535514: Reinforcement Learning

Lecture 3 — MDP and Optimal Policies

Ping-Chun Hsieh

February 26, 2024

Announcement

- No class this Thursday (2/29)

- To make up for the lecture:
  - **Plan A**: 10min extension on 3/4 (Mon.), 3/11 (Mon.), 3/18 (Mon.), 3/25 (Mon.), 4/1 (Mon.)
  - **Plan B**: Make-up lecture on 3/7 (Thursday), 4:30pm-5:20pm

Human Intelligence vs Machine Intelligence?

Robot PR1

Is this robot impressive to you?

(A video made by Pieter Abbeel back in 2013)

(Source: https://www.youtube.com/watch?v=6qBZPSTR96N4)

Human Intelligence vs Machine Intelligence?

Pieter Abbeel
14.3K subscribers

Subscribe

36 | Share | Download | ...

6,299 views May 29, 2013
This video showcases PR1, a robot developed by Keenan Wyrobek, Eric Berger, HFM Van der Loos, and Ken Salisbury at Stanford. It is the predecessor of the Willow Garage PR2. The PR1 was tele-operated during this entire video.

Reference: K. Wyrobek, E. Berger, H.F.M. Van der Loos, and K. Salisbury. Towards a personal robotics development platform: Rationale and design of an intrinsically safe personal robot. In the Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2008.

Show less

Actually not that impressive
(as PR1 was tele-operated by a human lol)

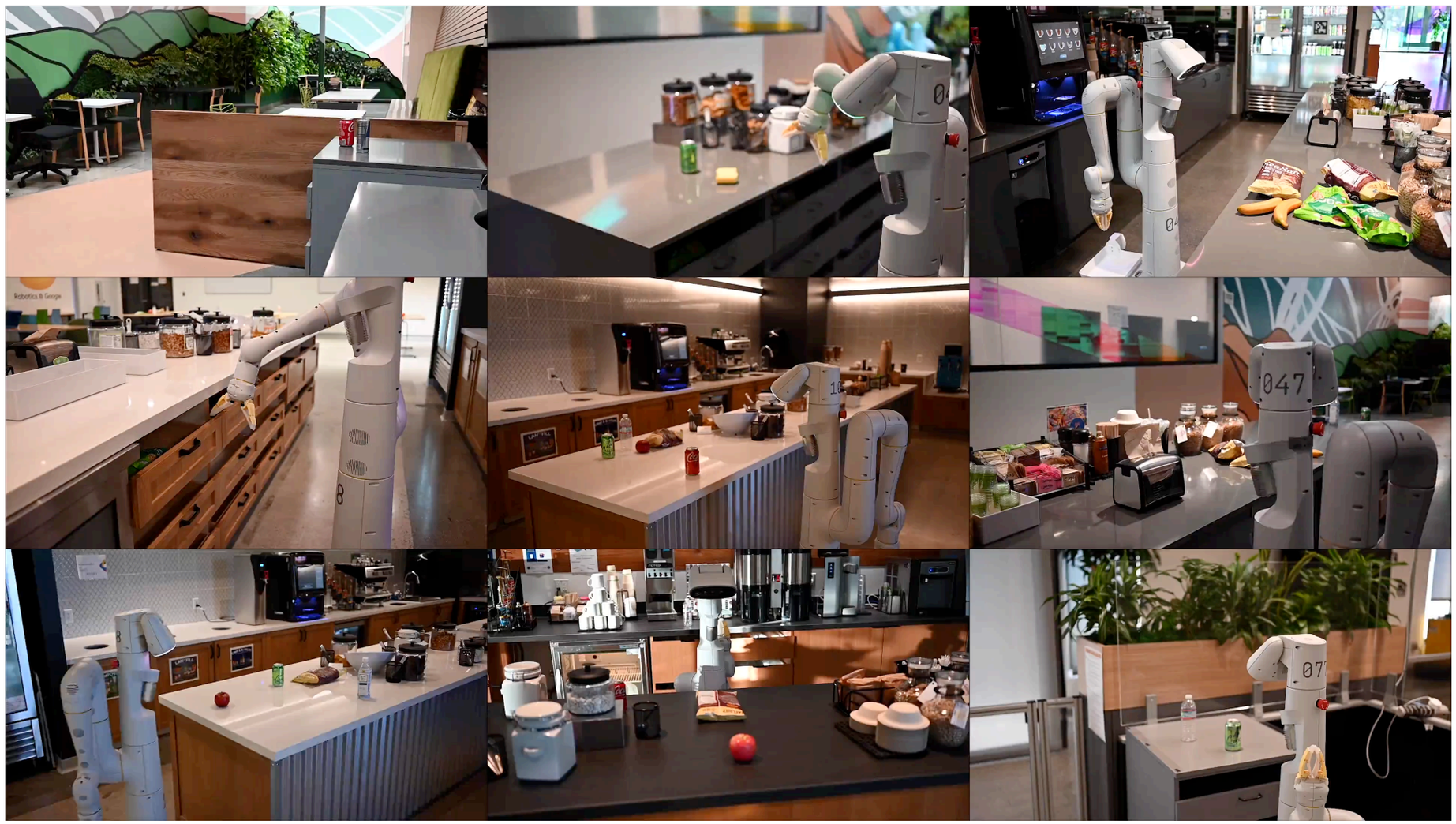
Takeaways:

1. Humans and machines are NOT very different in “physical capability”
2. The difference lies in the “mind”, rather than the “body”

That is why we love RL research! (To extend human mind!)

How About Now?

RT-1 (Robotics Transformer) by Google



(Source: https://blog.research.google/2022/12/rt-1-robotics-transformer-for-real.html)

Review: Markov Decision Process

- **Markov Decision Process (MDP)**: An MDP $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$ is specified by

Underlying Dynamics

1. State space $\mathcal{S}$ (assumed finite)
2. Action space $\mathcal{A}$ (assumed finite)
3. Transition matrix $P = [P^a_{ss'}]$ with $P^a_{ss'} = \mathbb{P}[s_{t+1} = s' \,|\, s_t = s, a_t = a]$

Task / Goal

4. Reward function $R_{s,a} = \mathbb{E}[r_{t+1} \,|\, s_t = s, a_t = a]$
5. Discount factor $\gamma \in [0,1]$

- In this lecture, we shall assume the model parameters P and R^a_s are known (i.e. no learning)

Review: How to Specify a Policy?

History-dependent policies

$\pi_t(a_t = a \mid s_1, s_2, \cdots, s_t)$

- Idea: “policy" is a lookup table specifying the action taken at any given state

- (Randomized) Policy: A policy π is a conditional distribution over possible actions given state s, i.e for any $s \in \mathcal{S}, a \in \mathcal{A}$

$$\pi(a \mid s) := \mathbb{P}(a_t = a \mid s_t = s)$$

- Let’s use Π to denote the set of all stationary policies

- Remark: Here we focus on stationary policies, i.e., π does not depend on time t

[Puterman, 1994]

- Question: What’s the intuition behind using stationary policies?

Markov property

Connection Between MDP and MRP

- Idea: Fix a policy $\pi(a \mid s)$ for an MDP $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$:

1. What is the probability of $s \to s'$ under π?

$$P^{\pi}_{ss'} = \sum_{a \in \mathcal{A}} \pi(a \mid s) P^{a}_{ss'}$$

(total probability theorem)

2. What is the expected reward of begin in s under π?

$$R^{\pi}_{s} = \sum_{a \in \mathcal{A}} \pi(a \mid s) R_{s,a}$$

- Under a fixed $\pi(a \mid s)$, we get an π-induced MRP $(\mathcal{S}, P^{\pi}, R^{\pi}, \gamma)$

Goals, Return, and State-Value Function of MDPs

- Goal: Given P and R, find a policy π that maximizes the expected cumulative reward (Question: this formulation can be viewed as optimal control, model-based RL, or model-free RL?)
- Return G_t: Cumulative discounted rewards over a single trajectory from t onwards (random)

$$G_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$$

- State-value function $V^\pi(s)$: Expected return if we start from state s

$$V^\pi(s) = \mathbb{E}[G_t \mid s_t = s; \pi]$$

- Question: The expectation above is taken w.r.t. randomness of ?

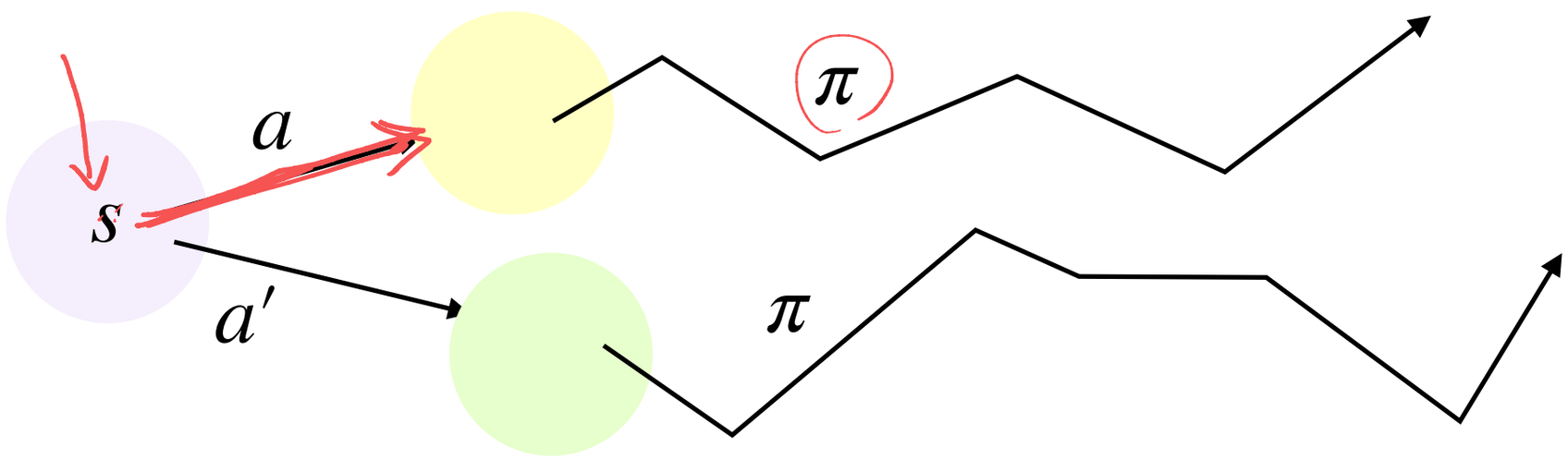
① π ② State transition ③ reward

Action-Value Function $Q^{\pi}(s, a)$

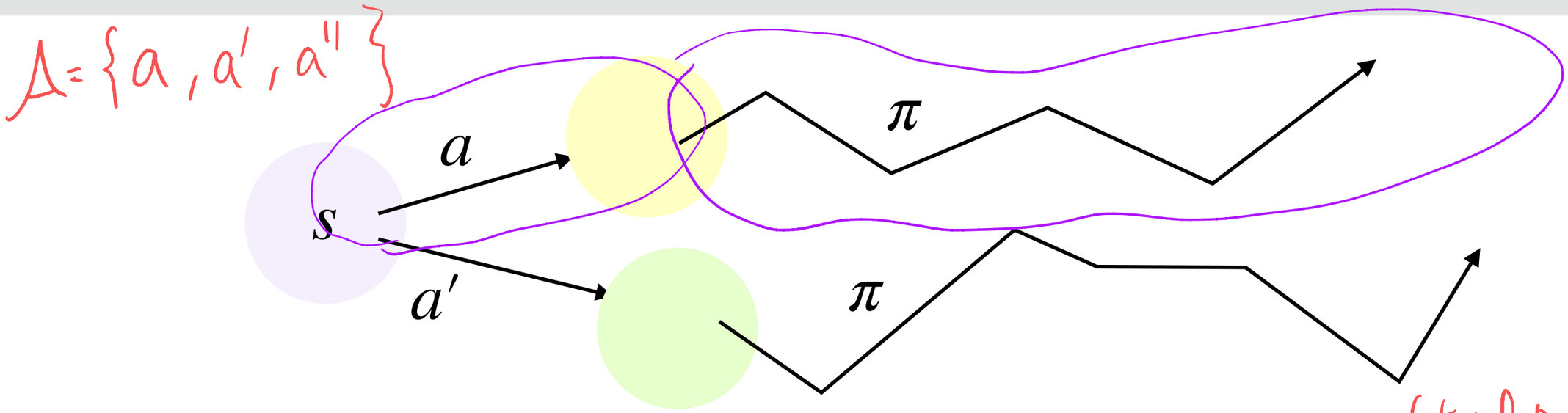
(Q-function)

- Action-value function $Q^{\pi}(s, a)$: Expected return if we start from state s and take action a, and then follow policy π

$$Q^{\pi}(s, a) = \mathbb{E}[G_t \mid s_t = s, a_t = a; \pi]$$



Natural Connection Between $V^{\pi}(s)$ and $Q^{\pi}(s,a)$



(1) V written in Q

$$V^{\pi}(s) = \sum_{a \in \mathcal{A}} \pi(a \mid s) Q^{\pi}(s,a)$$

(total probability theorem)

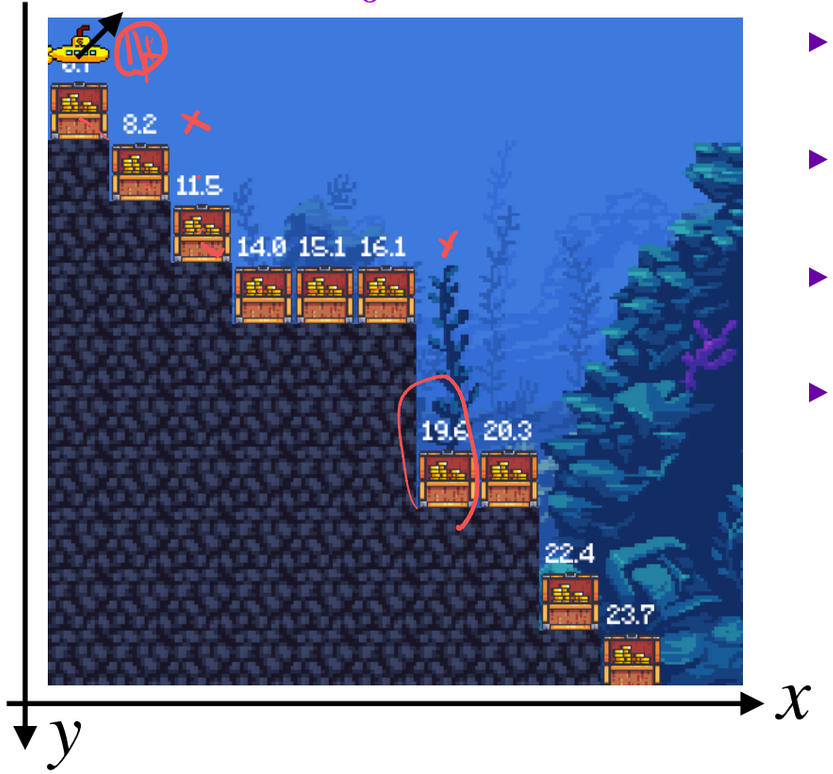
(2) Q written in V

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

Example: Value Functions

Deep Sea Treasure

Initial state s_0 = (0,0)



- **State**: Coordinate (x, y)
- **Actions**: Right or Down
- **Reward**: Values on the treasures
- Submarine stays at the same place after hitting a wall
- The episode ends after finding a treasure
- **Policy**: $\pi(\text{Right}|s) = \pi(\text{Down}|s) = 0.5$, for all states s
- **Question**: $Q^\pi(s_0, \text{Down})$=? $Q^\pi(s_0, \text{Right})$=? $V^\pi(s_0)$=?

$\tilde{Q}^\pi(s_0, \text{Down}) = 0.1$

$\tilde{Q}^\pi(s_0, \text{Right}) = \sum_i (\text{treasure}_i) \cdot (0.5)^{\#\text{steps to reach treasure } i} \cdot (\#\text{of possible paths to reach treasure } i)$

$(0.5)^2 \quad (0.5)^4 \cdot C_1^2 C_1^1$

$V^\pi(s_0) = \pi(\downarrow|s_0) Q^\pi(s_0, \text{Down}) + \pi(\rightarrow|s_0) Q^\pi(s_0, \text{Right})$

$\pi(\downarrow|s_0) = 0.5 \qquad \pi(\rightarrow|s_0) = 0.5$

(Figure source: MO-Gymnasium)

Recursions for Computing $V^{\pi}(s)$ and $Q^{\pi}(s, a)$

(1) V written in Q

$$V^{\pi}(s) = \sum_{a \in \mathcal{A}} \pi(a \mid s) Q^{\pi}(s, a)$$

(2) Q written in V

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

(3) V written in V

$$V^{\pi}(s) = \sum_{a \in \mathcal{A}} \pi(a|s) \cdot \left(R_{s,a} + \gamma \cdot \sum_{s' \in S} P^{a}_{ss'} V^{\pi}(s') \right)$$

(4) Q written in Q

$$Q^{\pi}(s,a) = R_{s,a} + \gamma \cdot \sum_{s' \in S} P^{a}_{ss'} \cdot \left(\sum_{a' \in A} \pi(a'|s') \cdot Q^{\pi}(s', a') \right)$$

Bellman equations

(Non-Iterative) MDP Policy Evaluation

For $V^{\pi}(s)$:

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

Consider π-induced MRP $(\mathcal{S}, P^{\pi}, R^{\pi}, \gamma)$:

$$R^{\pi}_{s} = \sum_{a \in \mathcal{A}} \pi(a \mid s) R_{s,a}$$

$$P^{\pi}_{ss'} = \sum_{a \in \mathcal{A}} \pi(a \mid s) P^{a}_{ss'}$$

Matrix form:

$$V^{\pi} = R^{\pi} + \gamma P^{\pi} V^{\pi}$$

... From Lecture 2

Solution of V^{π}:

$$V^{\pi} = \left(I - \gamma P^{\pi} \right)^{-1} R^{\pi}$$

Iterative MDP Policy Evaluation (IPE)

We know:

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

- Iterative policy evaluation for a fixed policy π:

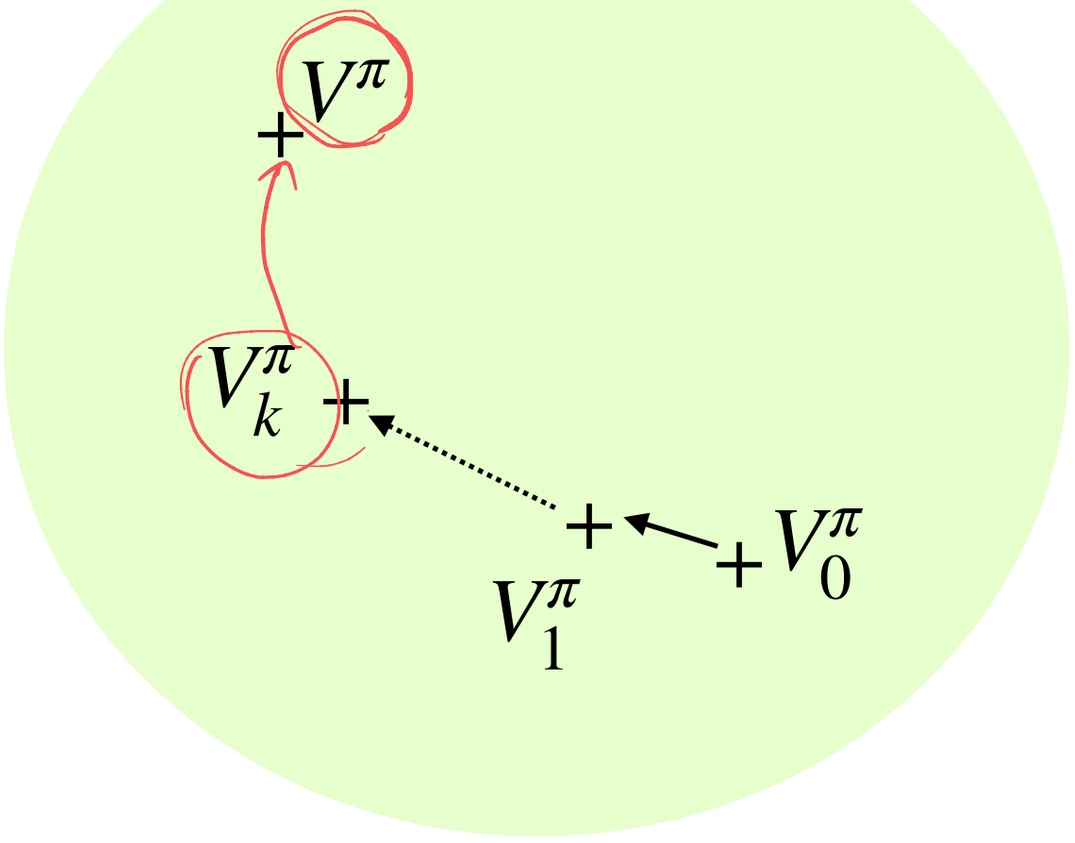
  1. Initialize $V^{\pi}_{0}(s) = 0$ for all s

  2. For $k = 1,2,\ldots$

$$V^{\pi}_{k}(s) = \sum_{a \in \mathcal{A}} \pi(a \mid s) \left(R_{s,a} + \gamma \sum_{s' \in \mathcal{S}} P^{a}_{ss'} V^{\pi}_{k-1}(s') \right) \quad \text{for all } s$$

- Question: What if we start from $V^{\pi}_{0}(s) = V^{\pi}(s), \forall s$?
- Question: In general, does $V^{\pi}_{k}(s)$ converge to the correct $V^{\pi}(s)$?

(Complete) Metric vector space $\mathbb{R}^{|\mathcal{S}|}$



- Question: What does IPE do to points in this space?

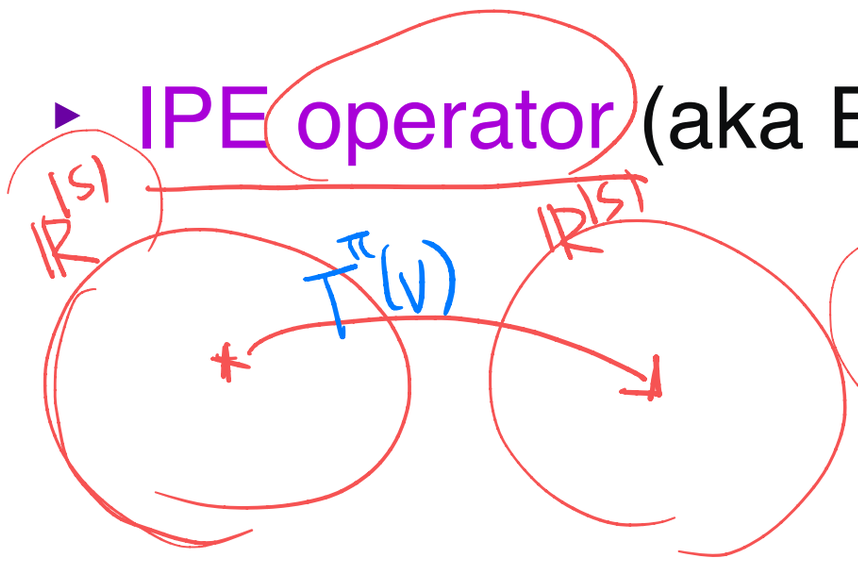
Prove convergence in 2 steps:

(A1): IPE brings points closer (formally, a "contraction" operator)

(A2): Under any contraction operator, the points converges to a unique fixed point

For (A1): IPE is a Contraction Map

- IPE operator (aka Bellman expectation backup operator):



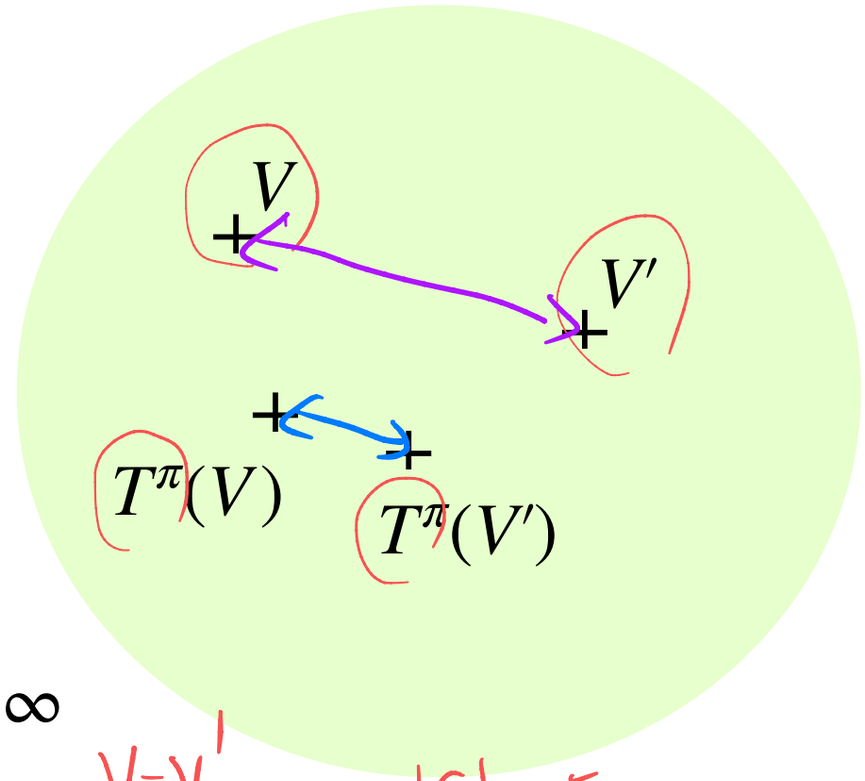
$$T^{\pi}(V) := R^{\pi} + \gamma P^{\pi} V$$

- Consider L_{∞}-norm to measure distance between any two value functions V, V'

$$||V - V'||_{\infty} := \max_{s \in \mathcal{S}} |V(s) - V'(s)|$$

For (A1): IPE is a Contraction Map (Cont.)

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



For any two value functions V and V',

$$||T^{\pi}(V) - T^{\pi}(V')||_{\infty}$$

$$= ||(R^{\pi} + \gamma P^{\pi} V) - (R^{\pi} + \gamma P^{\pi} V')||_{\infty}$$

$$= \gamma ||P^{\pi}(V - V')||_{\infty}$$

$$\leq \gamma ||(V - V')||_{\infty}$$

$P^{\pi}_{s^{(1)},\cdot}$ P^{π} $V - V'$ $V(s^{(1)}) - V'(s^{(1)})$ $V(s^{(|S|)}) - V'(s^{(|S|)})$ $|S| = 5$

$P^{\pi}_{s^{(1)},s^{(1)}}$

$\left| \left(P^{\pi}_{s^{(1)},\cdot} \right)^{T} (V - V') \right|$

$\leq \max_{s \in S} |V(s) - V'(s)|$

We say T^{π} is a γ-contraction operator ($\gamma < 1$)

For (A2): Banach Fixed-Point Theorem

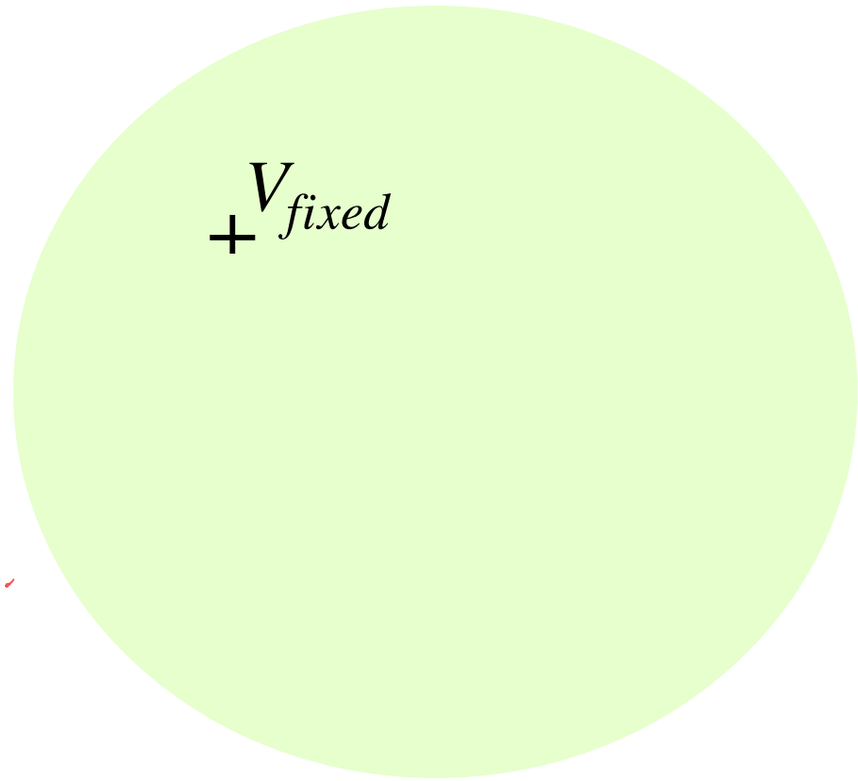
- **Banach Fixed-Point Theorem:** For any non-empty complete metric space, if T is a γ-contraction operator, then T has a unique fixed point.

$T(V) = V$

- Question: Why is this useful?

By Banach Fixed-Point Theorem,

true V^{π} is our "unique" fixed point under T^{π}.



(A_1): $\| T^\pi(V) - T^\pi(V') \|_\infty \leq \gamma \cdot \| V - V' \|_\infty$

(A_2): V^π is the unique fixed point (i.e., $T^\pi(V^\pi) = V^\pi$)

Show: $V_k^\pi \to V^\pi$

Set $V \equiv V_k^\pi$, $V' \equiv V^\pi$

$$\| \underbrace{T^\pi(V_k^\pi)}_{= V_{k+1}^\pi} - \underbrace{T^\pi(V^\pi)}_{= V^\pi \text{ by } (A_2)} \|_\infty \leq \gamma \cdot \| V_k^\pi - V^\pi \|_\infty \quad \cdots \text{ by } (A_1)$$

$$\Rightarrow \quad \| V_{k+1}^\pi - V^\pi \|_\infty \leq \gamma^{k+1} \cdot \| V_0^\pi - V^\pi \|_\infty$$

Quick Summary

- Under IPE, the value functions V_k^π converges to the correct V_k^π, for any initialization V_0^π

Contraction property is central to various RL algorithms:

A Theory of Regularized Markov Decision Processes

Matthieu Geist [1] Bruno Scherrer [2] Olivier Pietquin [1]

Abstract

Many recent successful (deep) reinforcement learning algorithms make use of regularization, generally based on entropy or Kullback-Leibler divergence. We propose a general theory of regularized Markov Decision Processes that generalizes these approaches in two directions: we consider a larger class of regularizers, and we consider the general modified policy iteration approach, encompassing both policy iteration and value iteration. The core building blocks of this theory are a notion of regularized Bellman operator and the Legendre-Fenchel transform, a classical tool of convex optimization. This approach allows for error propagation analyses of general algorithmic schemes of which (possibly variants of) classical algorithms such as Trust Region Policy Optimization, Soft Q-learning, Stochastic Actor Critic or Dynamic Policy Programming are special cases. This also draws connections to proximal convex optimization, especially to Mirror Descent.

Tsallis entropy (Lee et al., 2018 having a sparse regularized greedy are based on a notion of tempo somehow extending the notion o regularized case (Nachum et al. Nachum et al., 2018), or on policy Mnih et al., 2016).

This non-exhaustive set of algori ing regularization, but they are different principles, consider eac ization, and have ad-hoc analysis, a general theory of regularized M (MDPs). To do so, a key observa dynamic programming, or (A)D from the core definition of the B tor. The framework we propose i Bellman operator, and on an assc transform. We study the theoretic larized MDPs and of the related r This generalizes many existing th vides new ones. Notably, it allow

(Geist et al., ICML 2019)

A Generalized Algorithm for Multi-Objective Reinforcement Learning and Policy Adaptation

Runzhe Yang
Department of Computer Science
Princeton University
runzhey@cs.princeton.edu

Xingyuan Sun
Department of Computer Science
Princeton University
xs5@cs.princeton.edu

Karthik Narasimhan
Department of Computer Science
Princeton University
karthikn@cs.princeton.edu

Abstract

We introduce a new algorithm for multi-objective reinforcement learning (MORL) with linear preferences, with the goal of enabling few-shot adaptation to new tasks. In MORL, the aim is to learn policies over multiple competing objectives whose relative importance (*preferences*) is unknown to the agent. While this alleviates dependence on scalar reward design, the expected return of a policy can change significantly with varying preferences, making it challenging to learn a single model to produce optimal policies under different preference conditions. We propose a generalized version of the Bellman equation to learn a single parametric representation for optimal policies over the space of all possible preferences. After an initial learning phase, our agent can execute the optimal policy under any given preference, or automatically infer an underlying preference with very few samples. Experiments across four different domains demonstrate the effectiveness of our approach.[1]

(Yang et al., NeurIPS 2019)

Now we know how to evaluate a given policy $\pi(a \mid s)$

How about an optimal policy?

Optimal Value Functions

- **Optimal state-value function**:

$$V^*(s) = \max_{\pi \in \Pi} V^\pi(s)$$

↖ Markov stationary policies

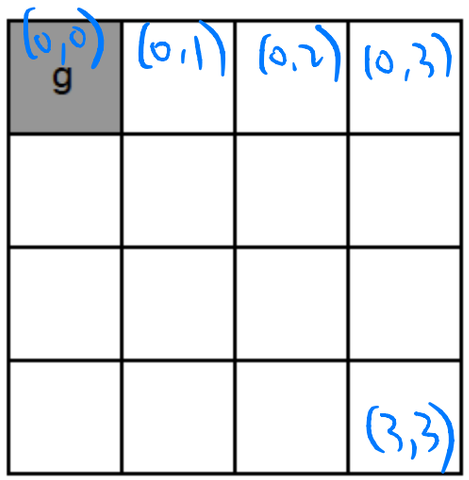
- **Optimal action-value function**:

$$Q^*(s,a) = \max_{\pi \in \Pi} Q^\pi(s,a)$$

- Question: Suppose we know $Q^*(s,a)$ for all pairs of (s,a). What's your guess about the "optimal" policy (though we haven't defined it yet)?

$$\pi^*(s) := \operatorname*{argmax}_{a \in A} Q^*(s,a)$$

Example: Shortest Path



Problem

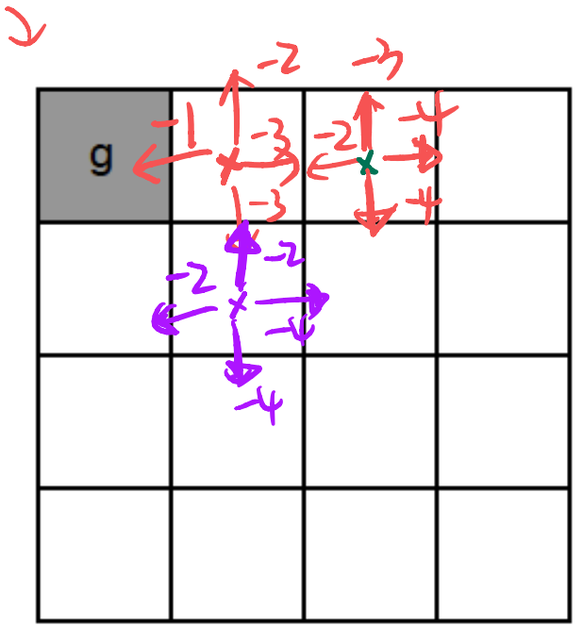
- **Actions**: Left / Right / Up / Down
- The learner stays at the same place after hitting a wall
- The episode ends after reaching the goal (denoted as g)
- **Reward**: $R(s, a) = -1$, for all (s, a)
- In this example, $V^*(s)$ captures the shortest distance from s to g

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

$V^*(s)$

- Question: Can you find $Q^*(s, a)$?
- Question: Suppose we know $Q_*(s, a)$ for all pairs of (s, a). What's your guess about the "optimal" policy (though we haven't defined it yet)?

Check: $\pi^*(s) = \arg\max_{a \in A} Q^*(s, a)$



$Q^*(s, a)$

Definition of an "Optimal" Policy (Formally)

- **Partial ordering of policies**:

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

- **An optimal policy**: A policy π^* is an optimal policy if it is better than or equal to all other policies, i.e.

$$\pi^* \geq \pi, \text{for all } \pi \in \Pi$$

- Question: Given an MDP, does such π^* always exist? Yes!

Existence of an Optimal Policy

- **Theorem (Existence of Optimal Policy)**: For any finite MDP, there always exists an optimal policy π^* such that

$$\pi^* \geq \pi, \text{ for all } \pi \in \Pi$$

(This will be shown shortly) [Puterman, 1994]

- A deterministic optimal policy can always be found by

$$\pi^*(a \mid s) = \begin{cases} 1, & \text{if } a = \arg\max_{a' \in \mathcal{A}} Q^*(s, a') \\ 0, & \text{else} \end{cases}$$

(To be proved shortly)

- Therefore, knowing Q^* is equivalent to knowing π^*
- Question: Number of deterministic policies?

M. Puterman, “Markov Decision Processes: Discrete Stochastic Dynamic Programming," 1994

How to find $Q^*(s, a)$ (and thereafter π^*)?

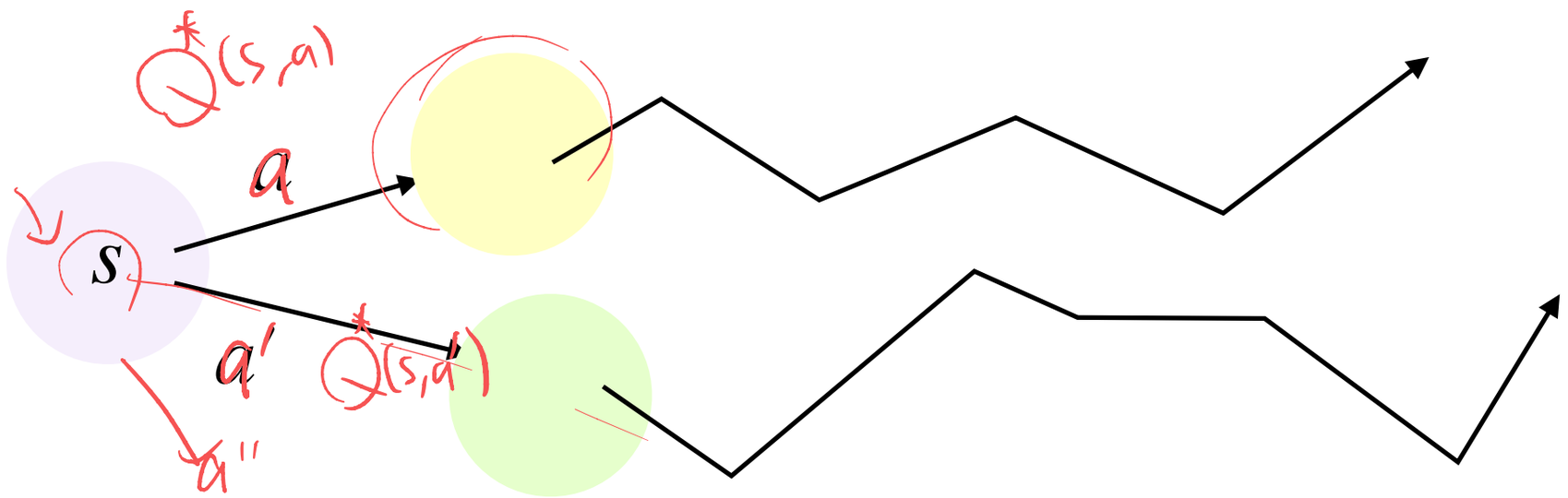
Bellman Optimality Equations

(1) V^* written in Q^*

$$V^*(s) = \max_a Q^*(s,a)$$

(2) Q^* written in V^*

$$Q^*(s,a) = R_s^a + \gamma \sum_{s'} P_{ss'}^a V^*(s')$$



(Proof: HW1 problems)

Bellman Optimality Equations (Cont.)

(1) V^* written in Q^*

$$V^*(s) = \max_a Q^*(s,a)$$

(2) Q^* written in V^*

$$Q^*(s,a) = R_s^a + \gamma \sum_{s'} P_{ss'}^a V^*(s')$$

(3) V^* written in V^*

$$V^*(s) = \max_a R_s^a + \gamma \sum_{s'} P_{ss'}^a V^*(s')$$

(4) Q^* written in Q^*

$$Q^*(s,a) = R_s^a + \gamma \sum_{s'} P_{ss'}^a \left(\max_{a'} Q^*(s',a') \right)$$

How to Solve the Bellman Optimality Equation?

$$V^*(s) = \max_a \left(R_s^a + \gamma \sum_{s'} P_{ss'}^a V^*(s') \right)$$

- Question: Solve this by linear algebra?

- The “max” operation makes it non-linear

- We need to resort to iterative methods:
  - Policy iteration
  - Value iteration

Value Iteration (VI)

From Bellman Optimality Equation to Bellman Optimality Backup Operator

- Recall: Bellman optimality equation

$$V^*(s) = \max_a \left(R_s^a + \gamma \sum_{s'} P_{ss'}^a V^*(s') \right)$$

- Define: Bellman optimality backup operator

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

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

Value Iteration: Pseudo Code

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

2. Repeat the following until convergence:

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

   - Equivalently: for each state s

$$V_{k+1}(s) = \max_a \left(R_s^a + \gamma \sum_{s'} P_{ss'}^a V_k(s') \right)$$

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

Example: Shortest Path

$$V_{k+1}(s) = \max_a \left(R_s^a + \gamma \sum_{s'} P_{ss'}^a V_k(s') \right)$$

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

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

V_3

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

V_4

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

V_5

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

V_6

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

V_7

Convergence of Value Iteration

- **Value iteration converges on V^*:** For any initial V_0, Value Iteration achieves that $V_k \to V^*$ (in L_∞-norm)

- Question: How to show this?

Convergence of Value Iteration

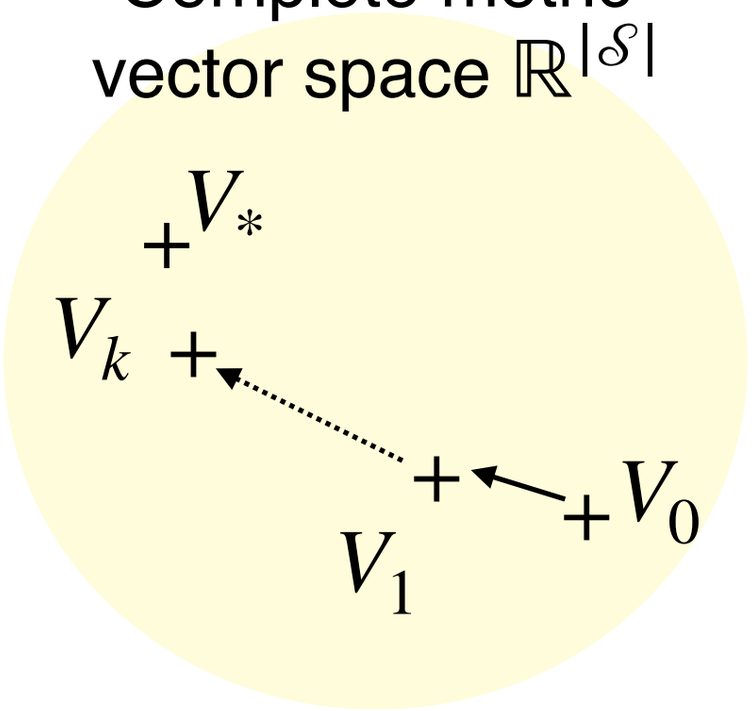
- **Value iteration converges on V^*:** For any initial V_0, Value Iteration achieves that $V_k \to V^*$ (in L_∞-norm)

- Proof: We prove convergence via the following 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



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

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

$$
\begin{aligned}
&||T^*(V) - T^*(\hat{V})||_\infty \\
&\quad = \max_s \left| T^*(V)(s) - T^*(\hat{V})(s) \right| \\
&\quad = \max_s \left| \max_a \left(R_s^a + \gamma \sum_{s'} P_{ss'}^a V(s') \right) - \max_{a'} \left(R_s^{a'} + \gamma \sum_{s'} P_{ss'}^{a'} \hat{V}(s') \right) \right| \\
&\quad \leq \max_s \max_a \left| \left(R_s^a + \gamma \sum_{s'} P_{ss'}^a V(s') \right) - \left(R_s^a + \gamma \sum_{s'} P_{ss'}^a \hat{V}(s') \right) \right| \\
&\quad \leq \max_s \max_a \left| \gamma \sum_{s'} P_{ss'}^a \left(V(s') - \hat{V}(s') \right) \right| \\
&\quad \leq \gamma ||(V - \hat{V})||_\infty
\end{aligned}
$$

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

For (B2)

- T^* is a γ-contraction operator in a complete metric space
- By Banach Fixed Point Theorem, T^* converges to the unique fixed point

Discussion: Issues With Value Iteration

- Question 1: In how many iterations will VI converge?

- Question 2: By applying VI, could we find an optimal policy?

- Question 3: By using VI, could we directly prove the existence of an optimal policy?

Some Historical Accounts

A seminal paper on “Value Iteration” (published in 1957)

A Markovian Decision Process

RICHARD BELLMAN

1. Introduction. The purpose of this paper is to discuss the asymptotic behavior of the sequence $\{f_N(i)\}$, $i = 1, 2, \cdots, M$, $N = 1, 2, \cdots$, generated by the non-linear recurrence relations

$$f_N(i) = \max_{\mathbf{q}} \left[b_i(\mathbf{q}) + \sum_{j=1}^{M} a_{ij}(\mathbf{q}) f_{N-1}(j) \right], \qquad N = 1, 2, \cdots, \tag{1}$$

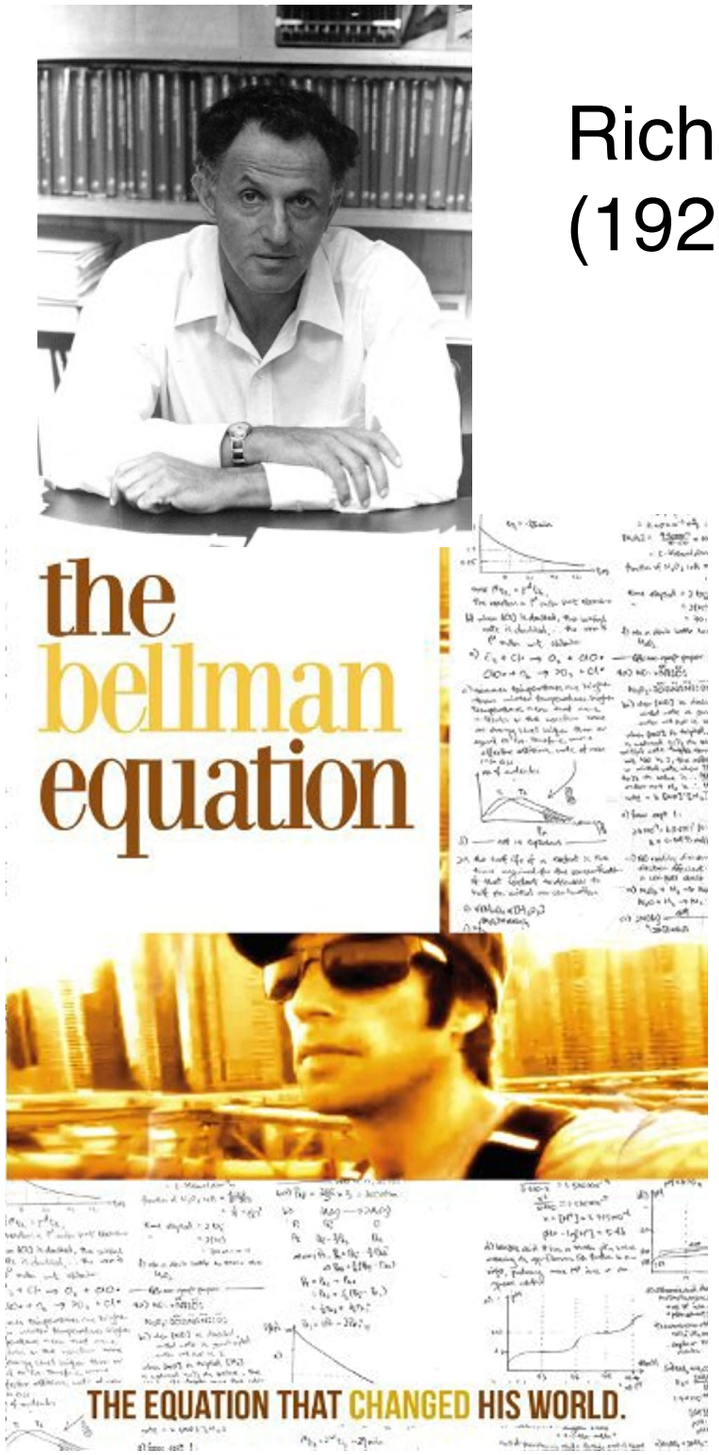
$$f_0(i) = c_i, \qquad i = 1, 2, \cdots, M.$$

Although these equations are non-linear, they possess certain quasi-linear properties which permit a more thorough discussion than might be imagined upon first glance.

As we shall discuss below, this question arises from the consideration of a dynamic programming process. A related process gave rise to an equation of the above form which was discussed in [1], under a particular set of assumptions concerning the functions $b_i(\mathbf{q})$ and the matrices $A(\mathbf{q}) = (a_{ij}(\mathbf{q}))$. Here we shall impose restrictions of a quite different type. Any complete discussion of relations of the foregoing type is at least as detailed as a corresponding discussion of the linear case, and, as in the linear case, the assumptions made determine the techniques employed to a considerable extent.

We shall discuss elsewhere some interesting quadratically non-linear recurrence relations which arise from specializing the forms of $b_i(\mathbf{q})$ and $a_{ij}(\mathbf{q})$. These are related to the types of differential equations discussed in [3], being essentially a particular type of discrete version.

It will be clear from what follows that similar techniques can be utilized to treat the determination of the asymptotic behavior of the sequences defined by

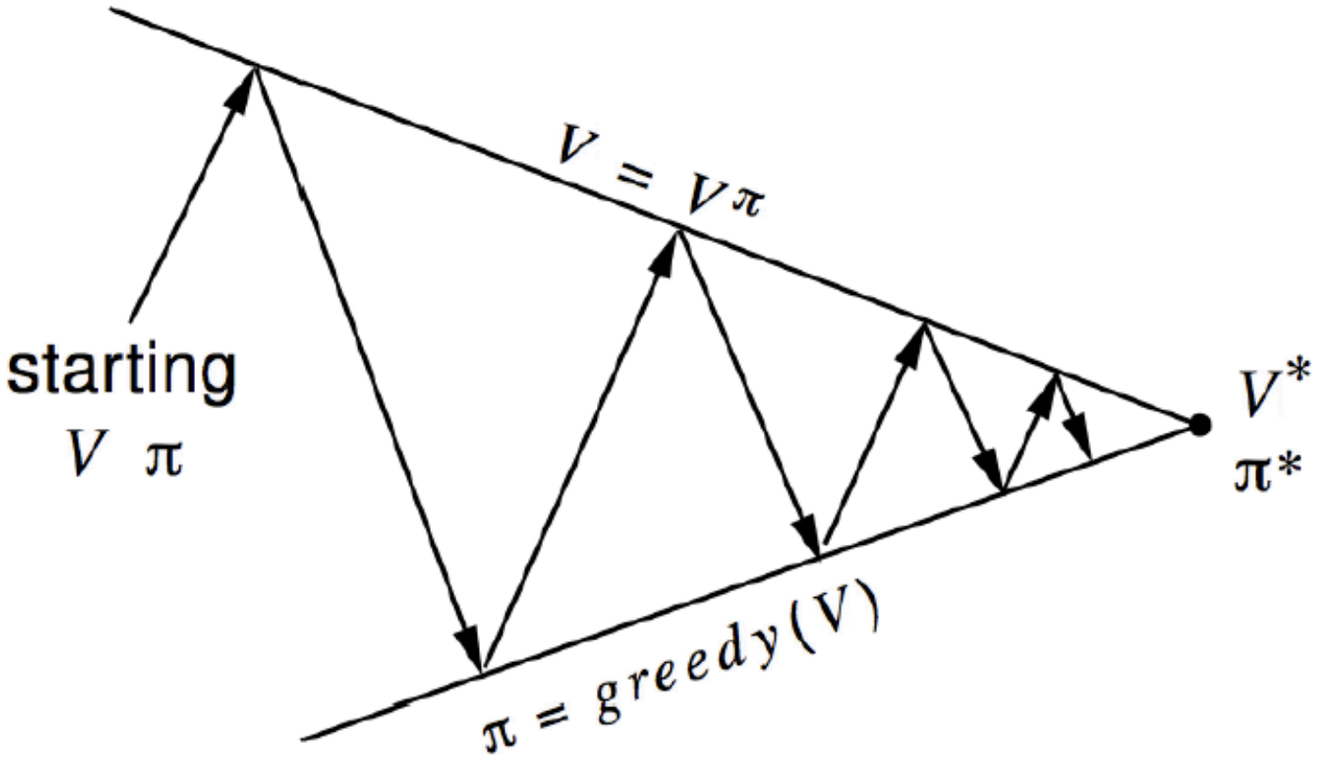


Richard Bellman (1920-1984)

A documentary released in 2011

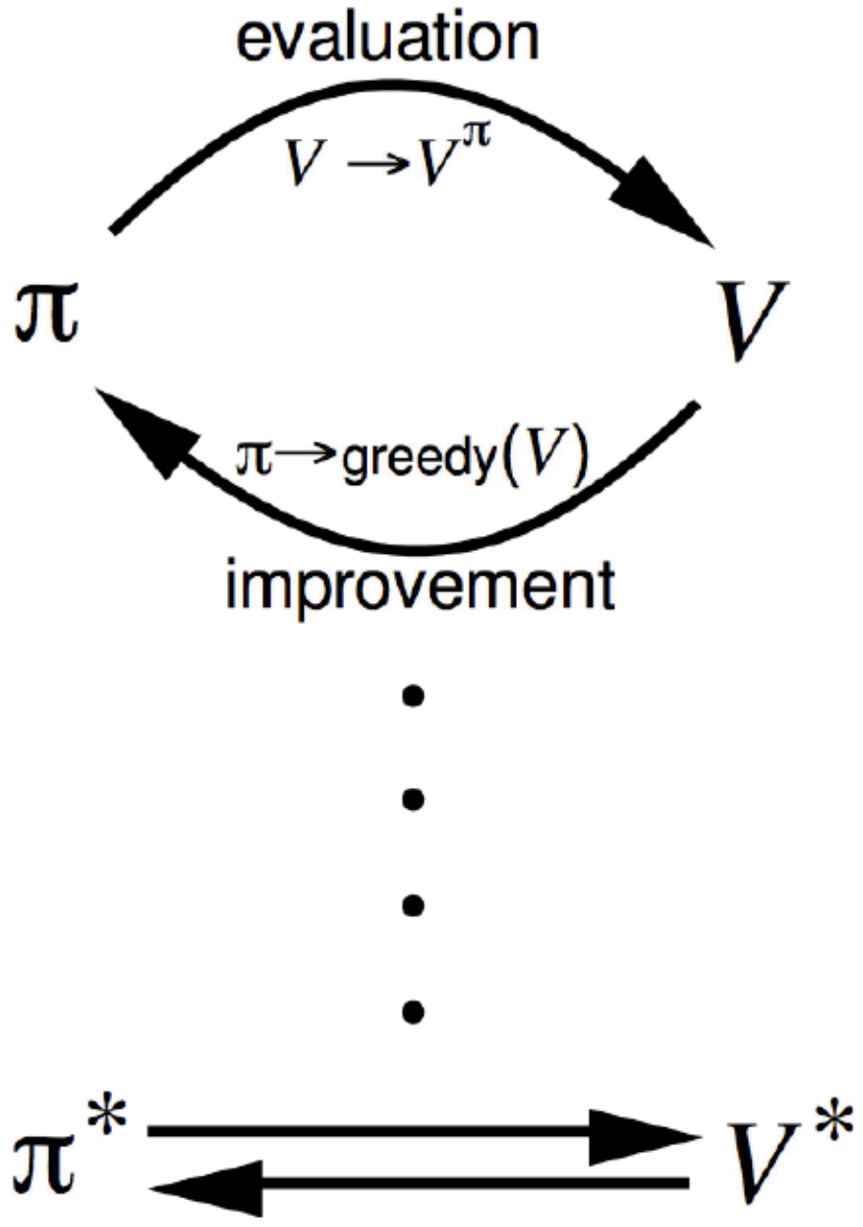
Policy Iteration (PI)

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)

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

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 $Q^{\pi_k}(s, a)$:

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

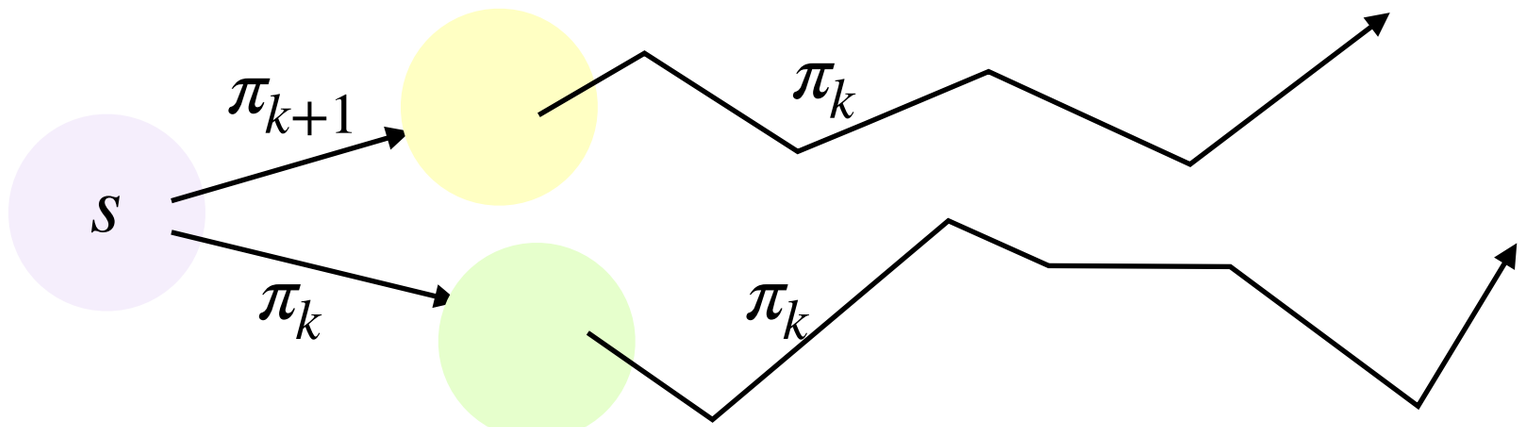
- Derive the new policy π_{k+1}: For all states s,

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

- 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?



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

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

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

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

Monotonic Improvement in Policy

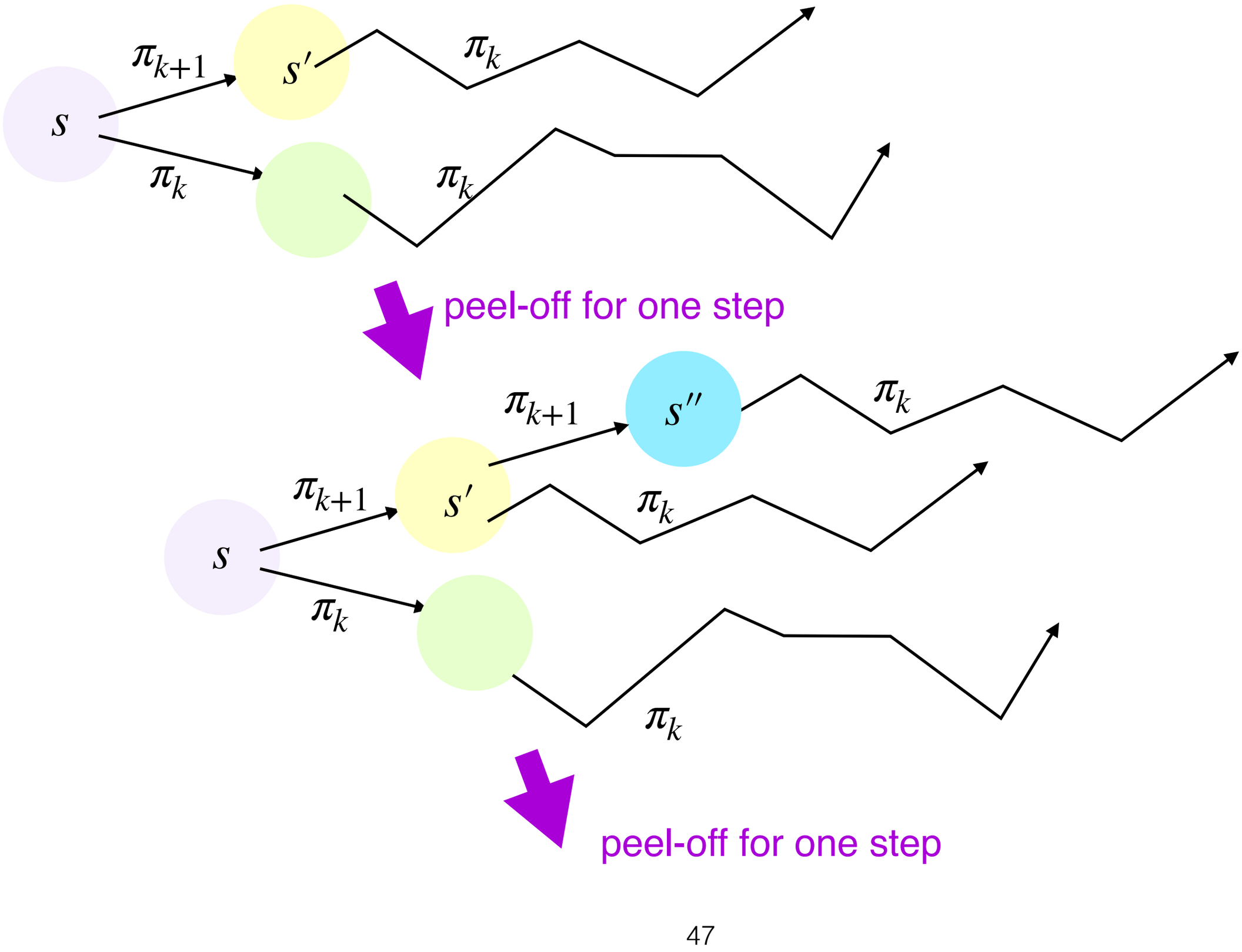
- Recall: Partial ordering of policies

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

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

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

Proof Idea: “Peeling off”



Proof: Monotonic Policy Improvement

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

Discussions: Policy Iteration

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

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

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

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

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+VI prove the existence of an optimal policy

Discussions: Policy Iteration (Cont.)

- Question: Will policy iteration terminate in finite number of iterations?

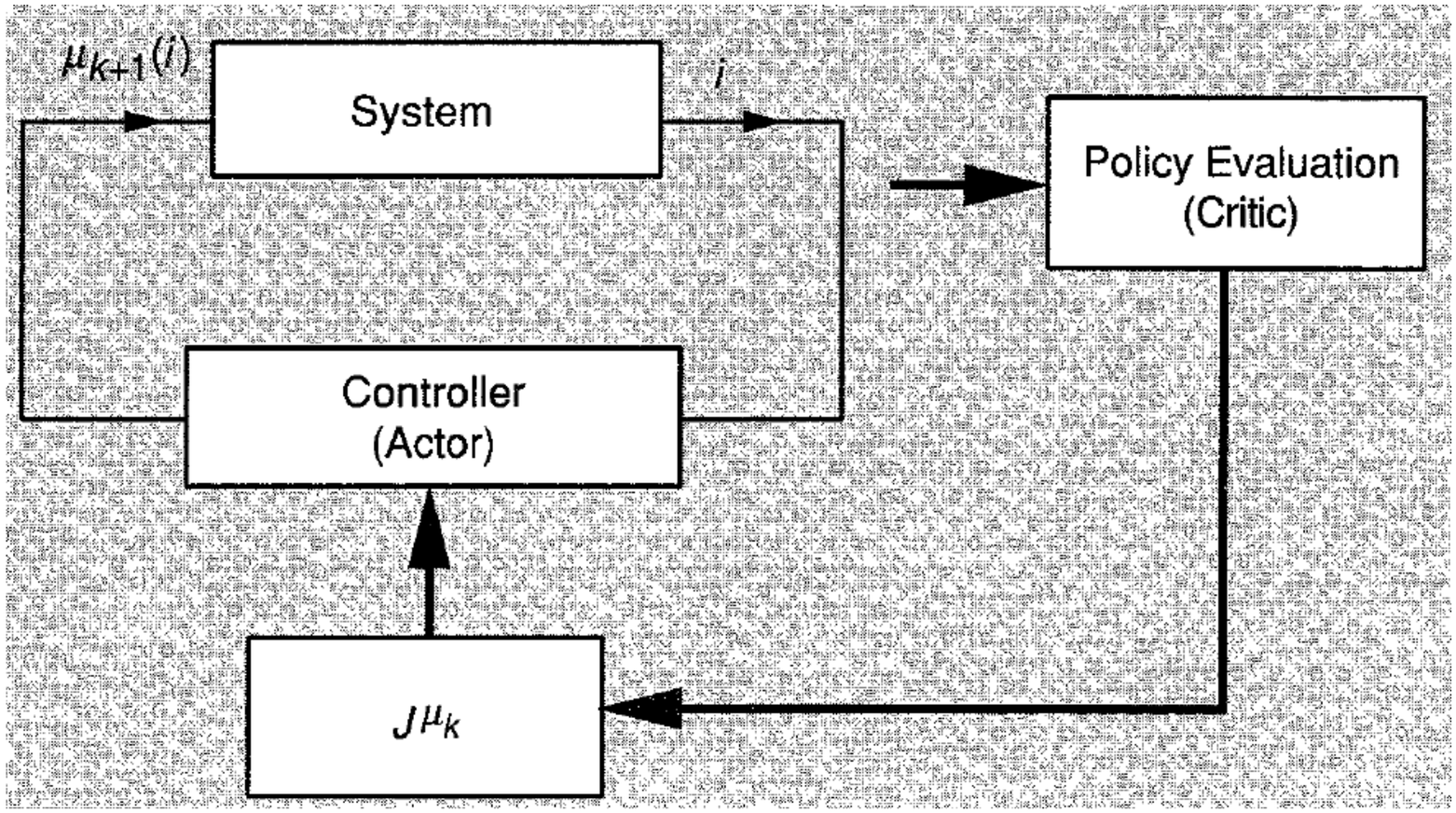
- Yes, in at most $|\mathcal{A}|^{|\mathcal{S}|}$ iterations (assume $|\mathcal{A}|, |\mathcal{S}|$ are finite)

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

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

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

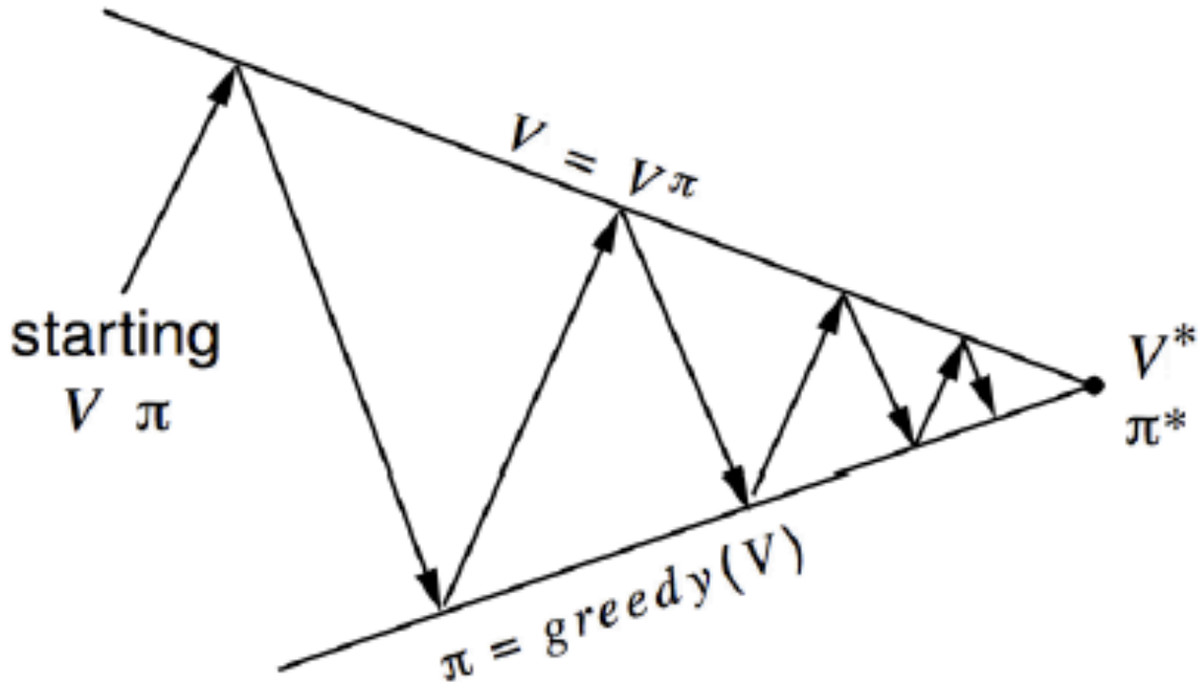
Policy Iteration: Actor-Critic Viewpoint

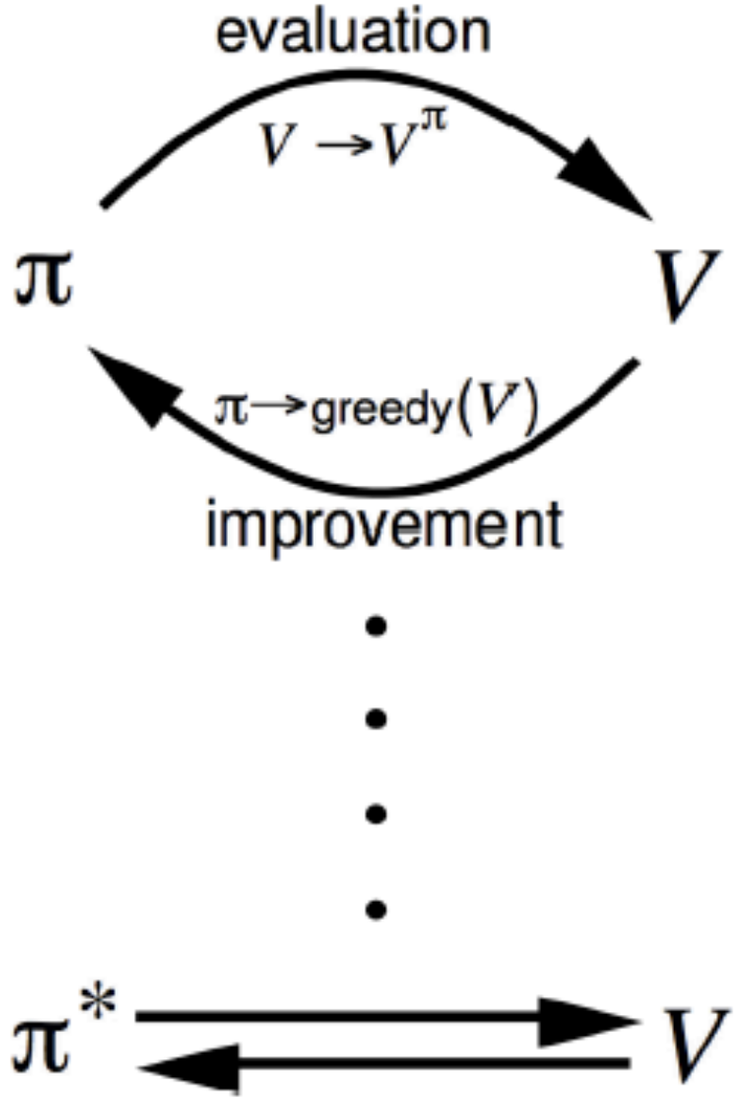


- Critic: evaluate the performance of the current policy
- Actor: take into account the latest evaluation and output an improved policy

(Figure Source: Bertsekas and Tsitsiklis, Section 2.2)

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)