Notion of Policy and Optimal Policies

Easwar Subramanian

TCS Innovation Labs, Hyderabad

cs5500.2020@iith.ac.in

August 19, 2023

Overview

Review

Markov Reward Process

Markov Reward Process

A Markov reward process is a tuple $< \mathcal{S}, \mathcal{P}, \mathcal{R}, \gamma >$ is a Markov chain with values

- $\mathcal{S}$: (Finite) set of states
- $\mathcal{P}$: State transition probablity
- $\mathcal{R}$: Reward for being in state s_t is given by a deterministic function $\mathcal{R}$

$$r_{t+1} = \mathcal{R}(s_t)$$

- γ : Discount factor such that $\gamma \in [0, 1]$

- In general, the reward function can also be an expectation $\mathcal{R}(s_t = s) = \mathbb{E}[r_{t+1}|s_t = s]$

No notion of action !

Value Function

The value function $V(s)$ gives the long-term value of state $s \in \mathcal{S}$

$$V(s) = \mathbb{E}\left(G_t | s_t = s\right) = \mathbb{E}\left(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s\right)$$

- Value function $V(s)$ determines the value of being in state s
- $V(s)$ measures the potential future rewards we may get from being in state s
- Observe that $V(s)$ is independent of t

Decomposition of Value Function

Let s and s' be successor states at time steps t and $t+1$, the value function can be decomposed into sum of two parts

- Immediate reward r_{t+1}
- Discounted value of next state s' (i.e. $\gamma V(s')$)

$$
\begin{aligned}
V(s) = \mathbb{E}(G_t | s_t = s) \quad &= \quad \mathbb{E}\left(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s\right) \\
&= \quad \mathbb{E}\left(r_{t+1} + \gamma V(s_{t+1}) | s_t = s\right)
\end{aligned}
$$

Bellman equation for value functions

$$
\boxed{V(s) = \mathbb{E}(r_{t+1} | s_t = s) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'} V(s')}
$$

Bellman Equation in Matrix Form

We have $\mathcal{S} = \{1, 2, \cdots, n\}$ and let $\mathcal{P}$, $\mathcal{R}$ be known. Then one can write the Bellman equation can as,

$$V = \mathcal{R} + \gamma \mathcal{P} V$$

where

$$\begin{bmatrix} V(1) \\ V(2) \\ \vdots \\ V(n) \end{bmatrix} = \begin{bmatrix} \mathcal{R}(1) \\ \mathcal{R}(2) \\ \vdots \\ \mathcal{R}(n) \end{bmatrix} + \gamma \begin{bmatrix} \mathcal{P}_{11} & \mathcal{P}_{12} & \cdots & \mathcal{P}_{1n} \\ \mathcal{P}_{21} & \mathcal{P}_{22} & \cdots & \mathcal{P}_{2n} \\ \vdots & & & \\ \mathcal{P}_{n1} & \mathcal{P}_{n2} & \cdots & \mathcal{P}_{nn} \end{bmatrix} \times \begin{bmatrix} V(1) \\ V(2) \\ \vdots \\ V(n) \end{bmatrix}$$

Solving for V, we get,

$$\boxed{V = (I - \gamma \mathcal{P})^{-1} \mathcal{R}}$$

The discount factor should be $\gamma < 1$ for the inverse to exist

Markov Decision Process

Markov decision process is a tuple $< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma >$ where

- $\mathcal{S}$: (Finite) set of states
- $\mathcal{A}$: (Finite) set of actions
- $\mathcal{P}$: State transition probability

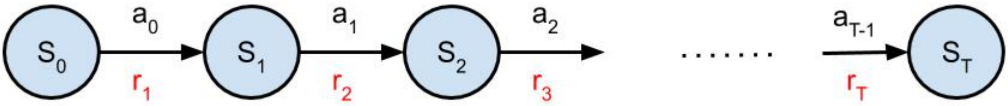
$$\mathcal{P}^a_{ss'} = \mathbb{P}(s_{t+1} = s' | s_t = s, a_t = a), a_t \in \mathcal{A}$$

- $\mathcal{R}$: Reward for taking action a_t at state s_t and transitioning to state s_{t+1} is given by the deterministic function $\mathcal{R}$

$$r_{t+1} = \mathcal{R}(s_t, a_t, s_{t+1})$$

- γ : Discount factor such that $\gamma \in [0, 1]$

Finite and Infinite Horizon MDPs



Depending on time horizon, a Markov decision process can be

- Finite horizon
- Infinite horizon
- Indefinite horizon (SSP)

For finite and (certain) indefinite MDPs with at least absorbing state, we can take the discount factor to be 1

Policy

Policy

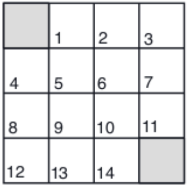
Let π denote a policy that maps state space $\mathcal{S}$ to action space $\mathcal{A}$

Policy

- Deterministic policy: $a = \pi(s), s \in \mathcal{S}, a \in \mathcal{A}$
- Stochastic policy $\pi(a|s) = P[a_t = a|s_t = s]$

Grid World : Revisited

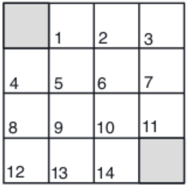
Consider a 4×4 grid world problem



- $\mathcal{S} : \{1, 2, \cdots, 14\}$ (non-terminal) + 2 terminal states (shaded)
- $\mathcal{A} : \{\text{Right, Left, Up, Down}\}$

Figure Source: David Silver's RL course

Grid World : Deterministic Policy



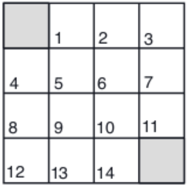
- $\mathcal{S}$: $\{1, 2, \cdots, 14\}$ (non-terminal) + 2 terminal states (shaded)
- $\mathcal{A}$: {Right, Left, Up, Down}
- **Deterministic policy** :

$$\pi(s) = \begin{cases} \text{Down}, & \text{if } s = \{3, 7, 11\} \\ \text{Right}, & \text{Otherwise} \end{cases}$$

- Example sequences : $\{\{8, 9, 10, 11, G\}, \{2, 3, 7, 11, G\}\}$

Figure Source: David Silver's RL course

Grid World : Stochastic Policy



- $\mathcal{S}: \{1, 2, \cdots, 14\}$ (non-terminal) + 2 terminal states (shaded)
- $\mathcal{A}: \{\text{Right, Left, Up, Down}\}$
- **Stochastic policy** : $\pi(a|s)$ could be a uniform random action between all available actions at state s
- Example sequences : $\{\{8, 4, 8, 9, 13, \cdots, \}, \{2, 6, 5, 9, 13, \cdots, \}\}$

Figure Source: David Silver's RL course

Stochastic Policy : Rock Scissors Paper

- Two player game of rock-paper-scissors
  - ★ Scissors beats paper
  - ★ Rock beats scissors
  - ★ Paper beats rock
- Consider policies for iterated rock-paper-scissors
  - ★ A deterministic policy is easily exploited
  - ★ A uniform random policy is optimal (i.e. Nash equilibrium)

Policy Evaluation

Value Functions with Policy

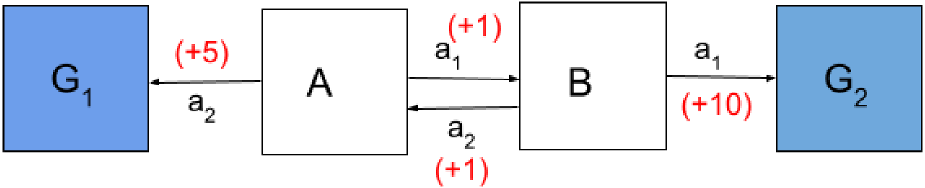
Given a MDP and a policy π, we define the value of a policy as follows :

State-value function

The value function $V^\pi(s)$ in state s is the expected (discounted) total return starting from state s and then following the policy π

$$V^\pi(s) = \mathbb{E}_\pi \left(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s \right)$$

Value Function Computation : Example



- States $\mathcal{S} = \{A, B, G_1, G_2\}$; States G_1 and G_2 are terminal states
- Two actions $\mathcal{A} = \{a_1, a_2\}$
- Value of states $\{A, B\}$ using forward policy π_f is given by, $V_{\pi_f}(A) = 11$, $V_{\pi_f}(B) = 10$
- Value of states $\{A, B\}$ using backward policy π_b is given by, $V_{\pi_b}(B) = 6$, $V_{\pi_b}(A) = 5$

Decomposition of State Value Function

The state-value function can be decomposed into immediate reward plus discounted value of successor state

$$V^\pi(s) = \mathbb{E}_\pi(r_{t+1} + \gamma V^\pi(s_{t+1}) | s_t = s)$$

Expanding the expectation, with $\mathcal{R}^a_{ss'} = \mathcal{R}(s, a, s')$ we get,

$$\mathbb{E}_\pi[r_{t+1} | s_t = s] = \sum_a \pi(a|s) \sum_{s'} \mathcal{P}^a_{ss'} \mathcal{R}^a_{ss'}$$

and

$$\mathbb{E}_\pi[\gamma V^\pi(s_{t+1}) | s_t = s] = \sum_a \pi(a|s) \sum_{s'} \mathcal{P}^a_{ss'} \gamma V^\pi(s')$$

Hence,

$$\boxed{V^\pi(s) = \sum_a \pi(a|s) \sum_{s'} \mathcal{P}^a_{ss'} \left[\mathcal{R}^a_{ss'} + \gamma V^\pi(s')\right]}$$

The above equation is called the Bellman Evaluation operator

Matrix Formulation of Bellman Evaluation Equation

$$V^\pi(s) = \sum_a \pi(a|s) \sum_{s'} \mathcal{P}^a_{ss'} \left[\mathcal{R}^a_{ss'} + \gamma V^\pi(s')\right]$$

Denote,

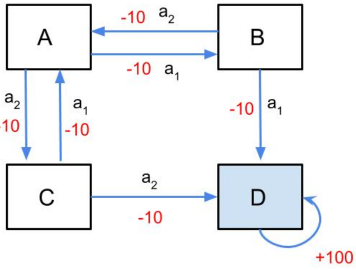
$$\begin{aligned}
\mathcal{P}^\pi(s'|s) &= \sum_{a \in \mathcal{A}} \pi(a|s) \mathcal{P}^a_{ss'} \\
\mathcal{R}^\pi(s) &= \sum_{a \in \mathcal{A}} \pi(a|s) \sum_{s'} \mathcal{P}^a_{ss'} \mathcal{R}^a_{ss'} = \mathbb{E}(r_{t+1}|s_t = s)
\end{aligned}$$

Using $\mathcal{P}^\pi$ and $\mathcal{R}^\pi$, for finite state MDP, one can rewrite the Bellman evaluation equation as

$$V^\pi = \mathcal{R}^\pi + \gamma \mathcal{P}^\pi V^\pi \implies V^\pi = (I - \gamma \mathcal{P}^\pi)^{-1} \mathcal{R}^\pi$$

Remark : Bellman Evaluation Equation for $V^\pi(s)$ is a system of $n = |\mathcal{S}|$ (linear) equations with n variables and can be solved if the model is known

Value Function Computation : Example



- States $\mathcal{S} = \{A, B, C, D\}$; State D is terminal state
- Two actions $\mathcal{A} = \{a_1, a_2\}$
- Stochastic Environment with action chosen succeeding 90% and failing 10%
- Upon failure, agent moves in the direction suggested by the other action

Value Function Computation : Example

- Consider a deterministic policy (π_1) that chooses action a_1 in all states
- Transition matrix corresponding to policy π_1 is given by

$$\begin{array}{cc} & \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{array}{c} A \\ B \\ C \\ D \end{array} & \begin{bmatrix} 0 & 0.9 & 0.1 & 0 \\ 0.1 & 0 & 0 & 0.9 \\ 0.9 & 0 & 0 & 0.1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

- Value of the states under the policy π_1 is given by,
  - ★ $V^{\pi_1}(D) = 100$
  - ★ $V^{\pi_1}(A) = 0.9 * [-10 + V^{\pi_1}(B)] + 0.1 * [-10 + V^{\pi_1}(C)]$
  - ★ $V^{\pi_1}(B) = 0.9 * [-10 + V^{\pi_1}(D)] + 0.1 * [-10 + V^{\pi_1}(A)]$
  - ★ $V^{\pi_1}(C) = 0.9 * [-10 + V^{\pi_1}(A)] + 0.1 * [-10 + V^{\pi_1}(D)]$
- $V^{\pi_1} = \{75.61, 87.56, 68.05, 100\};$

Value Function Computation : Example

- Consider a deterministic policy (π_2) that chooses action a_2 in all states
- Transition matrix corresponding to policy π_2 is given by

$$\begin{array}{c|cccc} & A & B & C & D \\ \hline A & 0 & 0.1 & 0.9 & 0 \\ B & 0.9 & 0 & 0 & 0.1 \\ C & 0.1 & 0 & 0 & 0.9 \\ D & 0 & 0 & 0 & 1 \end{array}$$

- Value of the states under the policy π_2 is given by,
  - ★ $V^{\pi_2}(D) = 100$
  - ★ $V^{\pi_2}(A) = 0.9 * [-10 + V^{\pi_2}(C)] + 0.1 * [-10 + V^{\pi_2}(D)]$
  - ★ $V^{\pi_2}(B) = 0.9 * [-10 + V^{\pi_2}(A)] + 0.1 * [-10 + V^{\pi_2}(D)]$
  - ★ $V^{\pi_2}(C) = 0.9 * [-10 + V^{\pi_2}(D)] + 0.1 * [-10 + V^{\pi_2}(A)]$
- $V^{\pi_2} = \{75.61, 68.05, 87.56, 100\};$

MDP + Policy = MRP

- MDP + policy = Markov Reward Process.
- Given a MDP $< \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma >$ and a policy π
- The MRP is given by $(\mathcal{S}, \mathcal{P}^{\pi}, \mathcal{R}^{\pi}, \gamma)$

Action Value Function

Action Value Function

Action-value function

The action-value function $Q(s, a)$ under policy π is the expected return starting from state s and taking action a and then following the policy π

$$Q^\pi(s, a) = \mathbb{E}_\pi \left(\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s, a_t = a \right)$$

The action-value function can similarly be decomposed as

$$Q^\pi(s, a) = \mathbb{E}_\pi (r_{t+1} + \gamma Q^\pi(s_{t+1}, a_{t+1}) | s_t = s, a_t = a)$$

Expanding the expectation we have $Q^\pi(s, a)$ to be

$$Q^\pi(s, a) = \sum_{s'} \mathcal{P}^a_{ss'} \left[\mathcal{R}^a_{ss'} + \gamma \sum_{a'} \pi(a'|s') Q^\pi(s', a') \right]$$

Relationship between $V^\pi(\cdot)$ and $Q^\pi(\cdot)$

Using definitions of $V^\pi(s)$ and $Q^\pi(s,a)$, we can arrive at the following relationships

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

$$Q^\pi(s,a) = \sum_{s' \in \mathcal{S}} \mathcal{P}^a_{ss'} \left[\mathcal{R}^a_{ss'} + \gamma V^\pi(s') \right]$$

Optimality in Policies

Optimal Policy

Define a partial ordering over policies

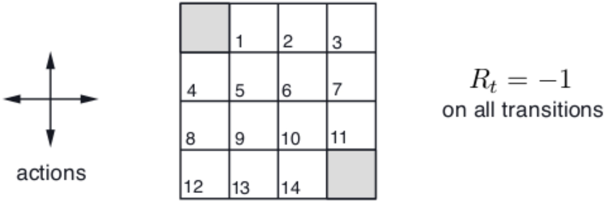
$$\pi \geq \pi', \quad \text{if} \quad V^{\pi}(s) \geq V^{\pi'}(s), \quad \forall s \in \mathcal{S}$$

Theorem

- There exists an optimal policy π_* that is better than or equal to all other policies.
- All optimal policies achieve the optimal value function, $V_*(s) = V^{\pi_*}(s)$
- All optimal policies achieve the optimal action-value function, $Q_*(s,a) = Q^{\pi_*}(s,a)$

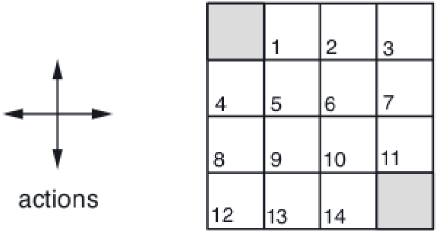
Grid World Problem

Consider a 4×4 grid world problem



- $\mathcal{S}$: $\{1, 2, \cdots, 14\}$ (non-terminal) + 2 terminal states (shaded)
- $\mathcal{A}$: {East, West, North, South}
- $\mathcal{P}$: Upon choosing an action from $\mathcal{A}$, state transitions are deterministic; except the actions that would take the agent off the grid in fact leave the state unchanged
- $\mathcal{R}$: Reward is -1 on all transitions until the terminal state is reached

Grid World Problem

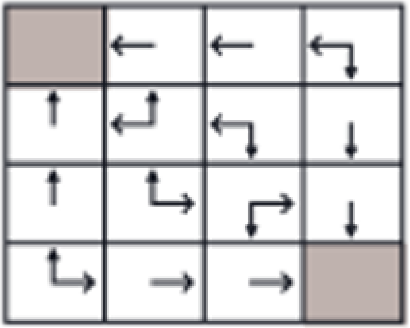


$R_t = -1$
on all transitions

Goal : Reach any of the goal state in as minimum plays as possible

Question : What could be an optimal policy to achieve the above objective ?

Grid World Problem : Optimal Policies



Question : How many optimal policies are there ?

Answer : There are infinite optimal policies (including some deterministic ones)

Solution to an MDP

Solving an MDP means finding a policy π_* as follows

$$\pi_* = \arg\max_{\pi} \left[\mathbb{E}_{\pi} \left(\sum_{t=0}^{\infty} \gamma^t r_{t+1} \right) \right]$$

is **maximum**

- Denote optimal value function $V_*(s) = V^{\pi_*}(s)$
- Denote optimal action value function $Q_*(s,a) = Q^{\pi_*}(s,a)$
- The main goal in RL or solving an MDP means finding an **optimal value function** V_* or **optimal action value function** Q_* or **optimal policy** π_*

Finding an Optimal Policy

Question : Suppose we are given $Q_*(s, a)$. Can we find an optimal policy ?

Answer : An optimal policy can be found by maximising over $Q_*(s, a)$

$$\pi_*(a|s) = \begin{cases} 1 & \text{if } a = \arg\max_{a \in \mathcal{A}} Q_*(s, a) \\ 0 & \text{Otherwise} \end{cases}$$

- If we know $Q_*(s, a)$, we immediately have an optimal policy
- There is always a deterministic optimal policy for any MDP

Greedy (Optimal) Policy

Greedy policy with respect to optimal (action) value function is an optimal policy

An optimal policy can be found by maximising over $Q_*(s, a)$

$$\pi_*(s) = \begin{cases} 1 & \text{if } a = \arg\max_{a \in \mathcal{A}} Q_*(s, a) \\ 0 & \text{Otherwise} \end{cases}$$

Greedy Policy

For a given $Q^{\pi}(\cdot,\cdot)$, define $\pi'(s)$ as follows

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

For a given $V^{\pi}(\cdot)$, define $\pi'(s)$ as follows

$$\pi'(s) = \text{greedy}(V) = \begin{cases} 1 & \text{if } a = \arg\max_{a\in\mathcal{A}} \left[\sum_{s'\in\mathcal{S}} \mathcal{P}_{ss'}^{a} \left(\mathcal{R}_{ss'}^{a} + \gamma V^{\pi}(s')\right)\right] \\ 0 & \text{Otherwise} \end{cases}$$

Relationship between $V_*(\cdot)$ and $Q_*(\cdot,\cdot)$

Question : Suppose we are given $Q_*(s,a), \forall s \in \mathcal{S}$. Can we find $V_*(s)$?

$$V_*(s) = \max_a Q_*(s,a)$$

Question : Suppose we are given $V_*(s), \forall s \in \mathcal{S}$. Can we find $Q_*(s,a)$?

$$Q_*(s,a) = \left[\sum_{s' \in \mathcal{S}} \mathcal{P}^a_{ss'} \left(\mathcal{R}^a_{ss'} + \gamma V_*(s')\right)\right]$$

Exact Methods

Policy Iteration

Question : Is there a way to arrive at π_* starting from an arbitrary policy π ?

Answer : Policy Iteration

- Evaluate the policy π
  - ★ Compute $V^\pi(s) = \mathbb{E}_\pi(r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots | s_t = s)$
- Improve the policy π

$$\pi'(s) = \text{greedy}(V^\pi(s))$$

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

Policy Evaluation

- **Problem** : Evaluate a given policy π
- Compute $V^{\pi}(s) = \mathbb{E}_{\pi}(r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots | s_t = s)$
- **Solution 1** : Solve a system of linear equations using any solver
- **Solution 2** : Iterative application of Bellman Evaluation Equation
- Iterative update rule :

$$V_{k+1}^{\pi}(s) \leftarrow \sum_{a} \pi(a|s) \sum_{s'} \mathcal{P}_{ss'}^{a} \left[\mathcal{R}_{ss'}^{a} + \gamma V_{k}^{\pi}(s')\right]$$

- The sequence of value functions $\{V_1^{\pi}, V_2^{\pi}, \cdots, \}$ converge to V^{π}

Policy Improvement

Suppose we know V^π. How to improve policy π ?

The answer lies in the definition of action value function $Q^\pi(s,a)$. Recall that,

$$\begin{aligned} Q^\pi(s,a) &= \mathbb{E}_\pi\left(\sum_{k=0}^{\infty}\gamma^k r_{t+k+1}|s_t = s, a_t = a\right) \\ &= \mathbb{E}(r_{t+1} + \gamma V^\pi(s_{t+1})|s_t = s, a_t = a) \\ &= \sum_{s'\in\mathcal{S}} \mathcal{P}^a_{ss'}\left[\mathcal{R}^a_{ss'} + \gamma V^\pi(s')\right] \end{aligned}$$

- If $Q^\pi(s,a) > V^\pi(s) \implies$ Better to select action a in state s and thereafter follow the policy π
- This is a special case of the policy improvement theorem

Policy Iteration : Algorithm

Algorithm Policy Iteration

1: Start with an initial policy π_1
2: **for** $i = 1, 2, \cdots, N$ **do**
3: Evaluate $V^{\pi_i}(s) \quad \forall s \in \mathcal{S}$. That is,
4: **for** $k = 1, 2, \cdots, K$ **do**
5: For all $s \in \mathcal{S}$ calculate

$$V_{k+1}^{\pi_i}(s) \leftarrow \sum_a \pi(a|s) \sum_{s'} \mathcal{P}_{ss'}^a \left[\mathcal{R}_{ss'}^a + \gamma V_k^{\pi_i}(s')\right]$$

6: **end for**
7: Perform policy Improvement

$$\pi_{i+1} = \text{greedy}(V^{\pi_i})$$

8: **end for**

Value Iteration

Question : Is there a way to arrive at V_* starting from an arbitrary value function V_0 ?

Answer : Value Iteration

Optimality Equation for State Value Function

Recall the Bellman Evaluation Equation for an MDP with policy π

$$V^{\pi}(s) = \sum_{a} \pi(s, a) \sum_{s'} \mathcal{P}_{ss'}^{a} \left[\mathcal{R}_{ss'}^{a} + \gamma V^{\pi}(s')\right]$$

<u>**Question**</u> : Can we have a recursive formulation for $V_*(s)$?

$$V_*(s) = \max_{a} Q_*(s, a) = \max_{a} \left[\sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^{a} \left(\mathcal{R}_{ss'}^{a} + \gamma V_*(s')\right)\right]$$

Value Iteration : Idea

- Suppose we know the value $V_*(s')$
- Then the solution $V_*(s)$ can be found by one step look ahead

$$V_*(s) \leftarrow \max_a \left[\sum_{s' \in \mathcal{S}} \mathcal{P}^a_{ss'} \left(\mathcal{R}^a_{ss'} + \gamma V_*(s') \right) \right]$$

- Idea of value iteration is to perform the above updates iteratively

Value Iteration : Algorithm

Algorithm Value Iteration

1: Start with an initial value function $V_1(\cdot)$;
2: **for** $k = 1, 2, \cdots, K$ **do**
3:     **for** $s \in \mathcal{S}$ **do**
4:         Calculate

$$V_{k+1}(s) \leftarrow \max_a \left[\sum_{s' \in \mathcal{S}} \mathcal{P}^a_{ss'} \left(\mathcal{R}^a_{ss'} + \gamma V_k(s') \right) \right]$$

5:     **end for**
6: **end for**
