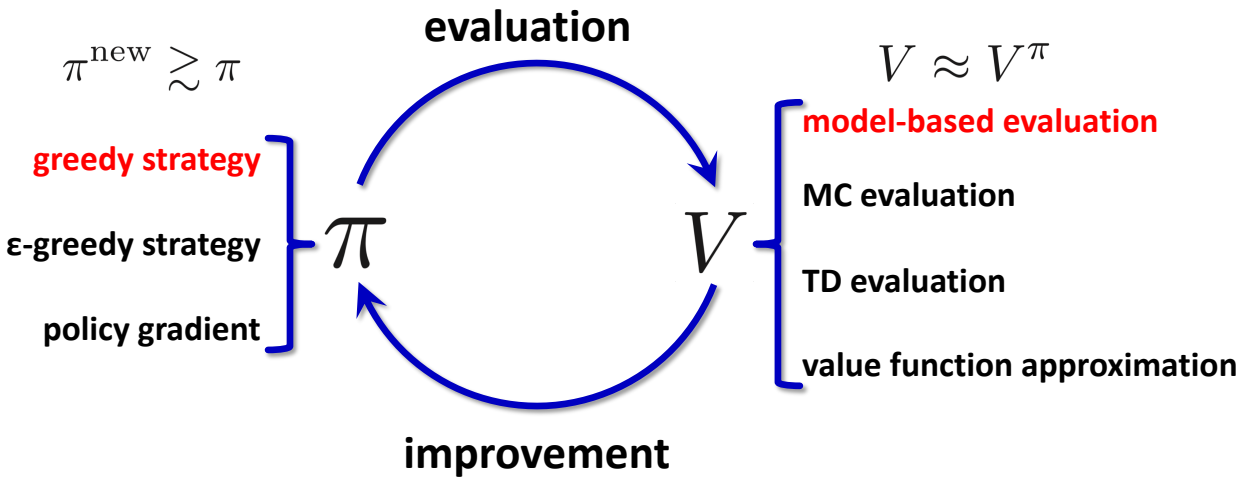
Algorithmic and Theoretical Foundations of RL

Value Iteration and Policy Iteration

Ke Wei
School of Data Science
Fudan University

General Framework



Overall, different RL algorithms can be viewed as implementing the idea of alternative update of value and policy in different ways. This lecture first presents the idea in the model based setting.

Recap: Bellman Operator and Bellman Optimality Operator

Bellman Operator

Elementwise form: $[\mathcal{T}^\pi V](s) = \mathbb{E}_{a\sim\pi(\cdot|s)}\mathbb{E}_{s'\sim P(\cdot|s,a)}\left[r(s,a,s') + \gamma V(s')\right]$

Matrix form: $\mathcal{T}^\pi V = r^\pi + \gamma P^\pi V$

- $\mathcal{T}^\pi$ is a contraction and V^π a fixed point of $\mathcal{T}^\pi$: $\mathcal{T}^\pi V^\pi = V^\pi$.

Bellman Optimality Operator

Elementwise form: $[\mathcal{T}V](s) = \max_a \mathbb{E}_{s'\sim P(\cdot|s,a)}\left[r(s,a,s') + \gamma V\left(s'\right)\right]$

Matrix form: $\mathcal{T}V = \max_\pi \mathcal{T}^\pi V = \max_\pi \{r^\pi + \gamma P^\pi V\}$

- $\mathcal{T}$ is a contraction and V^* a fixed point of $\mathcal{T}$: $\mathcal{T}V^* = V^*$.

Table of Contents

Value Iteration

Value Iteration (VI): Solve Bellman optimality equation by fixed point iteration,

$$V^{k+1}(s) = \max_a \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[r(s,a,s') + \gamma V^k(s') \right].$$

- To retrieve a policy after value iteration:

$$\pi_{k+1}(a|s) = \begin{cases} 1 & \arg\max_a \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[r(s,a,s') + \gamma V^k(s') \right], \\ 0 & \text{otherwise.} \end{cases}$$

Question: Note that $V^{k+1} = \mathcal{T}^{\pi_{k+1}} V^k = \mathcal{T} V^k$, but whether $V^{\pi_{k+1}} = V^{k+1}$?

Convergence of Value Iteration

Theorem 1

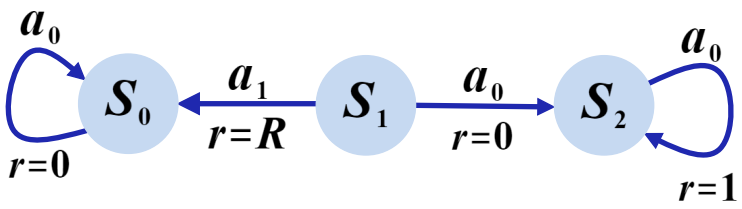
Let $\{V^k\}$ be the sequence of value functions produced by VI. Then for any $k \geq 1$,

$$\|V^k - V^*\|_\infty \leq \gamma^k \|V^0 - V^*\|_\infty,$$

which implies that $\lim_{k \to \infty} V^k = V^*$.

- The per iteration computational cost of value iteration is $O\left(|\mathcal{S}|^2 |\mathcal{A}|\right)$.
- After at most $k = O\left(\frac{\log(1/\varepsilon)}{\log(1/\gamma)}\right)$ iterations, one has $\|V^k - V^*\|_\infty \leq \varepsilon$.

Illustrative Example



- three states: $\mathcal{S} = \{s_0, s_1, s_2\}$
- two actions: $\mathcal{A} = \{a_0, a_1\}$

Each edge is associated with a deterministic transition and a reward.

Suppose we start from $V^0 = 0$. Then

$$V^k(s_0) = r(s_0, a_0, s_0) + \gamma V^{k-1}(s_0) = \gamma V^{k-1}(s_0) = \gamma^k V^0(s_0) = 0,$$

$$V^k(s_2) = r(s_2, a_0, s_2) + \gamma V^{k-1}(s_2) = 1 + \gamma V^{k-1}(s_2) = \frac{1-\gamma^k}{1-\gamma} + \gamma^k V^0(s_2) = \frac{1-\gamma^k}{1-\gamma},$$

$$\begin{aligned} V^k(s_1) &= \max\left\{r(s_1, a_0, s_2) + \gamma V^{k-1}(s_2), r(s_1, a_1, s_0) + \gamma V^{k-1}(s_0)\right\} \\ &= \max\left\{\frac{\gamma}{1-\gamma}\left(1-\gamma^{k-1}\right), R\right\}. \end{aligned}$$

Thus (assuming $R < \frac{\gamma}{1-\gamma}$),

$$V^*(s_0) = 0, \quad V^*(s_1) = \frac{\gamma}{1-\gamma}, \quad V^*(s_2) = \frac{1}{1-\gamma}.$$

Asynchronous Value Iteration

State values in VI are updated synchronously. An alternative is **asynchronous value iteration**: Rather than sweeping through all states to create a new value vector, only updates one state (an entry of vector) at a time.

Gauss-Seidel Value Iteration:

$$\text{for } s = 1, 2, 3, \ldots$$
$$V(s) \leftarrow \max_a \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[r\left(s, a, s'\right) + \gamma V\left(s'\right) \right]$$

Table of Contents

Policy Iteration

$$\pi_0 \xrightarrow{\text{E}} V^{\pi_0} \xrightarrow{\text{I}} \pi_1 \xrightarrow{\text{E}} V^{\pi_1} \xrightarrow{\text{I}} \pi_2 \xrightarrow{\text{E}} \cdots \xrightarrow{\text{I}} \pi^*$$

There are two ingredients in **Policy Iteration (PI)**.

Policy Evaluation:

$$V^{\pi_k} = r^{\pi_k} + \gamma P^{\pi_k} V^{\pi_k}.$$

Policy Improvement:

$$\pi_{k+1}(a|s) = \begin{cases} 1 & a = \arg\max_a \underbrace{\mathbb{E}_{s' \sim P(\cdot|s,a)} \left[r(s,a,s') + \gamma V^{\pi_k}(s') \right]}_{Q^{\pi_k}(s,a)}, \\ 0 & \text{otherwise.} \end{cases}$$

- It is clear that

$$\mathcal{T}^{\pi_{k+1}} V^{\pi_k} = \mathcal{T} V^{\pi_k} \quad \text{and} \quad V^{\pi_{k+1}} = \mathcal{T}^{\pi_{k+1}} V^{\pi_{k+1}} = \lim_{m \to \infty} (\mathcal{T}^{\pi_{k+1}})^m V^{\pi_k}.$$

  In addition, by policy improvement lemma in Lecture 1, one has $V^{\pi_{k+1}} \geq V^{\pi_k}$.

Convergence of Policy Iteration

Theorem 2

Let $\{\pi_k\}$ be the policy sequence produced by PI. Then for any $k \geq 1$,

$$\|V^{\pi_k} - V^*\|_\infty \leq \gamma^k \|V^{\pi_0} - V^*\|_\infty,$$

which implies that $\lim_{k \to \infty} V^{\pi_k} = V^*$.

- The per iteration computational cost of policy iteration is $O\left(|\mathcal{S}|^3\right)$ to evaluate V^{π_k} plus $O\left(|\mathcal{S}|^2|\mathcal{A}|\right)$ to produce a new policy.
- After at most $k = O\left(\frac{\log(1/\varepsilon)}{\log(1/\gamma)}\right)$ iterations, one has $\|V^{\pi_k} - V^*\|_\infty \leq \varepsilon$.

Proof of Theorem 2

First note that

$$V^{\pi_k} = \mathcal{T}^{\pi_k} V^{\pi_k} \geq \mathcal{T}^{\pi_k} V^{\pi_{k-1}} = \mathcal{T} V^{\pi_{k-1}}.$$

Iterating this procedure yields that

$$V^{\pi_k} \geq \mathcal{T} V^{\pi_{k-1}} \geq \cdots \geq \mathcal{T} V^{\pi_0}.$$

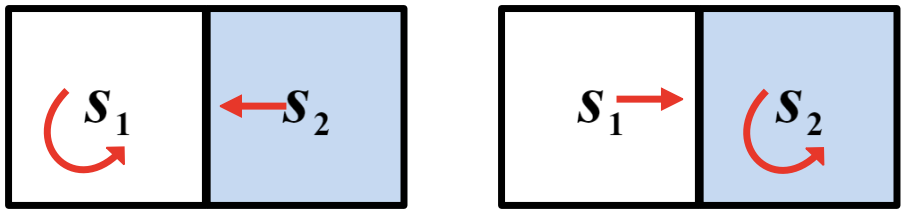
Therefore,

$$V^* - V^{\pi_k} \leq V^* - \mathcal{T}^k V^{\pi_0} = \mathcal{T}^k (V^* - V^{\pi_0}).$$

The assertion follows immediately by taking infinite norm on both sides.

Illustrative Example

Consider the example in following figure, where each state is associated with three possible actions: a_l, a_0, a_r (move leftwards, stay unchanged, and move rightwards). The reward is r_{s_1} = -1 and r_{s_2} = 1. The discount rate is γ = 0.9.



(a) (b)

Assume the initial policy π_0 is given in (a). This policy satisfies $\pi_0(a_0|s_1) = 1$ and $\pi_0(a_l|s_2) = 1$. This policy is not good because it does not move toward s_2. We next apply policy iteration problem.

"Mathematical Foundation of Reinforcement Learning" by Shiyu Zhao, 2022.

Illustrative Example

- Policy Evaluation

$$\begin{cases} V^{\pi_0}(s_1) = -1 + \gamma V^{\pi_0}(s_1) \\ V^{\pi_0}(s_2) = -1 + \gamma V^{\pi_0}(s_1) \end{cases} \Rightarrow \begin{cases} V^{\pi_0}(s_1) = -10 \\ V^{\pi_0}(s_2) = -10 \end{cases}$$

- Policy Improvement

$Q^{\pi_0}(s,a)$	a_ℓ	a_0	a_r
s_1	$-$	-10	-8
s_2	-10	-8	$-$

Since π_1 choose the action that maximize $Q^{\pi_0}(s,a)$, one has (see (b)):

$$\pi_1(a_r|s_1) = 1, \quad \pi_1(a_0|s_2) = 1.$$

It is evident that this is an optimal policy.

Remark

- In both VI an PI, all the state values are updated $\rightarrow$ full exploration.
- Current state values are used as backup for exploitation.
- VI and PI in terms of action values can also be similarly developed which are the versions that will be used in model free RL methods.

Policy Iteration as Newton's Method

For a nonlinear equation $F(V) = 0$, Newton's method solves a first order approximation at each iteration:

$$F(V) \approx F(V^k) + J^k(V - V^k) = 0,$$

which yields

$$V^{k+1} = V^k - (J^k)^{-1} F(V^k).$$

Here, J^k is the Jacobian of F at V^k.

Policy Iteration as Newton's Method

For the Bellman optimality equation, one has

$$F(V) = \max_{\pi}\{r^{\pi} + \gamma P^{\pi} V\} - V.$$

Given the current state value V^{π_k} associated with policy π_k, the Jacobian of F at V^{π_k} is given by $\gamma P^{\pi_{k+1}} - I$. The Newton's update gives

$$\begin{aligned}
&V^{\pi_k} - (\gamma P^{\pi_{k+1}} - I)^{-1}(r^{\pi_{k+1}} + \gamma P^{\pi_{k+1}} V^{\pi_k} - V^{\pi_k}) \\
&= V^{\pi_k} - (\gamma P^{\pi_{k+1}} - I)^{-1} r^{\pi_{k+1}} - V^{\pi_k} \\
&= (I - \gamma P^{\pi_{k+1}})^{-1} r^{\pi_{k+1}} \\
&= V^{\pi_{k+1}}.
\end{aligned}$$

Thus, PI can be viewed as Newton's method for nonlinear equation $F(V) = 0$.

Variants of PI: Truncated PI

Truncated policy iteration (TPI) is the same as PI except that it merely runs a finite number of iterations in the policy evaluation.

Truncated Policy Evaluation: Set $V^{k,0} = V^{k-1}$ and estimate V^{π_k} via

$$V^{k,j} = r^{\pi_k} + \gamma P^{\pi_k} V^{k,j-1},$$

where $1 \leq j \leq m_k$. Set $V^k = V^{k,m_k}$, or equivalently, $V^k = (\mathcal{T}^{\pi_k})^{m_k} V^{k-1}$.

Policy Improvement:

$$\pi_{k+1}(a|s) = \begin{cases} 1 & a = \arg\max_a \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[r(s,a,s') + \gamma V^k(s') \right], \\ 0 & \text{otherwise.} \end{cases}$$

- Letting $m_k = \infty$, then $V^k = \lim_{m_k \to \infty} (\mathcal{T}^{\pi_k})^{m_k} V^{k-1} = V^{\pi_k}$ and TPI is exactly PI. On the other hand, letting $m_k = 1$, then $V^k = \mathcal{T}^{\pi_k} V^{k-1}$ and TPI is exactly VI.

Variants of PI: Approximate PI

Approximate Policy Iteration (API) is an even more general framework than TPI, where V^{π_k} is evaluated approximately and π_{k+1} is obtained by approximate policy improvement.

Approximate Policy Evaluation: Given π_k, estimate V^{π_k} by V^k that satisfies

$$\|V^k - V^{\pi_k}\|_\infty \leq \delta.$$

Approximate Policy Improvement: Produce a policy π_{k+1} that satisfies

$$\|r^{\pi_{k+1}} + \gamma P^{\pi_{k+1}} V^k - \mathcal{T} V^k\|_\infty \leq \varepsilon.$$

Table of Contents

ε-Optimal Policy and Error Amplification

Definition 1 (ε-optimal policy)
A policy π is called ε-optimal policy if

$$V^{\pi} \geq V^{*} - \varepsilon \mathbf{1}.$$

Theorem 3 (Error amplification)

For any vector $V \in \mathbb{R}^{|\mathcal{S}|}$, let π be the greedy policy with respect to V, i.e,

$$\pi(a|s) = \begin{cases} 1 & a = \arg\max_{a} \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[r(s,a,s') + \gamma V(s') \right] \\ 0 & \text{otherwise.} \end{cases}$$

Then $V^{\pi} \geq V^{*} - \frac{2\gamma}{1-\gamma} \left\| V - V^{*} \right\|_{\infty} \mathbf{1}$.

A Useful Lemma

Lemma 1

For any policy π and a vector $V \in \mathbb{R}^{|\mathcal{S}|}$, there holds

$$V^\pi \geq V - \frac{1}{1-\gamma} \max_s \{V(s) - \mathcal{T}^\pi V(s)\} \mathbf{1}.$$

- Note that $V^\pi = \mathcal{T}^\pi V^\pi$. Thus, if V is a vector such that $V \approx \mathcal{T}^\pi V$, it should hold $V \approx V^\pi$. Lemma 1 validates this intuition by providing an entrywise result akin to Bellman error (a notation that will be discussed later in this course).

Proof of Lemma 1

First consider $(\mathcal{T}^\pi)^k V$,

$$
\begin{aligned}
(\mathcal{T}^\pi)^k V &= (\mathcal{T}^\pi)^{k-1} \mathcal{T}^\pi V \geq (\mathcal{T}^\pi)^{k-1} (V - \max_s \{V(s) - \mathcal{T}^\pi V(s)\} \mathbf{1}) \\
&\geq (\mathcal{T}^\pi)^{k-1} V - \gamma^{k-1} \max_s \{V(s) - \mathcal{T}^\pi V(s)\} \mathbf{1} \\
&\geq \cdots\cdots \\
&\geq V - (1 + \cdots + \gamma^{k-1}) \max_s \{V(s) - \mathcal{T}^\pi V(s)\} \mathbf{1} \\
&= V - \frac{1 - \gamma^k}{1 - \gamma} \max_s \{V(s) - \mathcal{T}^\pi V(s)\} \mathbf{1}.
\end{aligned}
$$

Taking a limit on both sides yields the result.

Proof of Theorem 3

First one has

$$\mathcal{T}^\pi V - (\mathcal{T}^\pi)^2 V = r^\pi + \gamma P^\pi V - r^\pi - \gamma P^\pi (\mathcal{T}_\pi V) = \gamma P^\pi (V - \mathcal{T}^\pi V).$$

It follows that

$$\begin{aligned}\max_s \{\mathcal{T}^\pi V(s) - (\mathcal{T}^\pi)^2 V(s)\} &\leq \gamma \max_s \{P^\pi (V - \mathcal{T}^\pi V)(s)\} \leq \gamma \max_s \{(V - \mathcal{T}^\pi V)(s)\} \\ &= \gamma \max_s \{(V - \mathcal{T} V)(s)\} \leq \gamma(1+\gamma)\|V - V^*\|_\infty,\end{aligned}$$

where the inequality follows from the fact $\mathcal{T}^\pi V = \mathcal{T} V$ by the definition of π. Thus, the application of Lemma 1 yields that

$$\begin{aligned}V^\pi &\geq \mathcal{T}^\pi V - \frac{1}{1-\gamma} \max_s \{\mathcal{T}^\pi V(s) - (\mathcal{T}^\pi)^2 V(s)\}\mathbf{1} \\ &\geq \mathcal{T} V - \frac{\gamma(1+\gamma)}{1-\gamma}\|V - V^*\|_\infty \mathbf{1} \\ &= \mathcal{T} V - \mathcal{T} V^* + V^* - \frac{\gamma(1+\gamma)}{1-\gamma}\|V - V^*\|_\infty \mathbf{1},\end{aligned}$$

from which the assertion follows directly.

ε-Optimal Policy and Error Amplification

Theorem 4 (Q-error amplification)

For any vector $Q \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|}$, let π be the greedy policy with respect to Q, i.e.,

$$\pi(a|s) = \begin{cases} 1 & a = \arg\max_{a\in\mathcal{A}} Q(s,a), \\ 0 & \text{otherwise.} \end{cases}$$

Then $V^{\pi} \geq V^{*} - \frac{2}{1-\gamma} \|Q - Q^{*}\|_{\infty} \mathbf{1}$.

Proof of Theorem 4

A direct calculation yields

$$
\begin{aligned}
V^*(s) - V^\pi(s) &= Q^*(s, \pi^*(s)) - Q^\pi(s, \pi(s)) \\
&= Q^*(s, \pi^*(s)) - Q^*(s, \pi(s)) + Q^*(s, \pi(s)) - Q^\pi(s, \pi(s)) \\
&= Q^*(s, \pi^*(s)) - Q^*(s, \pi(s)) + \gamma \mathbb{E}_{s' \sim P(\cdot|s, \pi(a))} \left[V^*(s') - V^\pi(s') \right] \\
&\leq Q^*(s, \pi^*(s)) - Q(s, \pi^*(s)) + Q(s, \pi(s)) - Q^*(s, \pi(s)) \\
&\quad + \gamma \mathbb{E}_{s' \sim P(\cdot|s, \pi(a))} \left[V^*(s') - V^\pi(s') \right] \\
&\leq 2 \|Q - Q^*\|_\infty + \gamma \|V^* - V^\pi\|_\infty .
\end{aligned}
$$

The proof is complete after rearrangement.

Computational Complexity for ε-Optimal Policy

Theorem 5 (Computational Complexity of Value Iteration)
Fix a target accuracy ε. Then after

$$O\left(\frac{|\mathcal{S}|^2 |\mathcal{A}|}{1-\gamma} \log\left(\frac{1}{(1-\gamma)\,\varepsilon}\right)\right)$$

elementary arithmetic operations, VI produces a ε-optimal π.

Theorem 6 (Computational Complexity of Policy Iteration)
Fix a target accuracy ε. Then after

$$O\left(\frac{|\mathcal{S}|^3 + |\mathcal{S}|^2 |\mathcal{A}|}{1-\gamma} \log\left(\frac{1}{\varepsilon}\right)\right)$$

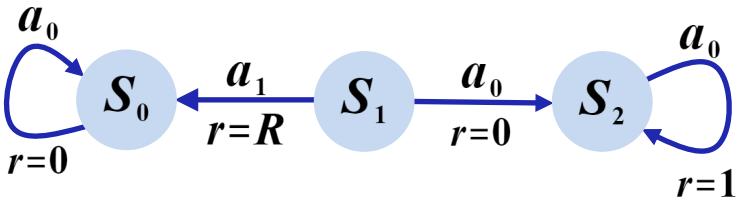
elementary arithmetic operations, PI produces a ε-optimal π.

Computational Complexity for Optimal Policy

Definition 2 (Strongly Polynomial)
*An algorithm is strongly polynomial if it is guaranteed to find an optimal policy with computation complexity **only** being polynomial in $|\mathcal{S}|$, $|\mathcal{A}|$, and the planning horizon $\frac{1}{1-\gamma}$.*

- VI is not strongly polynomial, but PI is strongly polynomial.

VI is Not Strongly Polynomial: Example



- three states: $\mathcal{S} = \{s_1, s_1, s_3\}$
- two actions: $\mathcal{A} = \{a_0, a_1\}$

Each edge is associated with a deterministic transition and a reward.

Recall that at k-th iteration, if starting from $V^0 = 0$ then one has

$$V^k(s_0) = 0, V^k(s_1) = \max\left\{\frac{\gamma}{1-\gamma}\left(1-\gamma^{k-1}\right), R\right\}, V^k(s_2) = \frac{1-\gamma^k}{1-\gamma}.$$

The greedy policy with respect to V^k at state s_1 satisfies:

$$\pi_k(s_1) = \begin{cases} a_0 & \text{if } \frac{\gamma}{1-\gamma}\left(1-\gamma^{k-1}\right) > R \\ a_1 & \text{otherwise.} \end{cases}$$

"Modified policy iteration algorithms are not strongly polynomial for discounted dynamic programming" by Eugene A. Feinberg, Jefferson Huang and Bruno Scherrer, 2014.

VI is Not Strongly Polynomial: Example

Assume $R < \frac{\gamma}{1-\gamma}$. Then $V^*(s_1) = \frac{\gamma}{1-\gamma}$ and the optimal action at s_1 is a_0. Thus the greedy policy is optimal only if:

$$\frac{\gamma}{1-\gamma}\left(1-\gamma^{k-1}\right) > R \Leftrightarrow \gamma^{k-1} < 1 - R\left(\frac{1-\gamma}{\gamma}\right) \Rightarrow k > 1 + \frac{\log\left(1 - R\left(\frac{1-\gamma}{\gamma}\right)\right)}{\log\gamma}.$$

Since $k \to \infty$ when $R \to \frac{\gamma}{1-\gamma}$, (nearly) infinite iterations are needed to produce an optimal policy. This basically means that when R is approaching $\frac{\gamma}{1-\gamma}$, it becomes difficult to tell optimal action from non-optimal one and the problem becomes difficult for VI.

Policy Iteration is Strongly Polynomial

Lemma 2 (Strict Progress Lemma)

Fix a suboptimal policy π_0 and let $\{\pi_k\}$ be the sequence of policies produced by PI. Then there exists a state s such that for any $k \geq \frac{1}{1-\gamma} \log\left(\frac{1}{1-\gamma}\right)$, one has

$$\pi_k(s) \neq \pi_0(s).$$

- The lemma shows that after every k iterations, policy iteration eliminates a suboptimal action at one state until there remains no suboptimal action to be eliminated. This can only be continued for at most $|\mathcal{S}||\mathcal{A}| - |\mathcal{S}|$ times: for every state, at least one action must be optimal.
- The strongly polynomial property of PI is related to the fact there are no repetition policies (without considering ties) during the iteration unless optimal policy is reached. If a different action is selected compared to last iteration, then basically strict progress will be made.

"Improved and generalized upper bounds on the complexity of policy iteration" by Bruno Scherrer, 2016.

Proof of Lemma 2

The first key question is about how to measure the progress of policies. To this end, consider

$$g(\pi', \pi) = \mathcal{T}^{\pi'} V^{\pi} - V^{\pi},$$

which can be viewed as advantage of π' relative to π in one-step lookahead. Recall from Lecture 1 that if $g(\pi', \pi) \geq 0$, then

$$V^{\pi'} - V^{\pi} = (I - \gamma P^{\pi'})^{-1}(r_{\pi'} - (I - \gamma P^{\pi'})V^{\pi}) = (I - \gamma P^{\pi'})^{-1} g(\pi', \pi) \geq 0.$$

Moreover, it can be shown that π^* is the optimal policy if and only if

$$g(\pi, \pi^*) \leq 0, \quad \forall\, \pi.$$

Thus, we can use $-g(\pi_k, \pi^*)$ to measure the progress of π_k, which is expected to decrease to zero. It is easy to see that if

$$-g(\pi_k, \pi^*)(s) < -g(\pi_0, \pi^*)(s),$$

then $\pi_k(s) \neq \pi_0(s)$.

Proof of Lemma 2 (Cont'd)

Moreover, we have

$$-g(\pi_k, \pi^*) = (I - \gamma P^{\pi_k})(V^* - V^{\pi_k}) = V^* - V^{\pi_k} - \gamma P^{\pi_k}(V^* - V^{\pi_k}) \leq V^* - V^{\pi_k}.$$

It follows that

$$\begin{aligned}
\|g(\pi_k, \pi^*)\|_\infty &\leq \|V^{\pi_k} - V^*\|_\infty \leq \gamma^k \|V^{\pi_0} - V^*\|_\infty \\
&= \gamma^k \|(I - \gamma P^{\pi_0})^{-1} g(\pi_0, \pi^*)\|_\infty \\
&\leq \frac{\gamma^k}{1-\gamma} \|g(\pi_0, \pi^*)\|_\infty
\end{aligned}$$

Thus, there exists an s such that

$$-g(\pi_k, \pi^*)(s) < -g(\pi_0, \pi^*)(s)$$

for sufficiently large k.

Runtime Bound for Policy Iteration

Theorem 7

Let $\{\pi_k\}$be the sequence of policies obtained by policy iteration starting from an arbitrary initial policy π_0. Then, after at most

$$O\left(\frac{|\mathcal{S}||\mathcal{A}| - |\mathcal{S}|}{1-\gamma}\log\left(\frac{1}{1-\gamma}\right)\right)$$

iterations, the policy produced by policy iteration is optimal. In particular, policy iteration can compute an optimal policy with at most

$$O\left(\frac{|\mathcal{S}|^4|\mathcal{A}| + |\mathcal{S}|^3|\mathcal{A}|^2}{1-\gamma}\log\left(\frac{1}{1-\gamma}\right)\right)$$

arithmetic and logic operations.

Another Strongly Polynomial Approach: Linear Programing (LP)

The linear programming approach is based on an interesting fact: If a vector V satisfies $\mathcal{T}V \leq V$ then $V^* \leq V$. This means that for all $s \in \mathcal{S}$,

$$V^*(s) = \min \{V(s) : \mathcal{T}V \leq V\}.$$

Thus V^* is the unique solution of following optimization problem:

$$\begin{aligned} \min \quad & \sum_{s \in \mathcal{S}} V(s) \\ \text{s.t.} \quad & \mathcal{T}V(s) = \max_{a \in \mathcal{A}} \sum_{s'} P\left(s'|s,a\right)\left(r\left(s,a,s'\right) + \gamma V\left(s'\right)\right) \leq V(s), \ \forall s \in \mathcal{S}. \end{aligned}$$

This is further equivalent to LP with $|\mathcal{S}|$ unknown variables and $|\mathcal{S}| \times |\mathcal{A}|$ inequality constraints:

$$\begin{aligned} \min \quad & \sum_{s \in \mathcal{S}} V(s) \\ \text{s.t.} \quad & \sum_{s' \in \mathcal{S}} p\left(s'|s,a\right)\left(r\left(s,a,s'\right) + \gamma V\left(s'\right)\right) \leq V(s), \ \forall s \in \mathcal{S}, a \in \mathcal{A}. \end{aligned}$$

"The Simplex and Policy-Iteration Methods are Strongly Polynomial for the Markov Decision Problem with a Fixed Discount Rate" by Yinyu Ye, 2011.

Questions?