Markov Decision Processes

Chen-Yu Wei

Sequence of Actions

To win the game, the learner has to take a sequence of actions $a_1 \to a_2 \to \cdots \to a_H$.

One option: view every sequence as a “meta-action”: $\bar{a} = (a_1, a_2, \cdots, a_H)$

Drawback:

- The number of actions is exponential in horizon
- In stochastic environments, this does not leverage intermediate observations

Solution idea: dynamic programming

Interaction Protocol: Fixed-Horizon Case

For **episode** $t = 1, 2, \dots, T$:

For **step** $h = 1, 2, \dots, H$:

Learner observes an observation $x_{t,h}$

Learner chooses an action $a_{t,h}$

Learner receives instantaneous reward $r_{t,h}$

General case:

$$\mathbb{E}[r_{t,h}] = R(x_{t,1}, a_{t,1}, \dots, x_{t,h}, a_{t,h}), \quad x_{t,h+1} \sim P(\cdot \mid x_{t,1}, a_{t,1}, \dots, x_{t,h}, a_{t,h})$$

$\Rightarrow$ Optimal decisions may depend on the entire history $\mathcal{H}_t = (x_{t,1}, a_{t,1}, \dots, x_{t,h})$

Interaction Protocol: Fixed-Horizon Case

For **episode** $t = 1, 2, \ldots, T$:

  For **step** $h = 1, 2, \ldots, H$:

    Learner observes an observation $x_{t,h}$

    Learner chooses an action $a_{t,h}$

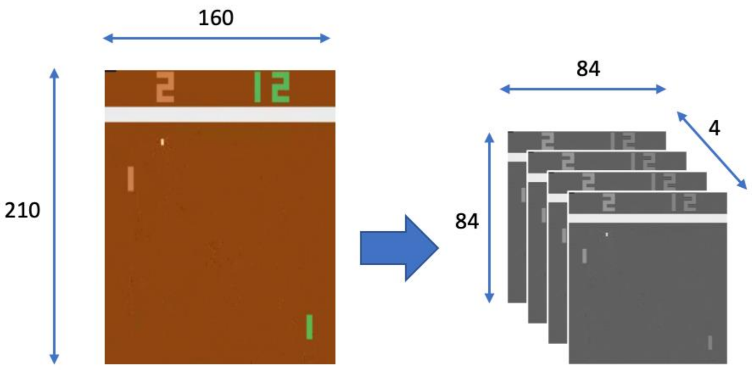
    Learner receives instantaneous reward $r_{t,h}$

We assume that the history $\mathcal{H}_t = (x_{t,1}, a_{t,1}, \ldots, x_{t,h})$ can be summarized as a **horizon-length-independent** representation $s_{t,h} = \Phi(x_{t,1}, a_{t,1}, \ldots, x_{t,h}) \in \mathcal{S}$ so that

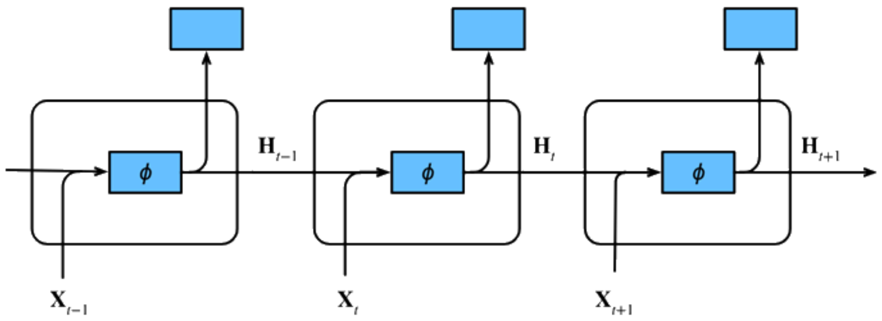
$$\mathbb{E}[r_{t,h}] = R(s_{t,h}, a_{t,h}), \quad x_{t,h+1} \sim P(\cdot \mid s_{t,h}, a_{t,h})$$

$s_{t,h}$ is called the "state" at the step h of episode t.

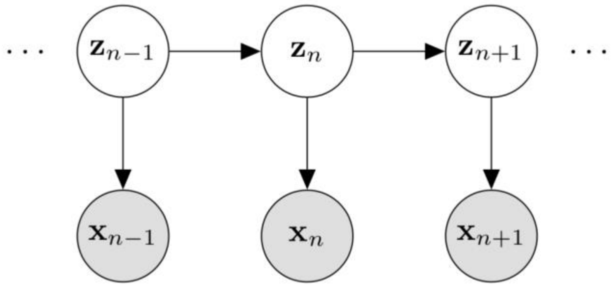
From Observations to States



Stacking recent observations



Recurrent neural network



Hidden Markov model

Interaction Protocol: Fixed-Horizon Case

For **episode** $t = 1, 2, \ldots, T$:

For **step** $h = 1, 2, \ldots, H$:

Environment reveals state $s_{t,h}$

Learner chooses an action $a_{t,h}$

Learner observes instantaneous reward $r_{t,h}$ with $\mathbb{E}[r_{t,h}] = R(s_{t,h}, a_{t,h})$

Next state is generated as $s_{t,h+1} \sim P(\cdot \mid s_{t,h}, a_{t,h})$

This is called the Markov decision process.

MDP as Contextual Bandits?

Viewing states as contexts, and viewing the problem as a contextual bandit problem with TH rounds (what's wrong?)

$(x_{t,1}, a_{t,1}, \ldots, x_{t,h})$

$$\text{Regret (contextual bandit)} = \sum_{t=1}^{T}\sum_{h=1}^{H} \max_a R(s_{t,h}, a) - \sum_{t=1}^{T}\sum_{h=1}^{T} R(s_{t,h}, a_{t,h})$$

$$\text{Regret (MDP)} = \sum_{t=1}^{T}\left[\sum_{h=1}^{H} R(s^*_{t,h}, a^*_{t,h})\right] - \sum_{t=1}^{T}\sum_{h=1}^{H} R(s_{t,h}, a_{t,h})$$

$$s^*_{t,1} = s_{t,1}$$

$$s^*_{t,h} \neq s_{t,h} \quad \text{for } h \geq 2$$

Formulations

- Interaction Protocol
  - Fixed-Horizon
  - Variable-Horizon (Goal-Oriented)
  - Infinite-Horizon
- Performance Metric
  - Total Reward
  - Average Reward
  - Discounted Reward
- Policy
  - History-dependent policy
  - Markov policy
  - Stationary policy

Horizon = Length of an episode

Interaction Protocols (1/3): Fixed-Horizon

Horizon length is a fixed number H

$h \leftarrow 1$

Observe initial state s_1

While $h \leq H$:

Choose action a_h

Observe reward r_h with $\mathbb{E}[r_h] = R(s_h, a_h)$

Observe next state $s_{h+1} \sim P(\cdot \mid s_h, a_h)$

Examples: games with a fixed number of time

Interaction Protocols (2/3): Goal-Oriented

The learner interacts with the environment until reaching **terminal states** $\mathcal{T} \subset \mathcal{S}$

$h \leftarrow 1$

Observe initial state s_1

While $s_h \notin \mathcal{T}$:

Choose action a_h

Observe reward r_h with $\mathbb{E}[r_h] = R(s_h, a_h)$

Observe next state $s_{h+1} \sim P(\cdot \mid s_h, a_h)$

$h \leftarrow h + 1$

Examples: video games, robotics tasks, personalized recommendations, etc.

Interaction Protocols (3/3): Infinite-Horizon

The learner continuously interacts with the environment

$h \leftarrow 1$

Observe initial state s_1

Loop forever:

Choose action a_h

Observe reward r_h with $\mathbb{E}[r_h] = R(s_h, a_h)$

Observe next state $s_{h+1} \sim P(\cdot\,|s_h, a_h)$

$h \leftarrow h + 1$

Examples: network management, inventory management

Formulations for Markov Decision Processes

- Interaction Protocol
  - Fixed-Horizon
  - Variable-Horizon (Goal-Oriented)
  - Infinite-Horizon

  (Fixed-Horizon and Variable-Horizon: Episodic setting)
- Performance Metric
  - Total Reward
  - Average Reward
  - Discounted Reward
- Policy
  - History-dependent policy
  - Markov policy
  - Stationary policy

Performance Metric

Total Reward (for episodic setting): $$\sum_{h=1}^{\tau} r_h$$ (τ: the step where the episode ends)

Average Reward (for infinite-horizon setting): $$\lim_{T\to\infty} \frac{1}{T} \sum_{h=1}^{T} r_h$$

If $|r_h| \leq 1$,

Discounted Total Reward (for episodic or infinite-horizon): $$\sum_{h=1}^{\tau} \gamma^{h-1} r_h \leq \frac{1}{1-\gamma}$$

τ: the step where the episode ends, or ∞ in the infinite-horizon case

$\gamma \in [0,1)$: discount factor

Interaction Protocols vs. Performance Metrics

"natural" objective

Fixed-Horizon ---------------------→ Total Reward

Goal-Oriented ---------------------→ Total Reward — Could be unbonded

Infinite-horizon ---------------------→ Average Reward — Could have constant change for an infinitesimal change in policy

Discounted Total Reward?

Focusing more on the **recent** reward

There is a potential mismatch between our ultimate goal and what we optimized.

Our Focus

In most of the following lectures, we focus on the **goal-oriented / infinite-horizon** setting with **discount total reward** as the performance metric.

Policy

A mapping from observations/contexts/states to (distribution over) actions

- Contextual bandits

$$a \sim \pi(\cdot \mid x) \quad \text{(randomized/stochastic)}$$

$$\text{or } a = \pi(x) \quad \text{(deterministic)}$$

- Multi-armed bandits

$$a \sim \pi$$

$$\text{or } a = a^\star$$

Policy for MDPs

History-dependent Policy

$$a_h \sim \pi(\cdot \mid s_1, a_1, r_1, s_2, a_2, r_2, \ldots, s_h)$$
$$a_h = \pi(s_1, a_1, r_1, s_2, a_2, r_2, \ldots, s_h)$$

Markov Policy

$$a_h \sim \pi(\cdot \mid s_h, h)$$
$$a_h = \pi(s_h, h)$$

← For **fixed-horizon + total reward** setting, there exists an optimal policy in this class

Stationary Policy

$$a_h \sim \pi(\cdot \mid s_h)$$
$$a_h = \pi(s_h)$$

← For **infinite-horizon/goal-oriented + discounted total reward** setting, there exists an optimal policy in this class

Fixed-Horizon + Total Reward

Dynamic Programming

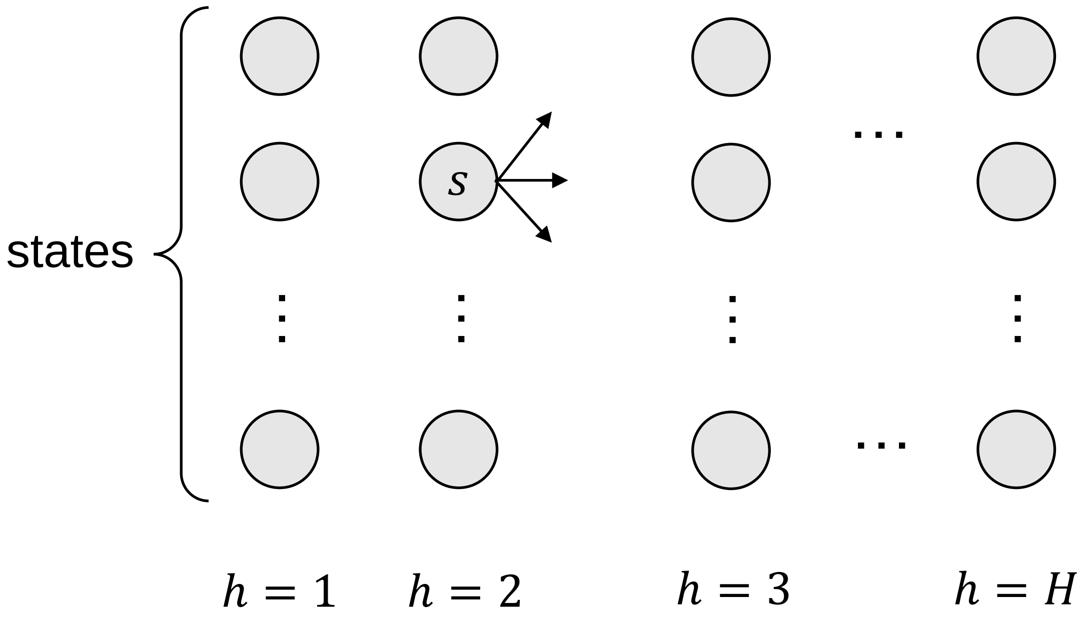
Goal: Calculate the expected total reward of a policy

A (Markov) policy is a mapping from (state, step index) to action distribution, written as

$$\pi_h(\cdot \mid s) \in \Delta(\mathcal{A}) \qquad \text{for } s \in \mathcal{S} \text{ and } h \in \{1, 2, \dots, H\}$$

Dynamic Programming

$V_h^\pi(s) = \mathbb{E}\left[\sum_{i=h}^{H} R(s_i, a_i) \,\middle|\, s_h = s,\ a_i \sim \pi_i(\cdot|s_i)\ \forall i \geq h \right]$



State transition: $P(s'|s,a)$

Reward: $R(s,a)$

Key quantity: $V_h^\pi(s)$ = the expected total reward of policy π starting from state s at step h.

Backward calculation:

$$V_H^\pi(s) = \sum_a \pi_H(a|s)\, R(s,a) \qquad \forall s$$

For $h = H-1, \ldots 1$: for all s

$$V_h^\pi(s) = \sum_a \pi_h(a|s) \left(R(s,a) + \underbrace{\sum_{s'} P(s'|s,a)\, V_{h+1}^\pi(s')}_{\text{Expected total reward from step } h+1} \right)$$

Bellman Equation

$$Q_h^\pi(s,a) = \mathbb{E}\left[\sum_{i=h}^{H} R(s_i, a_i) \,\middle|\, s_h = s,\ a_h = a,\ a_i \sim \pi_i(\cdot|s_i)\ \forall i \geq h+1 \right]$$

$$V_{H+1}^\pi(s) = 0$$

$$V_h^\pi(s) = \sum_a \pi_h(a|s) \underbrace{\left(R(s,a) + \sum_{s'} P(s'|s,a) V_{h+1}^\pi(s') \right)}_{Q_h^\pi(s,a)} \qquad \text{for } h = H, \dots, 1$$

$$V_h^\pi(s) = \sum_{a \in \mathcal{A}} \pi_h(a|\ s) Q_h^\pi(s,a)$$

$$Q_h^\pi(s,a) = R(s,a) + \sum_{s'} P(s'|s,a) V_{h+1}^\pi(s')$$

Occupancy Measures

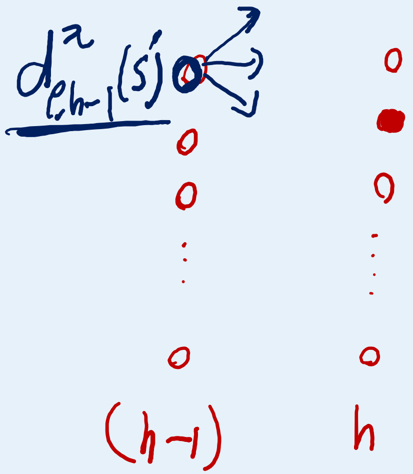
$$d_\rho^\pi(s) = \mathbb{E}\left[\sum_{h=1}^{H} \mathbb{I}\{s_h = s\} \,\middle|\, s_1 \sim \rho,\ a_i \sim \pi_i(\cdot|s_i)\ \forall i \geq 1\right]$$

$d_\rho^\pi(s)$: the expected number of times state s is visited, under policy π and initial state distribution ρ

$$d_{\rho,h}^\pi(s) = \Pr(s_h = s \mid \text{—}) = \mathbb{E}[\mathbb{I}\{s_h = s\} \mid \text{—}], \quad d_\rho^\pi(s) = \sum_{h=1}^{H} d_{\rho,h}^\pi(s)$$

Key quantity: $d_{\rho,h}^\pi(s)$ = the probability of state s being visited **at step *h***, under policy π and initial state distribution ρ

Forward calculation:



$d_{\rho,1}^\pi(s) = \rho(s) \quad \forall s$

For $h = 2, \ldots H$:

$$d_{\rho,h}^\pi(s) = \sum_{s'} d_{\rho,h-1}^\pi(s') \sum_{a'} \pi_{h-1}(a'|s') P(s|s',a') \quad \forall s$$

Reverse Bellman Equation

$$d^{\pi}_{\rho,1}(s) = \rho(s)$$

$$d^{\pi}_{\rho,h}(s) = \sum_{s',a'} \underbrace{d^{\pi}_{\rho,h-1}(s')\,\pi_{h-1}(a'|s')}_{d^{\pi}_{\rho,h-1}(s',a') \,\doteq\, \Pr(s_{h-1}=s',\, a_{h-1}=a' \,|\, s_1\sim\rho,\ \pi)} P(s|s',a') \qquad \text{for } h = 2, \ldots, H$$

$$d^{\pi}_{\rho,h}(s) = \sum_{s',a'} d^{\pi}_{\rho,h-1}(s',a') P(s|s',a')$$

$$d^{\pi}_{\rho,h}(s,a) = d^{\pi}_{\rho,h}(s)\pi_h(a|s)$$

Dynamic Programming

$V_h^\star(s) = \max_\pi V_h^\pi(s)$

Goal: Find the optimal policy

Key quantity: $V_h^\star(s)$ = the optimal expected total reward starting from state s at step h.

Backward calculation:

$$V_H^\star(s) = \max_a R(s,a) \quad \forall s$$

For $h = H-1, \ldots 1$:

$$V_h^\star(s) = \max_a \left(R(s,a) + \sum_{s'} P(s'|s,a)\, V_{h+1}^\star(s') \right) \quad \forall s$$

Value Iteration

$$\pi_h^\star(s) = \operatorname{argmax}_a R(s,a) + \sum_{s'} P(s'|s,a)\, V_{h+1}^\star(s')$$

Bellman Optimality Equation

$$V^\star_{H+1}(s) = 0$$

$$V^\star_h(s) = \max_a \underbrace{\left(R(s,a) + \sum_{s'} P(s'|s,a)\, V^\star_{h+1}(s') \right)}_{Q^\star_h(s,a)} \qquad \text{for } h = H, \dots, 1$$

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

$$Q^\star_h(s,a) = R(s,a) + \sum_{s'} P(s'|s,a)\, V^\star_{h+1}(s')$$

$$\pi^\star_h(s) = \operatorname{argmax}_a \; Q^\star_h(s,a)$$

Recap

$$V_h^\pi(s) = \sum_{a \in \mathcal{A}} \pi_h(a|\, s) Q_h^\pi(s,a)$$

$$Q_h^\pi(s,a) = R(s,a) + \sum_{s' \in \mathcal{S}} P(s'|s,a)\, V_{h+1}^\pi(s')$$

Bellman Equation
(Value Iteration for V^π)

$$d_{\rho,h}^\pi(s,a) = d_{\rho,h}^\pi(s)\pi_h(a|s)$$

$$d_{\rho,h}^\pi(s) = \sum_{s',a'} d_{\rho,h-1}^\pi(s',a')P(s|s',a')$$

Reverse Bellman Equation

$$V_h^\star(s) = \max_a Q_h^\star(s,a)$$

$$Q_h^\star(s,a) = R(s,a) + \sum_{s' \in \mathcal{S}} P(s'|s,a)\, V_{h+1}^\star(s')$$

Bellman Optimality Equation
(Value Iteration)

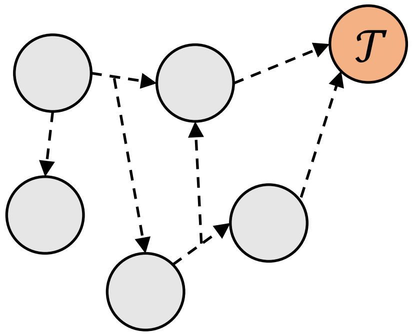
Infinite-Horizon / Goal-Oriented + Discounted Total Reward

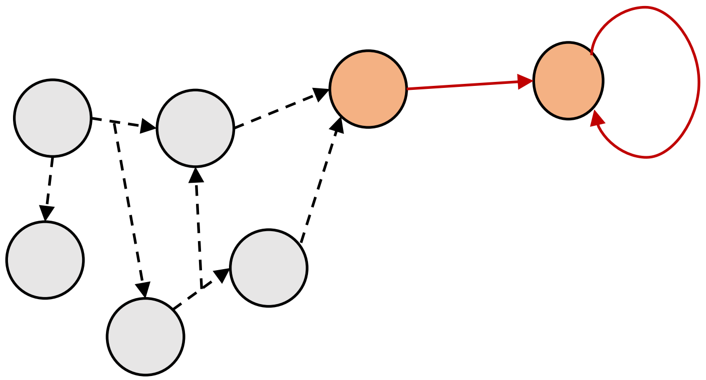
Equivalent Views

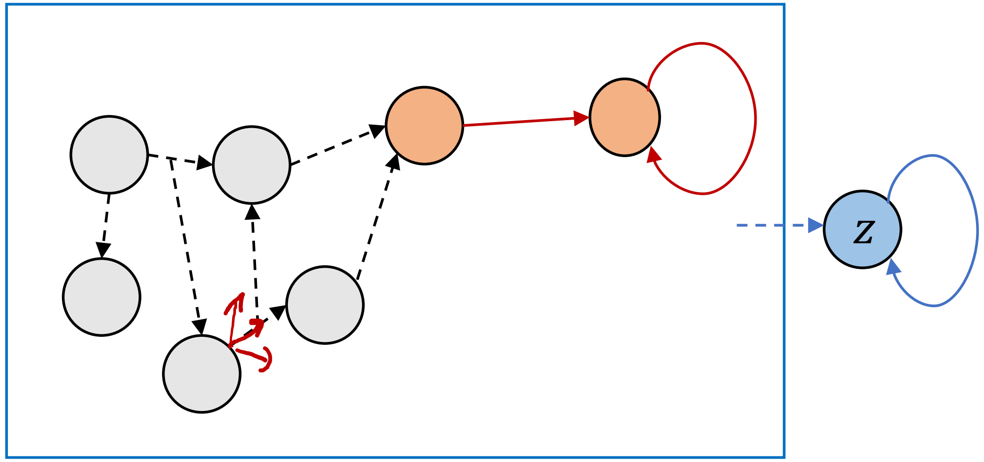
$P^{new}(s'|s,a) = \gamma P^{old}(s'|s,a)$, $P^{new}(z|s,a) = 1-\gamma$

Scale down all transitions by a factor of γ and add probability $1-\gamma$ transitioning to z

→ deterministic and zero-reward







Converting goal-oriented to infinite-horizon

Converting discounted total reward to total reward

$$\mathbb{E}^{\text{new}}\left[\sum_{h=1}^{\infty}\gamma^{h-1}r_h\right] = \mathbb{E}^{\text{old}}\left[\sum_{h=1}^{\tau}\gamma^{h-1}r_h\right]$$

$$\mathbb{E}^{\text{new}}\left[\sum_{h=1}^{\infty}r_h\right] = \mathbb{E}^{\text{old}}\left[\sum_{h=1}^{\infty}\gamma^{h-1}r_h\right]$$

Prob of staying in original MDP at step h

Dynamic Programming

$V^{\pi}(s) = \lim_{H\to\infty} \mathbb{E}\left[\sum_{h=1}^{H} \gamma^{h-1} R(s_h, a_h) \,\middle|\, s_1 = s,\ a_h \sim \pi(\cdot|s_h)\ \forall h \geq 1\right]$

Goal: Calculate the expected discounted total reward of a stationary policy π

$V^{\pi}(s)$ = the expected discounted total reward starting from state s, follow π

Key quantity: $V_i^{\pi}(s)$ = the expected discounted total reward starting from state s supposed that i more steps can be executed

$V_i^{\pi}(s) = \mathbb{E}\left[\sum_{h=1}^{i} \gamma^{h-1} R(s_h, a_h) \,\middle|\, \sim\right]$

$V_0^{\pi}(s) = 0 \quad \forall s$ (follow π)

For $i = 1, 2, 3$ ...

$$V_i^{\pi}(s) = \sum_a \pi(a|s)\left(R(s,a) + \gamma \sum_{s'} P(s'|s,a)\, V_{i-1}^{\pi}(s')\right) \quad \forall s$$

$+ (1-\gamma) \times 0$

$V^{\pi}(s) = \lim_{i\to\infty} V_i^{\pi}(s)$ (need to prove that the limit exists)

Value Iteration for V^π

$$\lim_{i \to \infty} \hat{V}_i(s) = V^\pi(s)$$

Arbitrary $\hat{V}_0(s) \quad \forall s$

For $i = 1, 2, 3$ ...

$$\hat{V}_i(s) = \sum_a \pi(a|s) \left(R(s,a) + \gamma \sum_{s'} P(s'|s,a)\, \hat{V}_{i-1}(s') \right) \quad \forall s$$

To show that this algorithm converges, we prove the following statement:

For any $\epsilon > 0$, there exists a large enough N such that

$$\left|\hat{V}_i(s) - \hat{V}_j(s)\right| \leq \epsilon$$

for any $i, j \geq N$.

Proof of Convergence

$$|\hat{V}_{i+1}(s) - \hat{V}_i(s)| \le O(\gamma^i) \quad \forall s.$$

$$|\hat{V}_i(s) - \hat{V}_j| \le \sum_{k=i}^{j-1} |\hat{V}_k(s) - \hat{V}_{k+1}(s)| = \sum_{k=i}^{j-1} O(\gamma^k) \le O\left(\frac{\gamma^i}{1-\gamma}\right)$$

$$\hat{V}_i(s) = \sum_a \pi(a|s)\left(R(s,a) + \gamma \sum_{s'} P(s'|s,a)\hat{V}_{i-1}(s')\right) \quad \forall s$$

$$\hat{V}_{i+1}(s) = \sum_a \pi(a|s)\left(R(s,a) + \gamma \sum_{s'} P(s'|s,a)\hat{V}_i(s')\right) \quad \forall s.$$

$$\Rightarrow \quad \hat{V}_{i+1}(s) - \hat{V}_i(s) = \gamma \sum_a \pi(a|s) \sum_{s'} P(s'|s,a)\left(\hat{V}_i(s') - \hat{V}_{i-1}(s')\right)$$

$$\Rightarrow \max_s |\hat{V}_{i+1}(s) - \hat{V}_i(s)| \le \gamma \sum_a \pi(a|s) \sum_{s'} P(s'|s,a) \underline{|\hat{V}_i(s') - \hat{V}_{i-1}(s')|}$$

$$= \gamma \underbrace{\sum_a \pi(a|s)}_{1} \underbrace{\sum_{s'} P(s'|s,a)}_{1} \max_{s^*} |\hat{V}_i(s^*) - \hat{V}_{i-1}(s^*)|$$

$$\le \gamma \max_{s^*} |\hat{V}_i(s^*) - \hat{V}_{i-1}(s^*)|$$

Proof of Convergence

For any $\epsilon > 0$, there exists a large enough N such that

$$\left|\hat{V}_i(s) - \hat{V}_j(s)\right| \leq \epsilon$$

for any $i, j \geq N$.

$$\hat{V}(s) = \lim_{i \to \infty} \inf\left\{\hat{V}_j(s) : j \geq i\right\}$$

For any $\epsilon > 0$, there exists a large enough N such that

$$\left|\hat{V}_i(s) - \hat{V}(s)\right| \leq \epsilon$$

for any $i \geq N$.

Proof of Uniqueness

No matter what the initial values of $\hat{V}_0(s)$ are, the limit $\lim_{i\to\infty} \hat{V}_i(s)$ is the same.

(This value is $V^\pi(s)$)

Assume $V^{(1)}_{(s)}$ and $V^{(2)}_{(s)}$ are different convergence point.

$$V^{(1)}(s) = \sum_a \pi(a|s)\left(R(s,a) + \gamma \sum_{s'} p(s'|s,a)\, V^{(1)}(s')\right)$$

$$V^{(2)}(s) \quad \text{———————————} \quad V^{(2)}(s')$$

$$V^{(1)}(s) - V^{(2)}(s) = \gamma \sum_a \pi(a|s) \sum_{s'} p(s'|s,a)\left(V^{(1)}(s') - V^{(2)}(s')\right)$$

$$\max_s \left|V^{(1)}(s) - V^{(2)}(s)\right| \leq \gamma \max_{s'} \left|V^{(1)}(s') - V^{(2)}(s')\right| \longrightarrow \left|V^{(1)}(s) - V^{(2)}(s)\right| = 0$$

Bellman Equation

$$V^\pi(s) = \sum_a \pi(a|s) \underbrace{\left(R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^\pi(s') \right)}_{Q^\pi(s,a)}$$

$$V^\pi(s) = \sum_a \pi(a|s) Q^\pi(s,a)$$

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

Approximate Bellman Equations

If $\left|\hat{V}(s) - \sum_a \pi(a|s)\left(R(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}[\hat{V}(s')]\right)\right| \le \epsilon \quad \forall s$

then $\left|\hat{V}(s) - V^\pi(s)\right| \le \dfrac{\epsilon}{1-\gamma} \quad \forall s$

$V^{(k-1)}$, $V^{(k-1)}$, $V^{(k)}$

$$V^\pi(s) = \sum_a \pi(a|s)\left(R(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}[V^\pi(s')]\right)$$

$$\Rightarrow \max_s |\hat{V}(s) - V^\pi(s)| = \left|\hat{V}(s) - \sum_a \pi(a|s)\left(R(s,a)+\gamma\mathbb{E}[V^\pi(s')]\right)\right|$$

$$\le \left|\sum_a \pi(a|s)\left(R(s,a)+\gamma\mathbb{E}[\hat{V}(s')]\right) - \sum_a \pi(a|s)\left(R(s,a)+\gamma\mathbb{E}[V^\pi(s')]\right)\right| + \epsilon$$

$$\le \gamma \max_{s'} |\hat{V}(s') - V^\pi(s')| + \epsilon \Rightarrow (1-\gamma)\max_s |\hat{V}(s) - V^\pi(s)| \le \epsilon$$

$$\left|\gamma \sum_a \pi(a|s) \mathbb{E}_{s'\sim P(\cdot|s,a)}[\hat{V}(s') - V^\pi(s')]\right|$$

$$\le \left|\gamma \sum_a \pi(a|s)\mathbb{E}\left[\max_{s'}|\hat{V}(s') - V^\pi(s')|\right]\right|$$

$$= \gamma \sum_a \pi(a|s) \max_{s'} |\hat{V}(s') - V^\pi(s')|$$

$$= \gamma \max_{s'} |\hat{V}(s') - V^\pi(s')|$$

Occupancy Measures

$$d_\rho^\pi(s) = \mathbb{E}\left[\sum_{h=1}^{\infty} \gamma^{h-1} \mathbb{I}\{s_h = s\} \,\middle|\, s_1 \sim \rho,\ a_h \sim \pi(\cdot|s_h)\ \forall h \geq 1\right]$$

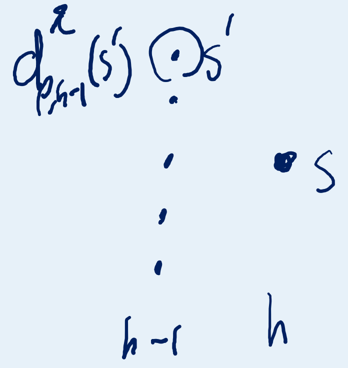
$d_\rho^\pi(s)$: the expected discounted number of times state s is visited, under policy π and initial state distribution ρ

$$d_{\rho,h}^\pi(s) = \mathbb{E}\left[\gamma^{h-1} \mathbb{I}\{s_h = s\} \,\middle|\, \text{———}\right]$$

Key quantity: $d_{\rho,h}^\pi(s) =$ the discounted probability of state s being visited at step h, under policy π and initial state distribution ρ

$$d_\rho^\pi(s) = \sum_{h=1}^{\infty} d_{\rho,h}^\pi(s)$$

Forward calculation:



$d_{\rho,1}^\pi(s) = \rho(s) \quad \forall s$

For $h = 2, 3, \ldots$

$$d_{\rho,h}^\pi(s) = \gamma \sum_{s'} d_{\rho,h-1}^\pi(s') \sum_{a'} \pi(a'|s') P(s|s',a') \quad \forall s$$

$$d^{\pi}_{\rho,h}(s) = \gamma \sum_{s'} d^{\pi}_{\rho,h-1}(s') \sum_{a'} \pi(a'|s')\, p(s|s',a')$$

$$\Rightarrow \sum_{h=2}^{\infty} d^{\pi}_{\rho,h}(s) = \gamma \sum_{s'} \underbrace{\sum_{h=2}^{\infty} d^{\pi}_{\rho,h-1}(s')} \sum_{a'} \pi(a'|s')\, p(s|s',a')$$

$$\Rightarrow d^{\pi}_{\rho}(s) - \underbrace{d^{\pi}_{\rho,1}(s)}_{\rho(s)} = \gamma \sum_{s'} d^{\pi}_{\rho}(s') \sum_{a'} \pi(a'|s')\, p(s|s',a')$$

Reverse Bellman Equation

$$d_\rho^\pi(s,a) = \mathbb{E}\left[\sum_{h=1}^{\infty} \gamma^{h-1}\mathbb{I}\{s_h=s, a_h=a\} \,\middle|\, \begin{matrix} s_1\sim\rho \\ \text{follow } \pi \end{matrix}\right]$$

$$d_\rho^\pi(s) = \rho(s) + \gamma \sum_{s',a'} \underbrace{d_\rho^\pi(s')\,\pi(a'|s')}_{d_\rho^\pi(s',a')} P(s|s',a')$$

$$\sum_s d_\rho^\pi(s) = \frac{1}{1-\gamma}$$

$$d_\rho^\pi(s) = \rho(s) + \gamma \sum_{s',a'} d_\rho^\pi(s',a') P(s|s',a')$$

$$d_\rho^\pi(s,a) = d_\rho^\pi(s)\pi(a|s)$$

$$\sum_s d_\rho^\pi(s) = 1$$

Another (more common) version makes $d_\rho^\pi(s)$ a distribution over s

→ Just change the $\rho(s)$ in the first equation by $(1-\gamma)\rho(s)$

Dynamic Programming

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

Goal: find optimal policy

Key quantity: $V_i^{\star}(s)$ = the optimal discounted total reward starting from state s supposed that i more steps can be executed

$$V_0^{\star}(s) = 0 \quad \forall s$$

For $i = 1, 2, 3$ ...

$$V_i^{\star}(s) = \max_a \left(R(s,a) + \gamma \sum_{s'} P(s'|s,a)\, V_{i-1}^{\star}(s') \right) \quad \forall s$$

} Value Iteration

$$V^{\star}(s) = \lim_{i \to \infty} V_i^{\star}(s) \qquad \pi^{\star}(s) = \operatorname{argmax}_a \; R(s,a) + \gamma \sum_{s'} P(s'|s,a)\, V^{\star}(s')$$

Bellman Optimality Equation

$$V^\star(s) = \max_a \underbrace{\left(R(s,a) + \gamma \sum_{s'} P(s'|s,a)\, V^\star(s') \right)}_{Q^\star(s,a)}$$

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

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

$$\pi^\star(s) = \operatorname{argmax}_a \; Q^\star(s,a)$$

Approximate Bellman Optimality Equations

Suppose that $\left|\hat{V}(s) - \max_a \left(R(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}\left[\hat{V}(s')\right]\right)\right| \leq \epsilon \quad \forall s$

Then

(1) $\left|\hat{V}(s) - V^\star(s)\right| \leq \frac{\epsilon}{1-\gamma} \quad \forall s$

(2) $V^\star(s) - V^{\hat{\pi}}(s) \leq \frac{2\epsilon}{1-\gamma} \quad \forall s$

where $\hat{\pi}(s) = \operatorname{argmax}_a \left(R(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}\left[\hat{V}(s')\right]\right)$

Summary

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

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

$$d_\rho^\pi(s,a) = d_\rho^\pi(s)\pi(a|s)$$

$$d_\rho^\pi(s) = (1-\gamma)\rho(s) + \gamma \sum_{s',a'} d_\rho^\pi(s',a') P(s|s',a')$$

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

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

Guarantees for approximate solutions

$$\left|\hat{V}(s) - \sum_{a\in\mathcal{A}} \pi(a|\,s) \left(R(s,a) + \gamma \sum_{s'\in\mathcal{S}} P(s'|s,a)\, \hat{V}(s') \right)\right| \leq \epsilon \quad \forall s$$

$$\Rightarrow\ \left|\hat{V}(s) - V^\pi(s)\right| \leq \frac{\epsilon}{1-\gamma} \quad \forall s$$

https://www.youtube.com/watch?v=XVuRQWXtxLA

$$\left|\hat{V}(s) - \max_a \left(R(s,a) + \gamma \sum_{s'\in\mathcal{S}} P(s'|s,a)\, \hat{V}(s') \right)\right| \leq \epsilon \quad \forall s$$

$\Rightarrow\ \left|\hat{V}(s) - V^\star(s)\right| \leq \frac{\epsilon}{1-\gamma}$ and $V^\star(s) - V^{\hat{\pi}}(s) \leq \frac{2\epsilon}{1-\gamma}$ $\forall s$

Policy Iteration

Policy Iteration

value iteration for V^π (Q^π)

for V^*

Policy Iteration

$Q^\pi(s,a) = R(s,a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[V^\pi(s')]$, $V^\pi(s) = \sum_a \pi(a|s) Q^\pi(s,a)$

For $k = 1, 2, \ldots$

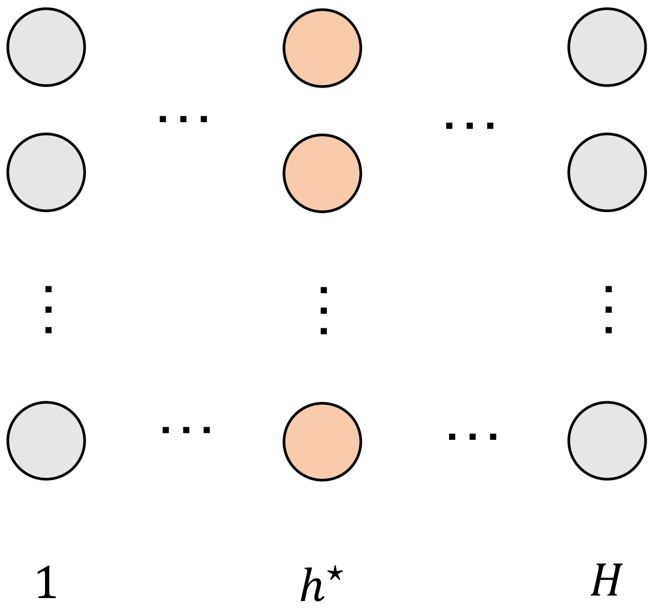
$$\forall s, \qquad \pi^{(k+1)}(s) \leftarrow \operatorname*{argmax}_a Q^{\pi^{(k)}}(s,a)$$

Theorem (monotonic improvement). Policy Iteration ensures

$$\forall s, \qquad V^{\pi^{(k+1)}}(s) \geq V^{\pi^{(k)}}(s)$$

Below, we will establish a more general lemma (not only show monotonic improvement, but also quantify *how much* the improvement is).

Single-Step Policy Modification under Fixed Horizon



$\pi'_h(\cdot\,|s) = \pi_h(\cdot|s) = \pi_{\text{in}}(\cdot\,|s)$ for $h < h^\star$

$\pi'_h(\cdot\,|s) = \pi_h(\cdot|s) = \pi_{\text{out}}(\cdot\,|s)$ for $h > h^\star$

Assume $\pi'_h(\cdot\,|s) = \pi_h(\cdot|s)$ for all $h \neq h^\star$

$$\mathbb{E}_{s\sim\rho}\left[V_1^{\pi'}(s)\right] - \mathbb{E}_{s\sim\rho}[V_1^{\pi}(s)] = ?$$

$$= \mathbb{E}\left[\sum_{h=1}^{H} R(s_h, a_h)\;\middle|\; s_1 \sim \rho, \pi'\right] - \mathbb{E}\left[\sum_{h=1}^{H} R(s_h, a_h)\;\middle|\; s_1 \sim \rho, \pi\right]$$

$$= \mathbb{E}\left[\sum_{h=h^\star}^{H} R(s_h, a_h)\;\middle|\; s_1 \sim \rho, \pi'\right] - \mathbb{E}\left[\sum_{h=h^\star}^{H} R(s_h, a_h)\;\middle|\; s_1 \sim \rho, \pi\right]$$

$$= \mathbb{E}\left[\sum_{h=h^\star}^{H} R(s_h, a_h)\;\middle|\; s_{h^\star} \sim d^{\pi_{\text{in}}}_{\rho,h^\star}, \pi'\right] - \mathbb{E}\left[\sum_{h=h^\star}^{H} R(s_h, a_h)\;\middle|\; s_h \sim d^{\pi_{\text{in}}}_{\rho,h^\star}, \pi\right]$$

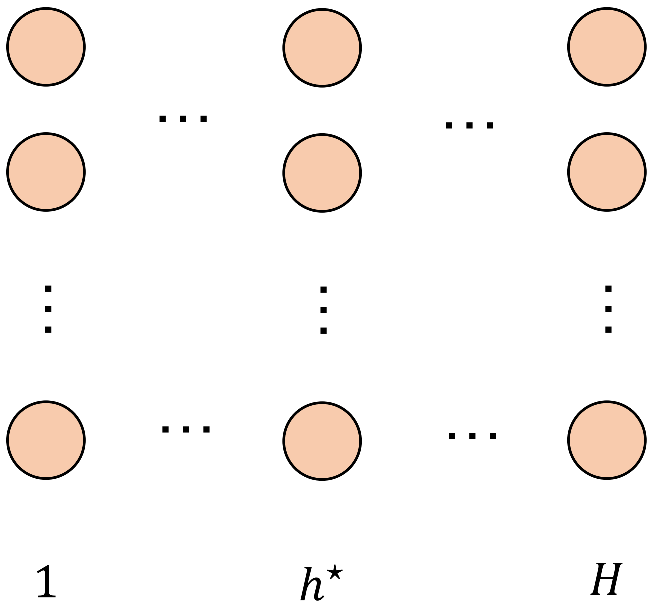
$$= \mathbb{E}_{s_{h^\star}\sim d^{\pi_{\text{in}}}_{\rho,h^\star}} \mathbb{E}_{a_{h^\star}\sim \pi'_{h^\star}(\cdot|s_{h^\star})}\left[Q^{\pi_{\text{out}}}_{h^\star}(s_{h^\star}, a_{h^\star})\right] - \mathbb{E}_{s_{h^\star}\sim d^{\pi_{\text{in}}}_{\rho,h^\star}} \mathbb{E}_{a_{h^\star}\sim \pi_{h^\star}(\cdot|s_{h^\star})}\left[Q^{\pi_{\text{out}}}_{h^\star}(s_{h^\star}, a_{h^\star})\right]$$

$$= \sum_{s,a} d^{\pi_{\text{in}}}_{\rho,h^\star}(s)\pi'_{h^\star}(a|s)Q^{\pi_{\text{out}}}_{h^\star}(s,a) - \sum_{s,a} d^{\pi_{\text{in}}}_{\rho,h^\star}(s)\pi_{h^\star}(a|s)Q^{\pi_{\text{out}}}_{h^\star}(s,a)$$

$$= \sum_{s,a} d^{\pi_{\text{in}}}_{\rho,h^\star}(s)\left(\pi'_{h^\star}(a|s) - \pi_{h^\star}(a|s)\right)Q^{\pi_{\text{out}}}_{h^\star}(s,a)$$

All-Step Policy Modification under Fixed Horizon

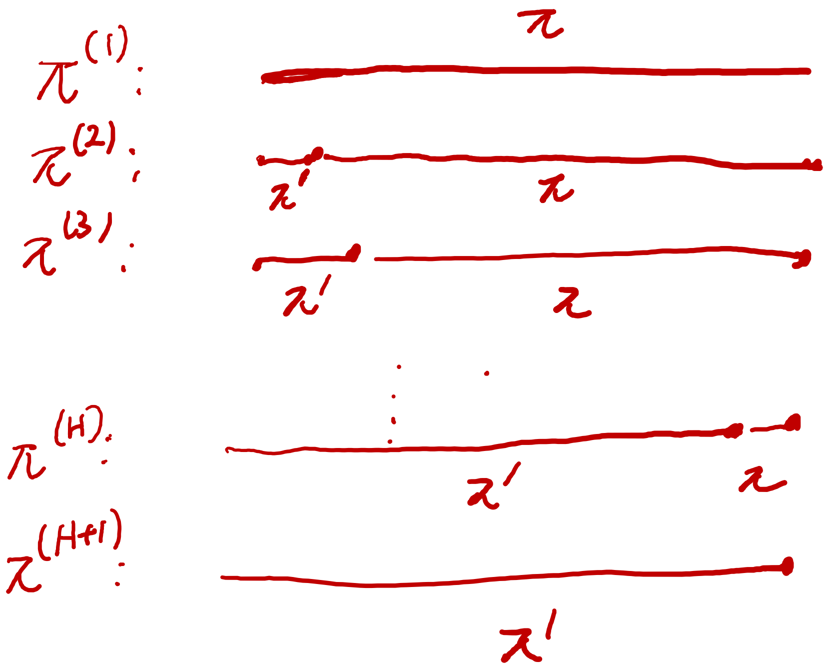
$\mathbb{E}_{s\sim\rho}\left[V_1^{\pi'}(s) - V_1^{\pi}(s)\right]$ for arbitrary π', π



Let $\pi^{(h)}$ be a Markov policy such that it is

- same as π' in steps 1 to $h-1$
- same as π in steps h to H

$\pi' = \pi^{(H+1)}$ and $\pi = \pi^{(1)}$

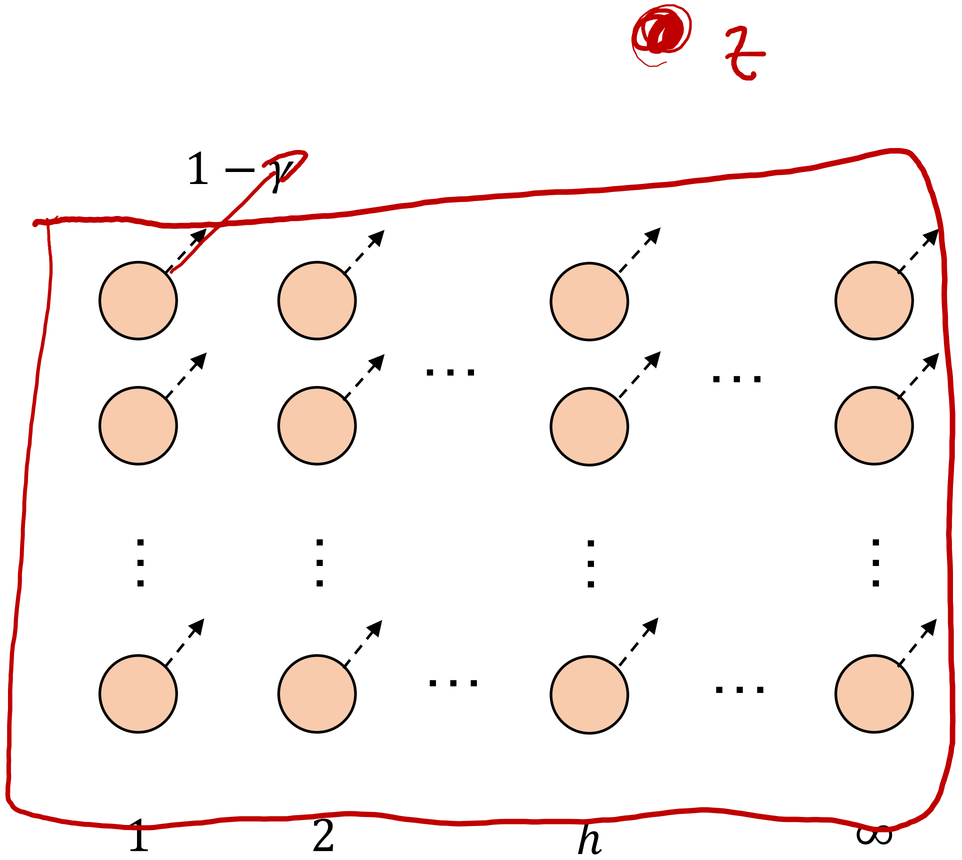


$= \mathbb{E}\left[V^{\pi^{(H+1)}}(s) - V^{\pi^{(1)}}(s)\right]$

$= \mathbb{E}\left[\sum_{h=1}^{H}\left(V^{\pi^{(h+1)}}(s) - V^{\pi^{(h)}}(s)\right)\right]$

$= \mathbb{E}\left[\sum_{h=1}^{H}\sum_{s,a} d_{\rho,h}^{\pi'}(s)\left(\pi'_h(a|s) - \pi_h(a|s)\right) Q_h^{\pi}(s,a)\right]$

Discounted Total Reward Setting



Define Markov policy

$$\pi^{(h)} = \begin{cases} \pi' & \text{in step } 1 \sim h-1 \\ \pi & \text{in step } h \to \infty \end{cases}$$

$$\pi^{(1)} = \pi, \quad \pi^{(\infty)} = \pi'$$

$$\mathbb{E}_{s\sim\rho}\left[V^{\pi'}(s) - V^{\pi}(s)\right] = \mathbb{E}_{s\sim\rho}\left[\sum_{h=1}^{\infty} V^{\pi^{(h+1)}}(s) - V^{\pi^{(h)}}(s)\right]$$

$$= \sum_{h=1}^{\infty}\sum_{s,a} d_{\rho h}^{\pi'}(s)\left(\pi'(a|s) - \pi(a|s)\right)Q^{\pi}(s,a)$$

$$= \sum_{s,a} d_{\rho}^{\pi'}(s)\left(\pi'(a|s) - \pi(a|s)\right)Q^{\pi}(s,a)$$

Performance / Value Difference Lemma

For any two stationary policies π' and π in the discounted total reward setting,

$$\mathbb{E}_{s\sim\rho}\left[V^{\pi'}(s)\right] - \mathbb{E}_{s\sim\rho}[V^{\pi}(s)] = \sum_{s,a} d_\rho^{\pi'}(s)\left(\pi'(a|s) - \pi(a|s)\right)Q^{\pi}(s,a)$$

$$= \sum_{s,a} d_\rho^{\pi'}(s,a)\left(Q^{\pi}(s,a) - V^{\pi}(s)\right)$$

$$= \sum_{s,a} d_\rho^{\pi'}(s)\pi'(a|s)Q^{\pi}(s,a) - \sum_{s} d_\rho^{\pi'}(s) V^{\pi}(s)$$

$$\sum_{s,a} d_\rho^{\pi'}(s)\pi'(a|s)V^{\pi}(s)$$

$$d_\rho^{\pi'}(s)\pi'(a|s)$$

e.g. $\pi'(s) = \arg\max Q^{\pi}(s,a)$

$$-\mathbb{E}_{s\sim}\left[V^{\pi}(s)\right] + \mathbb{E}_{s}\left[V^{\pi'}(s)\right] = \sum_{s,a} d_\rho^{\pi}(s)\left(\pi(a|s) - \pi'(a|s)\right)Q^{\pi'}(s,a)$$

$$\pi'(a|s) - \pi(a|s)$$

$$= \sum_{s,a} d_\rho^{\pi'}(s,a)\, A^{\pi}(s,a)$$

$$A^{\pi}(s,a) = Q^{\pi}(s,a) - \sum_{a'} \pi(a'|s)Q^{\pi}(s,a')$$

Modified Policy Iteration

Bellman Operator $\mathcal{T}^\pi$

$$(\mathcal{T}^\pi V)(s) = \sum_a \pi(a|s)\left(R(s,a) + \gamma \sum_{s'} P(s'|s,a)\,V(s')\right)$$

Greedy Policy Operator $\mathcal{G}$

$$(\mathcal{G}V)(s) = \operatorname*{argmax}_a \left(R(s,a) + \gamma \sum_{s'} P(s'|s,a)\,V(s')\right)$$

Value Iteration:

$$\pi_{k+1} = \mathcal{G}V_k$$
$$V_{k+1} = \mathcal{T}^{\pi_{k+1}} V_k$$

Policy Iteration:

$$\pi_{k+1} = \mathcal{G}V_k$$
$$V_{k+1} = (\mathcal{T}^{\pi_{k+1}})^{\infty} V_k$$

MPI:

$$\pi_{k+1} = \mathcal{G}V_k$$
$$V_{k+1} = (\mathcal{T}^{\pi_{k+1}})^{m} V_k$$

Policy update

Value update

Difference: Relative speed between policy and value updates

Summary for the Basics of MDPs

- MDPs model decision-making problems where the return depends on sequences of actions.
- “State” summarizes all the information needed to make decisions (in the fixed-horizon setting, the step index is also important).
- Interaction Protocols: fixed-horizon, goal-oriented, infinite-horizon
- Performance Metrics: total reward, average reward, discounted total reward
- Policies: history-dependent, Markov, stationary
- While the **number of action sequence is exponential** in the horizon length, the optimal policy can be computed in **poly(#state, #actions, horizon length)** time using dynamic programming techniques (Value Iteration).
- The dynamic programing here is slightly more complicated since it involves infinite horizon and recursive states.
- Bellman equation, Reverse Bellman equation, Bellman optimality equation
- Approximate Bellman optimality → Approximate optimal policy
- Policy Iteration and Performance Difference Lemma
- Unifying Value Iteration and Policy Iteration

Summary for the Basics of MDPs

- MDPs model decision-making problems where the return depends on sequences of actions.
- “State” summarizes all the information needed to make decisions (in the fixed-horizon setting, the step index is also important).
- Interaction Protocols: fixed-horizon, goal-oriented, infinite-horizon
- Performance Metrics: total reward, average reward, discounted total reward
- Policies: history-dependent, Markov, stationary
- While the **number of action sequence is exponential** in the horizon length, the optimal policy can be computed in **poly(#state, #actions, horizon length)** time using dynamic programming techniques (Value Iteration).
- The dynamic programing here is slightly more complicated since it involves infinite horizon and recursive states.
- Bellman equation, Reverse Bellman equation, Bellman optimality equation
- Approximate Bellman optimality → Approximate optimal policy
- Policy Iteration and Performance Difference Lemma
- Unifying Value Iteration and Policy Iteration