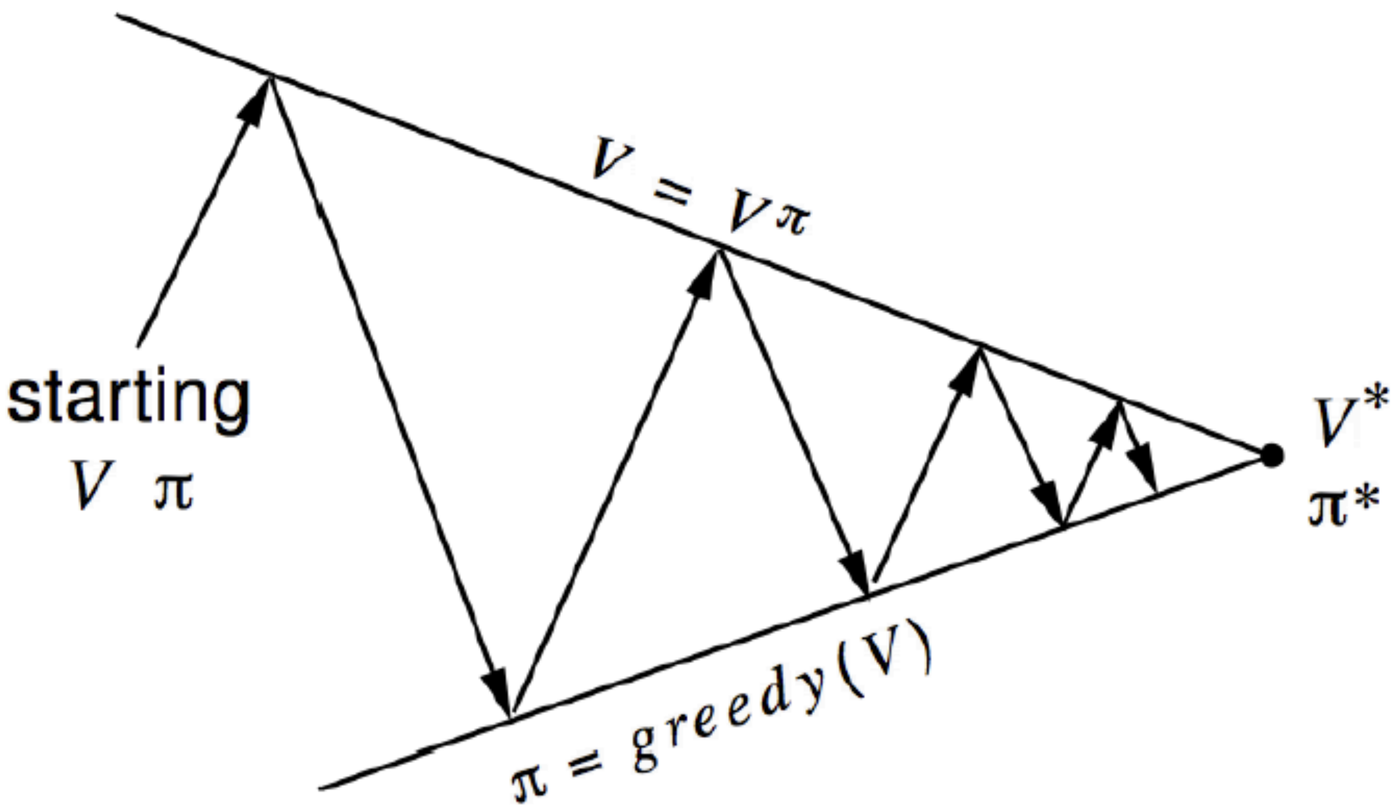
- **Asymptotic Convergence:** $V_k \to V^*$, as $k \to \infty$

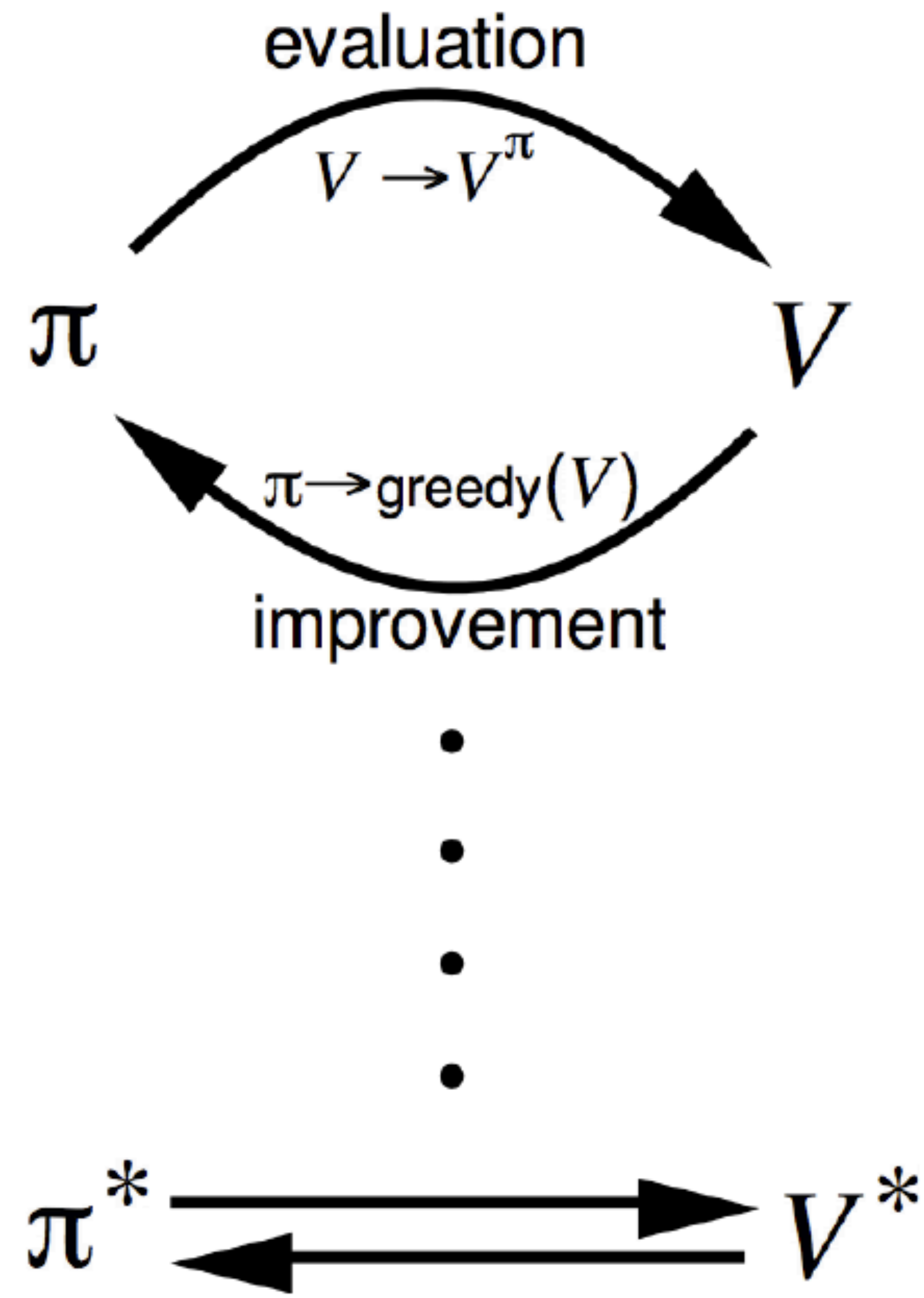
- **Convergence Rate:** $\|V_k - V^*\|_\infty \leq \gamma^k \cdot \|V_0 - V^*\|_\infty$

- Question: Which one is stronger?

2. Policy Iteration (PI)

Policy Iteration: Generic Procedure





Policy evaluation Estimate v_π
  Iterative policy evaluation

Policy improvement Generate $\pi' \geq \pi$
  Greedy policy improvement

(Slide Credit: David Silver)

Policy Iteration: Pseudo Code

(We focus on deterministic policies)

Step 1. Initialize $k = 0$ and set $\pi_0(s)$ arbitrarily for all states

Step 2. While k is zero or $\pi_k \neq \pi_{k-1}$:

- Derive V^{π_k} via policy evaluation for π_k (iterative/non-iterative)
- Derive π_{k+1} by greedy one-step policy improvement

One-Step Policy Improvement

- Given V^{π_k}, compute $Q^{\pi_k}(s, a)$:

$$Q^{\pi_k}(s, a) = R(s, a) + \gamma \sum_{s} P(s' \,|\, s, a) V^{\pi_k}(s')$$

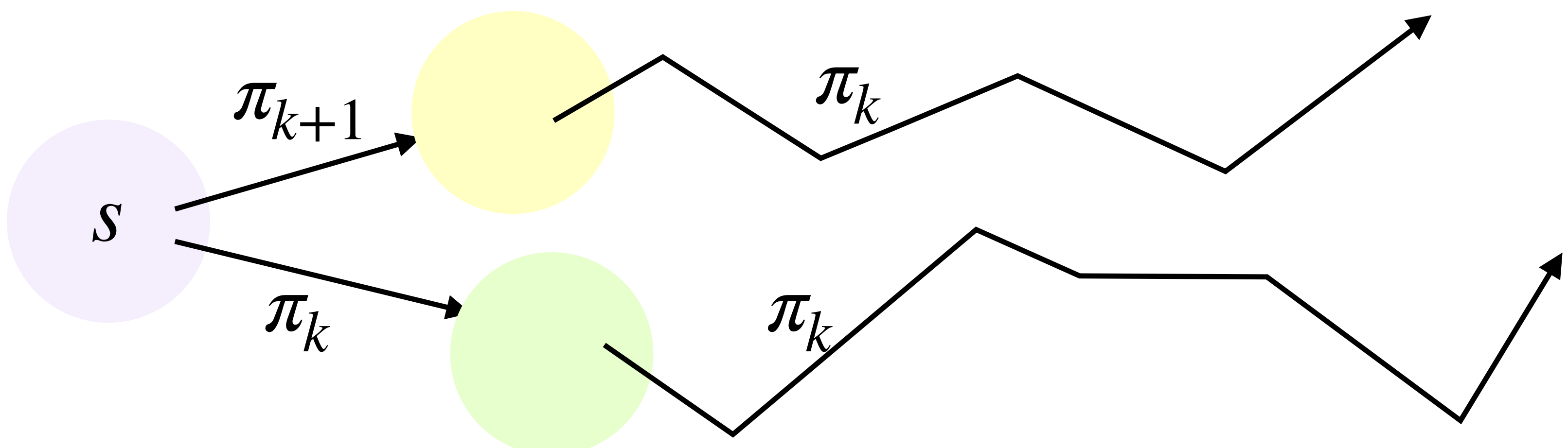
- Derive the new policy π_{k+1}: For all states s,

$$\pi_{k+1}(s) = \arg\max_{a \in \mathcal{A}} Q^{\pi_k}(s, a)$$

- Note: We will use $R(s, a)$ and $R_{s,a}$ interchangeably

Why is One-Step Policy Improvement Reasonable?

- Question: Suppose we take $\pi_{k+1}(s)$ for one step and then follow π_k subsequently. Is this better than just following π_k?



$$Q^{\pi_k}(s,a) = R(s,a) + \gamma \sum_{s} P(s' \mid s,a) V^{\pi_k}(s')$$

$$\max_{a \in \mathscr{A}} Q^{\pi_k}(s,a) \geq R(s, \pi_k(s)) + \gamma \sum_{s} P(s' \mid s,a) V^{\pi_k}(s') \;=\; V^{\pi_k}(s)$$

$$\pi_{k+1}(s) = \arg\max_{a \in \mathscr{A}} Q^{\pi_k}(s,a)$$

- Question: But how about following π_{k+1} all the way?

Monotonic Improvement in Policy

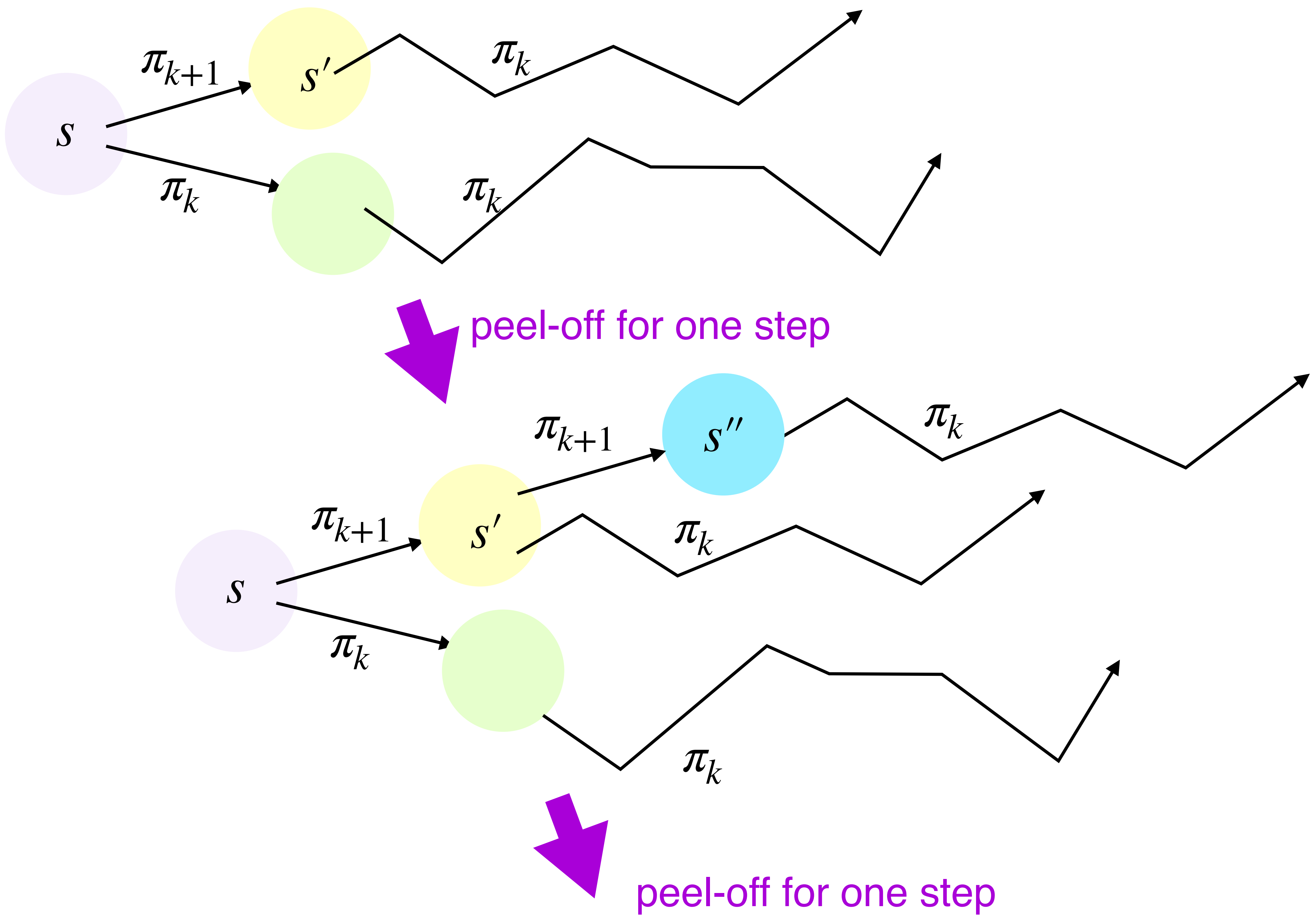
- **Recall**: Partial ordering of policies

$$\pi \geq \pi' \quad \text{if} \quad V^{\pi}(s) \geq V^{\pi'}(s), \forall s$$

- **Question**: Do we have $\pi_{k+1} \geq \pi_k$?

- **Theorem (Monotonic Policy Improvement)**: Under the one-step policy improvement step, we have $V^{\pi_{k+1}}(s) \geq V^{\pi_k}(s)$ for all $s \in \mathcal{S}$ and hence $\pi_{k+1} \geq \pi_k$.

Proof Idea: “Peeling off”



Proof: Monotonic Policy Improvement

$$
\begin{aligned}
V^{\pi_k}(s) &\leq \max_{a \in \mathscr{A}} Q^{\pi_k}(s, a) \\
&= \max_{a \in \mathscr{A}} R(s, a) + \gamma \sum_{s'} P(s' \mid s, a) V^{\pi_k}(s') \\
&= R(s, \pi_{k+1}(s)) + \gamma \sum_{s'} P(s' \mid s, \pi_{k+1}(s)) V^{\pi_k}(s') \\
&\leq R(s, \pi_{k+1}(s)) + \gamma \sum_{s'} P(s' \mid s, \pi_{k+1}(s)) \boxed{\max_{a' \in \mathscr{A}} Q^{\pi_k}(s', a')} \\
&= R(s, \pi_{k+1}(s)) + \gamma \sum_{s'} P(s' \mid s, \pi_{k+1}(s)) \\
&\qquad \times \left(R(s', \pi_{k+1}(s)) + \gamma \sum_{s''} P(s'' \mid s', \pi_{k+1}(s')) V^{\pi_k}(s'') \right) \\
&\ldots \\
&= V^{\pi_{k+1}}(s)
\end{aligned}
$$

Discussions: Policy Iteration

- Question 1: Will policy iteration terminate in finitely many iterations?

- Yes, in at most $|\mathscr{A}|^{|\mathscr{S}|}$ iterations (assume $|\mathscr{A}|, |\mathscr{S}|$ are finite)

  (There are $|\mathscr{A}|^{|\mathscr{S}|}$ deterministic policies)

  (If $\pi_{k+1} \neq \pi_k$, then they must differ by at least 1 entry)

  (Monotonic policy improvement: $\pi_{k+1} \geq \pi_k$)

Discussions: Policy Iteration (Cont.)

- Question 2: If we have $\pi_{k+1} = \pi_k$, what shall we expect about π_{k+2}?

- Recall:

$$Q^{\pi_k}(s,a) = R(s,a) + \gamma \sum_{s} P(s'\,|\,s,a) V^{\pi_k}(s')$$

$$\pi_{k+1}(s) = \arg\max_a Q^{\pi_k}(s,a)$$

$$\pi_{k+2}(s) = \arg\max_a Q^{\pi_{k+1}}(s,a)$$

Therefore, policy iteration can terminate when $\pi_{k+1} = \pi_k$

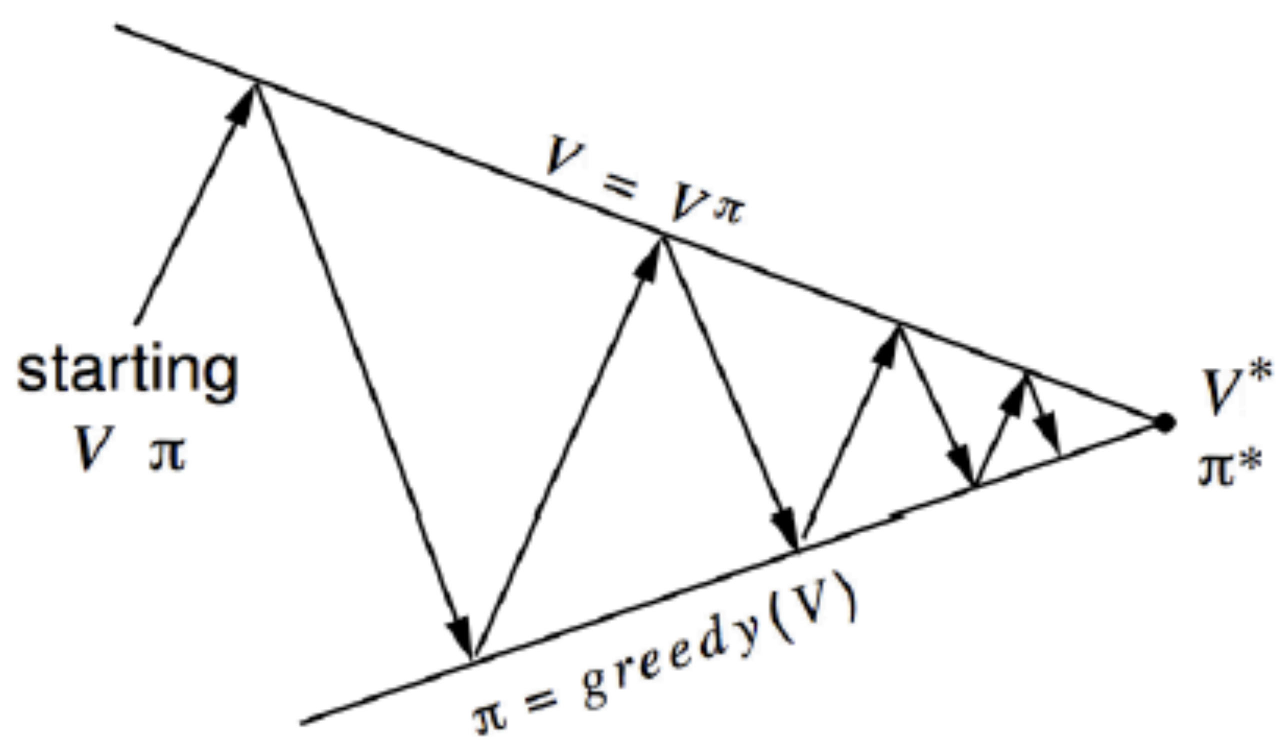
Moreover, $\pi_{k+1} = \pi_k$ implies Bellman optimality equation is satisfied by π_k:

$$V^{\pi_k}(s) = \max_a Q^{\pi_k}(s,a)$$

Therefore, π_k must be a (deterministic) optimal policy (Why?)

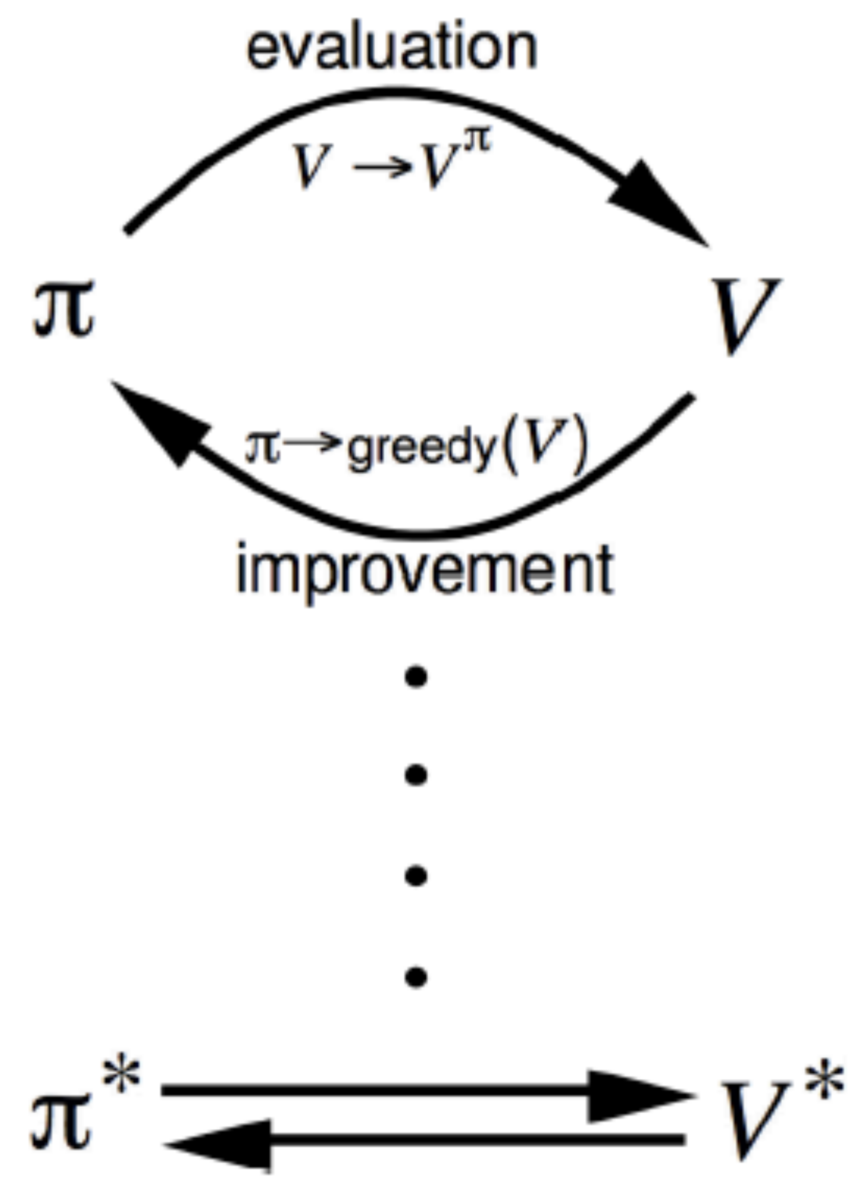
Hence, PI proves the existence of an optimal policy

Extension: Generalized Policy Iteration



Policy evaluation Estimate v_π
Any policy evaluation algorithm

Policy improvement Generate $\pi' \geq \pi$
Any policy improvement algorithm



- Remark: Policy gradient methods can be interpreted as an instance of generalized policy iteration (discussed in Lectures 5-7)

 (Figure Source: Sutton & Barto, Section 4.6)

Summary

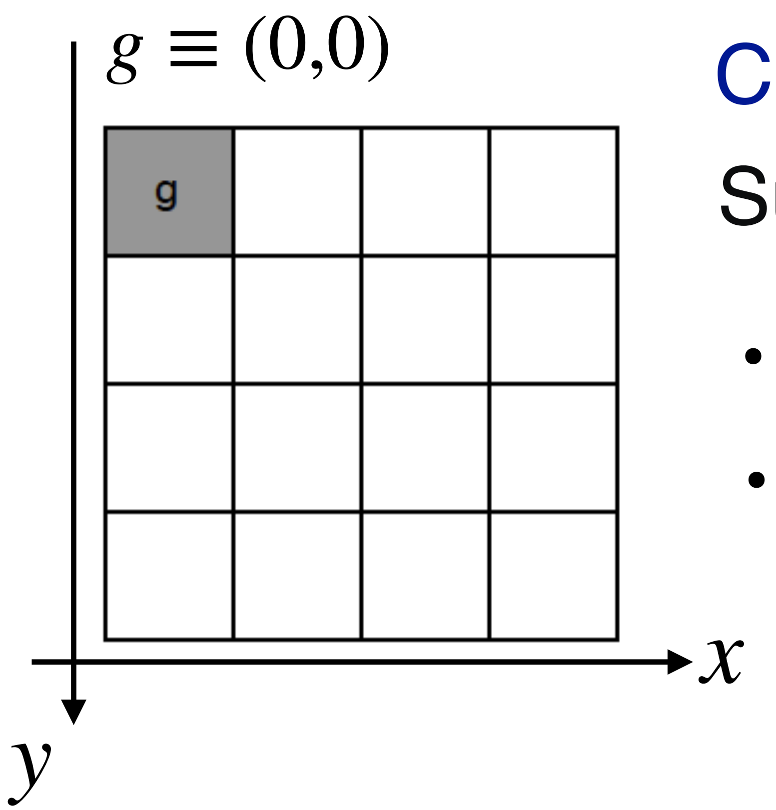
Problem	Bellman Equation	Algorithm
Prediction	Bellman Expectation Equation	Iterative Policy Evaluation
Control	Bellman Expectation Equation + Greedy Policy Improvement	Policy Iteration
Control	Bellman Optimality Equation	Value Iteration

(Table Source: David Silver's lecture slides)

Extension: Regularized MDPs

Geist et al., “A Theory of Regularized Markov Decision Processes,” ICML 2019

Motivation: ***Reward Shaping*** for Faster Learning



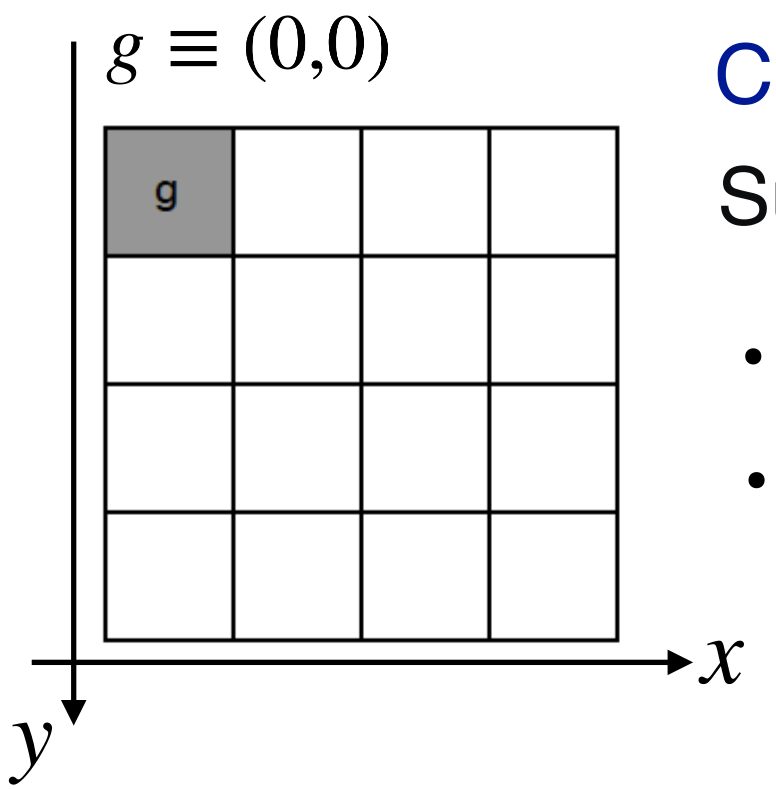
Consider our favorite “shortest path” problem

Suppose the reward function of the environment is

- $R((1,0), \leftarrow) = R((0,1), \uparrow) = 1$
- $R(s, a) = 0$, otherwise

Is this problem easy to learn?

Motivation: ***Reward Shaping*** for Faster Learning



Consider our favorite "shortest path" problem

Suppose the reward function of the environment is

- $R((1,0), \leftarrow) = R((0,1), \uparrow) = 1$
- $R(s, a) = 0$, otherwise

Is this problem easy to learn?

What if we augment the reward function (denoted by $\tilde{R}$) as follows:

- $\tilde{R}((1,0), \leftarrow) = \tilde{R}((0,1), \uparrow) = 1$
- $\tilde{R}(s, a) = \|s - g\|_1$, otherwise

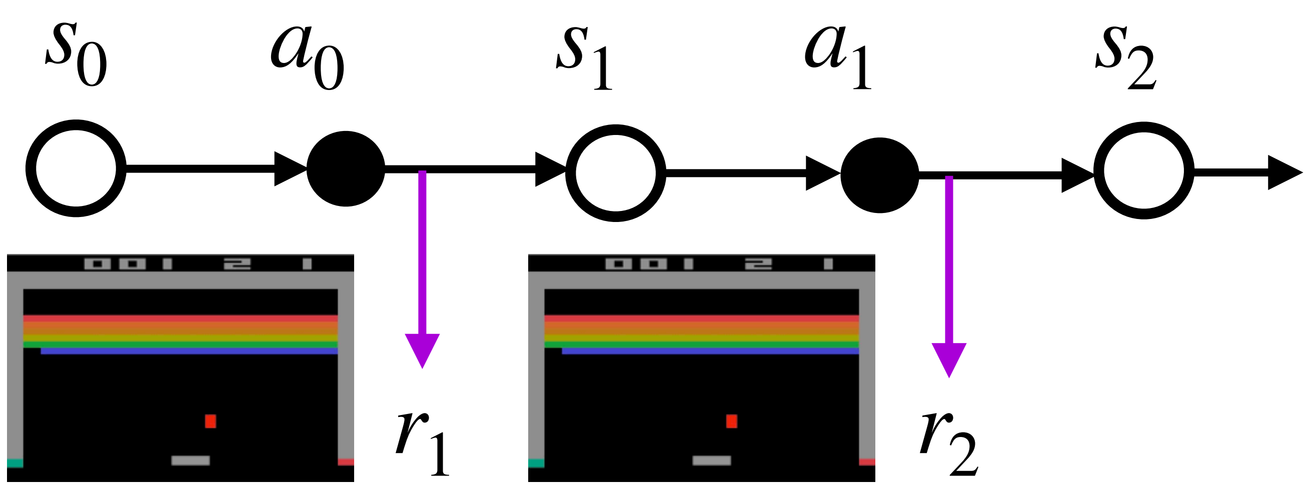
Question: Does this lead to the same optimal policy?

Question: Is this problem easier to learn?

Example: ***Entropy*** as Intrinsic Reward

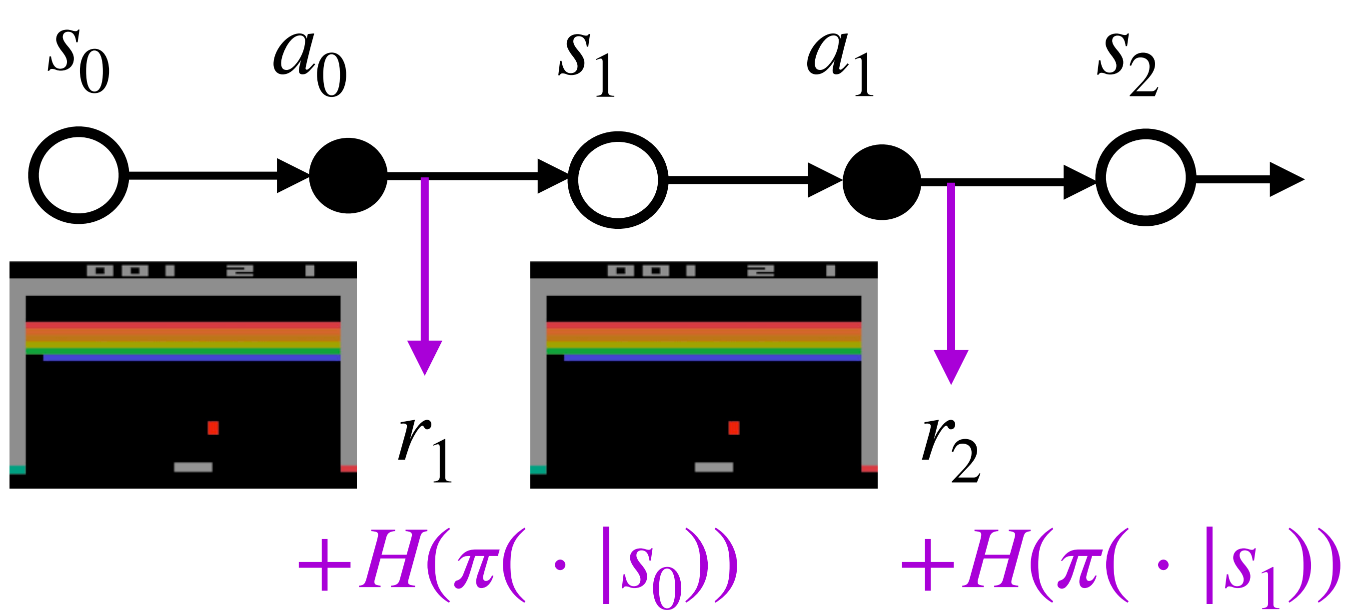
- **Extrinsic** Rewards & **Intrinsic** Rewards

Standard MDP



Extrinsic rewards

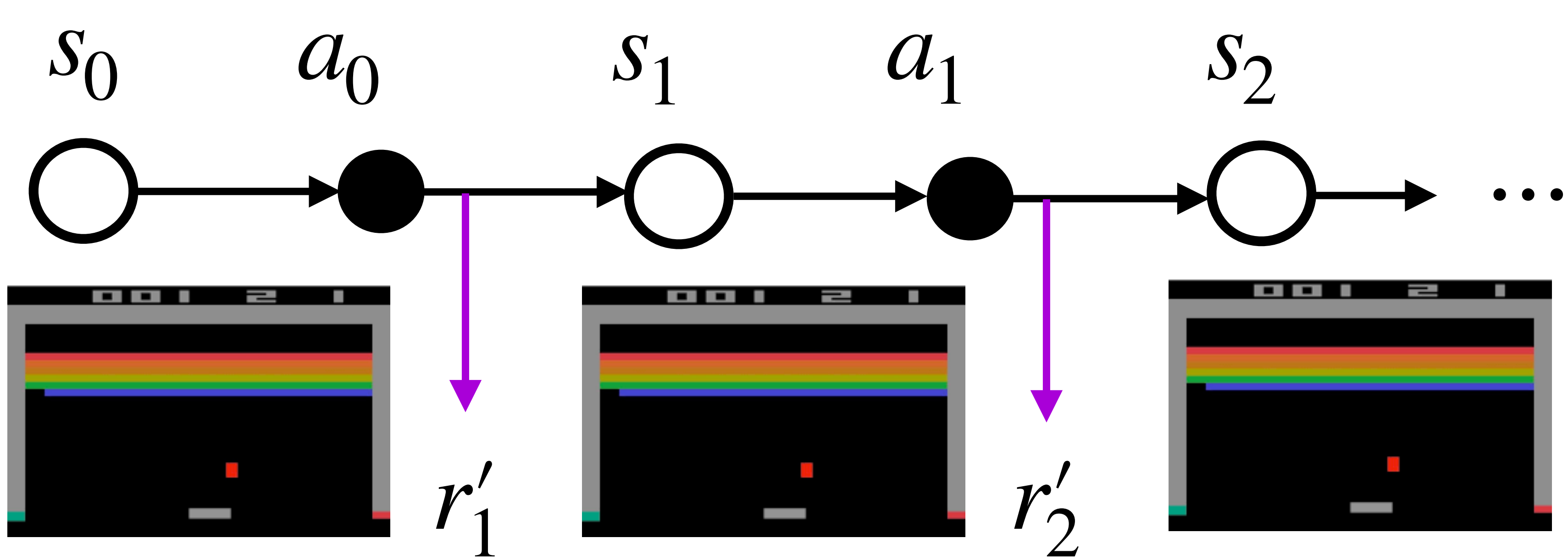
MDP with entropy bonus



Entropy as intrinsic rewards for better exploration

More Generally: *Regularized MDPs*

Regularized MDP = Standard MDP + Regularized rewards!



$$r'_1 = r_1 + \Omega(\pi(\cdot \mid s_0)) \qquad r'_2 = r_2 + \Omega(\pi(\cdot \mid s_1))$$

- A regularized MDP can be specified by $(\mathcal{S}, \mathcal{A}, P, R, \Omega, \gamma)$
  - $\Omega(\cdot)$: A function that maps an *action distribution* to a *real number*

Value Functions of *Regularized MDPs*

	Unregularized MDP	Regularized MDP
Return	$G_t := r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots$	
Value function	$V^\pi(s) := \mathbb{E}[G_t \mid s_t = s; \pi]$	
Q function	$Q^\pi(s,a) := \mathbb{E}[G_t \mid s_t = s, a_t = a; \pi]$	
Bellman expectation equations	$V^\pi(s) = \sum_{a \in \mathcal{A}} \pi(a \mid s) Q^\pi(s,a)$ $Q^\pi(s,a) = R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^a_{ss'} V^\pi(s')$	

Next Question: *How to find* V^{π}_{Ω}*?*

Regularized Bellman Expectation Operator

- Regularized Bellman Expectation Operator

$$[T^{\pi}_{\Omega}V](s) := [T^{\pi}V](s) - \underbrace{\Omega(\pi(\cdot\,|\,s))}_{\text{regularization term}}$$

$$= \underbrace{R^{\pi}_{s} - \Omega(\pi(\cdot\,|\,s))}_{\text{regularized immediate reward}} + \gamma P^{\pi}_{ss'}V$$

- Question: Is T^{π}_{Ω} a contraction? *Yes! (in L_{∞}-norm)*
- Therefore, under T^{π}_{Ω}, there is a unique fixed point, which is the regularized value function V^{π}_{Ω}
- To find V^{π}_{Ω}, we can use the regularized IPE method

Geist et al., "A Theory of Regularized Markov Decision Processes," ICML 2019

Next Question: *How to define "optimality" for regularized MDPs?*

Regularized Bellman Optimality Operator

- Recall: Bellman optimality operator for ***unregularized*** MDPs

$$[T^*V](s) = \max_{a \in \mathscr{A}} R_s^a + \gamma P_s^a V = \max_{\pi} \underbrace{R_s^\pi + \gamma P_s^\pi V}_{=[T^\pi V](s)}$$

- Regularized Bellman optimality equations

$$[T^*_\Omega V](s) = \max_{\pi} \{[T^\pi_\Omega V](s)\}$$ ←a greedy step!

- Useful Facts

1. T^*_Ω is also a contraction map (in L_∞-norm)
2. There is a unique fixed point of T^*_Ω
3. We define regularized optimal value function V^*_Ω as the fixed point of T^*_Ω

(See HW1 problem for more details)

Regularized Q-functions and Policy Iteration

Regularized Q-function	$Q^{\pi}_{\Omega}(s,a) := R^a_s + \gamma E_{s' \sim P(\cdot \mid s,a)}[V^{\pi}_{\Omega}(s')]$
Regularized optimal Q-function	$Q^{*}_{\Omega}(s,a) := R^a_s + \gamma E_{s' \sim P(\cdot \mid s,a)}[V^{*}_{\Omega}(s')]$

- Property: $[T^{\pi}_{\Omega} V^{\pi}_{\Omega}](s) = V^{\pi}_{\Omega}(s) = \langle \pi(\cdot \mid s), Q^{\pi}_{\Omega}(s, \cdot) \rangle - \Omega(\pi(\cdot \mid s))$
- Question: Now we are ready for "regularized policy iteration". How?

(See HW1 problem for more details)

Regularized Policy Iteration (Regularized PI)

Regularized Policy Iteration

1. Initialize $k = 0$ and set $\pi_0(\cdot \mid s)$ arbitrarily for all states
2. While $\underline{k \text{ is zero}}$ or $\underline{\pi_k \neq \pi_{k-1}}$:
   - Derive $V_\Omega^{\pi_k}$ and $Q_\Omega^{\pi_k}$ via policy evaluation
   - Derive π_{k+1} by greedy policy improvement:

$$\pi_{k+1}(\cdot \mid s) = \arg\max_{\pi} \left\{ \langle \pi(\cdot \mid s), Q_\Omega^{\pi_k}(s, \cdot) \rangle - \Omega(\pi(\cdot \mid s)) \right\}$$

Regularized PI + Entropy Regularizer

- Soft Policy Iteration: A special case of regularized PI with negative entropy

$$\Omega(\pi(\cdot \mid s)) := \sum_a \pi(a \mid s) \log \pi(a \mid s)$$

- **Theorem**: Under Soft Policy Iteration, we have

$$\pi_{k+1}(\cdot \mid s) = \arg\max_{\pi} \left\{ \langle \pi(\cdot \mid s), Q_{\Omega}^{\pi_k}(s, \cdot) \rangle - \Omega(\pi(\cdot \mid s)) \right\}$$
$$= \frac{\exp(Q_{\Omega}^{\pi_k}(s, \cdot))}{\sum_{a \in \mathcal{A}} \exp(Q_{\Omega}^{\pi_k}(s, a))}$$

- Proof: HW1 problem