Izmir Institute of Technology

CENG 461 – Artificial Intelligence

Markov Decision Processes

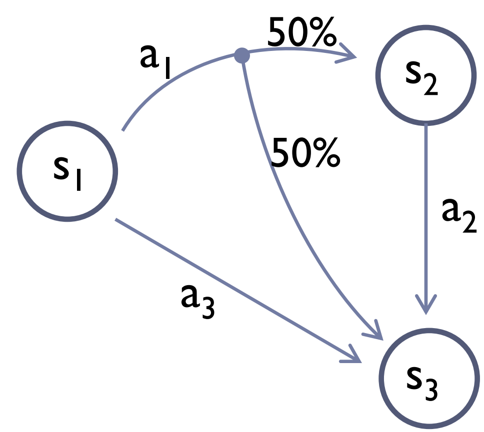
2015-16 Fall Term

Planning under Uncertainty

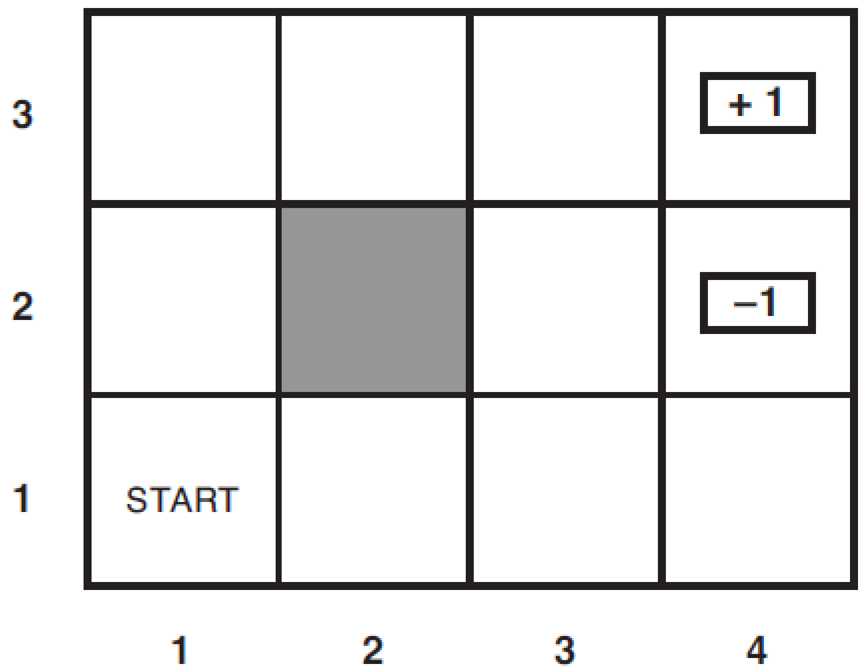
	Deterministic	Stochastic
Fully Observable	A*, Depth-first, Breadth-first	MDP
Partially Observable		POMDP

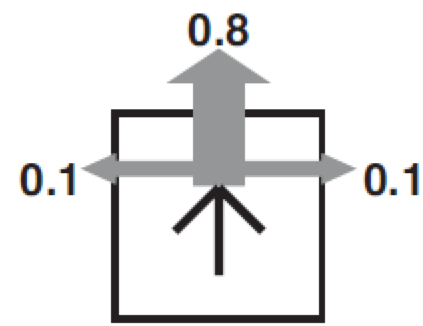
Markov Decision Process (MDP)

- An MDP is defined by
  - a set of states $s_1, \ldots, s_N$,
  - a set of actions $a_1, \ldots, a_K$
  - a transition model $T(s, a, s') = P(s' | s, a)$
  - a reward function $R(s)$ (reward of being at a state)
- The solution must specify an action for each state and such a solution is called a policy $\pi(s)$.
- The optimal policy $\pi^*(s)$ maximizes the expected reward.



Grid World



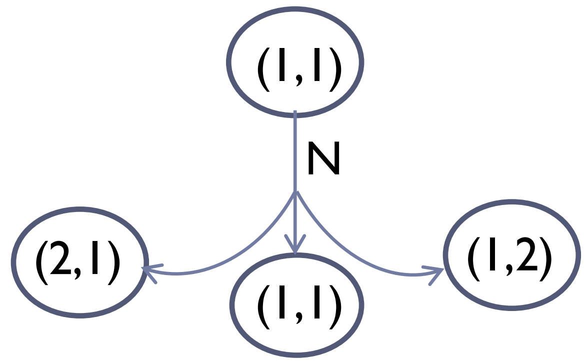


+1 and -1 are the absorbing states, i.e. the agent leaves the environment

Going N may also result in E (10%) or W(10%).
We represent as p=(0.8,0.1,0.1).

Planning @ Stochastic Environments

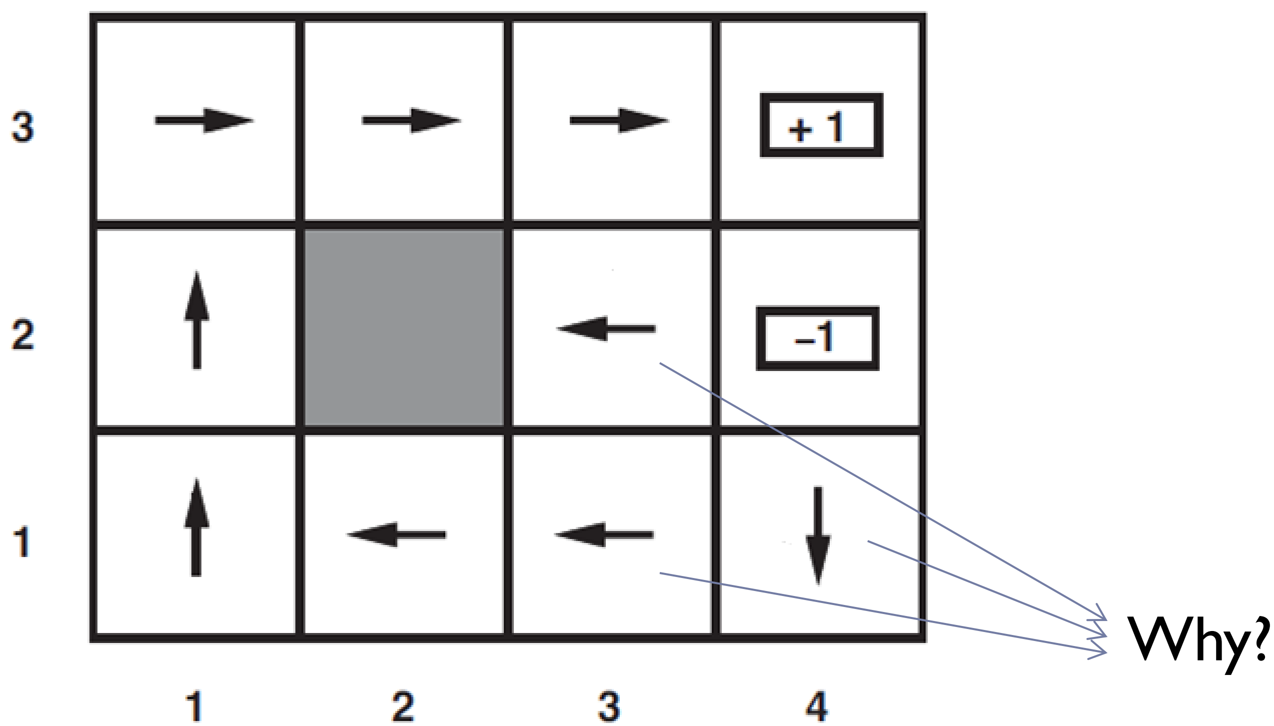
- Conventional planning has some problems in stochastic environments
  - Branching factor is huge since T(s, a, s') have different possible outcomes



  - States are visited many times. Algorithms like A* does not solve the problem since those states are not planned but occurred due to the stochastic environment

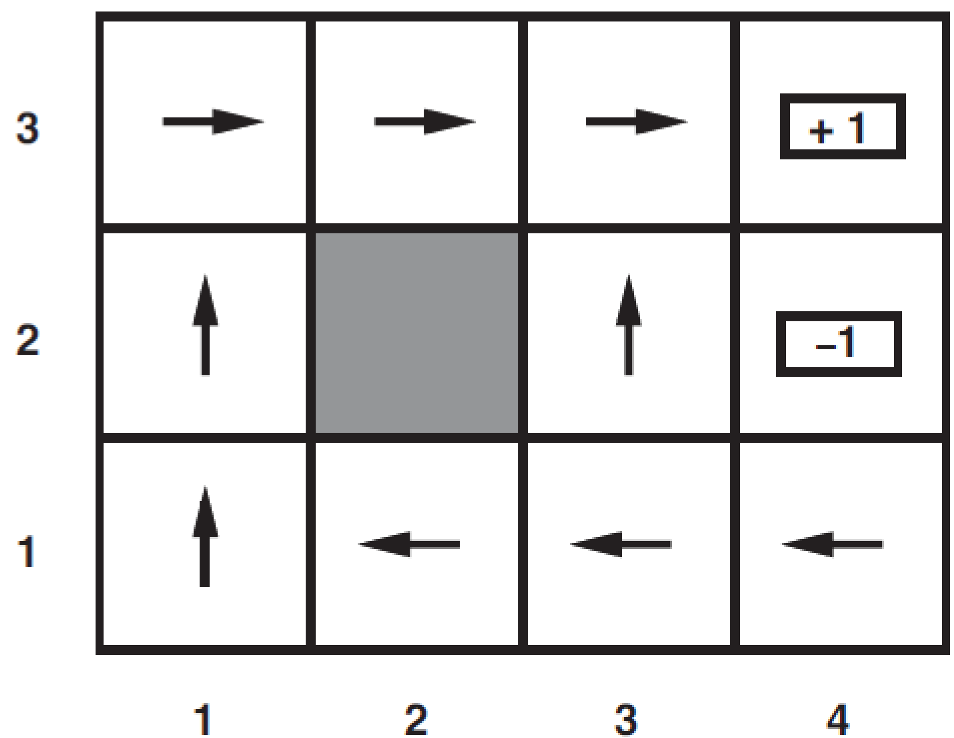
Optimal Policy $\pi^*(s)$

When R(s)=0 for states other than absorbing states, i.e. there is no cost for wandering around

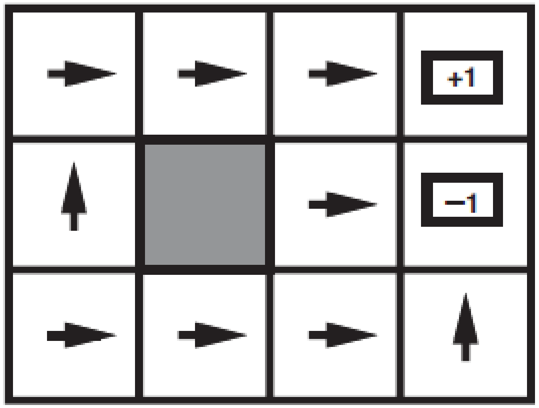


Optimal Policy for R(s) = -0.03

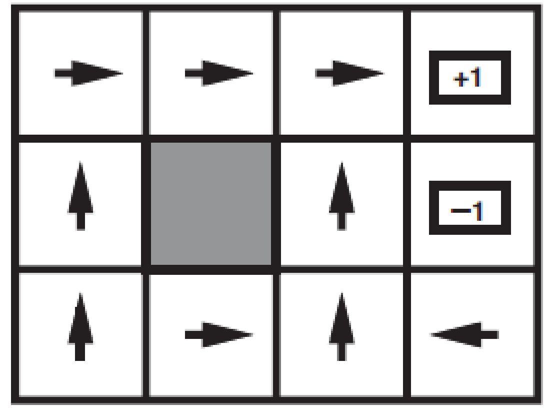
R(s)=-0.03 for states other than absorbing states, i.e. moving within the grid world has a step cost.



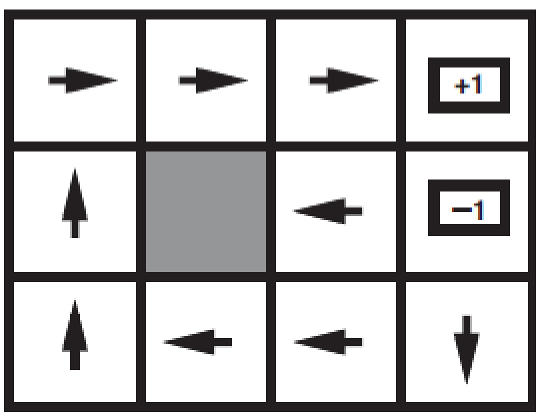
Optimal Policies for Different R(s)



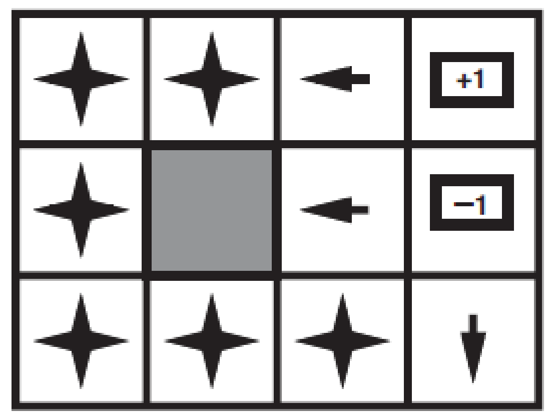
R(s) < -1.6284



-0.4278 < R(s) < -0.0850



-0.0221 < R(s) ≤ 0



R(s) > 0

Finding the Optimal Policy

- An optimal policy satisfies:

$$\pi^* = \underset{\pi}{\operatorname{argmax}}\ E\left[\sum_{t=0}^{\infty} \gamma^t R(s_t)\right]$$

- This maximizes the expected sum of discounted rewards
- γ^t, is the discount factor, describes the preference of an agent for current awards over future awards. When γ=1, future rewards are equivalent to current awards. Usually values like 0.9 are chosen.

Utility of a state

- Given a certain policy, the utility of a state:

$$U^{\pi}(s) = E\left[\sum_{t=0}^{\infty} \gamma^{t} R(s_t)\right]$$

where $s_0 = s$.

- This can also be written on the condition that agent chooses the optimal action

$$U(s) = R(s) + \gamma \max_{a} \sum_{s'} P(s'|a, s)U(s')$$

In short: The utility of a state is the immediate reward plus the expected discount utility for the next state.

Utility for States in Grid World

3	0.85	0.89	0.93	+1
2	0.81		0.68	–1
1	0.77	0.73	0.70	0.47
	1	2	3	4

Calculated with $R(s)$=-0.03 , γ=1 , p=(0.8,0.1,0.1)

Value Iteration Algorithm

- Calculating the final utilities is an iterative process and can be performed by 'value iteration algorithm'.
- We can start with $U(s) = 0$ for all states.
- We iterate the values of each state with the following equation

$$U_{i+1}(s) = R(s) + \gamma \max_{a} \sum_{s'} P(s'|s,a) U_i(s')$$

- Convergence to the actual state utilities is guaranteed.
- The optimal policy action is the action that maximizes the max part of the update equations.

Value Iteration in a Deterministic World

- Start with $U(s)$ = 0 for all states except for absorbing ones. Take $\mathrm{R}(s)$=-0.03 and γ=1. p=(1,0,0). Apply value iteration.

$$U_{i+1}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a)U_i\,(s')$$

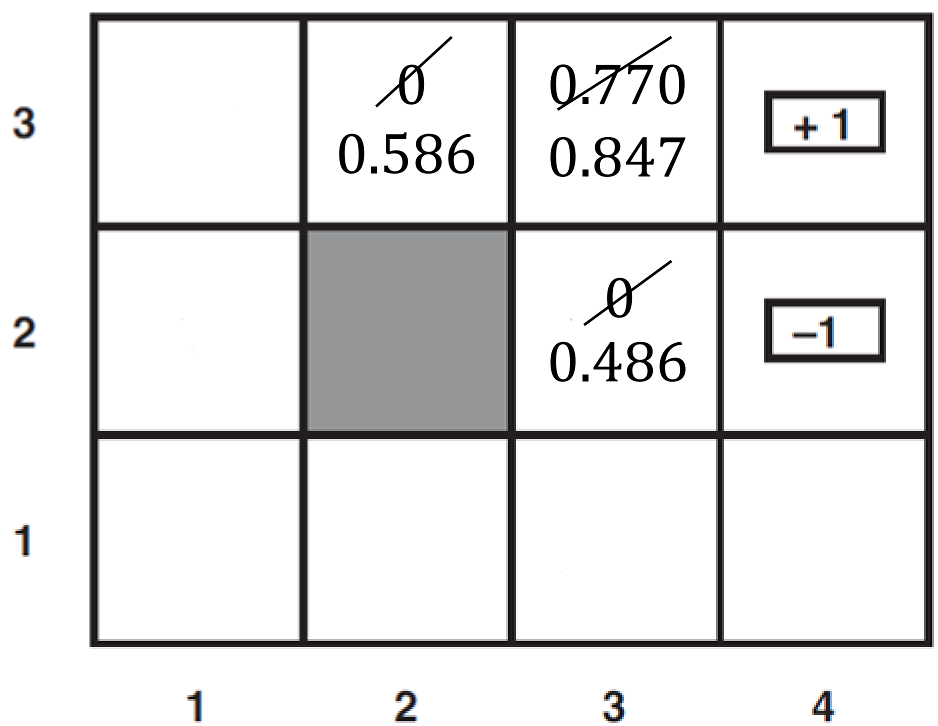
3	0.91	0.94	0.97	+1
2	0.88		0.94	–1
1	0.85	0.88	0.91	0.88
	1	2	3	4

$U(3,3)$ = ?
$U(3,2)$ = ?
$U(3,1)$ = ?
.
.
$U(1,1)$ = ?

Value Iteration in a Stochastic World

- Start with $U(s)$ = 0 for all states except for absorbing ones. Take R(s)=-0.03 and γ=1. p=(0.8,0.1,0.1). Apply value iteration

$$U_{i+1}(s) = R(s) + \gamma \max_{a} \sum_{s'} P(s'|s,a) U_i(s')$$

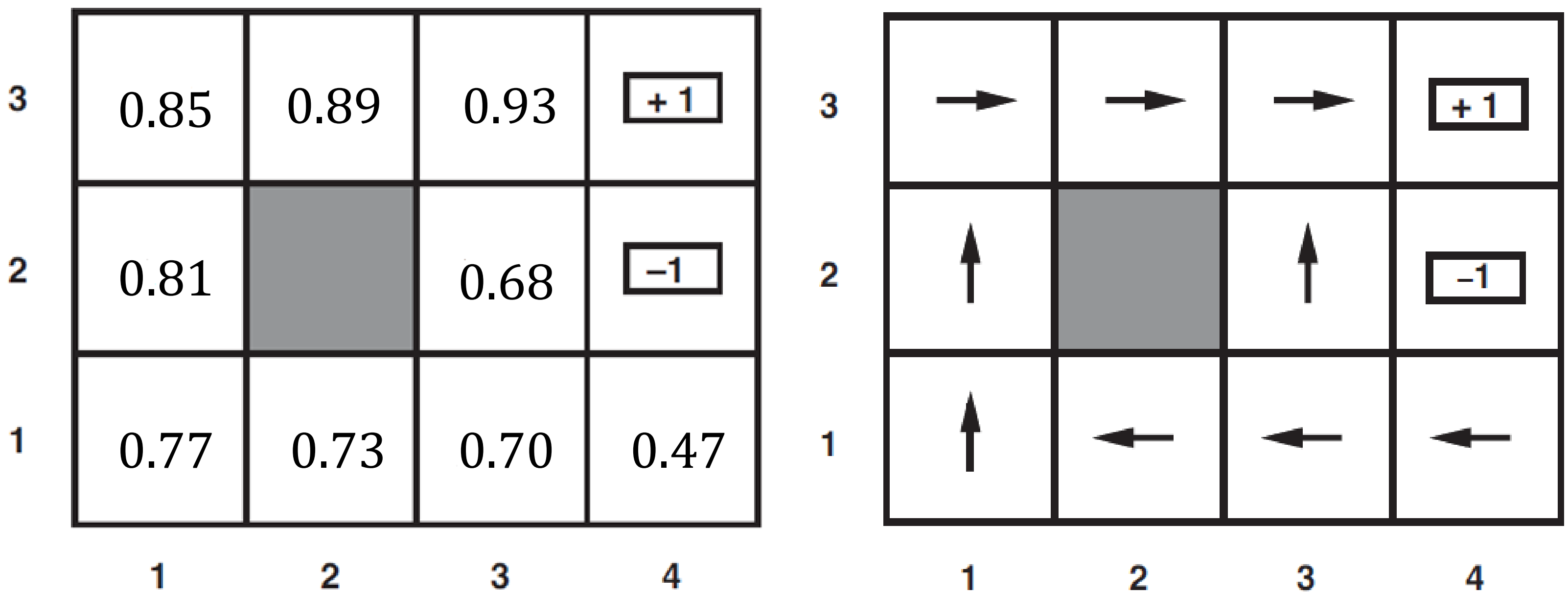


$U(3,3)$ = ?
$U(3,2)$ = ?
$U(2,3)$ = ?
At first iteration?

At second iteration?

Value Iterations and Policy

Converged value iteration algorithm and corresponding policy



Review

- Fully observable $s_1, \ldots, s_N, a_1, \ldots, a_K$
- Stochastic $P(s'|s, a)$
- Reward $R(s)$
- Objective:

$$\max \ E[\sum_{t=0}^{\infty} \gamma^t R(s_t)]$$

- Value iteration:

$$U_{i+1}(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) U_i\ (s')$$

- Policy: Converged value iteration provides us a solution called a policy $\pi(s)$.

Partially Observable MDPs (POMDPs)

- When the environment is partially observable the agent has to alternate between
  - Information gathering actions
  - Goal oriented actions