Học phần RBE3043: Các thuật toán thích nghi

Buổi 4: Quá trình ra quyết định Markov hữu hạn (cont.)

Giảng viên: TS. Nguyễn Thế Hoàng Anh

Hà Nội, ngày 11 tháng 10 năm 2023

The Agent-Environment Interface

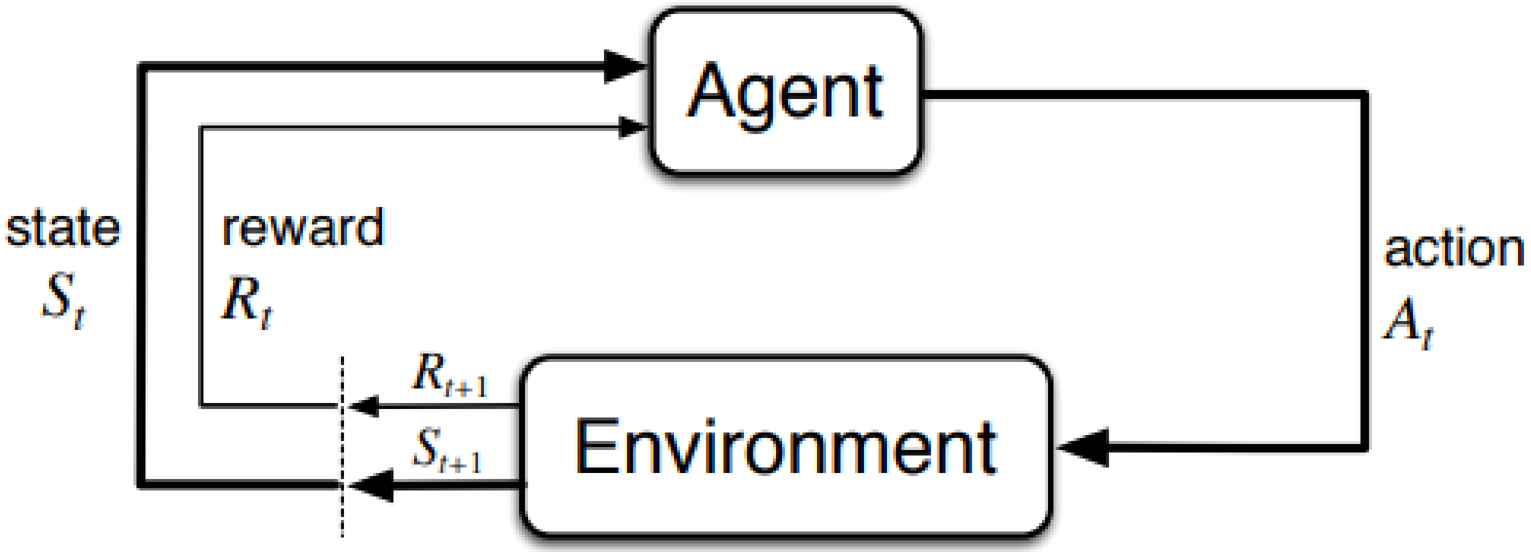


Figure 3.1: The agent–environment interaction in a Markov decision process.

MDPs are meant to be a straightforward framing of the problem of learning from interaction to achieve a goal

MDP Tuple: <S, A, P, R,γ>

- *S: State of the agent on the grid (4,3)*
  - Note that cell denoted by (x,y)
- *A: Actions of the agent, i.e., N, E, S, W*

- *P: Transition function*
  - Table P(s'| s, a), prob of s' given action "a" in state "s"
  - E.g., P((4,3) | (3,3), N) = 0.1
  - E.g., P((3, 2) | (3,3), N) = 0.8
  - (Robot movement, uncertainty of another agent's actions,…)

- *R: Reward (more comments on the reward function later)*
  - *R((3, 3), N) =* -1/25
  - R (4,1) = +1
- *γ: Discounted* factor

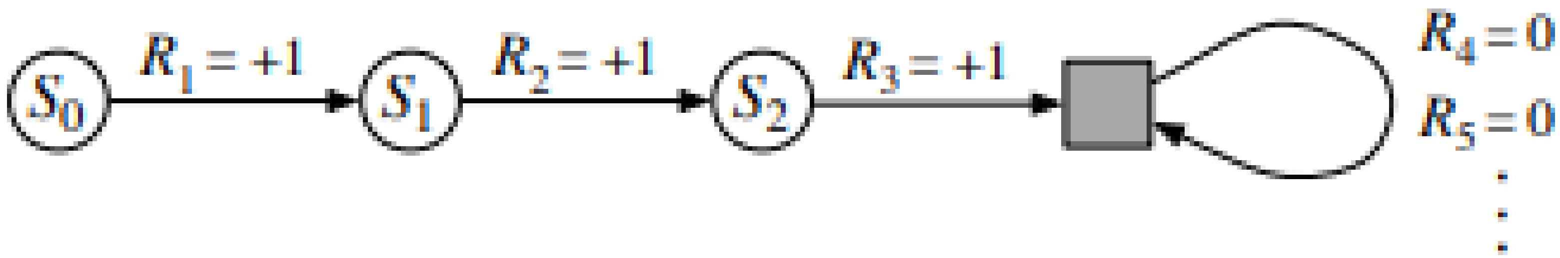
Episodic and Continuing

- **Episodic tasks**: Reinforcement learning tasks that agent interact with environment within a sequence of separate episodes ~ have a terminal state
  - ✓Playing Go, Dota, Atari 2600
  - ✓Bioreactor
  - ✓Pic-and-place robot
  - ✓UAV
  - ✓Self-driving car
  - ✓Incheon airport robot

- **Continuing tasks**: others (not break down into episodes) ~ continue indefinitely ~ no terminal state
  - ✓Cart-pole
  - ✓Cycling robot
  - ✓Studying

Episodic tasks

Episodic tasks have the followed type of state transition diagram



Absorbing state: transition to itself with zero reward → R4 - R5 - …

Return of episodic MDP

$$G_t \doteq \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k,$$

including the possibility that $T = \infty$ or $\gamma = 1$ (but not both).

If reward is constant +1 and the MDP is infinite

$$G_t = \sum_{k=0}^{\infty} \gamma^k = \frac{1}{1-\gamma}.$$

Policies and Value functions

v_π the *state-value function for policy* π

The *value* of a state s under a policy π, denoted $v_\pi(s)$, is the expected return when starting in s and following π thereafter. For MDPs, we can define v_π formally by

$$v_\pi(s) \doteq \mathbb{E}_\pi[G_t \mid S_t = s] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s\right], \text{ for all } s \in \mathcal{S}, \tag{3.12}$$

q_π the *action-value function for policy* π.

the value of taking action a in state s under a policy π, denoted $q_\pi(s, a)$, as the expected return starting from s, taking the action a, and thereafter following policy π:

$$q_\pi(s, a) \doteq \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s, A_t = a\right]. \tag{3.13}$$

These value functions measure how good it is for the agent to be in a given state or how good it is to perform a given action in a given state

Bellman equation

Relationship between the value of a state and the values of its successor states

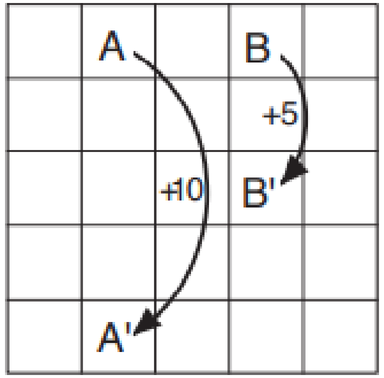
$$
\begin{aligned}
v_\pi(s) &\doteq \mathbb{E}_\pi[G_t \mid S_t = s] \\
&= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} \mid S_t = s] && \text{(by (3.9))} \\
&= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r|s, a)\Big[r + \gamma \mathbb{E}_\pi[G_{t+1}|S_{t+1} = s']\Big] \\
&= \sum_a \pi(a|s) \sum_{s', r} p(s', r|s, a)\Big[r + \gamma v_\pi(s')\Big], \quad \text{for all } s \in \mathcal{S}, && (3.14)
\end{aligned}
$$

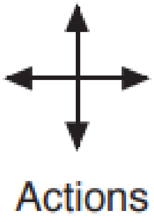
agent is following policy π at time t, then $\pi(a|s)$ is the probability that $A_t = a$ if $S_t = s$.

Meaning of Bellman equation: value of the start state must equal the (discounted) value of the expected next state plus the reward expected in the future

Examples

Example 3.5: Gridworld Figure 3.2 (left) shows a rectangular gridworld representation of a simple finite MDP. The cells of the grid correspond to the states of the environment. At each cell, four actions are possible: `north`, `south`, `east`, and `west`, which deterministically cause the agent to move one cell in the respective direction on the grid. Actions that would take the agent off the grid leave its location unchanged, but also result in a reward of -1. Other actions result in a reward of 0, except those that move the agent out of the special states A and B. From state A, all four actions yield a reward of $+10$ and take the agent to A$'$. From state B, all actions yield a reward of $+5$ and take the agent to B$'$.





3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

Figure 3.2: Gridworld example: exceptional reward dynamics (left) and state-value function for the equiprobable random policy (right).

Suppose the agent selects all four actions with equal probability in all states. Figure 3.2 (right) shows the value function, v_π, for this policy, for the discounted reward case with $\gamma = 0.9$. This value function was computed by solving the system of linear equations (3.14).

Exercise 3.12

1	2	3	4	5
10	9	8	7	6
11	12	13	14	15
20	19	18	17	16
21	22	23	24	25

		2.3		
	0.7	?	0.4	
		-0.4		

$v_\pi(s' = 12)$ = 0.7

$v_\pi(s' = 18)$ = -0.4

$v_\pi(s' = 14)$ = 0.4

$v_\pi(s' = 8)$ = 2.3

$v_\pi(s = 13)$ = ?

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

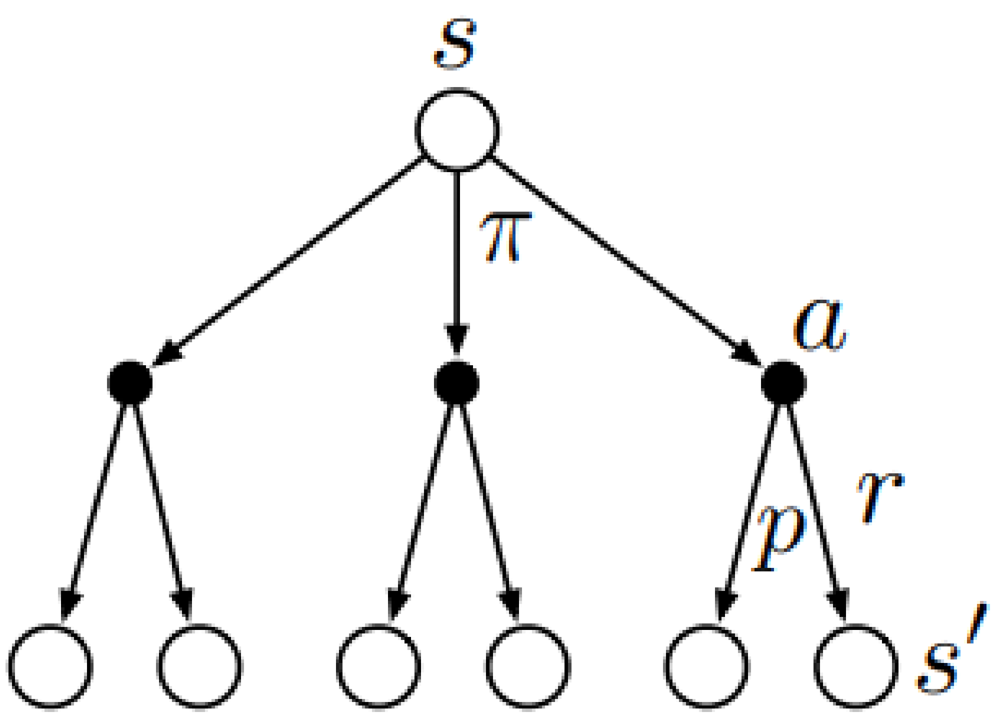
$$= \sum_a \pi(a|s) \sum_{s'} p(s', r=0|s,a)\left[r = 0 + \gamma\, v_\pi(s')\right]$$

$$= \sum_a \pi(a|s) \sum_{s'} \gamma\, v_\pi(s')$$

$$= \sum_a \pi(a|s)\, \gamma \sum_{s'} v_\pi(s')$$

$$= \sum_a \pi(a|s) \times 0.9 \times \left[v_\pi(s'=12) + v_\pi(s'=14) + v_\pi(s'=18) + v_\pi(s'=8)\right]$$

Backup diagram

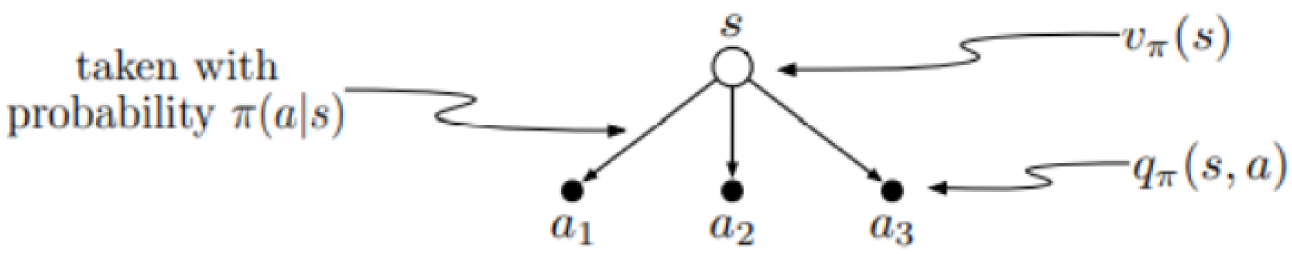


Backup diagram for v_π

Backup diagram: shows relationship between two successive states
Open circle: a state
Solid circle: a state-action pair
Policy: π

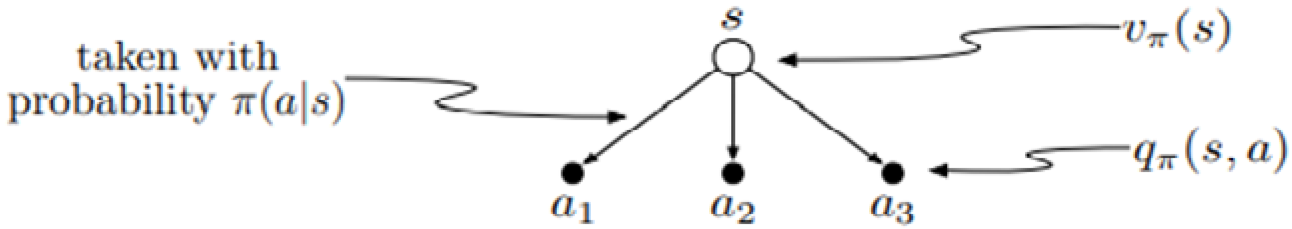
Backup (update) operations transfer value information back to a state (or a state-action pair) from its successor states (or state-action pairs)

Exercise 3.16 The value of a state depends on the values of the actions possible in that state and on how likely each action is to be taken under the current policy. We can think of this in terms of a small backup diagram rooted at the state and considering each possible action:



Give the equation corresponding to this intuition and diagram for the value at the root node, $v_\pi(s)$, in
af node, $q_\pi(s,a)$, given $S_t = s$. This equation should include an terms of the value at the expected le
the policy, π. Then give a second equation in which the expected expectation conditioned on following
ns of $\pi(a|s)$ such that no expected value notation appears in the value is written out explicitly in terr
equation.

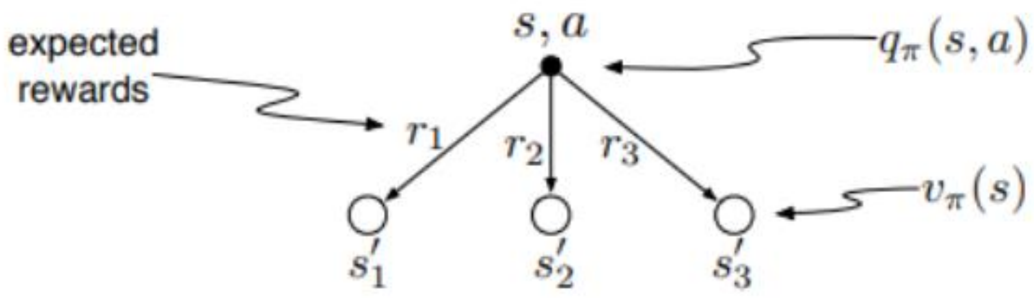
$$v_\pi(s) = ?$$



$$v_\pi(s) = \pi(a_1|s)q_\pi(s|a_1) + \pi(a_2|s)q_\pi(s|a_2) + \pi(a_3|s)q_\pi(s|a_3)$$

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

Exercise 3.17 The value of an action, $q_\pi(s,a)$, depends on the expected next reward and the expected sum of the remaining rewards. Again we can think of this in terms of a small backup diagram, this one rooted at an action (state–action pair) and branching to the possible next states:



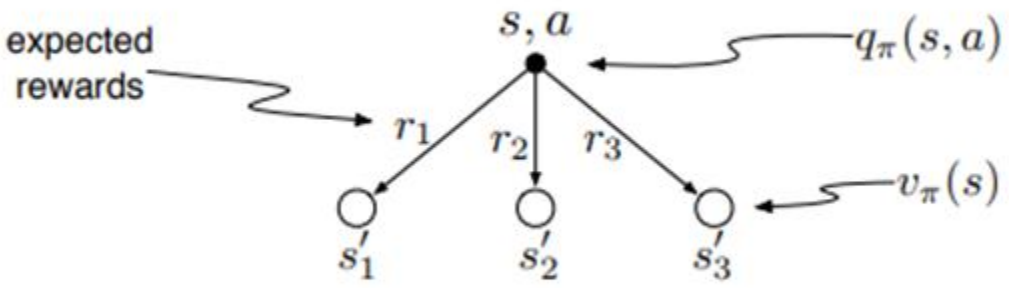
Give the equation corresponding to this intuition and diagram for the action value, $q_\pi(s,a)$, in terms of the expected next reward, R_{t+1}, and the expected next state value, $v_\pi(S_{t+1})$, given that $S_t = s$ and $A_t = a$. This equation should include an expectation but *not* one conditioned on following the policy. Then give a second equation, writing out the expected value explicitly in terms of $p(s', r \mid s, a)$ defined by (3.2), such that no expected value notation appears in the equation. □

$$q_\pi(s,a) = ?$$

The action value $q(s,a)$ depends on expected next reward and expected sum of remaining rewards. Writing in terms of expectation we get

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

For all branches we get values as follows



$$q^1_\pi(s,a) = p(s_1, r_1|s,a)\left[r_1 + \gamma v_\pi(s_1)\right]$$

$$q^2_\pi(s,a) = p(s_2, r_2|s,a)\left[r_2 + \gamma v_\pi(s_2)\right]$$

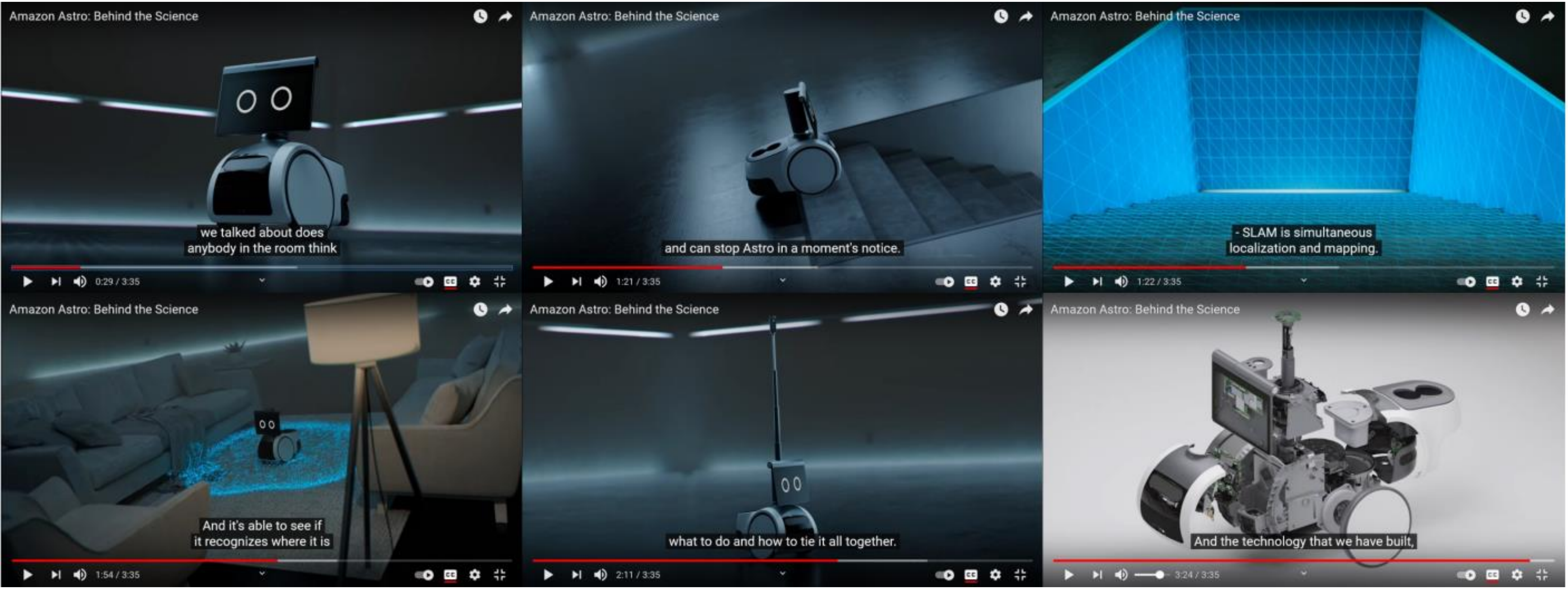
$$q^3_\pi(s,a) = p(s_3, r_3|s,a)\left[r_3 + \gamma v_\pi(s_3)\right]$$

The final value will be sum of the three

$$q_\pi(s,a) = q^1_\pi(s,a) + q^2_\pi(s,a) + q^3_\pi(s,a)$$

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

Amazon Astro



Amazon Astro	https://www.youtube.com/watch?v=sj1t3msy8dc
Behind the Science	https://www.youtube.com/watch?v=Zy4If8-Wth4

Amazon Astro: Technologies

2 camera (front) + 2 additional cameras, follow people

2 wheels, autonomous driving, SLAM – Simultaneous localization and mapping (reinforcement learning?), automatic charging

Robot understanding & communication

See
Infer
Hear
Understanding
Moving

Speech recognition and processing: window break, sound alarm

Image processing: object detection, smoke detection

video streaming

Optimal Policies and Optimal Value Functions

Better policy: Expected return of policy π is greater than or equal to expected return of policy π' for all states → policy π better or equal to policy π'

Optimal policy π_*: best policy that is better or equal to all other policies

Optimal state-value function

$$v_*(s) \doteq \max_\pi v_\pi(s) \qquad \text{for all } s \in \mathcal{S}.$$

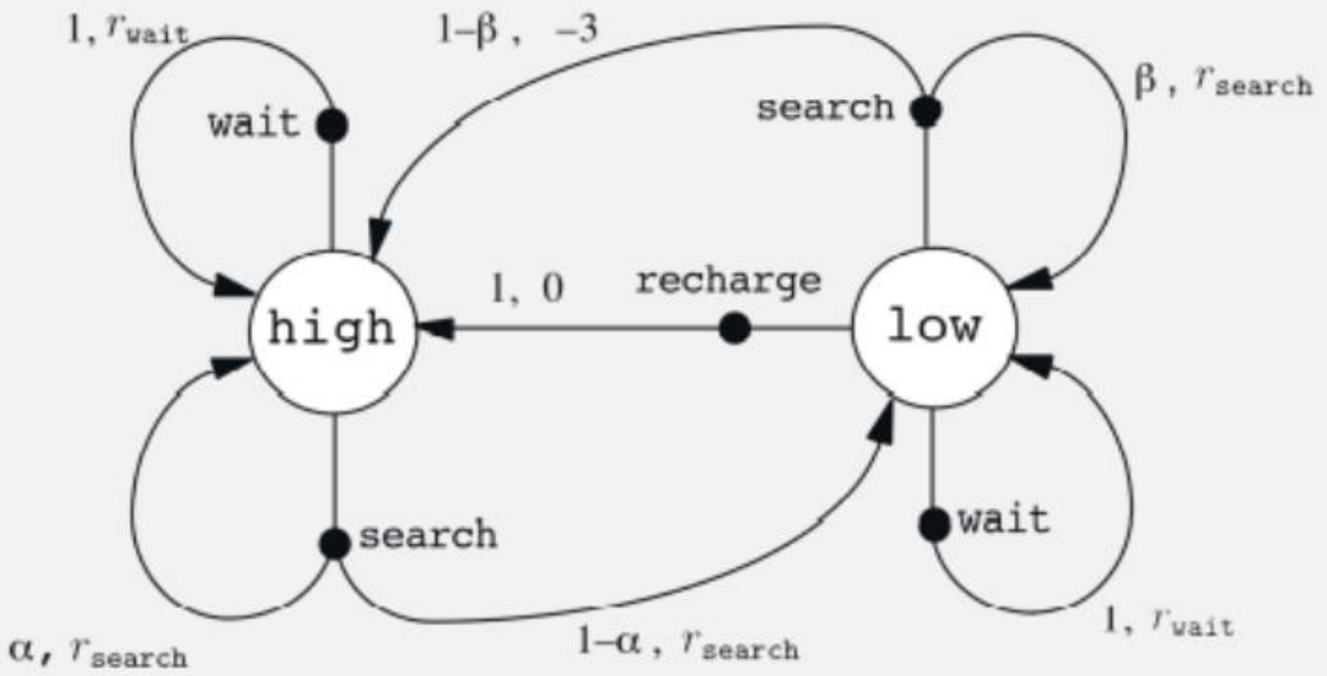
Optimal action-value function

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

$$q_*(s,a) = \mathbb{E}[R_{t+1} + \gamma v_*(S_{t+1}) \mid S_t = s, A_t = a]$$

Example 3.8: Bellman Optimality Equations for the Recycling robot

s	a	s'	$p(s' \mid s, a)$	$r(s, a, s')$
high	search	high	α	$r_{\texttt{search}}$
high	search	low	$1 - \alpha$	$r_{\texttt{search}}$
low	search	high	$1 - \beta$	-3
low	search	low	β	$r_{\texttt{search}}$
high	wait	high	1	$r_{\texttt{wait}}$
high	wait	low	0	$r_{\texttt{wait}}$
low	wait	high	0	$r_{\texttt{wait}}$
low	wait	low	1	$r_{\texttt{wait}}$
low	recharge	high	1	0
low	recharge	low	0	0



Example 3.8: cont.

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

States: {h, l} ~ {high, low}
Actions: {s, w, re} ~ {search, wait, recharge}
Reward: $\{r_s, r_w\}$ ~ $\{r_{search}, r_{wait}\}$

$$\begin{aligned} v_*(\mathtt{h}) &= \max \left\{ \begin{array}{l} p(\mathtt{h}|\mathtt{h},\mathtt{s})[r(\mathtt{h},\mathtt{s},\mathtt{h}) + \gamma v_*(\mathtt{h})] + p(\mathtt{l}|\mathtt{h},\mathtt{s})[r(\mathtt{h},\mathtt{s},\mathtt{l}) + \gamma v_*(\mathtt{l})], \\ p(\mathtt{h}|\mathtt{h},\mathtt{w})[r(\mathtt{h},\mathtt{w},\mathtt{h}) + \gamma v_*(\mathtt{h})] + p(\mathtt{l}|\mathtt{h},\mathtt{w})[r(\mathtt{h},\mathtt{w},\mathtt{l}) + \gamma v_*(\mathtt{l})] \end{array} \right\} \\ &= \max \left\{ \begin{array}{l} \alpha[r_\mathtt{s} + \gamma v_*(\mathtt{h})] + (1-\alpha)[r_\mathtt{s} + \gamma v_*(\mathtt{l})], \\ 1[r_\mathtt{w} + \gamma v_*(\mathtt{h})] + 0[r_\mathtt{w} + \gamma v_*(\mathtt{l})] \end{array} \right\} \\ &= \max \left\{ \begin{array}{l} r_\mathtt{s} + \gamma[\alpha v_*(\mathtt{h}) + (1-\alpha) v_*(\mathtt{l})], \\ r_\mathtt{w} + \gamma v_*(\mathtt{h}) \end{array} \right\}. \end{aligned}$$

$$v_*(\mathtt{l}) = \max \left\{ \begin{array}{l} \beta r_\mathtt{s} - 3(1-\beta) + \gamma[(1-\beta)v_*(\mathtt{h}) + \beta v_*(\mathtt{l})] \\ r_\mathtt{w} + \gamma v_*(\mathtt{l}), \\ \gamma v_*(\mathtt{h}) \end{array} \right\}.$$

For any choice of $r_\mathtt{s}$, $r_\mathtt{w}$, α, β, and γ, with $0 \le \gamma < 1$, $0 \le \alpha, \beta \le 1$, there is exactly one pair of numbers, $v_*(\mathtt{h})$ and $v_*(\mathtt{l})$, that simultaneously satisfy these two nonlinear equations. ■

Example 3.9: solving the Gridworld

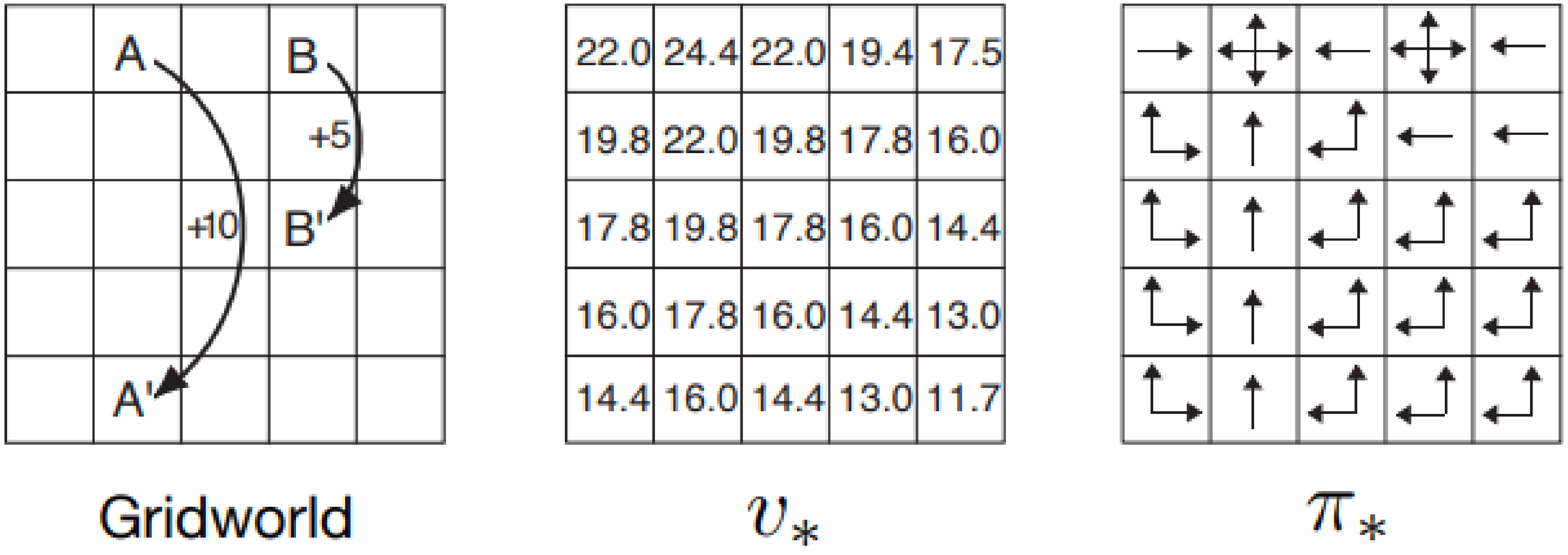
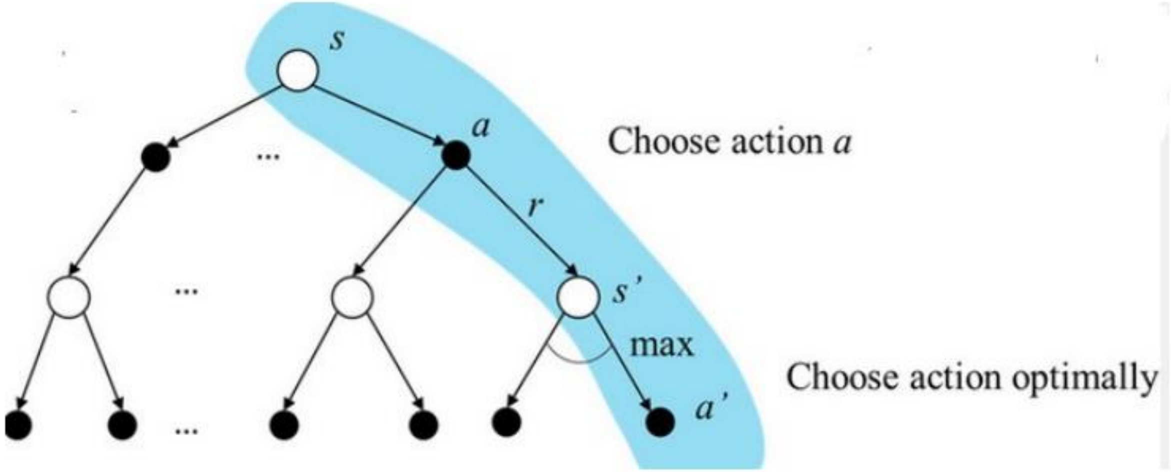


Figure 3.5: Optimal solutions to the gridworld example.

Optimal state-value function and Optimal action-value function relationship

$$v_*(s) = \max_a \; q_*(s,a)$$



Nhớ giữ sức khỏe! Take care!

Thank you!
Q&A