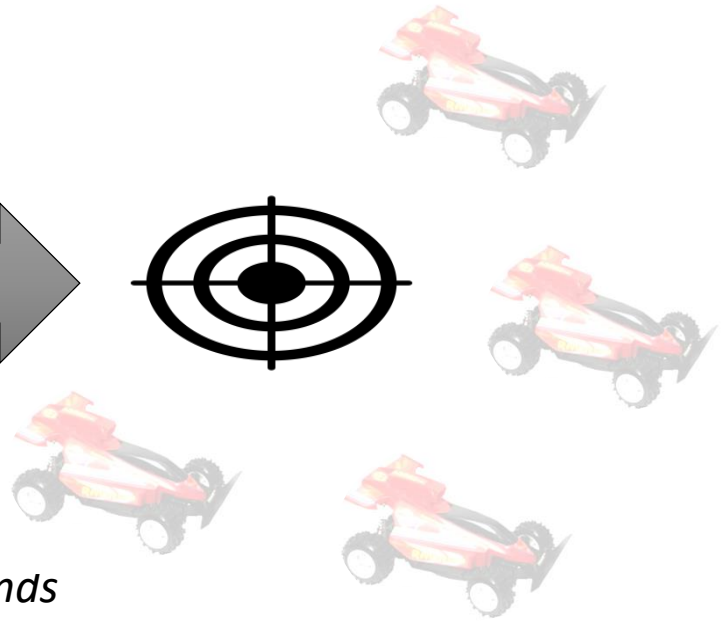
Sistemi Intelligenti
Corso di Laurea in Informatica, A.A. 2019-2020
Università degli Studi di Milan

An introduction to MDPs

Planning under uncertainty

- Action selection is often affected by uncertainty
- Example:



Fast forward for 2 seconds

Planning under uncertainty

Are the effects of my actions perfectly predictable?

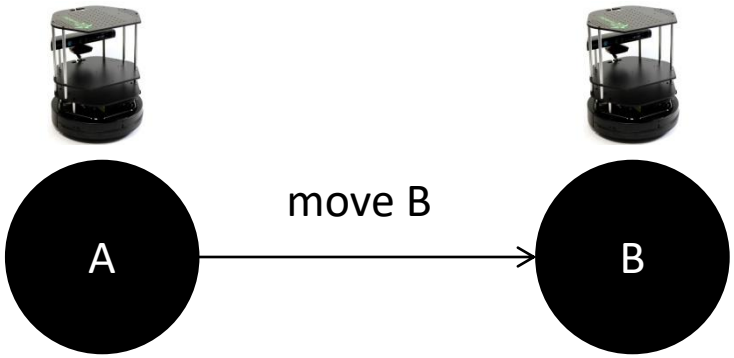
Am I always sure about what's going on?

Deterministic vs Stochastic transitions

Fully observable vs Partially observable states

Examples

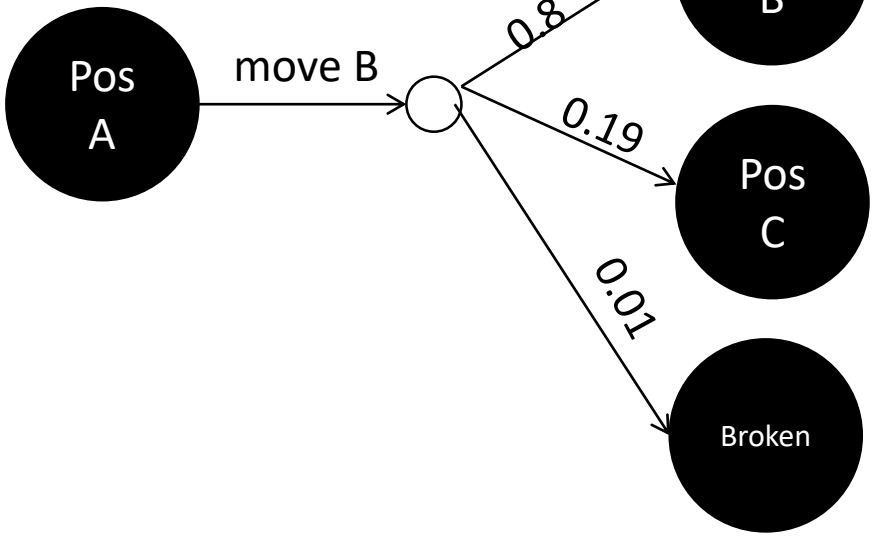
- Deterministic transitions, fully observable states
- Only actuation is needed, no sensing!



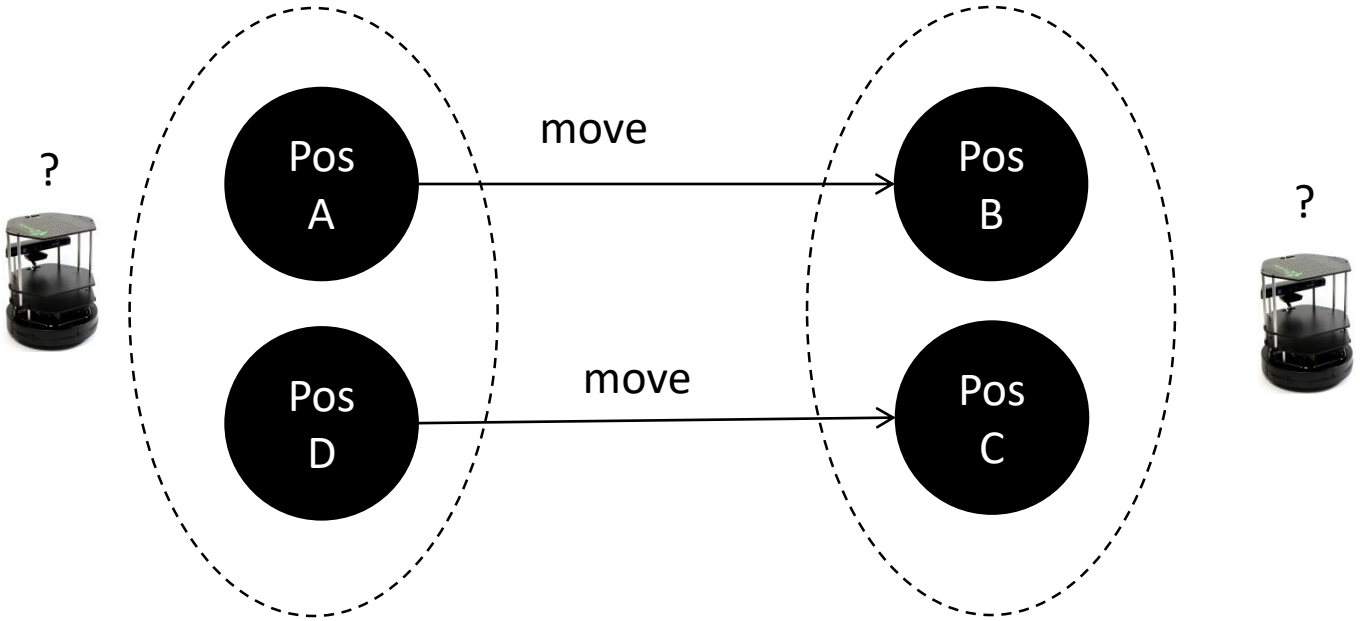
Examples

- Stochastic transitions, fully observable states

Surveillance robot

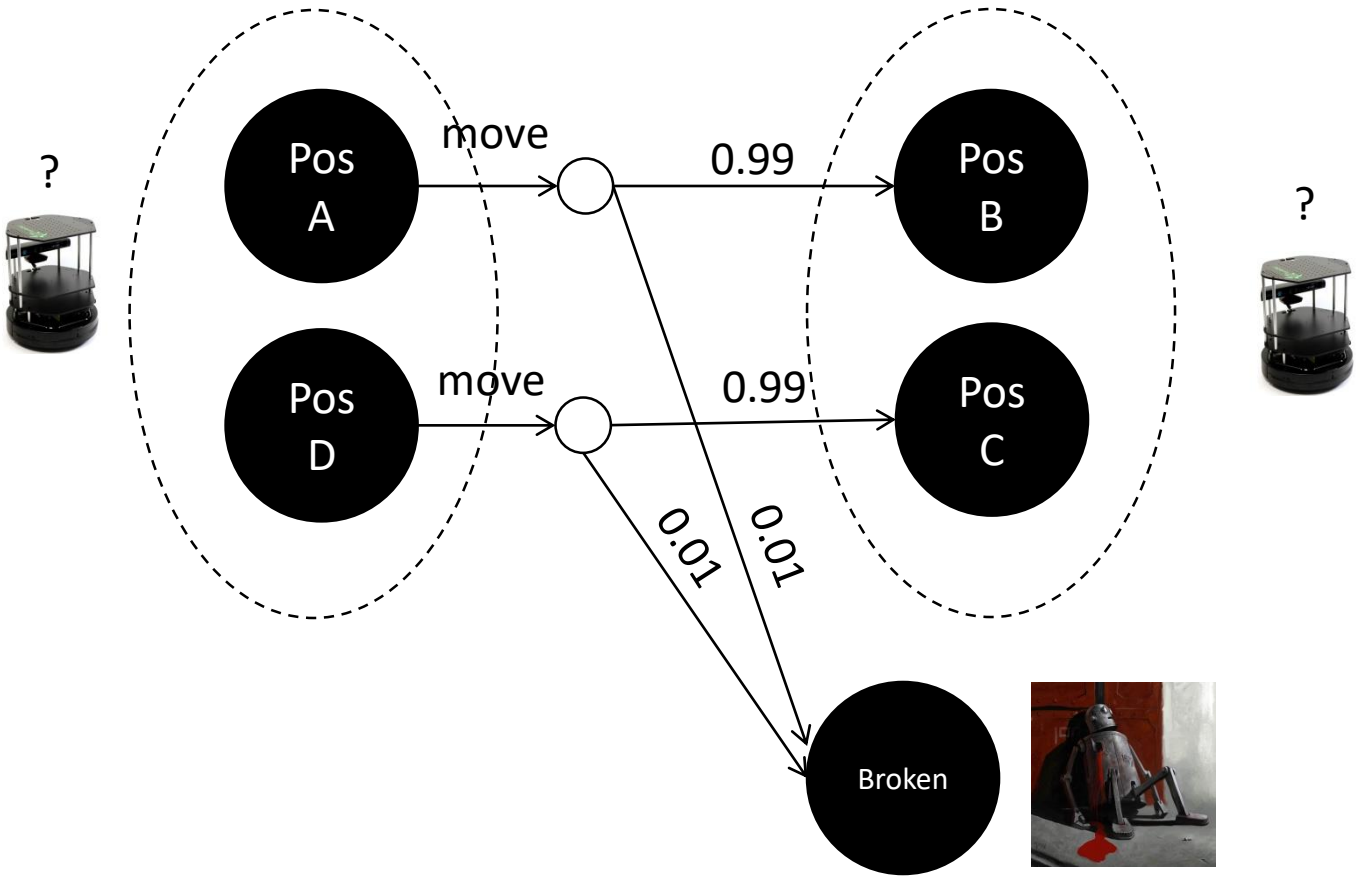


Examples



Deterministic transitions, partially observable states

Examples



Stochastic transitions, partially observable states

Markov Decision Processes (MDP)

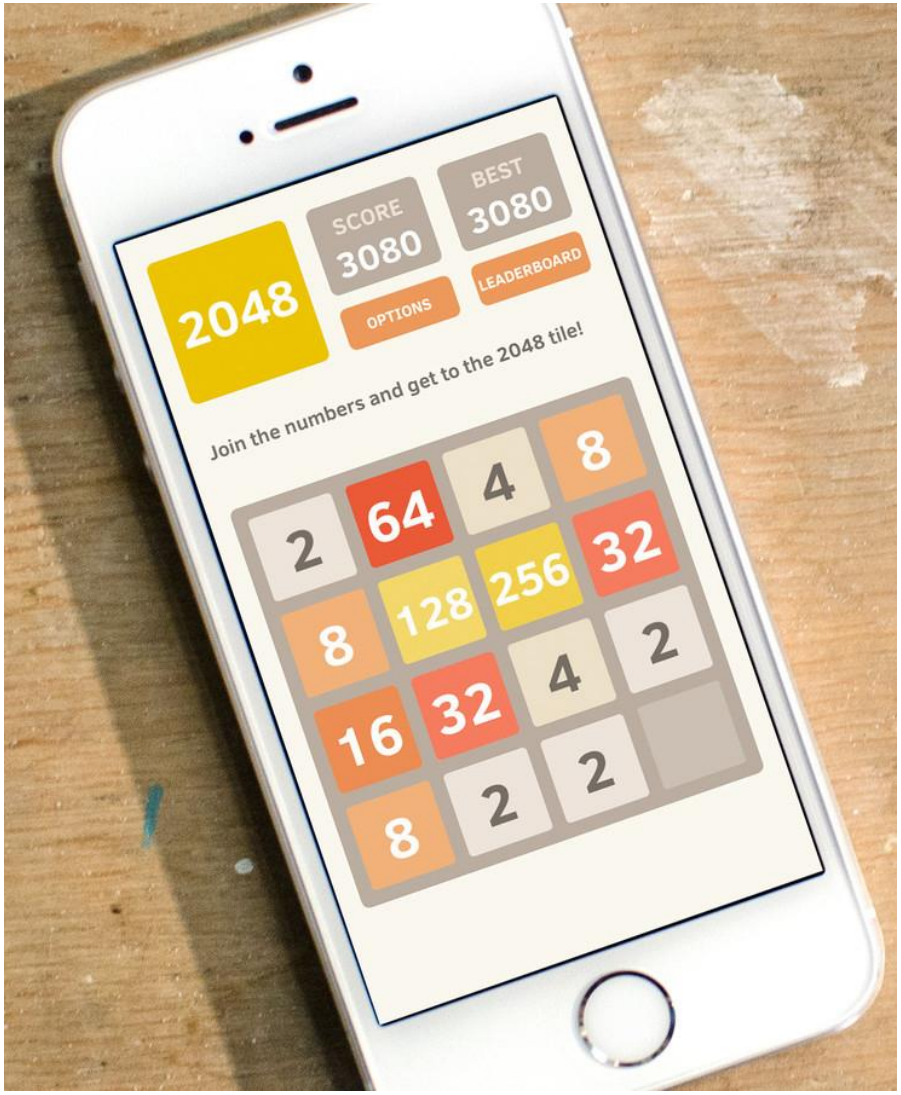
- We assume full observability of states, but non-deterministic actions
- We cannot specify a transition function like before, instead we give a set of transition probabilities

$$P(s'|s,a)$$ Probability of reaching state s', given that current state is s and action a is taken

- State transitions satisfy the **Markov property**: they depend only on the current state and not on states visited before

- The Markov property can be stated more generally: the state encodes all the information we need to pick an optimal action

Example (Markovian, deterministic)



Example (Markovian (?), deterministic)

If the snake world is finite, then …

MDPs

- Can we formulate the problem asking for a plan?

- Plans are unfit for this situation: we cannot tell how to reach some goal by giving a mere sequence of actions

- We need a **policy**

$$\pi : X \rightarrow a$$

Given the current state, it returns what action to play (deterministic)

- Policy execution:
  1. Observe current state s
  2. Execute
  3. Repeat from 1

MDPs

- We previously spoken about action costs, in MDPs we speak about immediate rewards

$R_a(s, s')$ The payoff that an agent gets when she goes from state s to state s' with an action a

- Rewards can be thought as a generalization of what before we described by means of goal states specification

- Solving the MDP means finding a policy that maximizes the expected reward over some finite or infinite time horizon; such policy is called the optimal policy (π^*)

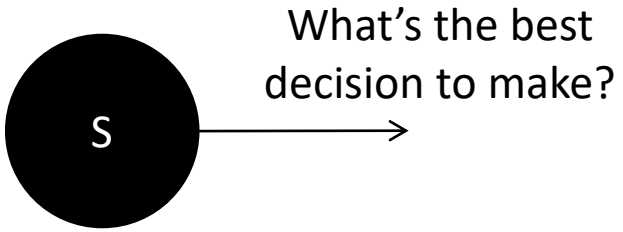
Some features of policies

- **Deterministic**: given a state it does not randomize on which action to take
- **Stationary** (or memoryless): it does not change over time

In MDPs, up to some reasonable properties or common operative choices, these assumptions are not restrictive: there always exists an optimal policy that is deterministic and stationary

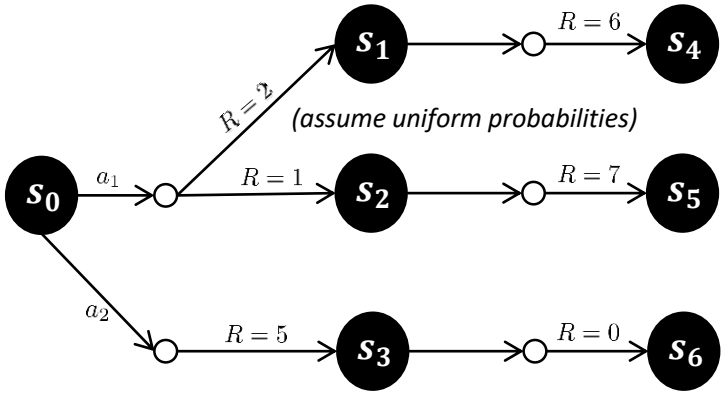
Stationary optimal policies

- In infinite horizon MDPs, the optimal policy is stationary (or memoryless):



- The Markov property says that all the information (transitions, rewards, …) we need to decide is encoded in the state. Thus, the answer to the above question only depends on the state, not on the time of visit. *Does this hold in finite horizon settings?*

- Is this true in finite horizon MDPs?

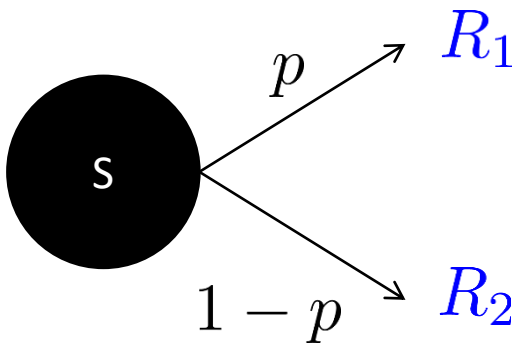


$\pi(s_0)$	$H=1$	$H=2$
a_1	1.5	π^* 8
a_2	π^* 5	5

The optimal policy depends on the time of visit!

Deterministic optimal policies

- Optimal policy is deterministic (intuition):



Suppose that in some state the optimal policy π^* prescribes to randomize over two actions

By following such policy the agent will get an expected reward of

$$\mathbb{E}_{\pi^*}[R] = pR_1 + (1-p)R_2$$

where R_1 (R_2) is the expected reward from S when playing the top (bottom) action

Three cases:

1. $R_1 = R_2$: the agent is indifferent, she can pick always the same action as well
2. $R_1 > R_2$: the agent can gain pushing probability to the top action
3. $R_1 < R_2$: the agent can gain pushing probability to the bottom action

MDP Value iteration

- Let's introduce the concept of **value function**

- How does it work?

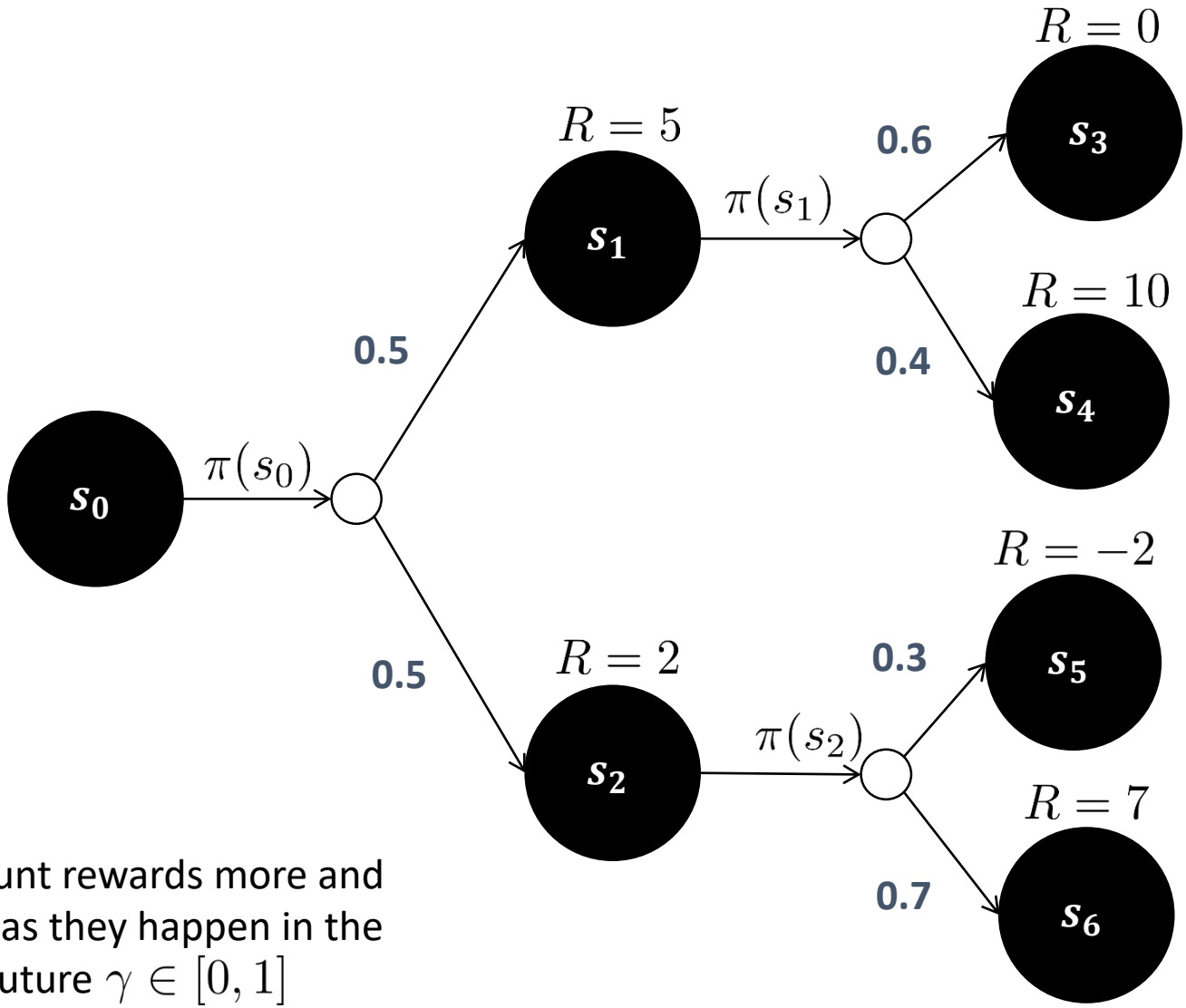
$$V_\pi : X \rightarrow \mathbb{R}$$

An agent executing policy π is in state s: how happy is she?

$$V_\pi(s)$$

- This quantity is defined as **the expected cumulative reward that can be obtained by executing π from s**

Example with H=2



Discount rewards more and more as they happen in the long future $\gamma \in [0, 1]$

$$V_\pi(s_0) = 0.5(\gamma^0 5 + (0.4(\gamma^1 10))) + 0.5(\gamma^0 2 + (0.3\gamma^1(-2) + 0.7\gamma^1(7)))$$

MDP Value iteration (intuition)

$V^*_{\pi,k}(s)$ Expected value of the optimal policy from state s when H=k

$V^*_{\pi,0}(s) = 0 \quad \forall s \in X$ The horizon is zero, no action no reward

The horizon is 1, there's room for just one action. The best thing to do is selecting the action that maximizes the **immediate expected reward**

$$V^*_{\pi,1}(s) = \max_a \left\{ \sum_{s' \in X} P(s'|s,a) R_a(s,s') \right\} \forall s \in X$$

Now the horizon is 2. The optimal policy would select the action that maximizes the immediate expected reward plus the **expected discounted reward of acting optimally from the arrival state**

$$V^*_{\pi,2}(s) = \max_a \left\{ \sum_{s' \in X} P(s'|s,a) \left(R_a(s,s') + \gamma V^*_{\pi,1}(s') \right) \right\} \forall s \in X$$

$$V^*_{\pi,3}(s) = \max_a \left\{ \sum_{s' \in X} P(s'|s,a) \left(R_a(s,s') + \gamma V^*_{\pi,2}(s') \right) \right\} \forall s \in X$$

$$\vdots$$

MDP Value iteration

- We obtain a recursive definition:

$$V^*_{\pi,H}(s) = \max_a \left\{ \sum_{s' \in X} P(s'|s,a)\big(R_a(s,s') + \gamma V^*_{\pi,H-1}(s')\big) \right\}$$

$$\pi^*(s) = \arg\max_a \left\{ \sum_{s' \in X} P(s'|s,a)\big(R_a(s,s') + \gamma V^*_{\pi,H-1}(s')\big) \right\}$$

- In infinite horizon settings the above definition becomes the *Bellman Equation*

$$V^*_{\pi}(s) = \max_a \left\{ \sum_{s' \in X} P(s'|s,a)\big(R_a(s,s') + \gamma V^*_{\pi}(s')\big) \right\}$$

- Solved with iterative methods
- Encodes the Bellman's principle of optimality