

Reinforcement Learning (AI644)

Prepared by Dr. Ashraf Elsayed

Lecture 4





مجموعة جامعة
الجميع الذكية
للتعليم القابضة

MIDOCEAN
UNIVERSITY

Reinforcement Learning

Lecture 4

Prepared by
Dr. Ashraf Elsayed

www.midocean.ae

Bellman Equations

The objectives of this lecture is to:

- Derive the Bellman equation for state-value and action-value functions
- Use the Bellman equations to compute value functions
- Define an optimal policy
- Derive the Bellman optimality equation for state-value functions
- Derive the Bellman optimality equation for action-value functions
- Understand how the Bellman optimality equations relate to the previously introduced Bellman equations

Bellman Equation Derivation

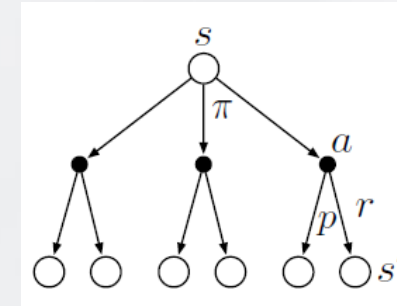
- In reinforce learning *Bellman equations* allows us to relate the value of the current state to the value of future states without waiting to observe all the future rewards.
- We use *Bellman equations* to formalize this connection between the value of a state and its possible successors.
- The *Bellman equation* for the *state value function* defines a relationship between the value of a state and the value of its possible successor states.
- Recall that the *return* is defined as the discounted sum of future rewards

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

State-Value Bellman Equation Derivation

$$v_{\pi}(s) \doteq \mathbb{E}_{\pi} [G_t | S_t = s]$$

$$= \mathbb{E}_{\pi} [R_{t+1} + \gamma G_{t+1} | S_t = s]$$



$$= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r | s, a) \left[r + \gamma \mathbb{E}_{\pi} [G_{t+1} | S_{t+1} = s'] \right]$$

$$= \sum_a \pi(a|s) \sum_{s', r} p(s', r | s, a) \left[r + \gamma v_{\pi}(s') \right], \quad \text{for all } s \in \mathcal{S}$$

- Note that R_{t+1} is a random variable, while the little r represents each possible reward outcome.

↓
The Immediate Reward

- Note how the final expression can be read easily as an expected value. It is really a sum over all values of the three variables, a , s' , and r . For each triple, we compute its probability, $\pi(a|s)p(s', r | s, a)$, weight the quantity in brackets by that probability, then sum over all possibilities to get an expected value.

Action-Value Bellman Equation Derivation

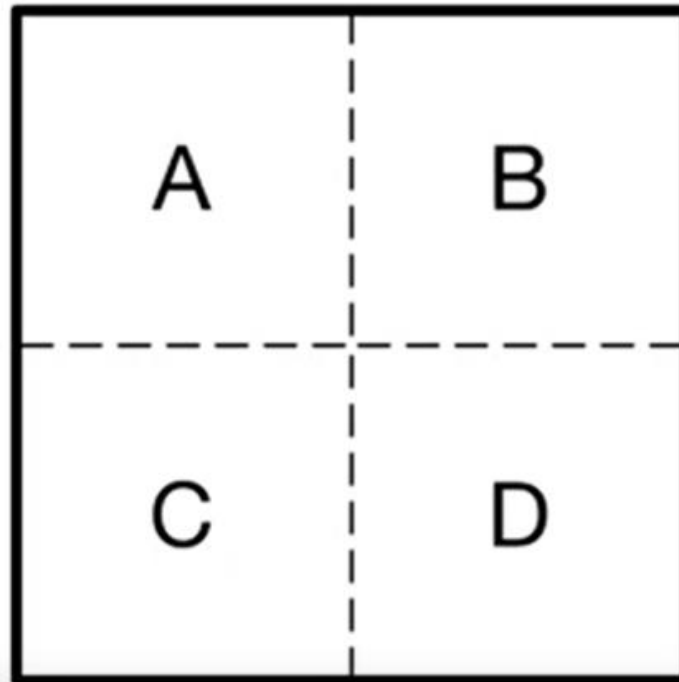
$$\begin{aligned}
 q_{\pi}(s, a) &\doteq \mathbb{E}_{\pi} [G_t | S_t = s, A_t = a] \\
 &= \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma \mathbb{E}_{\pi} [G_{t+1} | S_{t+1} = s']] \\
 &= \sum_{s'} \sum_r p(s', r | s, a) \left[r + \gamma \sum_{a'} \pi(a', s') \mathbb{E}_{\pi} [G_{t+1} | S_{t+1} = s', A_{t+1} = a'] \right] \\
 q_{\pi}(s, a) &= \sum_{s'} \sum_r p(s', r | s, a) \left[r + \gamma \sum_{a'} \pi(a', s') q_{\pi}(s', a') \right]
 \end{aligned}$$

- These equations provide relationships between the values of a state or state action pair and the possible next states or next state action pairs.

Why Bellman Equations?

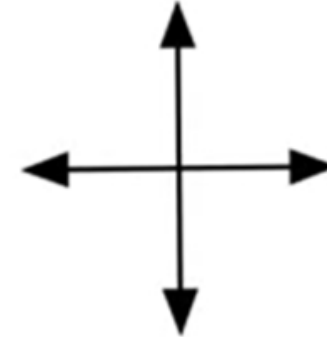
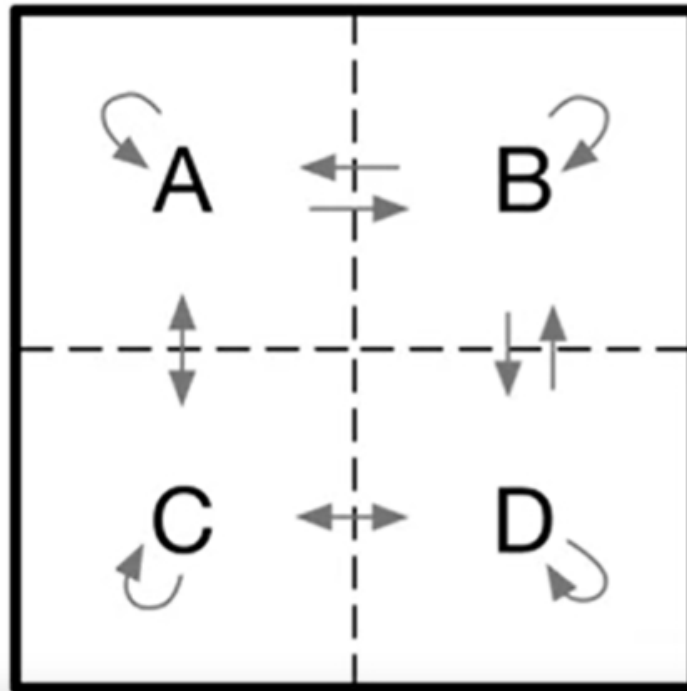
- Consider the following simple Gridworld problem consisting of just four states

Example: Gridworld



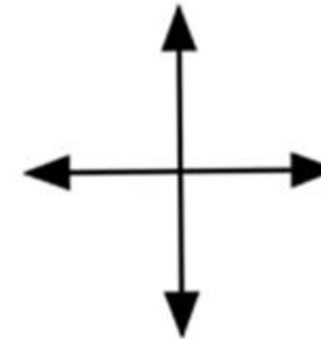
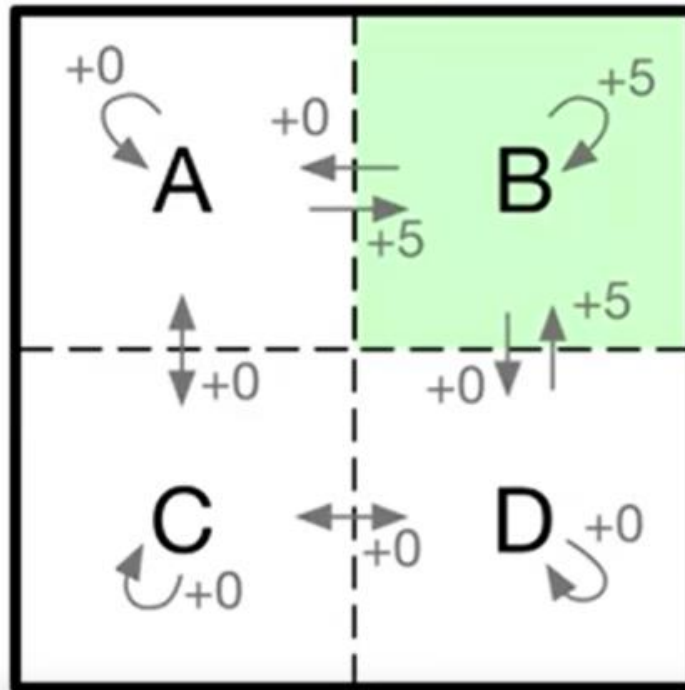
Why Bellman Equations?

Example: Gridworld



Why Bellman Equations?

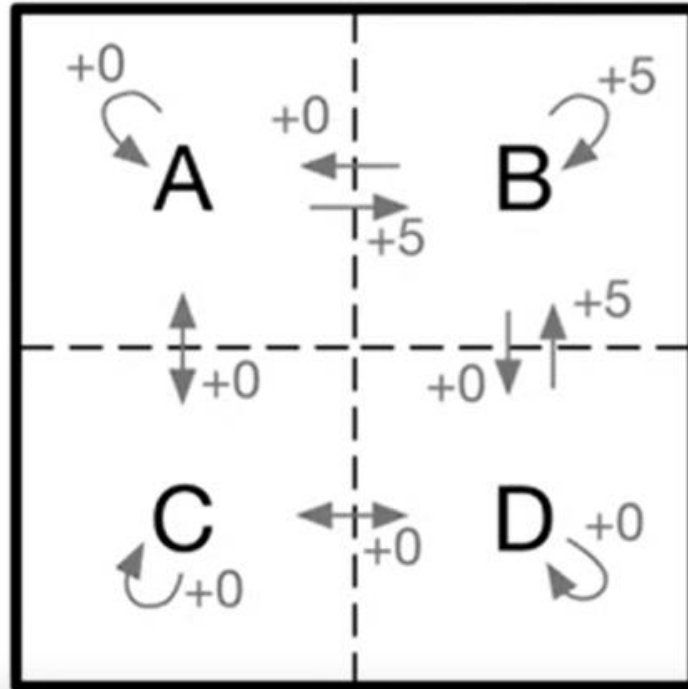
Example: Gridworld



Why Bellman Equations?

Example: Gridworld

$\gamma = 0.7$



Uniform action policy



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

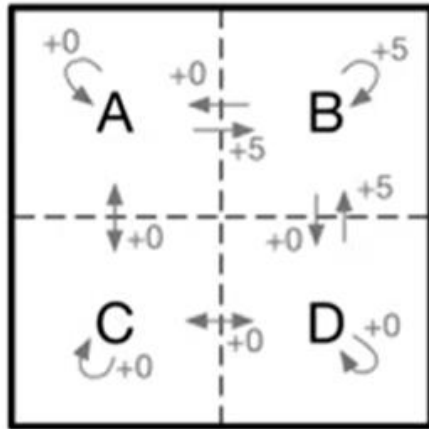
$$V_{\pi}(B) \doteq \mathbb{E}_{\pi}[G_t \mid S_t = B]$$

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

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

Why Bellman Equations?

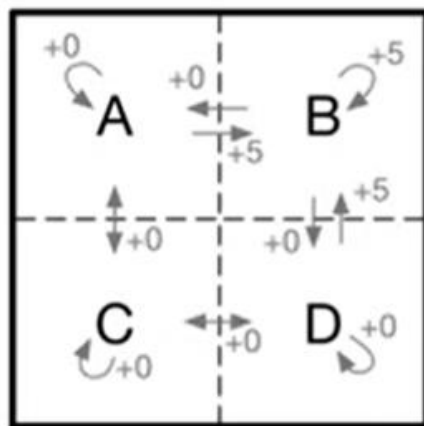
Example: Gridworld



$$V_{\pi}(s) = \sum_a \pi(a | s) \sum_r \sum_{s'} p(s', r | s, a) [r + \gamma V_{\pi}(s')]$$

Why Bellman Equations?

Example: Gridworld

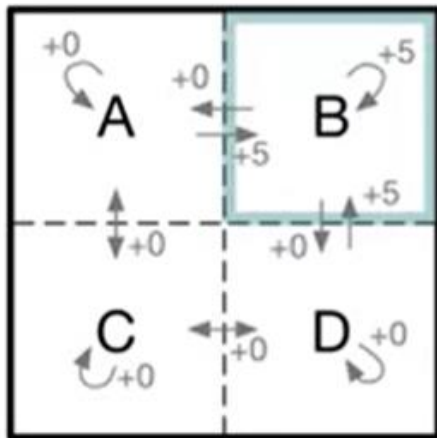


$$V_{\pi}(s) = \sum_a \pi(a | s) \sum_r \sum_{s'} p(s', r | s, a) [r + \gamma V_{\pi}(s')]$$

$$V_{\pi}(A) = \sum_a \pi(a | A) (r + 0.7 V_{\pi}(s'))$$

Why Bellman Equations?

Example: Gridworld



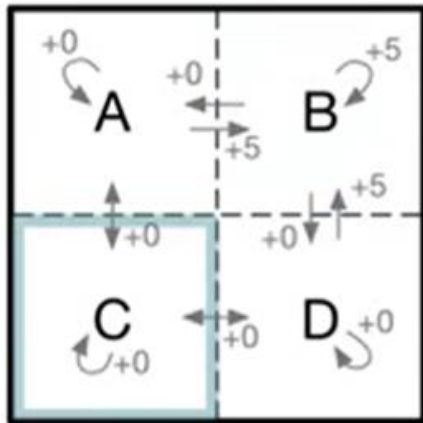
$$V_{\pi}(s) = \sum_a \pi(a | s) \sum_r \sum_{s'} p(s', r | s, a) [r + \gamma V_{\pi}(s')]$$

$$V_{\pi}(A) = \sum_a \pi(a | A) (r + 0.7 V_{\pi}(s'))$$

$$V_{\pi}(A) = \frac{1}{4} (5 + 0.7 V_{\pi}(B))$$

Why Bellman Equations?

Example: Gridworld



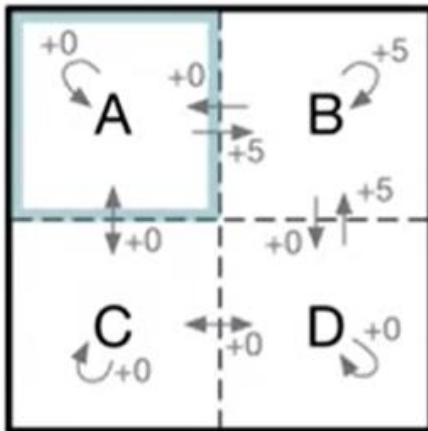
$$V_{\pi}(s) = \sum_a \pi(a | s) \sum_r \sum_{s'} p(s', r | s, a) [r + \gamma V_{\pi}(s')]$$

$$V_{\pi}(A) = \sum_a \pi(a | A) (r + 0.7 V_{\pi}(s'))$$

$$V_{\pi}(A) = \frac{1}{4}(5 + 0.7 V_{\pi}(B)) + \frac{1}{4}0.7 V_{\pi}(C)$$

Why Bellman Equations?

Example: Gridworld



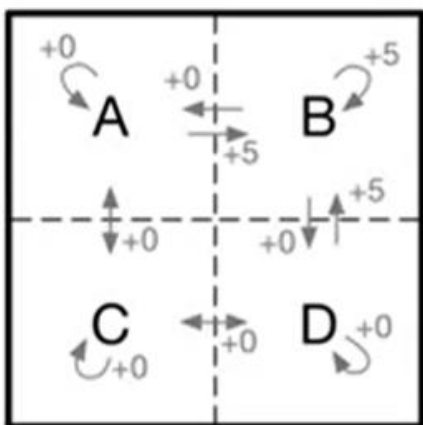
$$V_{\pi}(s) = \sum_a \pi(a | s) \sum_r \sum_{s'} p(s', r | s, a) [r + \gamma V_{\pi}(s')]$$

$$V_{\pi}(A) = \sum_a \pi(a | A) (r + 0.7 V_{\pi}(s'))$$

$$V_{\pi}(A) = \frac{1}{4}(5 + 0.7 V_{\pi}(B)) + \frac{1}{4}0.7 V_{\pi}(C) + \frac{1}{2}0.7 V_{\pi}(A)$$

Why Bellman Equations?

Example: Gridworld



$$V_{\pi}(s) = \sum_a \pi(a | s) \sum_r \sum_{s'} p(s', r | s, a) [r + \gamma V_{\pi}(s')]$$

$$V_{\pi}(A) = \frac{1}{4}(5 + 0.7V_{\pi}(B)) + \frac{1}{4}0.7V_{\pi}(C) + \frac{1}{2}0.7V_{\pi}(A)$$

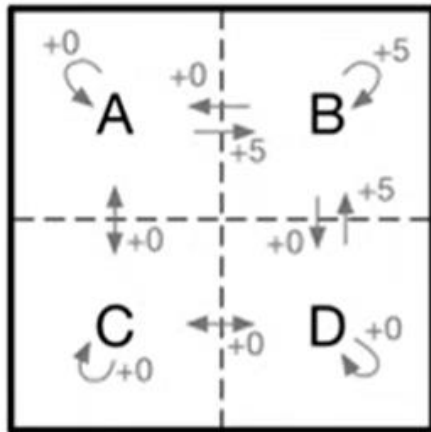
$$V_{\pi}(B) = \frac{1}{2}(5 + 0.7V_{\pi}(B)) + \frac{1}{4}0.7V_{\pi}(A) + \frac{1}{4}0.7V_{\pi}(D)$$

$$V_{\pi}(C) = \frac{1}{4}0.7V_{\pi}(A) + \frac{1}{4}0.7V_{\pi}(D) + \frac{1}{2}0.7V_{\pi}(C)$$

$$V_{\pi}(D) = \frac{1}{4}(5 + 0.7V_{\pi}(B)) + \frac{1}{4}0.7V_{\pi}(C) + \frac{1}{2}0.7V_{\pi}(D)$$

Why Bellman Equations?

Example: Gridworld



$$V_{\pi}(s) = \sum_a \pi(a | s) \sum_r \sum_{s'} p(s', r | s, a) [r + \gamma V_{\pi}(s')]$$

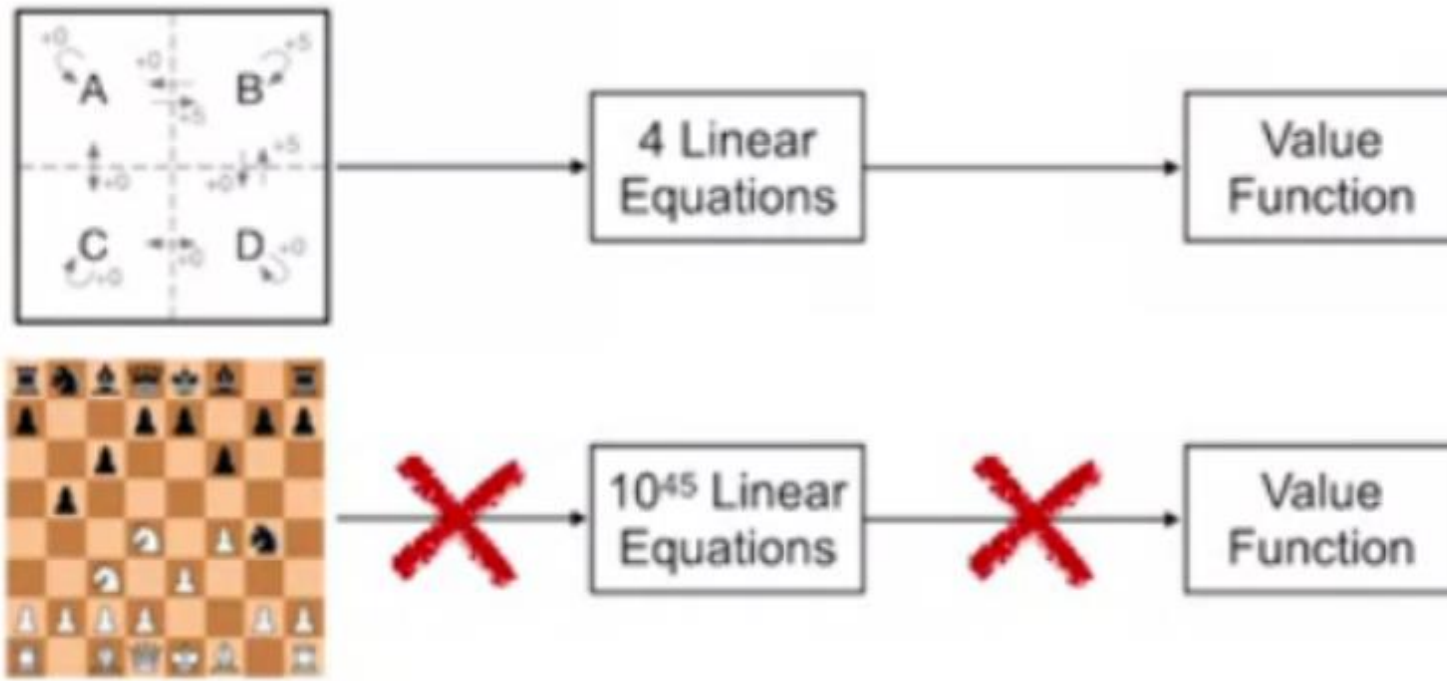
$$V_{\pi}(A) = 4.2$$

$$V_{\pi}(B) = 6.1$$

$$V_{\pi}(C) = 2.2$$

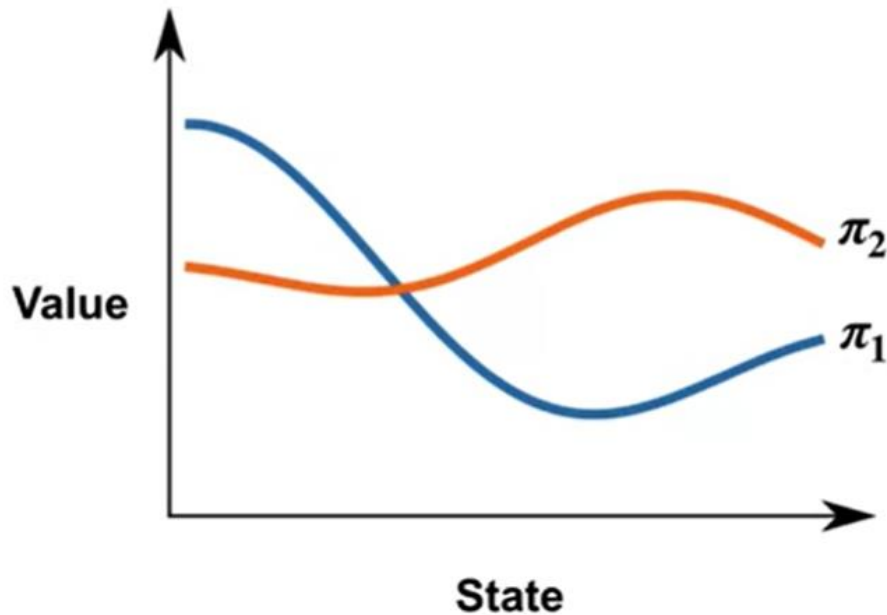
$$V_{\pi}(D) = 4.2$$

Solve only small MDP



Optimal Policies

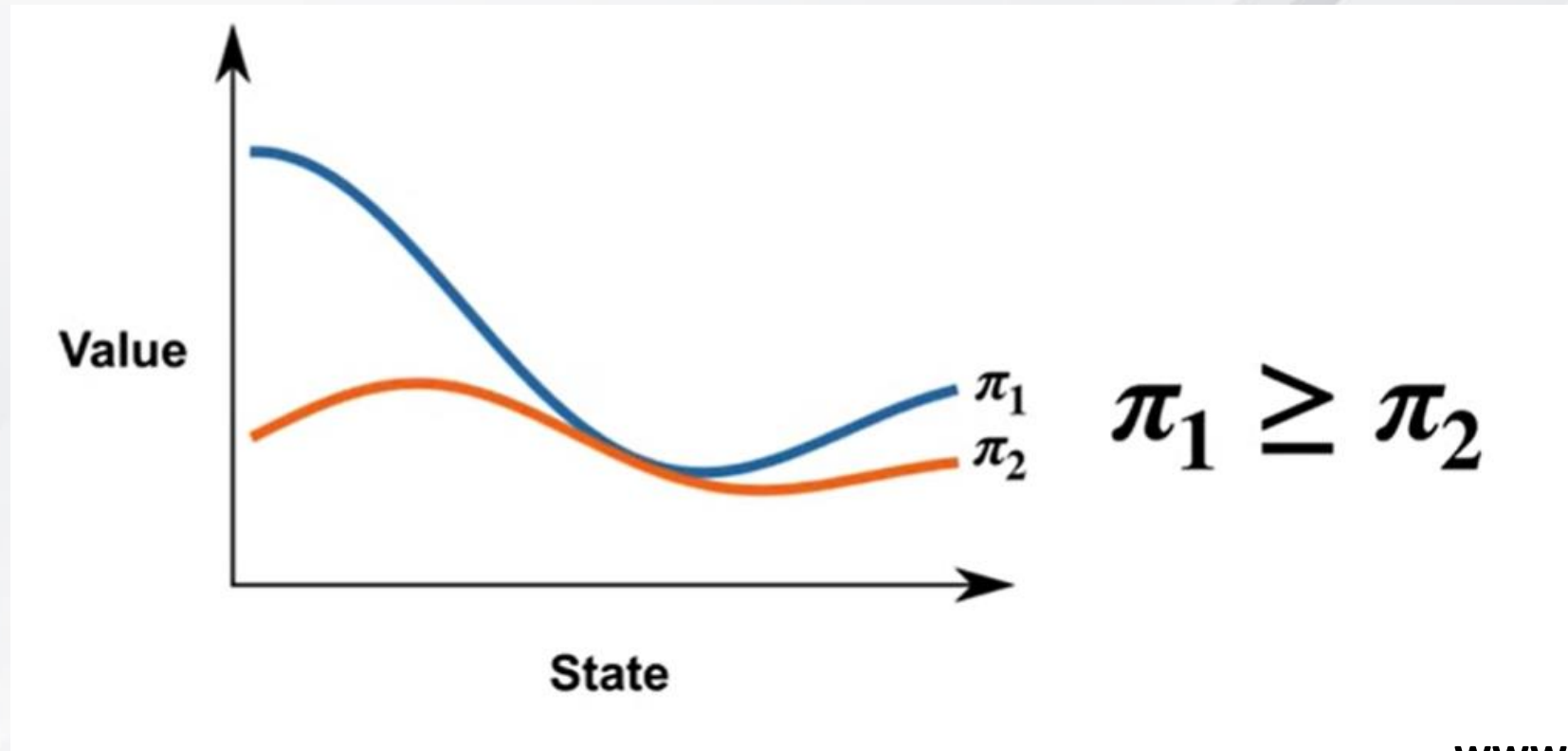
- In order to define an optimal policy, we need first to understand how to compare between two policies.



Here both policies are incomparable, since in some states π_1 achieves better values than π_2 while in other states π_2 achieves better values than π_1 .

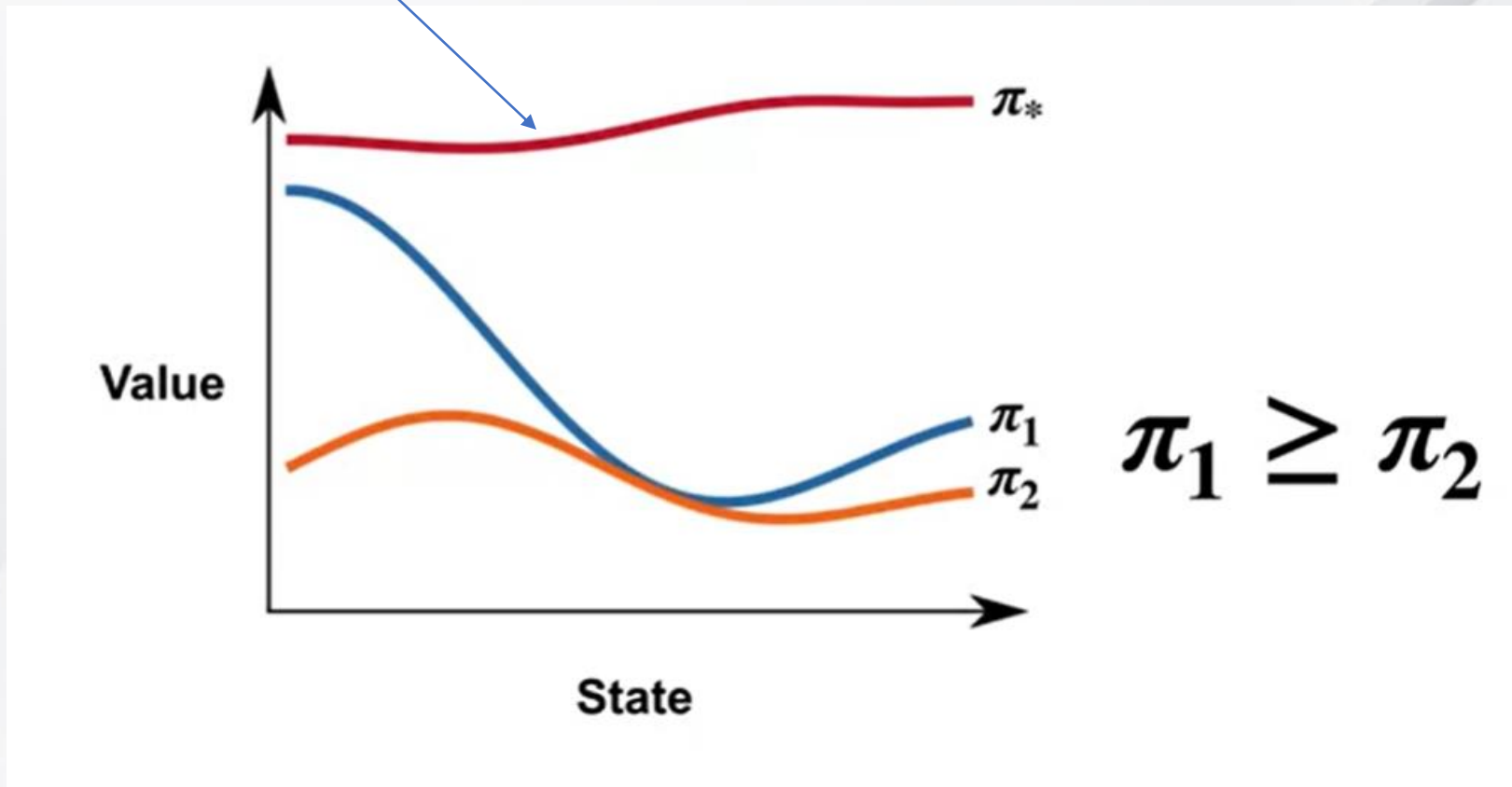
Optimal Policies

- In this example we can say that both policies are comparable and $\pi_1 \geq \pi_2$. This means that π_1 is as good as or better than π_2 if and only if the value under π_1 is greater than or equal the value under π_2 for every state.



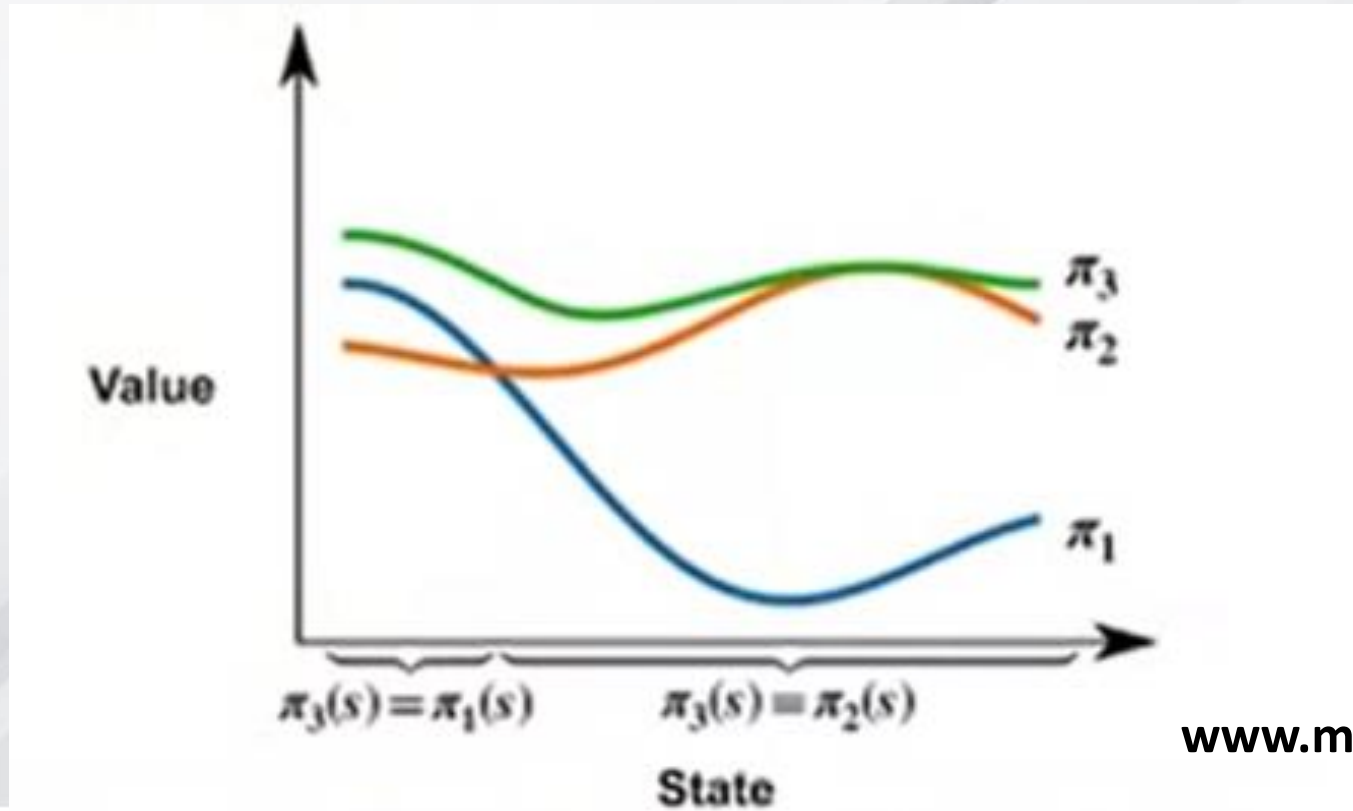
Optimal Policies

- An *optimal policy* is the policy that is as good as or better than all other policies.

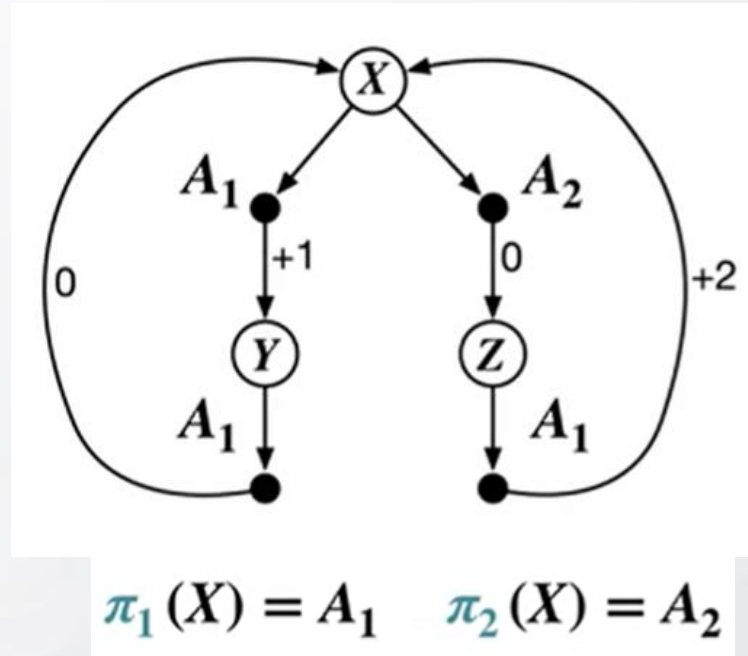


Optimal Policies

- There is always at least one *optimal policy* exists.
- Assume that there is a policy π_1 that is doing better in some states and a policy π_2 that is doing better in other states then we combine them into a policy π_3 that is doing better in all states.

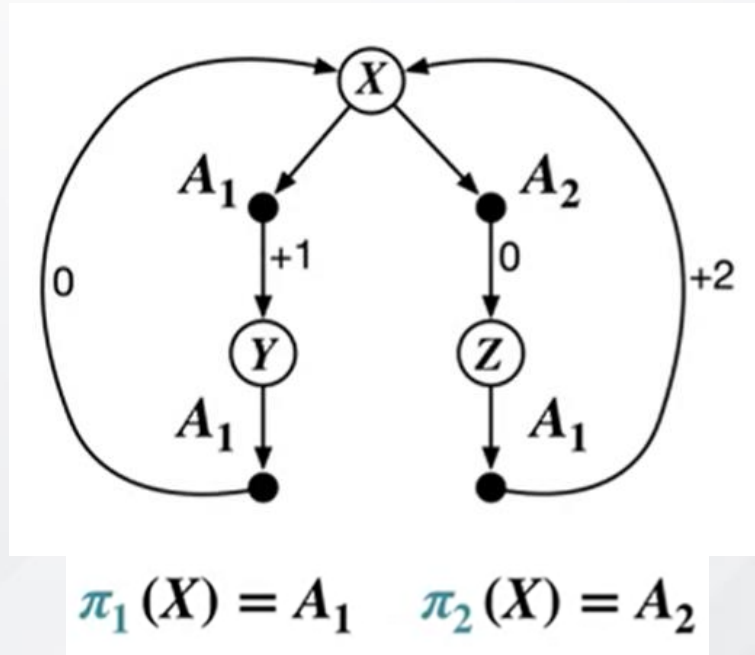


Example (Optimal Policies)



Which of these two policies is optimal?

Example (Optimal Policies)



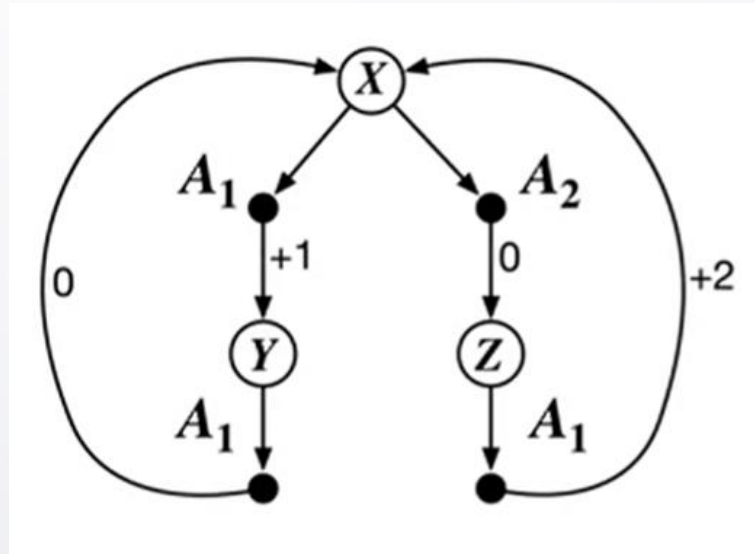
$$\gamma = 0$$

$$v_{\pi_1}(X) = 1 \quad \checkmark$$

$$v_{\pi_2}(X) = 0$$

In this case π_1 is the optimal policy since $\gamma=0$

Example (Optimal Policies)



$$\pi_1(X) = A_1 \quad \pi_2(X) = A_2$$

$$\gamma = 0.9$$

$$v_{\pi_1}(X) = 1 + 0.9 * 0 + (0.9)^2 * 1 + \dots$$

$$= \sum_{k=0}^{\infty} (0.9)^{2k} = \frac{1}{1 - 0.9^2} \approx 5.3$$

$$v_{\pi_2}(X) = 0 + 0.9 * 2 + (0.9)^2 * 0 + \dots$$

$$= \sum_{k=0}^{\infty} (0.9)^{2k+1} * 2 = \frac{0.9}{1 - 0.9^2} * 2 \approx 9.5$$



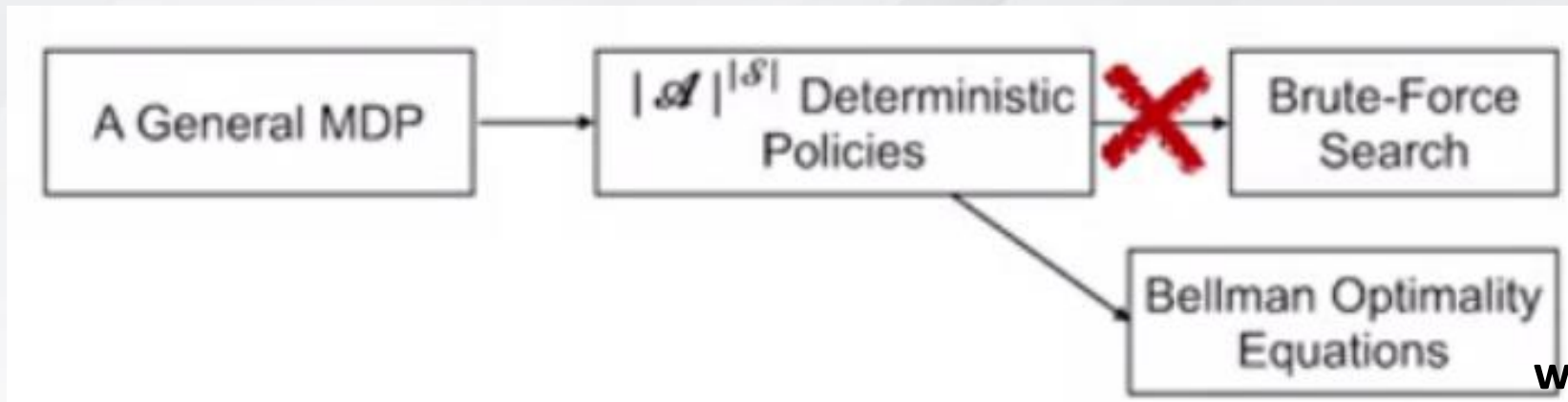
In this case π_2 is the optimal policy

Optimal Policies in MDP

- We can only directly solve small MDPs.



- In general, for large MDPs it is intractable to solve using Brute-Force search.



Optimal Value Functions

- An optimal policy is one which is as good as or better than every other policy.
- The value function for the optimal policy thus has the greatest value possible in every state.

v_*

$$v_{\pi_*}(s) \doteq \mathbb{E}_{\pi_*}[G_t \mid S_t = s] = \max_{\pi} v_{\pi}(s) \text{ for all } s \in \mathcal{S}$$

q_*

$$q_{\pi_*}(s, a) = \max_{\pi} q_{\pi}(s, a) \text{ for all } s \in \mathcal{S} \text{ and } a \in \mathcal{A}$$

Bellman optimally equation

$$v_*(s) = \sum_a \pi_*(a|s) \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$v_*(s) = \max_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

Bellman Optimality Equation for v_*

Bellman optimally equation

$$q_*(s, a) = \sum_{s'} \sum_r p(s', r | s, a) \left[r + \gamma \sum_{a'} \pi_*(a', s') q_*(s', a') \right]$$

$$q_*(s, a) = \sum_{s'} \sum_r p(s', r | s, a) \left[r + \gamma \max_{a'} \pi_*(a', s') q_*(s', a') \right]$$

Bellman Optimality Equation for q_*

- The Bellman's optimality equation gives us a similar system of equations for the optimal value.
- One natural question is, can we solve this system in a similar way to find the optimal state-value function? Unfortunately, the answer is no.
- Taking the maximum over actions is not a linear operation. So standard techniques from linear algebra for solving linear systems won't apply

$$v_{\pi}(s) = \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a) [r + \gamma v_{\pi}(s')]$$

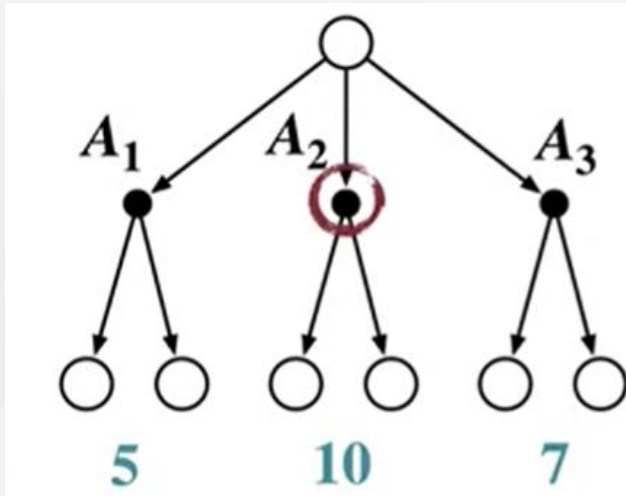


$$v_{*}(s) = \max_a \sum_{s'} \sum_r p(s', r|s, a) [r + \gamma v_{*}(s')] \rightarrow \text{the term } \max_a \text{ is not linear}$$



Using Optimal Value Functions to Get Optimal Policies

- In general, having v_* makes it relatively easy to work out the optimal policy as long as we also have access to the dynamics function p .
- For any state, we can look at each available action and evaluate the boxed term.
- There will be some action for which this term obtains a maximum.



$$v_*(s) = \max_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

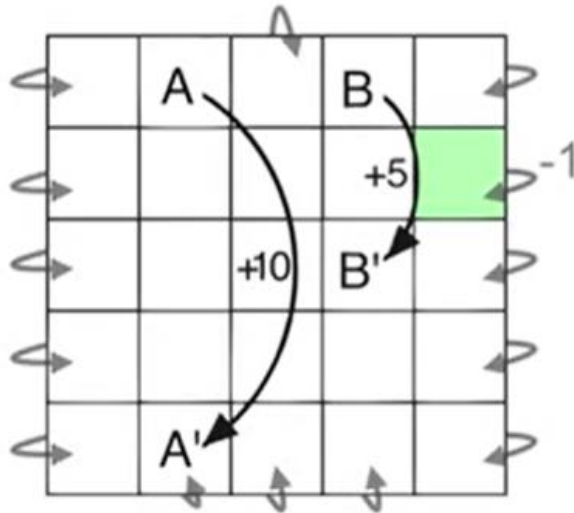
$$\pi_*(s) = \operatorname{argmax}_a \boxed{\sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]}$$

Look ahead

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$\gamma = 0.9$$



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

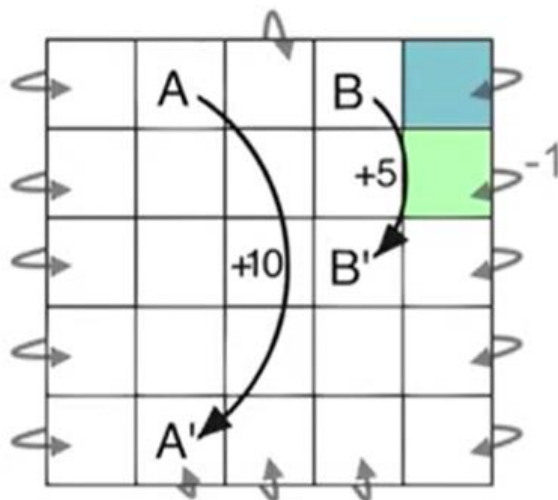
π_*

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$\gamma = 0.9$$

$$0 + 0.9 * 17.5 = 14.0$$



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

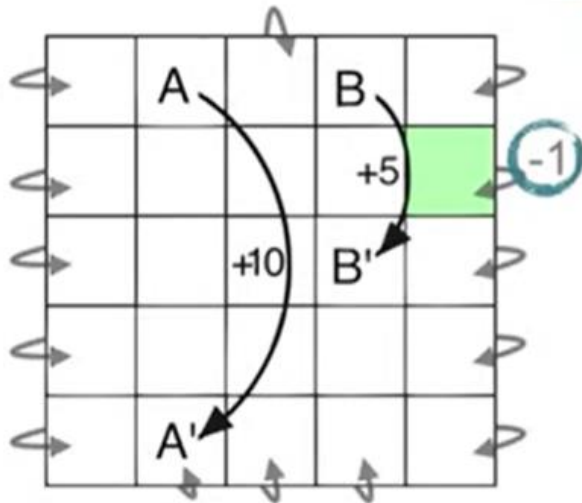
π_*

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$\gamma = 0.9$

$$-1 + 0.9 * 16.0 = 13.4$$



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

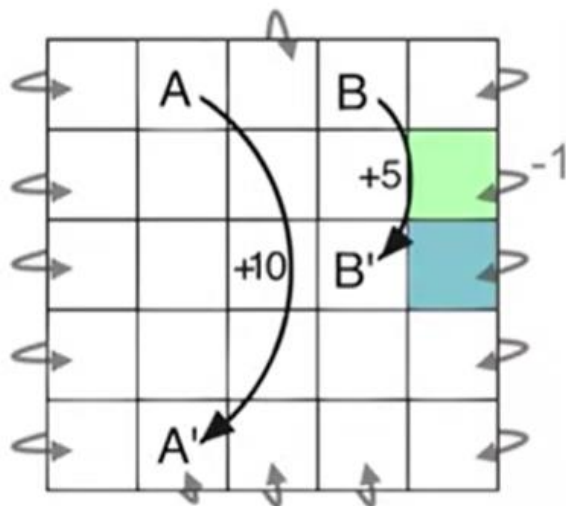
π_*

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$\gamma = 0.9$$

$$0 + 0.9 * 14.4 = 13.0$$



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

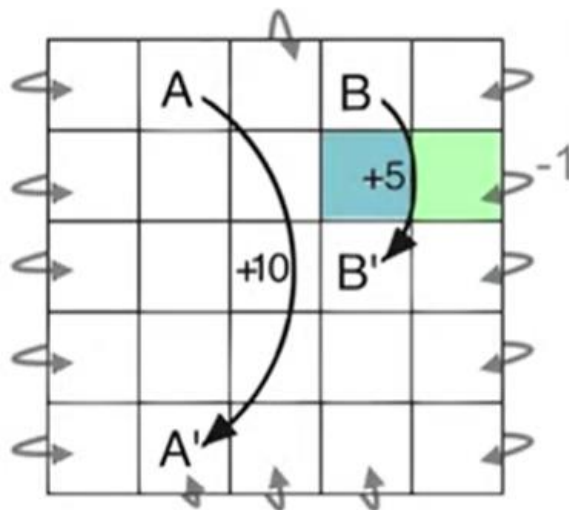
π_*

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$\gamma = 0.9$$

$$0 + 0.9 * 17.8 = 16.0$$



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

V_*

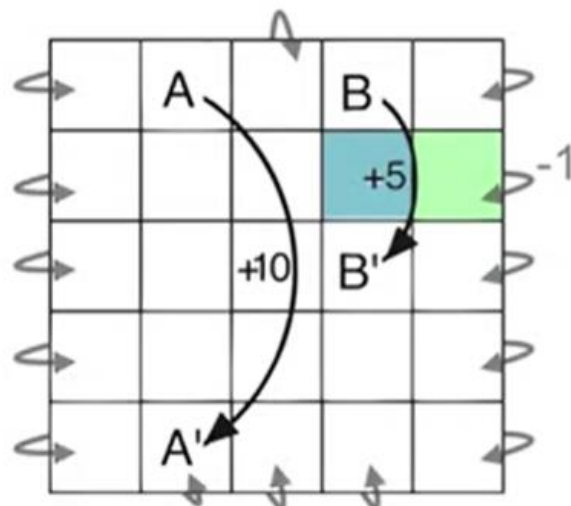
π_*

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$\gamma = 0.9$$

$$0 + 0.9 * 17.8 = 16.0$$



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

				←

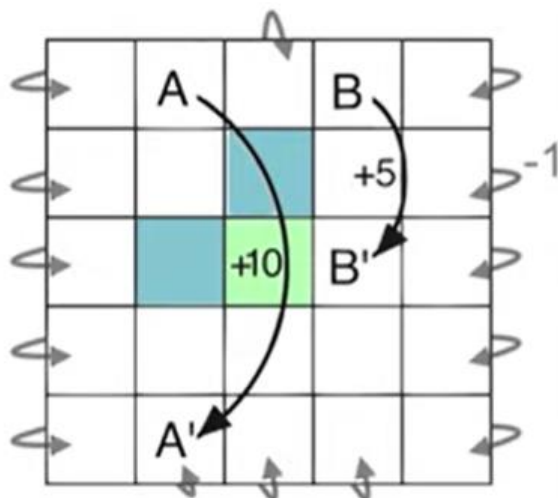
π_*

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$\gamma = 0.9$$

$$0 + 0.9 * 19.8 = 17.8$$



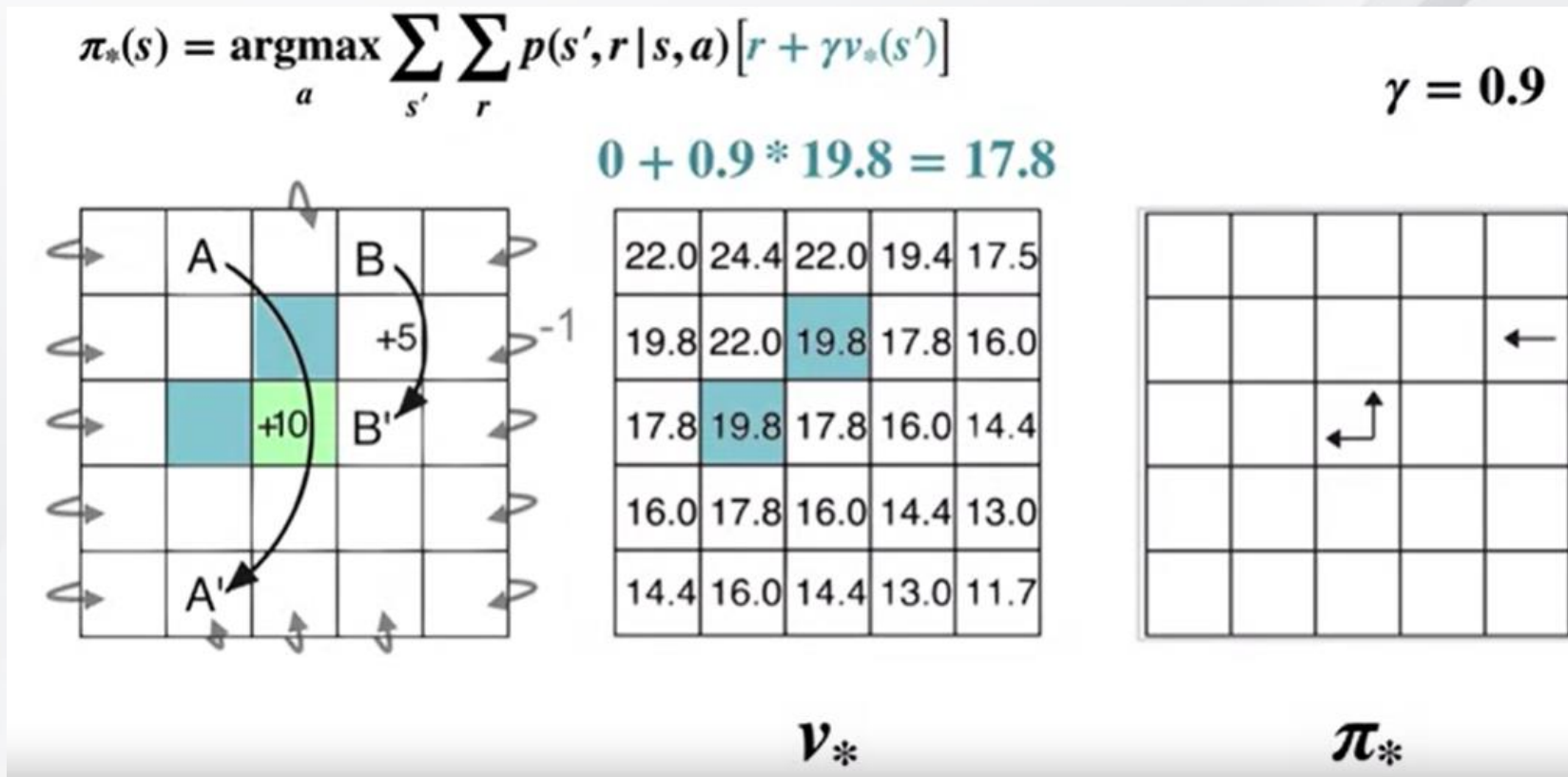
22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

				←

π_*

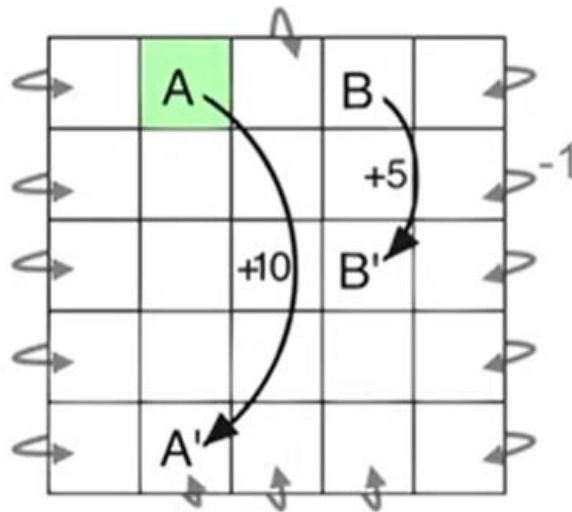
Example: Gridworld



Example: Gridworld

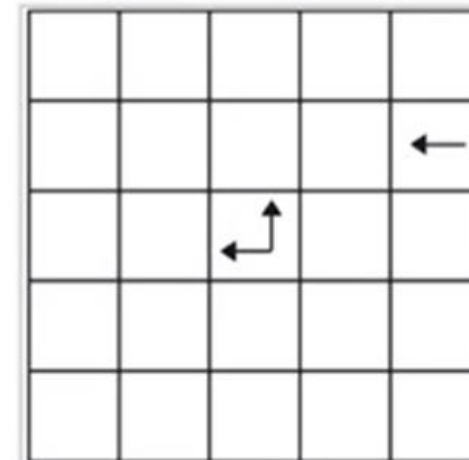
$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$\gamma = 0.9$$



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*



π_*

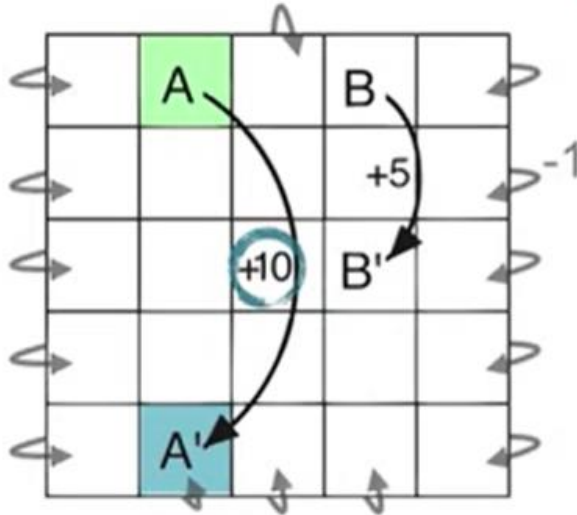
Remember that regardless of the action we pick in state A, we transition to state A' with a reward of +10. This means that in state A every action is optimal.

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

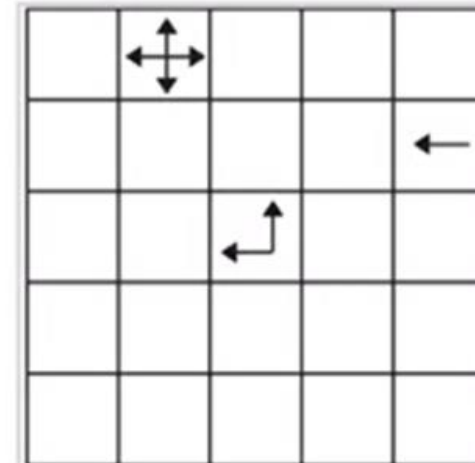
$$\gamma = 0.9$$

$$10 + 0.9 * 16.0 = 24.4$$



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

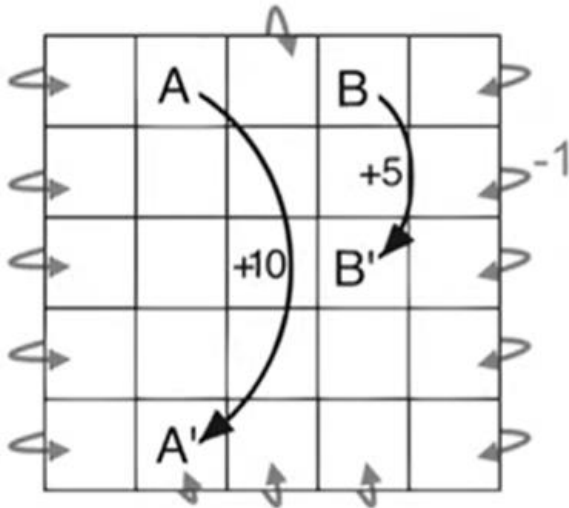


π_*

Example: Gridworld

$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$

$$\gamma = 0.9$$



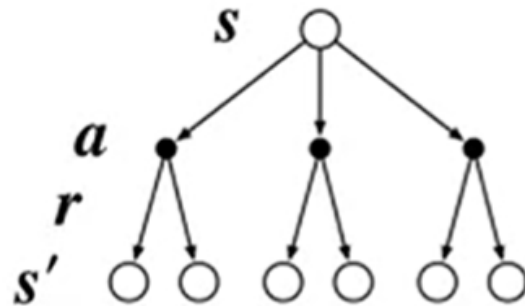
22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

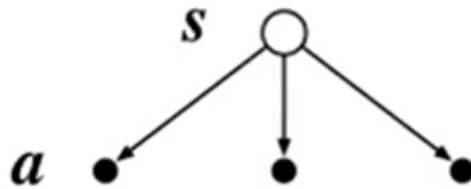
→	↕	←	↕	←
↶	↑	↶	←	←
↶	↑	↶	↶	↶
↶	↑	↶	↶	↶
↶	↑	↶	↶	↶

π_*

Determining an Optimal Policy



$$\pi_*(s) = \operatorname{argmax}_a \sum_{s'} \sum_r p(s', r | s, a) [r + \gamma v_*(s')]$$



$$\pi_*(s) = \operatorname{argmax}_a q_*(s, a)$$

Simple Example

$$\mathbb{S} = \{P, Q, R, S\} \quad \mathbb{A} = \{a, b, c\} \quad \gamma = 0.8$$

Simple Example

$p(s, a, s')$					$p(s, b, s')$					$p(s, c, s')$				
$s \backslash s'$	P	Q	R	S	$s \backslash s'$	P	Q	R	S	$s \backslash s'$	P	Q	R	S
P	0.9	0.1	0.0	0.0	P	0.0	0.6	0.0	0.4	P	0.0	1.0	0.0	0.0
Q	0.0	0.9	0.1	0.0	Q	0.4	0.0	0.6	0.0	Q	1.0	0.0	0.0	0.0
R	0.0	0.0	0.9	0.1	R	0.0	0.4	0.0	0.6	R	0.0	0.0	0.0	1.0
S	0.1	0.0	0.0	0.9	S	0.6	0.0	0.4	0.0	S	0.5	0.5	0.0	0.0

state transition probabilities

$r(s, a)$			
$s \backslash a$	a	b	c
P	0	0	3
Q	0	5	5
R	0	0	0
S	10	10	10

rewards

policy

$$\pi(P) = a$$

$$\pi(Q) = a$$

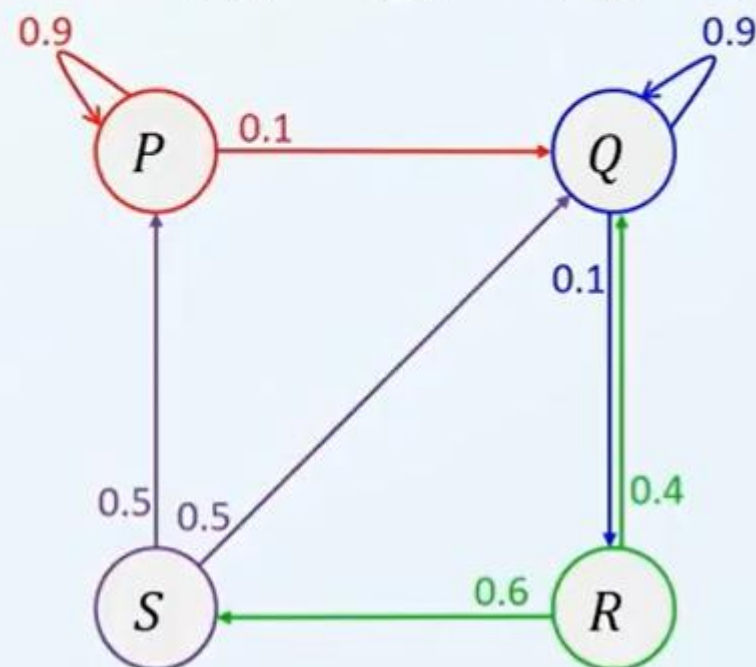
$$\pi(R) = b$$

$$\pi(S) = c$$

Simple Example

Policy: $\pi(P) = a$, $\pi(Q) = a$, $\pi(R) = b$, $\pi(S) = c$

Find: $V^\pi(P)$, $V^\pi(Q)$, $V^\pi(R)$, $V^\pi(S)$



$\mathbf{P}^\pi$				$\mathbf{r}^\pi$
0.9	0.1	0.0	0.0	0
state transition probability matrix				reward matrix

Simple Example

$p(s, a, s')$					$p(s, b, s')$					$p(s, c, s')$				
$s \backslash s'$	P	Q	R	S	$s \backslash s'$	P	Q	R	S	$s \backslash s'$	P	Q	R	S
P	0.9	0.1	0.0	0.0	P	0.0	0.6	0.0	0.4	P	0.0	1.0	0.0	0.0
Q	0.0	0.9	0.1	0.0	Q	0.4	0.0	0.6	0.0	Q	1.0	0.0	0.0	0.0
R	0.0	0.0	0.9	0.1	R	0.0	0.4	0.0	0.6	R	0.0	0.0	0.0	1.0
S	0.1	0.0	0.0	0.9	S	0.6	0.0	0.4	0.0	S	0.5	0.5	0.0	0.0

state transition probabilities

$r(s, a)$			
$s \backslash a$	a	b	c
P	0	0	3
Q	0	5	5
R	0	0	0
S	10	10	10

rewards

policy

$$\pi(P) = a$$

$$\pi(Q) = a$$

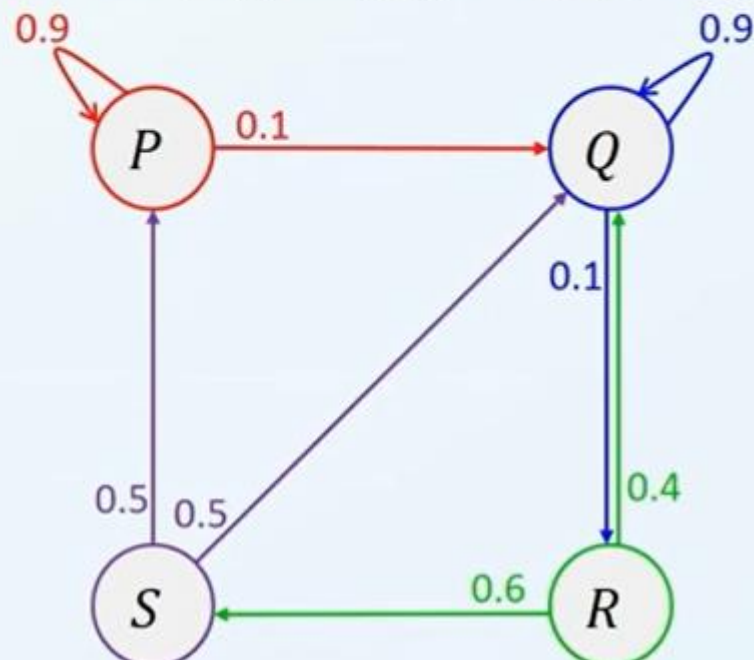
$$\pi(R) = b$$

$$\pi(S) = c$$

Simple Example

Policy: $\pi(P) = a, \pi(Q) = a, \pi(R) = b, \pi(S) = c$

Find: $V^\pi(P), V^\pi(Q), V^\pi(R), V^\pi(S)$



$\mathbf{P}^\pi$				$\mathbf{r}^\pi$
0.9	0.1	0.0	0.0	0
0.0	0.9	0.1	0.0	0
state transition probability matrix				reward matrix

Simple Example

$p(s, a, s')$					$p(s, b, s')$					$p(s, c, s')$				
$s \backslash s'$	P	Q	R	S	$s \backslash s'$	P	Q	R	S	$s \backslash s'$	P	Q	R	S
P	0.9	0.1	0.0	0.0	P	0.0	0.6	0.0	0.4	P	0.0	1.0	0.0	0.0
Q	0.0	0.9	0.1	0.0	Q	0.4	0.0	0.6	0.0	Q	1.0	0.0	0.0	0.0
R	0.0	0.0	0.9	0.1	R	0.0	0.4	0.0	0.6	R	0.0	0.0	0.0	1.0
S	0.1	0.0	0.0	0.9	S	0.6	0.0	0.4	0.0	S	0.5	0.5	0.0	0.0

state transition probabilities

$r(s, a)$			
$s \backslash a$	a	b	c
P	0	0	3
Q	0	5	5
R	0	0	0
S	10	10	10

rewards

policy

$$\pi(P) = a$$

$$\pi(Q) = a$$

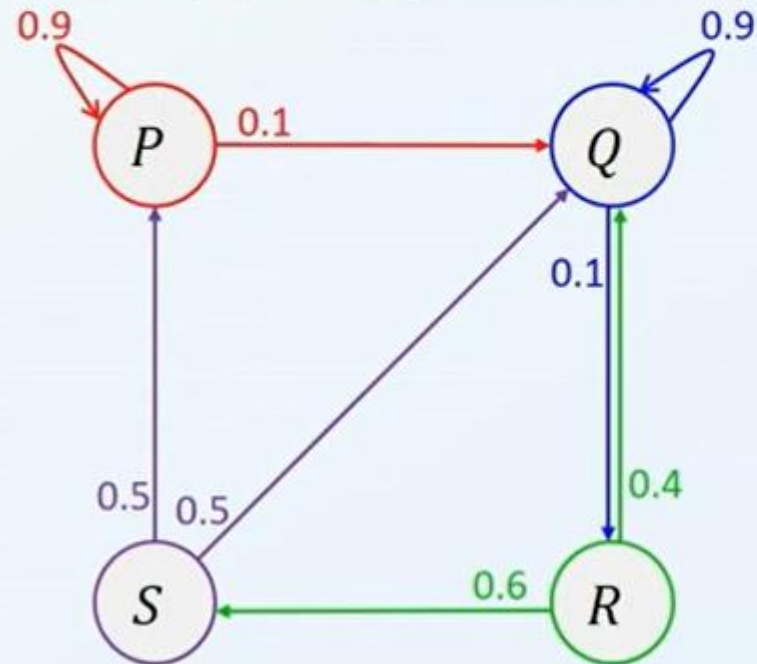
$$\pi(R) = b$$

$$\pi(S) = c$$

Simple Example

Policy: $\pi(P) = a, \pi(Q) = a, \pi(R) = b, \pi(S) = c$

Find: $V^\pi(P), V^\pi(Q), V^\pi(R), V^\pi(S)$



$$\mathbf{P}^\pi$$

0.9	0.1	0.0	0.0
0.0	0.9	0.1	0.0
0.0	0.4	0.0	0.6

state transition
probability matrix

$$\mathbf{r}^\pi$$

0
0
0

reward
matrix

Simple Example

$p(s, a, s')$					$p(s, b, s')$					$p(s, c, s')$				
$s \backslash s'$	P	Q	R	S	$s \backslash s'$	P	Q	R	S	$s \backslash s'$	P	Q	R	S
P	0.9	0.1	0.0	0.0	P	0.0	0.6	0.0	0.4	P	0.0	1.0	0.0	0.0
Q	0.0	0.9	0.1	0.0	Q	0.4	0.0	0.6	0.0	Q	1.0	0.0	0.0	0.0
R	0.0	0.0	0.9	0.1	R	0.0	0.4	0.0	0.6	R	0.0	0.0	0.0	1.0
S	0.1	0.0	0.0	0.9	S	0.6	0.0	0.4	0.0	S	0.5	0.5	0.0	0.0

state transition probabilities

$r(s, a)$			
$s \backslash a$	a	b	c
P	0	0	3
Q	0	5	5
R	0	0	0
S	10	10	10

rewards

policy

$$\pi(P) = a$$

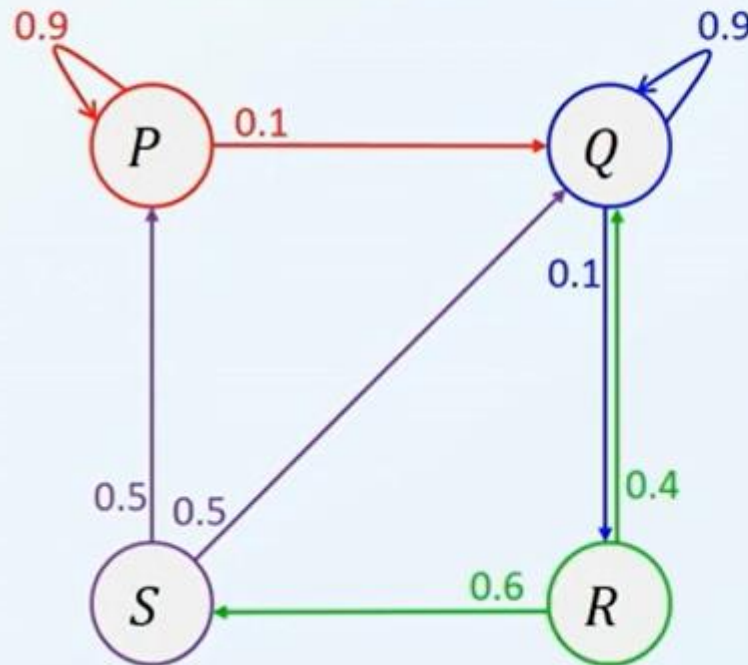
$$\pi(Q) = a$$

$$\pi(R) = b$$

$$\pi(S) = c$$

Simple Example

Policy: $\pi(P) = a, \pi(Q) = a, \pi(R) = b, \pi(S) = c$
Find: $V^\pi(P), V^\pi(Q), V^\pi(R), V^\pi(S)$



$$\mathbf{P}^\pi$$

0.9	0.1	0.0	0.0
0.0	0.9	0.1	0.0
0.0	0.4	0.0	0.6
0.5	0.5	0.0	0.0

state transition
probability matrix

$$\mathbf{r}^\pi$$

0
0
0
10

reward
matrix

Simple Example

Let: $p = V^\pi(P)$, $q = V^\pi(Q)$, $r = V^\pi(R)$, $s = V^\pi(S)$

$$V^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} p(s, a, s') V^\pi(s')$$

Solve: $\gamma = 0.8$

$$p = 0 + \gamma(0.9p + 0.1q)$$

$$q = 0 + \gamma(0.9q + 0.1r)$$

$$r = 0 + \gamma(0.4q + 0.6s)$$

$$s = 10 + \gamma(0.5p + 0.5q)$$

$\mathbf{P}^\pi$				$\mathbf{r}^\pi$
0.9	0.1	0.0	0.0	0
0.0	0.9	0.1	0.0	0
0.0	0.4	0.0	0.6	0
0.5	0.5	0.0	0.0	10
state transition probability matrix				reward matrix



مجموعة جامعة
الجميع الذكية
للتعليم القابضة

MIDOCEAN
UNIVERSITY



Thank You

