Introduction to Reinforcement Learning based Control and Deep Reinforcement Learning based control

Associate Professor
(Maitre de Conférences)
Automatic Control, Reliability of Systems
mayank-shekhar.jha{at}univ-lorraine.fr

Research
Centre de Recherche en Automatique de Nancy
(CRAN) UMR 7039,
Faculté des Sciences et Technologies
Boulevard des Aiguillettes - BP 70239 - Bât.
1er cycle 54506 Vandoeuvre-lès-Nancy
Cedex France

Teaching
Polytech Nancy (ESSTIN),
2 Rue Jean Lamour 54509
Vandoeuvre-lès-Nancy,
Cedex France

Email: mayank-shekhar.jha [at] univ-lorraine.fr

Markov Decision process

Markov Decision Process (MDP)

Markov process

Markov Reward Process

Markov Decision Process

JHA Mayank , Email: mayank-shekhar.jha [at] univ-lorraine.fr

MDP

- Markov Decision Process (MDP)
  - describes fully an environment for RL.
  - current state characterises the process
  - all essential information is contained by MDP
  - Almost all RL problems can be formulated by MDPs.

Markov Property

Definition: S_t is Markov if and only if:

$$\mathbb{P}(S_{t+1} \mid S_t) = \mathbb{P}[S_{t+1} \mid S_1, \dots S_t]$$

- state describes all the relevent information.
- state is known → history can be thrown away.

Markov Property

Consider s and s'

State transition probability

$$P(s, s') = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s \right]$$

State transition Matrix $\mathcal{P}$

$$\mathcal{P} = \begin{matrix} \text{from} \\ \\ \\ \end{matrix} \begin{bmatrix} P_{11} & \cdots & P_{1n} \\ \vdots & & \\ P_{n1} & & P_{nn} \end{bmatrix} \text{ to}$$

Markov Process

- is a tuple (S, P)
  - $S \rightarrow$ finite set of states
  - $P \rightarrow$ state transition matrix.

Markov Reward Process.

- is a tuple (S, P, R, γ)
  - $R \rightarrow$ reward function $R_S = \mathbb{E}\left[R_{t+1} \mid S_t = S \right]$
  - $\gamma \rightarrow$ is discount factor. $\gamma \in [0, 1]$

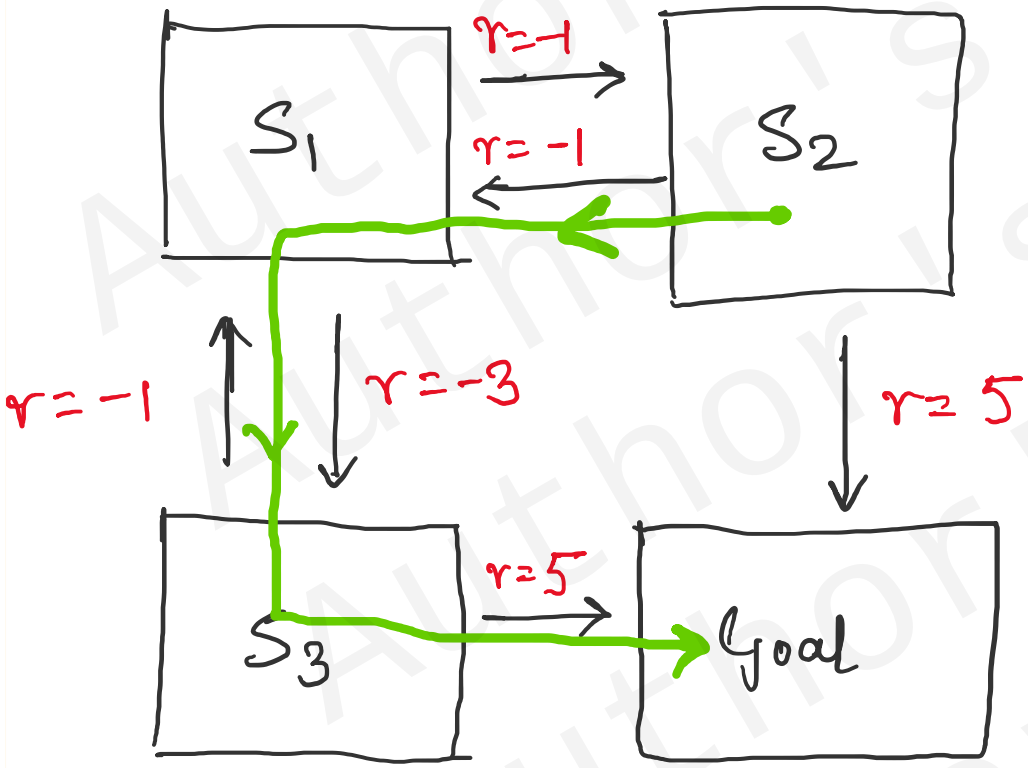
Return

Return G_t ⟶ total discounted reward from time-step 't'.

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

- $\gamma \in [0, 1]$
- Value of reward 'R' after '$k+1$' time-steps is $\gamma^k \cdot R$.
- If
  - $\gamma = 0$ ⟶ near sighted ness
  - $\gamma = 1$ ⟶ far-sighted evaluation.

Grid World Example



$$V_\pi(s) = \mathbb{E}_\pi\left[R_{t+1} + R_{t+2} + R_{t+3} + \cdots \mid S_t = s\right]$$

Assume: A movement from S2 as:

$$V_\pi(S_{2,t}) = \mathbb{E}_\pi\left(-1 + -3 + 5 \mid S_t = S_2\right)$$

$$= (-1 + -3 + 5) = 1$$

→ we consider only one π

if, $\pi = \{\pi_1, \pi_2, \pi_3 \cdots\}$

then average over all π.

Value Function

Value Function

$V(s)$ ⟶ long term value of state 's'.

State value function is expected return starting from state 's'.

$$V(s) = \mathbb{E}\left[G_t \mid S_t = s \right]$$

Value Function

- Value Function can be divided into two parts
  - Immidiate reward R_{t+1}
  - discounted value of successor state $\gamma v(S_{t+1})$

$$v(s) = \mathbb{E}[G_t \mid S_t = s]$$

$$= \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} \cdots \mid S_t = s]$$

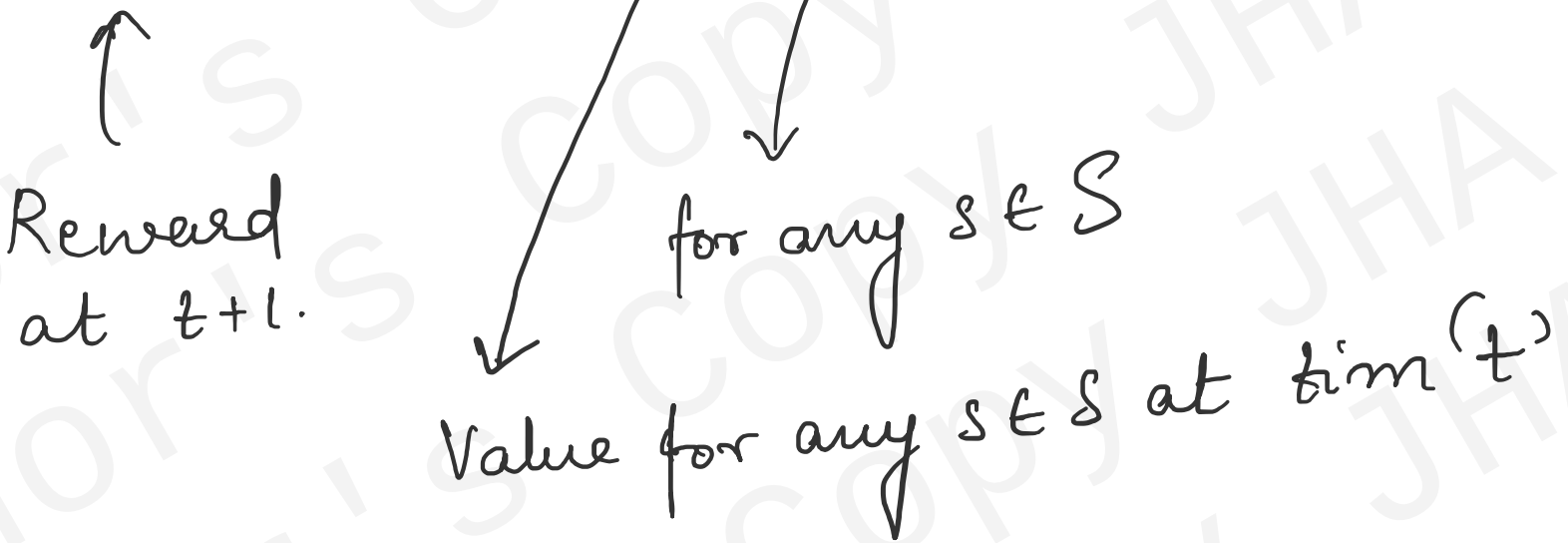
$$= \mathbb{E}[R_{t+1} + \gamma(R_{t+2} + \gamma R_{t+3} + \gamma^2 R_{t+4} \cdots) \mid S_t = s]$$

$$= \mathbb{E}[R_{t+1} + \gamma G_{t+1} \mid S_t = s] = \mathbb{E}[R_{t+1} + \gamma v(S_{t+1}) \mid S_t = s]$$

POLYTECH NANCY · UNIVERSITÉ DE LORRAINE · CRAN
JHA Mayank , Email: mayank-shekhar.jha [at] univ-lorraine.fr

Value Function

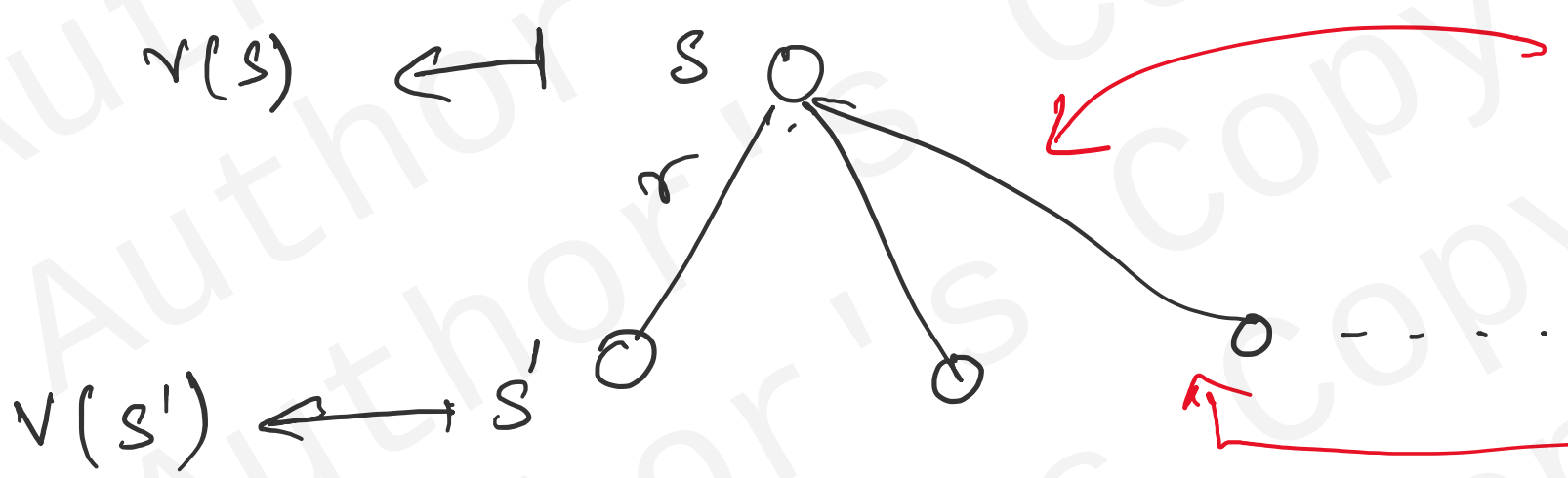
$$V(s) = \mathbb{E}\left[R_{t+1} + \gamma \cdot V(S_{t+1}) \mid S_t = s \right]$$



(This is the Iterative Equation)

But passage from $s \longrightarrow s'$ is a probabilistic process.

(time t) (time $t+1$)



Look ahead process

average all the transitions

$$V(s) = R_s + \gamma \sum_{s \in S} P(s, s') \cdot V(s')$$

Bellman Equation

$$V = R + \gamma \mathcal{P} V \qquad \text{(using matrices)}$$

$$\begin{bmatrix} V(1) \\ \vdots \\ V(n) \end{bmatrix} = \begin{bmatrix} R_1 \\ \vdots \\ R_n \end{bmatrix} + \begin{bmatrix} \mathcal{P}_{11} & \cdots & \mathcal{P}_{1n} \\ \vdots & & \\ \mathcal{P}_{11} & \cdots & \mathcal{P}_{nn} \end{bmatrix} \begin{bmatrix} V(1) \\ \vdots \\ V(n) \end{bmatrix}$$

Markov Decision Process.

Markov process with Decisions!!

(S, A, P, R, γ)

↓

action! = Decision (what to do when in a state to obtain the objective)

Markov Decision Process. (S, A, P, R, γ)

Markov process with Decisions!!

- $S \rightarrow$ set of finite states
- $A \rightarrow$ set of finite actions
- $P \rightarrow$ state transition prob. matrix

$$P_a(s, s') = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s, A_t = a\right]$$

- R is reward function,

$$R_a(s) = \mathbb{E}\left(R_{t+1} \mid S_t = s, A_t = a\right)$$

Policy

Policy

A policy π is a distribution over actions given states:

$$\pi(a|s) = \mathbb{P}[A_t = a \mid S_t = s]$$

→ A policy defines the behaviour of an agent.

→ Policies are stationary

$$A_t \sim \pi(\cdot|S_t), \forall t > 0$$

Policies

- Given an MDP $M = (S, A, P, R, \gamma)$ and a policy π
- State sequence $S_1, S_2, \ldots$ Markov Process (S, P^{π})
- State and reward sequence $(S, P^{\pi}, R^{\pi}, \gamma)$ with respect to policy π is a Markov Reward Process.

$$P^{\pi}(s,s') = \sum_{a \in A} \pi(a|s) P_a(s,s')$$

Average over all possible actions and associated probs.

$$R^{\pi}(s,s') = \sum_{a \in A} \pi(a|s) R_a(s)$$

Average over all possible actions.

→ Given a MDP (with respect to various policies)

↓ given a policy π

MRP.

Value Functions ➔ V-functions and Q-functions

JHA Mayank , Email: mayank-shekhar.jha [at] univ-lorraine.fr

Value Function (V-function and Q-function)

State Value Function (V function)

V - function $V_\pi(s)$ → expected 'return' from state 's' and following policy π

$$V_\pi(s) = \mathbb{E}_\pi [G_t \mid S_t = s]$$

(Remember: $G_t = R_{t+1} + \gamma R_{t+2} + \dots = \sum_{k=0}^{\infty} \gamma^k \cdot R_{t+k+1}$)

Action Value Function (Q-function)

Q-function → expected return starting from state 's', taking action 'a', and follow policy π.

$$q_\pi(s,a) = \mathbb{E}_\pi [G_t \mid S_t = s, A_t = a]$$

V-function can be decomposed → immidiate reward + all future states/rewards

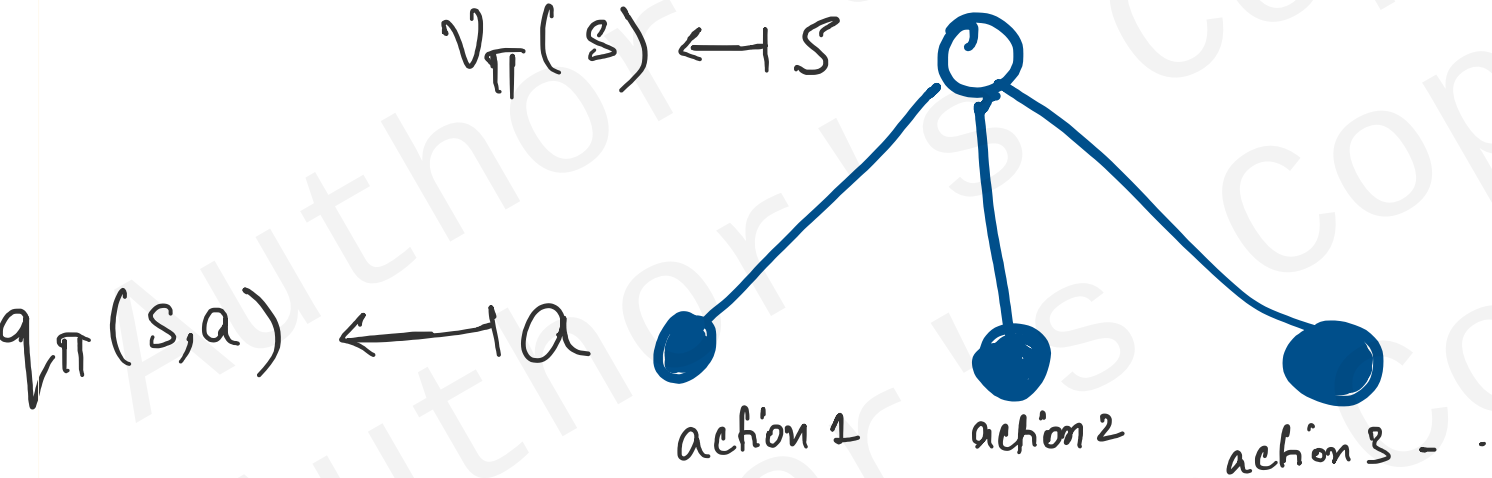
$$v_\pi(s) = \mathbb{E}_\pi\left(R_{t+1} + \gamma\, v_\pi(S_{t+1}) \mid S_t = s\right)$$

Q-function can also be decomposed.

$$q_\pi(s,a) = \mathbb{E}_\pi\left(R_{t+1} + \gamma\, q_\pi(S_{t+1}, A_{t+1}) \mid S_t = s, A_t = a\right)$$

→ take some action from state s
→ get immidiate reward for that action
→ ask Q-value of next state we end up in, under the next action chosen from that time.

How are V-function and Q-function related?



$$v_\pi(s) = \sum_{a \in A} \pi(a|s) \cdot q_\pi(s,a)$$

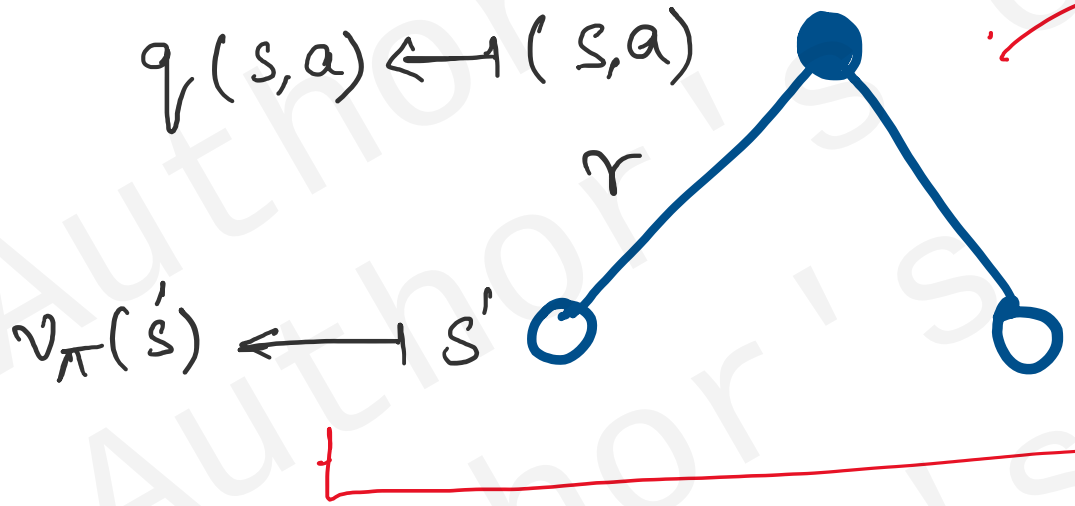
→ from state 's' under policy π

→ three actions taken.

→ each action has action value.

⟹ average over all such values → value $v_\pi(s)$ of being in state 's'. (at top)

How to interpret Q^{π}



→ When in state 's' how good is it to take action 'a'.

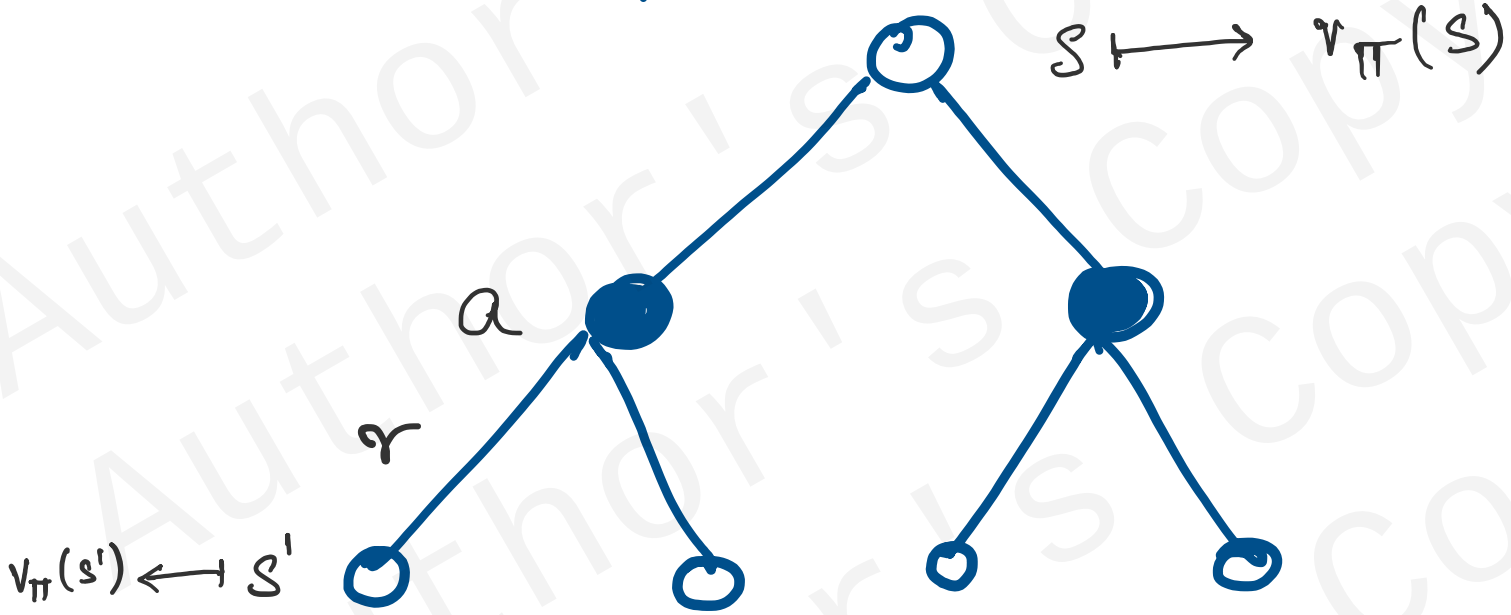
→ after transition with reward 'r'

$$q_\pi(s,a) = R_a(s) + \gamma \sum_{s' \in S} P_a(s,s') \cdot v_\pi(s')$$

how good to be in that state

Bellman Equation

Bellman Equation (Expectation) for v_π



$$v_\pi(s) = \sum_{a \in A} \pi(a|s) \left(R_a(s) + \gamma \sum_{s' \in S} P_a(s,s') \, v_\pi(s') \right)$$

Bellman Equation (Expectation) for V_π

$$V_\pi(s) = \sum_{a \in A} \pi(a|s) \left(R_a(s) + \gamma \sum_{s' \in S} P_a(s,s') V_\pi(s') \right)$$

- $V_\pi(s)$: Value of being in state 's'
- $\sum_{a \in A}$: average over all actions
- $\pi(a|s)$: probability of taking action a in s
- $R_a(s)$: Reward when in s
- $\sum_{s' \in S}$: Average over all states
- $P_a(s,s')$: Prob of being in s' from s
- $V_\pi(s')$: Value of being in s'

Bellman Equation (Expectation) for V_π

$$V_\pi(s) = \sum_{a \in A} \pi(a|s) \left(R_a(s) + \gamma \sum_{s' \in S} P_a(s,s') V_\pi(s') \right)$$

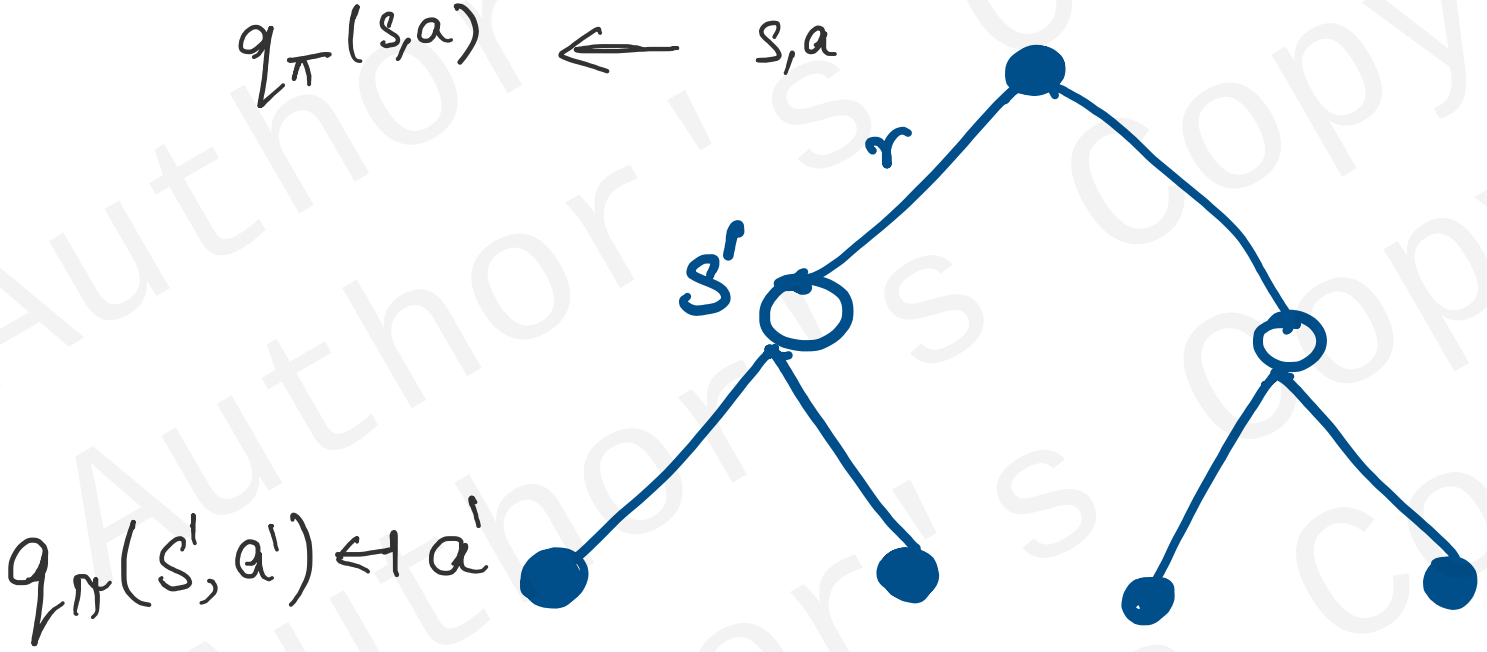
Value of being in state 's' at time 't'

sum over all actions

sum over all states !!

Value of being in state s' at t+1

Bellman Equation for Q-function.



$$q_\pi(s,a) = R_a(s) + \gamma \sum_{s' \in S} P_a(s,s') \sum_{a' \in A} \pi(a'|s') q_\pi(s',a')$$

Bellman Equation for Q-function.

$$q_\pi(s,a) = R_a(s) + \gamma \sum_{s' \in S} P_a(s,s') \sum_{a' \in A} \pi(a'|s') \, q_\pi(s',a')$$

- $q_\pi(s,a)$: Value of being in state s with respect to action a
- $\sum_{s' \in S}$: sum over all states
- $P_a(s,s')$: Probability of passing from s to s'
- $\pi(a'|s')$: Prob of action a' when in state s'
- $q_\pi(s',a')$: Value of being in s' with respect to a'.

$s \rightarrow$ at time t

$s' \rightarrow$ at time $t+1$.

Bellman Eq in Matrix form.

$$V_\pi = R_\pi + \gamma P^\pi \cdot V_\pi$$

with solution

$$V_\pi = (I - \gamma P^\pi)^{-1} \cdot R^\pi$$

$$V = R + \gamma P V \quad \text{(using matrices)}$$

$$\begin{bmatrix} V(1) \\ \vdots \\ V(n) \end{bmatrix} = \begin{bmatrix} R_1 \\ \vdots \\ R_n \end{bmatrix} + \begin{bmatrix} P_{11} & \cdots & P_{1n} \\ \vdots & & \\ P_{11} & \cdots & P_{nn} \end{bmatrix} \begin{bmatrix} V(1) \\ \vdots \\ V(n) \end{bmatrix}$$

If summation over all states be achieved,
if all values are known → this eq can be solved !

Optimal Value Function & Optimal Policy

Optimal Value Function

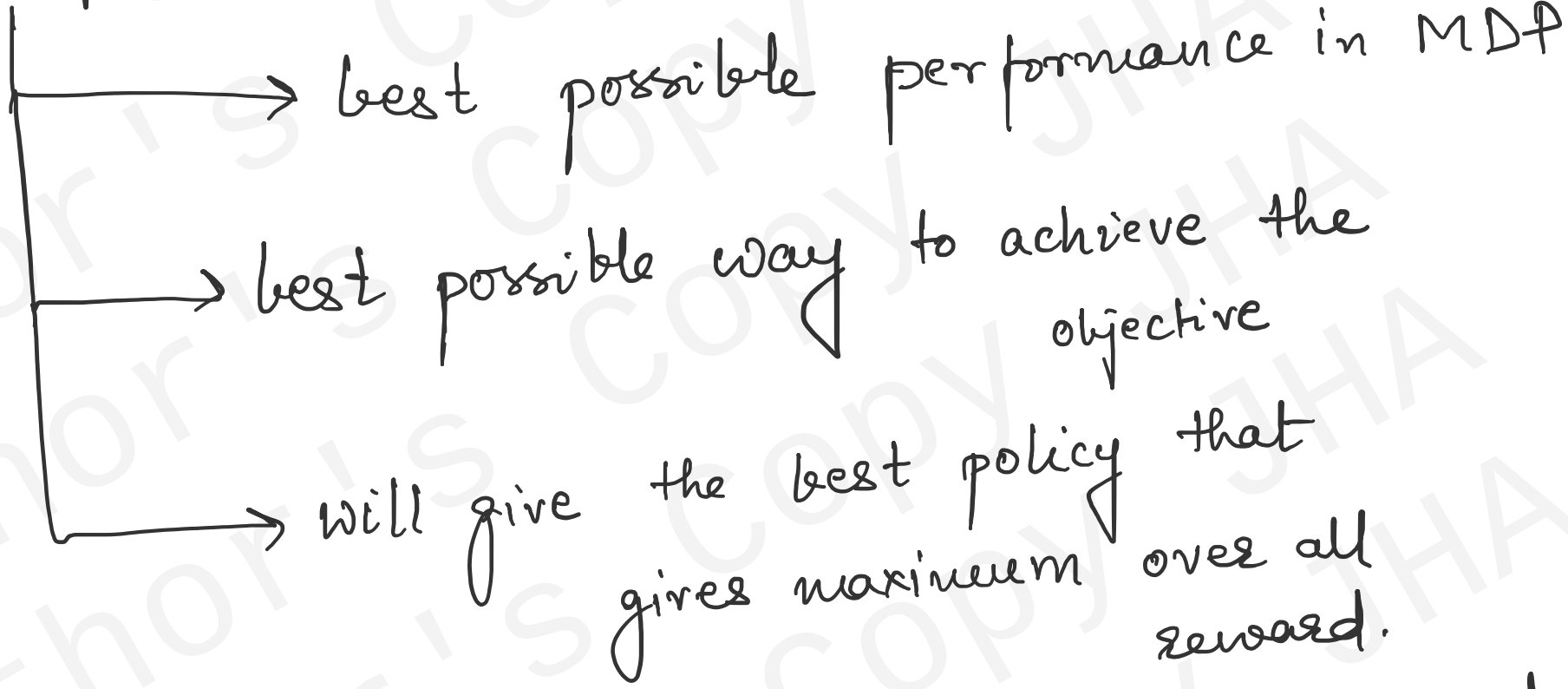
Optimale state-value function $v_*(s)$ → maximum V-function function over all policies.

$$v_*(s) = \max_{\pi} v_\pi(s)$$

Optimale action-value function $q_*(s)$ → max Q-fn over all policies

$$q_*(s,a) = \max_{\pi} q_\pi(s,a)$$

Optimal Value function



MDP is solved when optimal value function is found.

Optimal Policy

Partial Ordering:

$$\pi \geq \pi' \quad \text{if} \quad V_\pi(s) \geq V_{\pi'}(s) \;;\; \forall s$$

Theorem: For any MDP

1. There exists π_* such that $\pi_* \geq \pi \;, \forall \pi$
2. All optimale policy achieve optimal value function (V-fun)
   ie. $V_{\pi_*}(s) = V_*(s)$
3. All optimal policies achieve optimal Q-function (Q-fun)
   $q_{\pi_*}(s,a) = q_*(s,a)$

How to find Optimale Policy ?

By maximising over $q_*(s,a)$

$$\pi_*(a|s) = \begin{cases} 1 & \text{if } a = \underset{a \in A}{\operatorname{argmax}}\ q_*(s,a) \\ 0 & \text{otherwise} \end{cases}$$

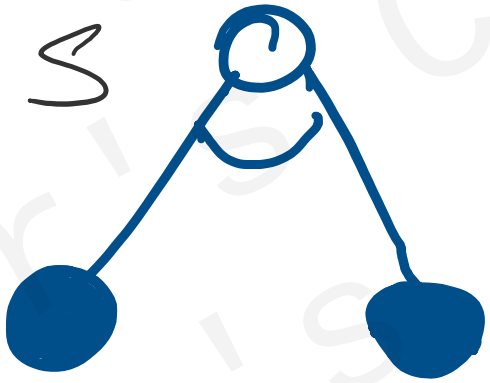
Note!: For any MDP, there is always an optimale policy

If $q_*(s,a)$ is known, optimal policy can be found directly !!

How is V_* and q_* related ?

$V_*(s) \longleftrightarrow s$

$q_*(s,a) \longleftrightarrow a$



From q_*

↓

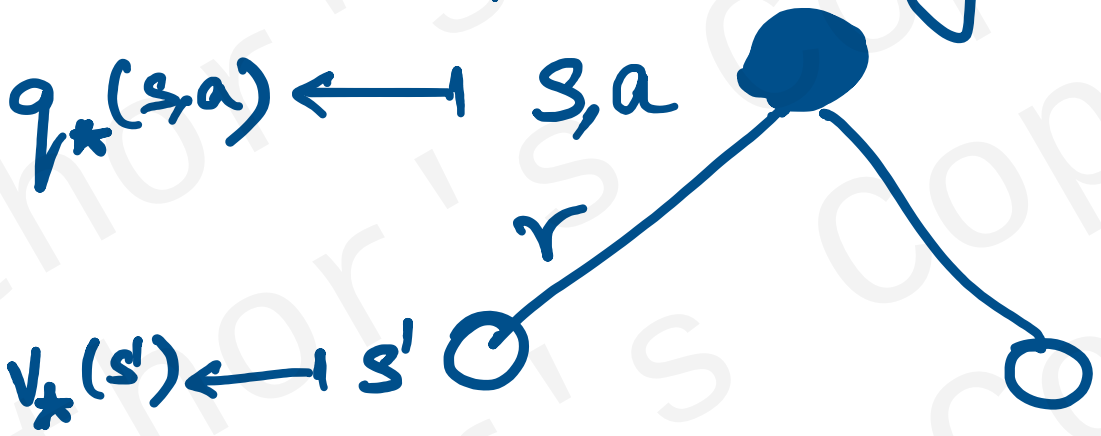
V_*

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

↓

Find max value of $q(s,a)$ for all $a \in A$

Bellman Optimality for $q_*(1)$

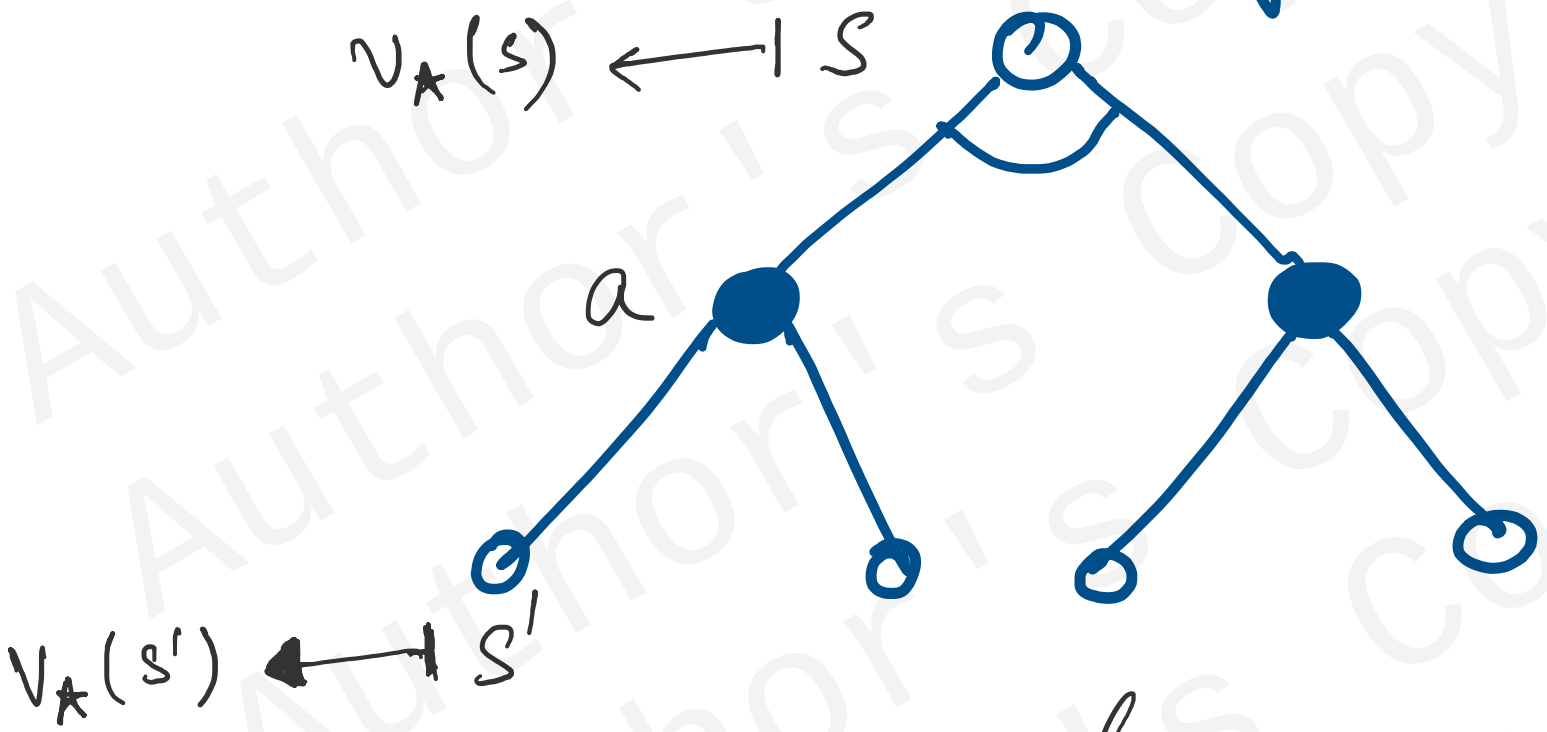


q_* is given as

$$q_*(s,a) = R_a(s) + \gamma \sum_{s' \in S} P_a(s,s')\, v_*(s')$$

Average over all states

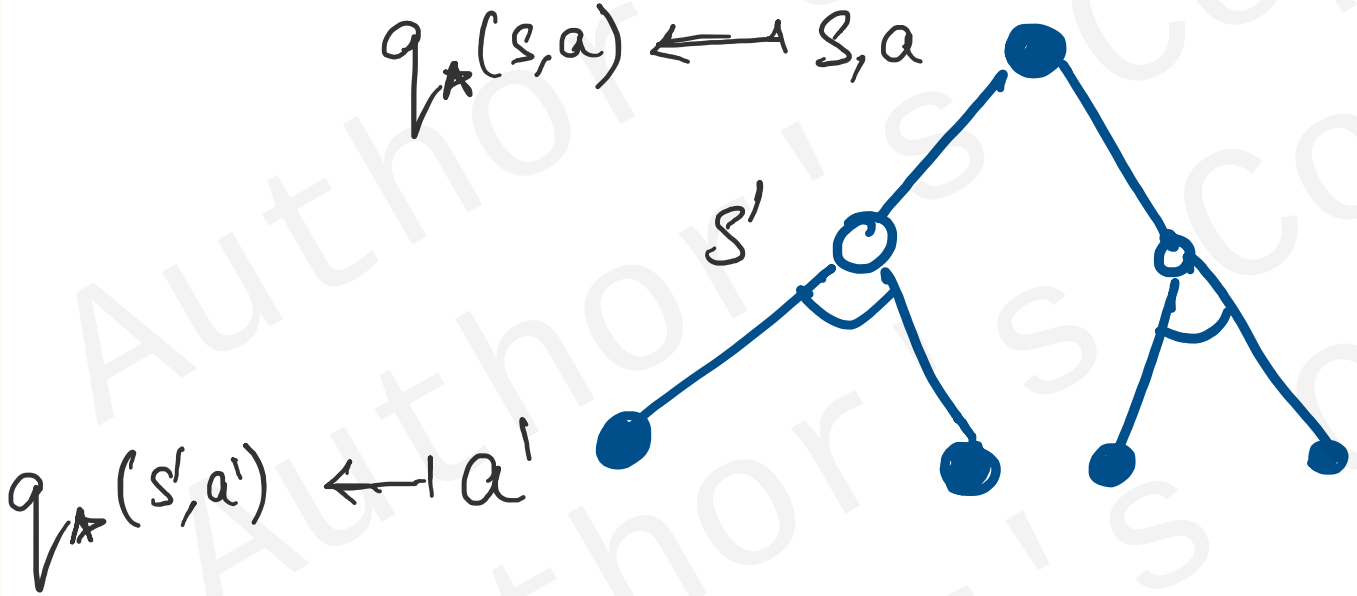
Bellman Optimality for v_*



q_*

$$v_*(s) = \max_a \left(R_a(s) + \gamma \sum_{s' \in S} P_a(s, s') v_*(s') \right)$$

Bellmann Optimality for q_* (2)



Another manner to find q_*

$$q_*(s,a) = R_a(s) + \gamma \sum_{s' \in S} P_a(s,s') \max_{a'} q_*(s',a')$$

So far:

- Value functions : V funs, Q-funs
- Optimality Equations
- Optimal Value Function : $V_*(s)$, $q_*(s)$
- From optimal value function → optimale policy

How to solve optimality equation to find optimal value → optimal policy ?

Bellman Optimality Equation

→ is non-linear

→ no closed form solution

→ Iterative approach required:

① Value Iteration

② Policy Iteration

→ Actor-Critic Methods.

③ Q-learning

④ SARSA.