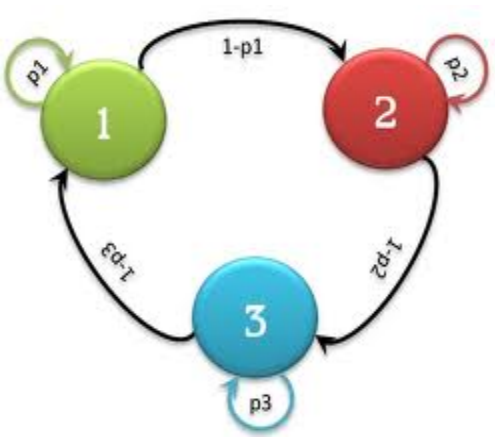
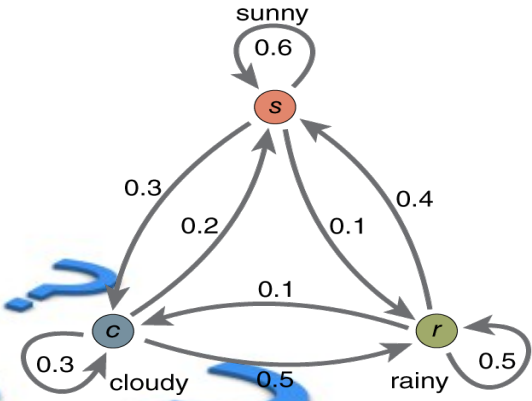
PARTIALLY OBSERVABLE MARKOV DECISION PROCESSES (POMDPS)

Presenter: Sharaf Malebary
Instructor: Dr. Ioannis Rekleitis
CSCE 790: ML. for Robotics

POMDPs ?!?!?!

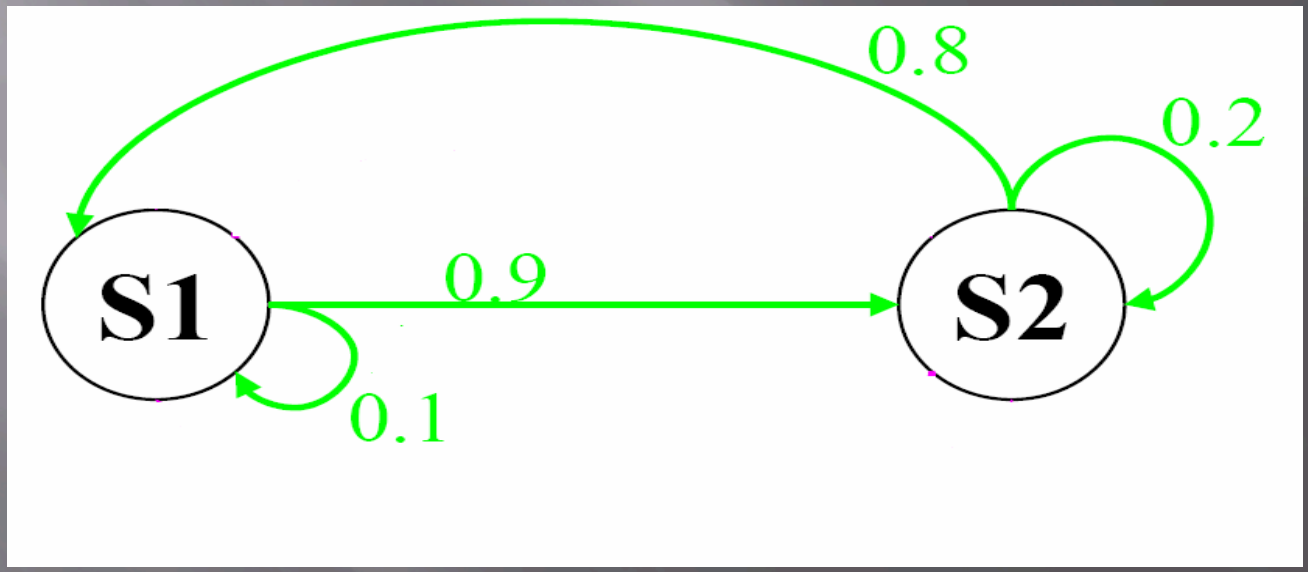




MC | HMM | MDP

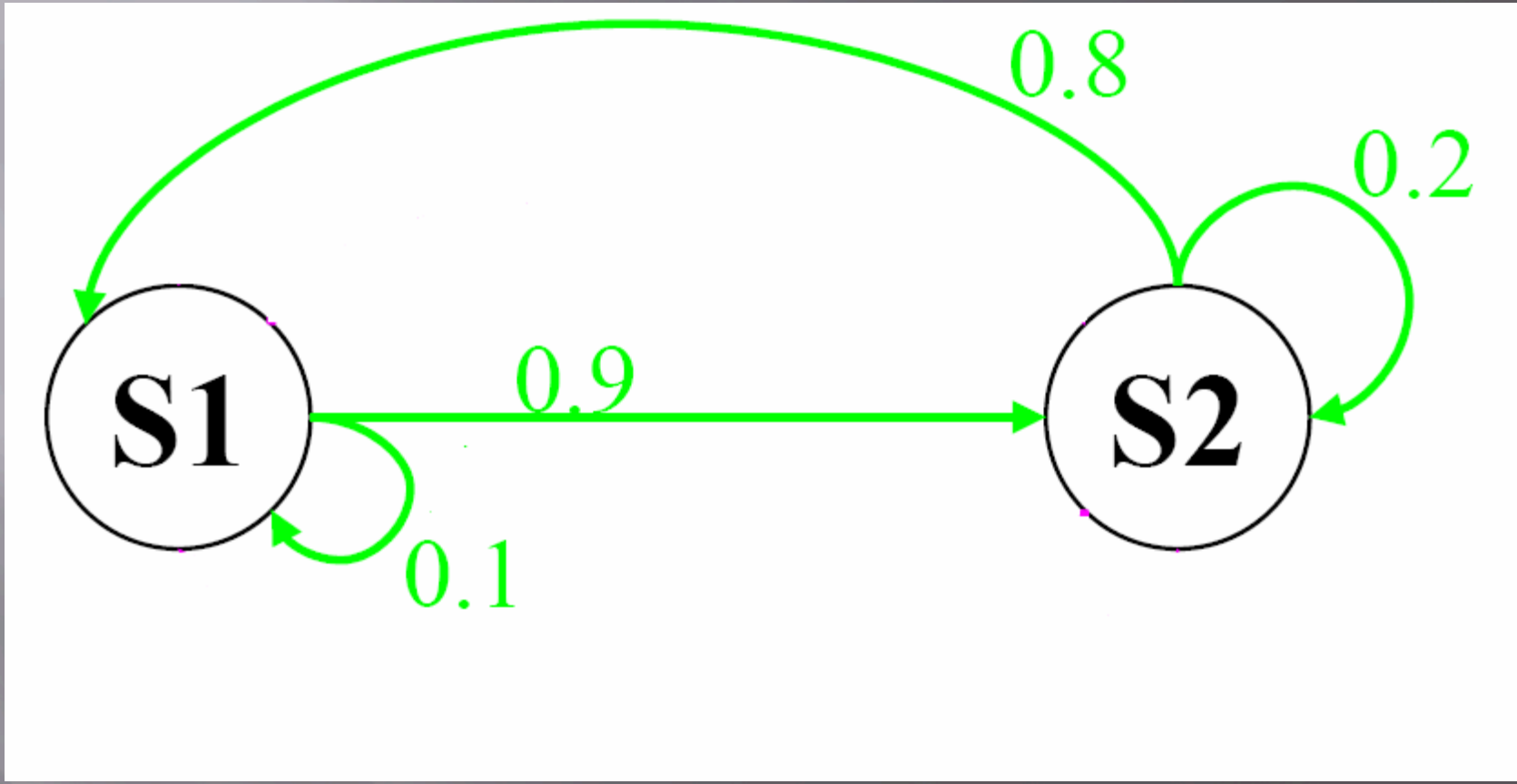
Markov Chain (MC)

- Finite number of discrete states.
- Probabilistic transitions between states.
- Next state determined only by current.



$$Rewards$$: S1 = 10, S2 = 0

Hidden Markov Model (HMM)

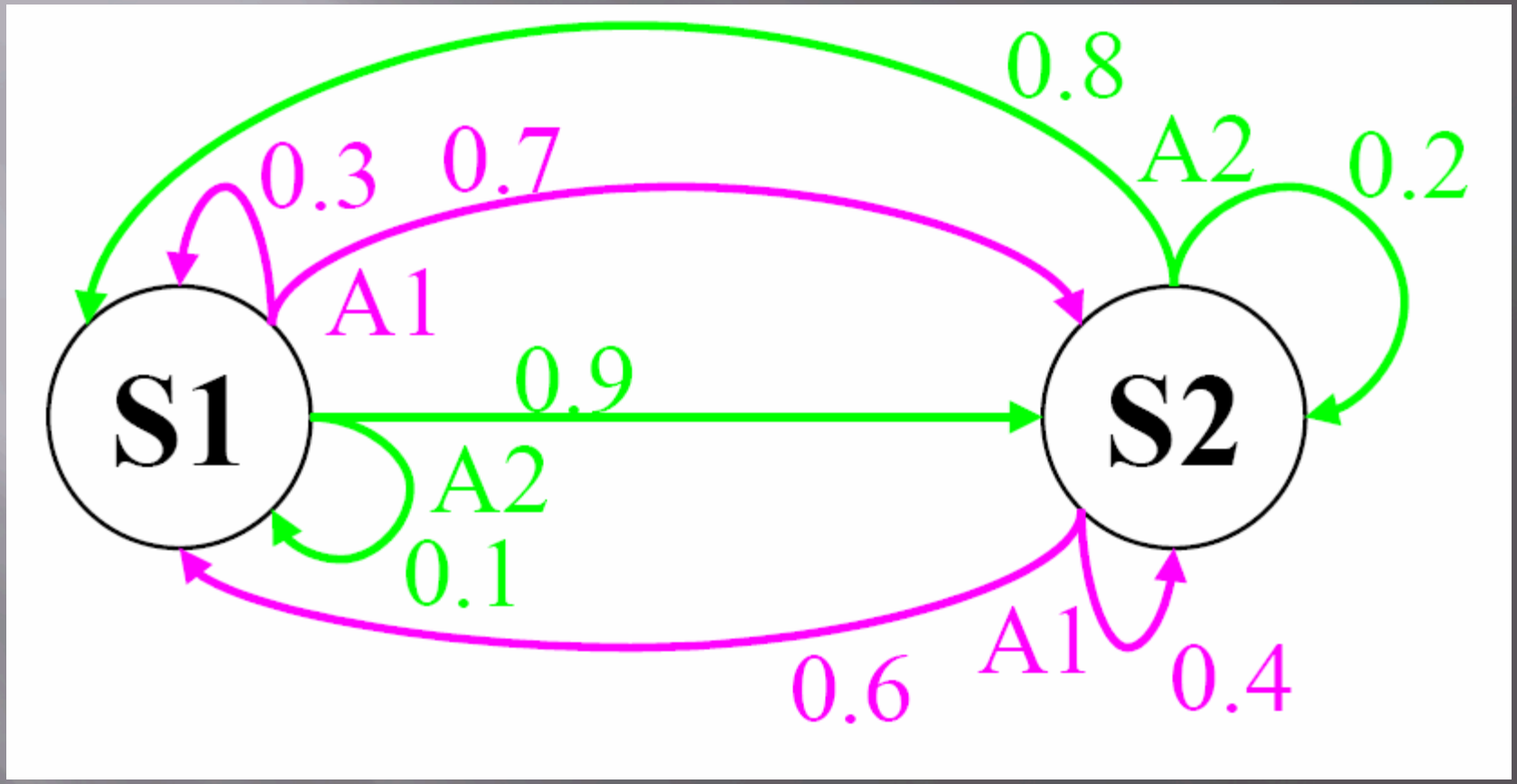


$$Rewards$$: S1 = 10, S2 = 0

S1 emits O1 with prob 0.75

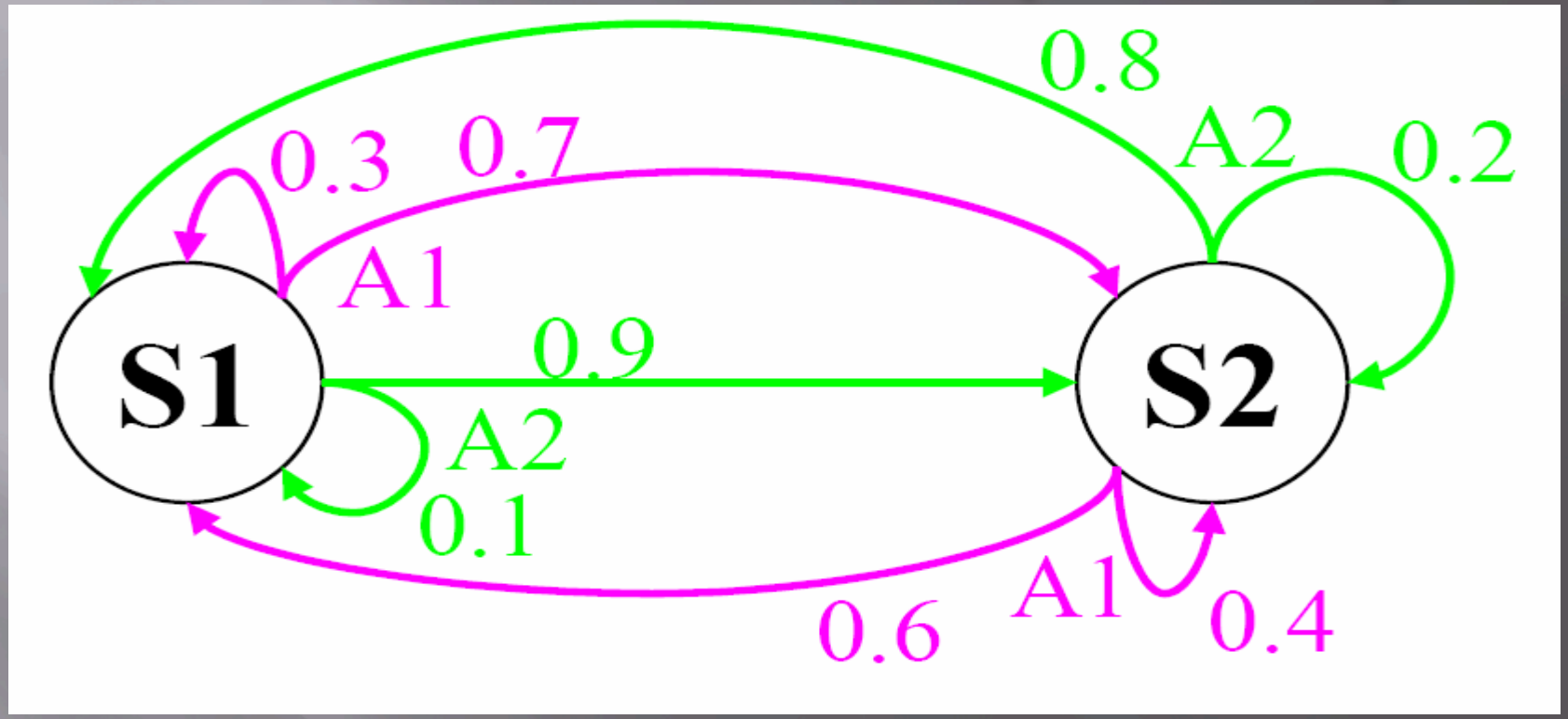
S2 emits O2 with prob 0.75

Markov Decision Process (MDP)



$$Rewards$$: S1 = 10, S2 = 0

Partially Observable Markov Decision Process (POMDP)



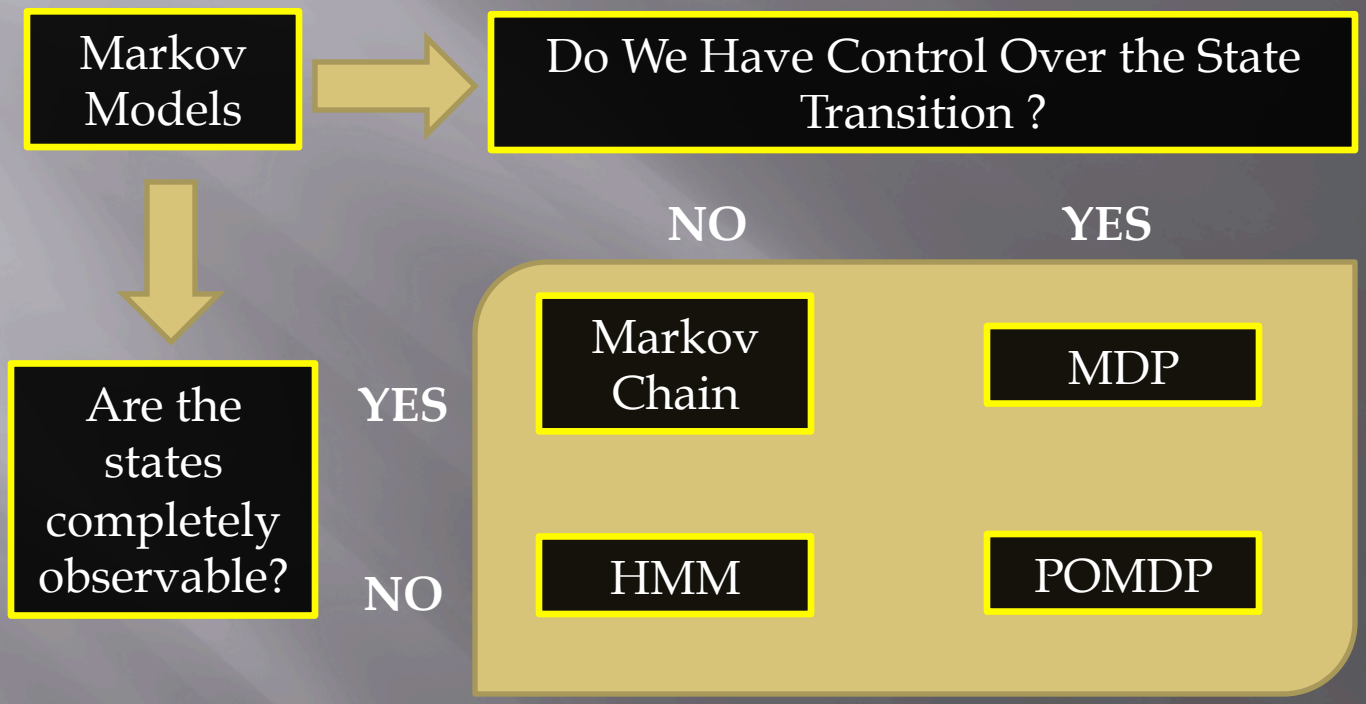
$$Rewards$$: S1 = 10, S2 = 0

S1 emits O1 with prob 0.75

S2 emits O2 with prob 0.75

Don't Know State

In General…



MDP vs. POMDP ?!!

POMDP vs. MDP

- **MDP**
  - +Tractable to solve
  - +Relatively easy to specify
  - Given: S, A, T, R. Goal to maximize the sum of r.
  - -Assumes perfect knowledge of state
- **POMDP**
  - +Treats all sources of uncertainty uniformly
  - +Allows for information gathering actions
  - Additionally given (MDP + conditional observation prob.)
  - -Hugely intractable to solve optimally

Time for some Formalism

- POMDP model:
  - Finite set of states $s_1, \ldots, s_n \in S$
  - Finite set of actions: $a_1, \ldots, a_m \in A$
  - Probabilistic state-action transitions: $p(s_i \mid a, s_j)$
  - Reward for each state/action pair: $r(s, a)$
  - Conditional observation probabilities: $p(o \mid s)$

Why POMDP in Robotics ….?

- Real robots in the field do not know their state exactly –this is a universal truth
- Control and planning methods so far have assumed knowledge of the state, x!
- Robots can be designed such that uncertainty is quite low in practice: we are sometimes safe
- This will always be a limitation, causing "dumb" actions in some cases (kill kitten)

Uncertainty……

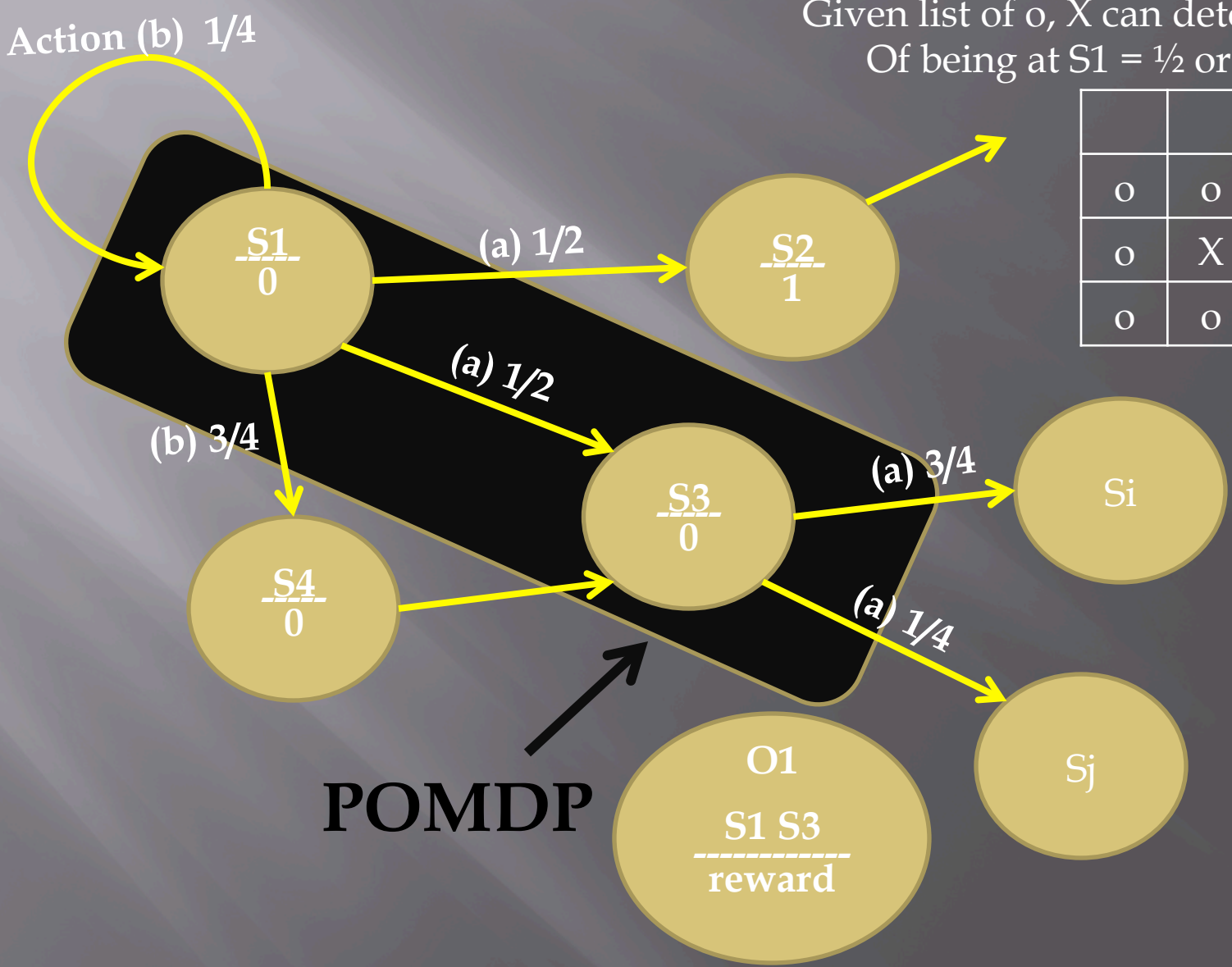
In summary:

POMDP = MDP but sensory measurements rather than knowledge of state

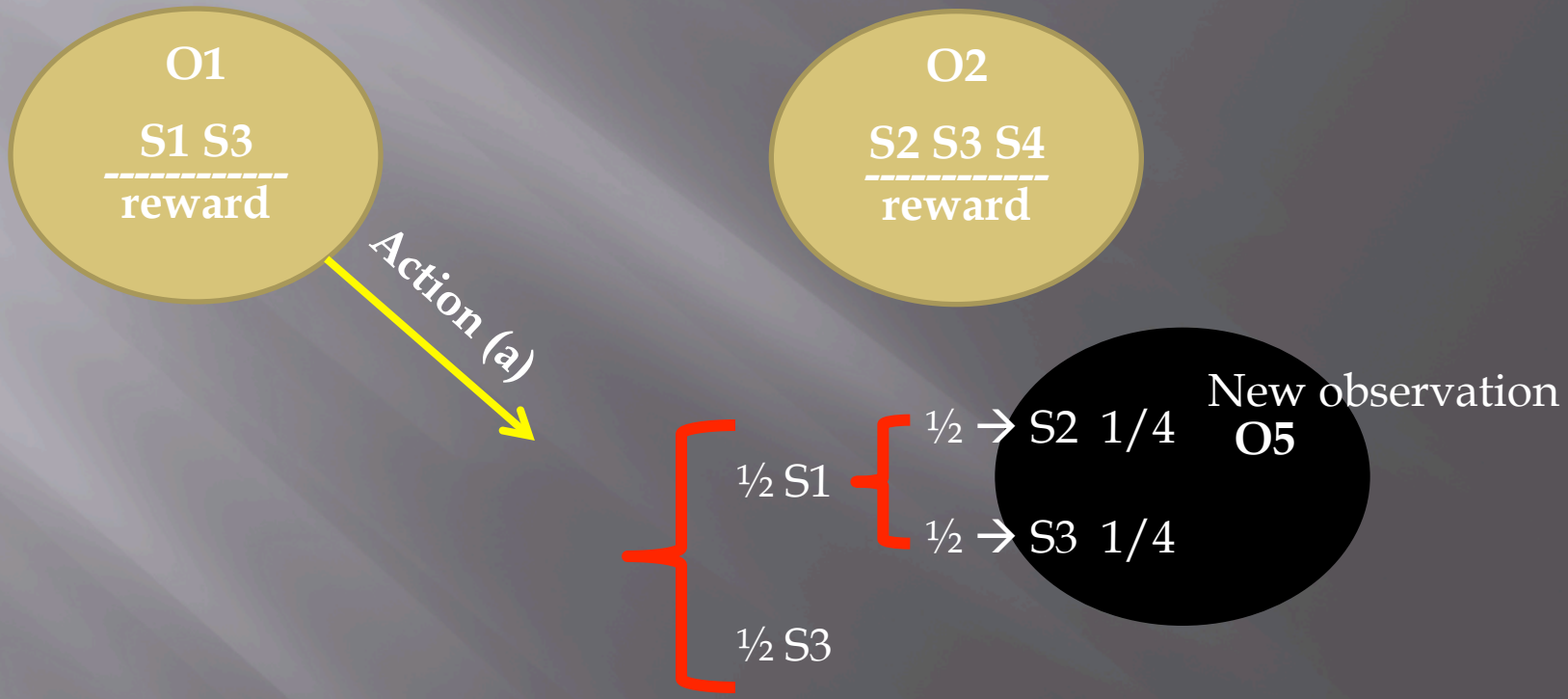
	Stochastic
Fully O.	MDP
P.O.	POMDP

Now let's take a look at an example

Let's start with MDP

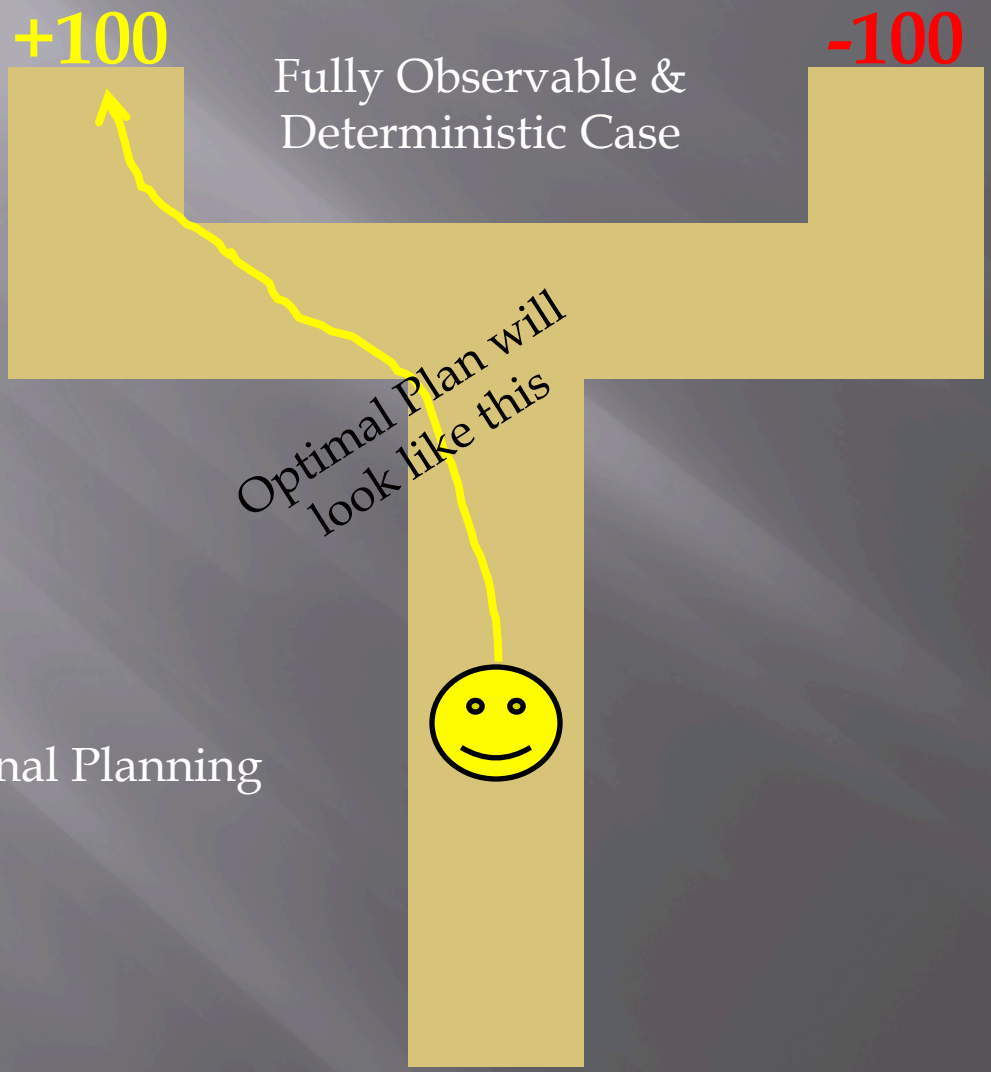


POMDP



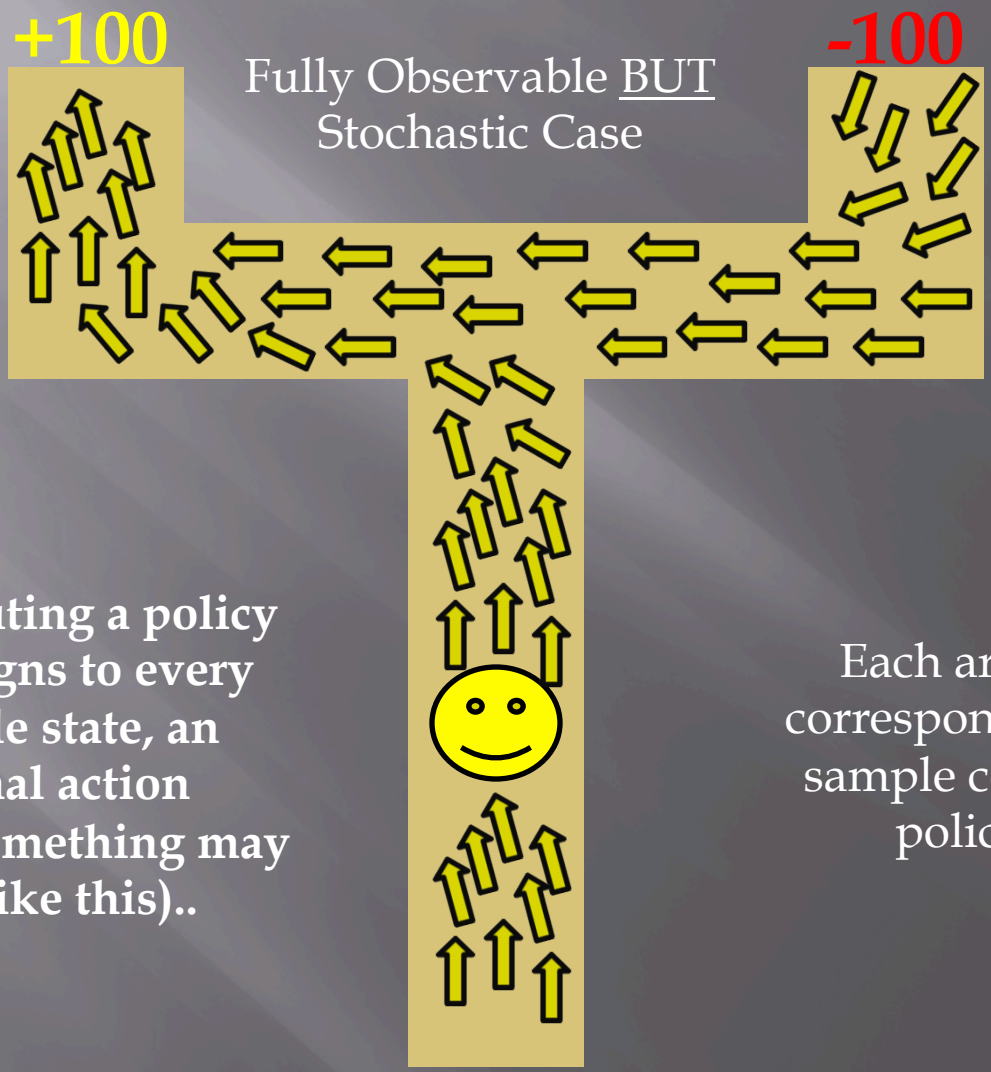
As you can see, the number of states can be infinity (extremely large). Using different Algorithms to reduce it to more manageable numbers.

Another Example



Conventional Planning

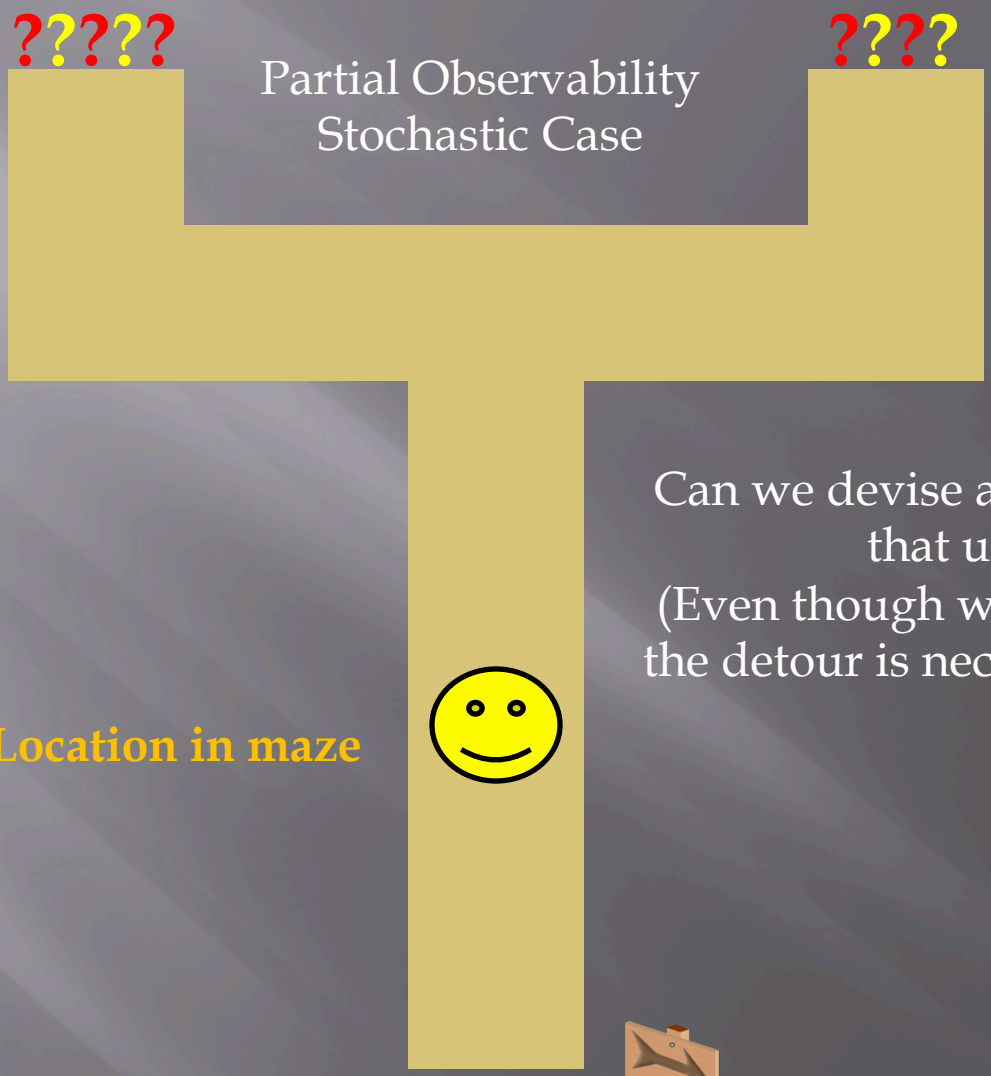
MDP with value iteration



By computing a policy that assigns to every possible state, an optimal action
We get (something may look like this)..

Each arrow corresponds to a sample control policy

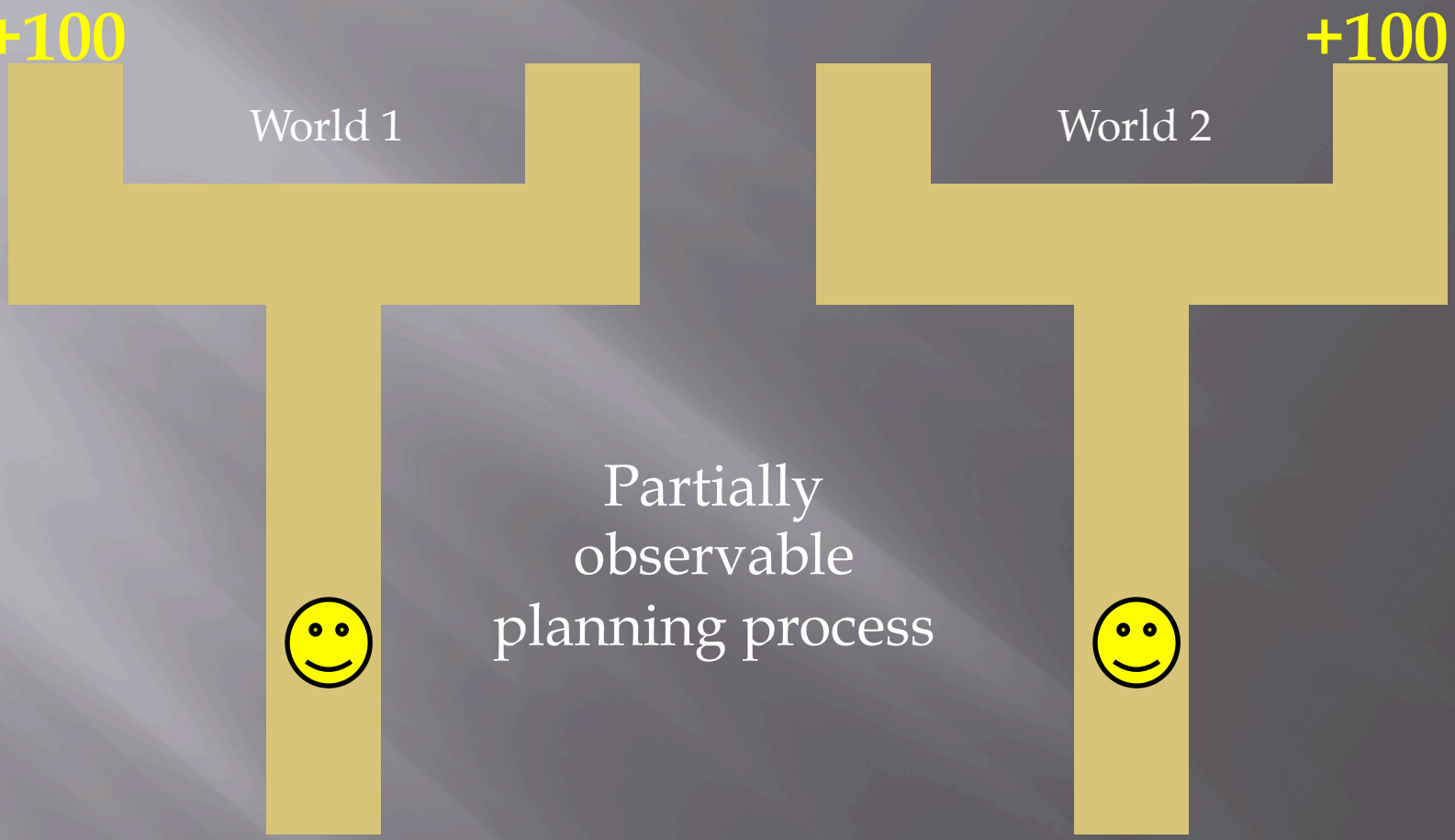
Partial Observability
Stochastic Case



Can we devise a method for planning that understands:
(Even though we want to receive 100, the detour is necessary to gather info)?

Observable Location in maze

Optimal policy to go south, check sign, and then head to target.
Exclusively goes south to gather info.



Partially observable planning process

What doesn't work is, to solve these two problems then put the solution together; for example by averaging!!!!
What works is ……

Information/Belief Space

- Planning not in the set of physical world states, but what we might know about those states.(Multitude of BS)
- If we move around and either reach one of the exits or the sign, then we know for sure where +100 → belief state change.
- Belief state Formally:
  - Probability distribution over world states: $b(s) = p(s)$
  - Action update rule: $b'(s) = \sum_{s' \in S} p(s \mid a, s') \cdot b(s')$
  - Observation update rule: $b'(s) = p(o \mid s) \cdot b(s)/k$

POMDP as Belief-State MDP

- Equivalent belief-state MDP
  - Each MDP state is a probability distribution (continuous belief state b) over the states of the original POMDP
  - State transitions are products of actions and observations

$$b'(s') = \mathrm{p}(s' \mid a, o, b) = \mathrm{p}(o \mid s', a, b) \cdot \mathrm{p}(s' \mid a, b) / \mathrm{p}(o \mid a, b)$$

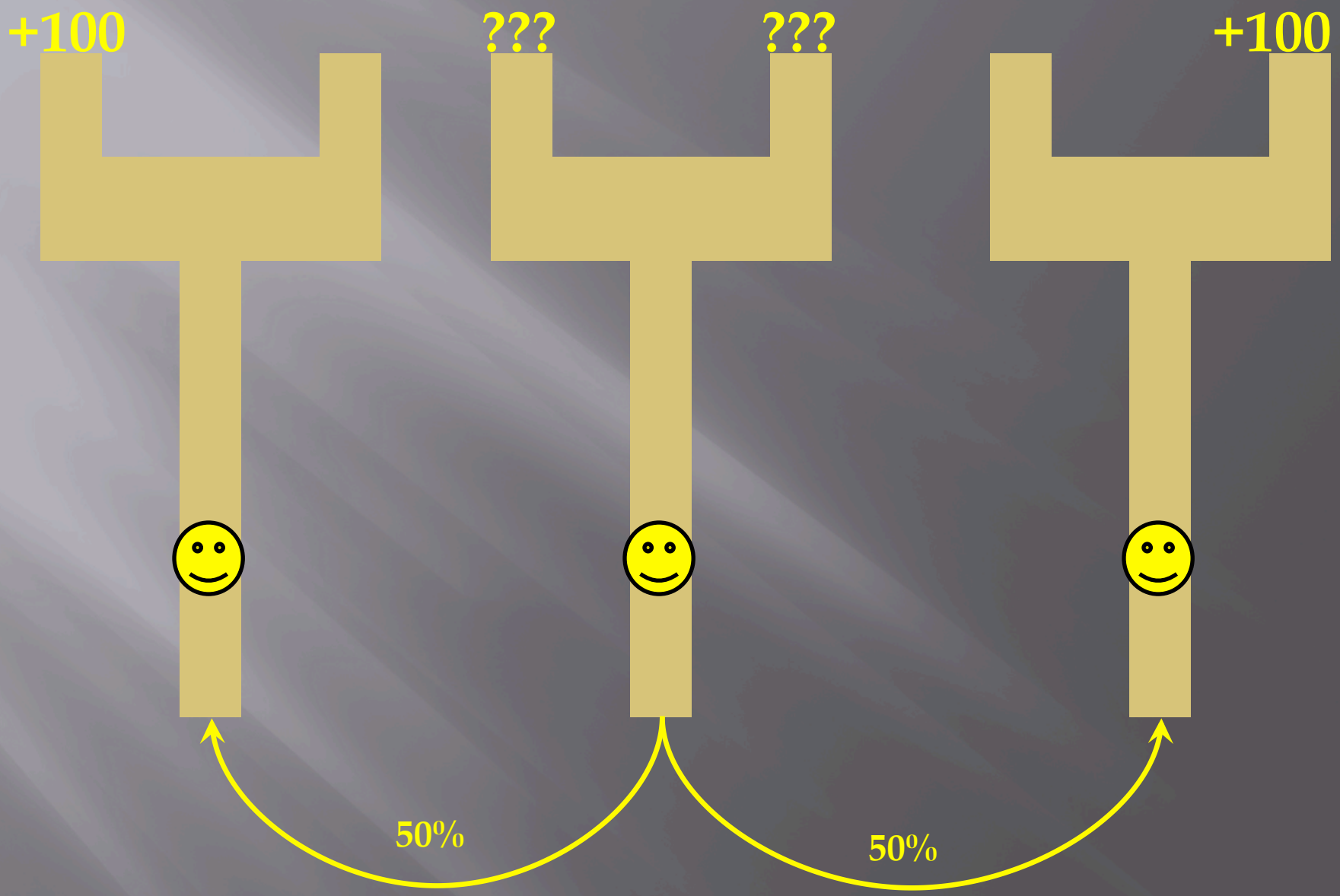
$$\mathrm{p}(o \mid s', a, b) = \mathrm{p}(o \mid s')$$

$$\mathrm{p}(s' \mid a, b) = \sum_{s \in S} \mathrm{p}(s' \mid a, s) \cdot b(s)$$

$$\mathrm{p}(o \mid a, b) = \sum_{s' \in S} \mathrm{p}(o \mid s') \cdot \mathrm{p}(s' \mid a, b)$$

  - Rewards are expected rewards of original POMDP

$$\mathrm{R}(a, b) = \sum_{s \in S} \mathrm{r}(a, s) \cdot b(s)$$


+100
???
???
+100
50%
50%

Summary for up to now

- In POMDPs we apply the very same idea as in MDPs.
- Since the state is not observable, the agent has to make its decisions based on the belief state which is a posterior distribution over states.
- Let *b* be the belief of the agent about the state under consideration.
- POMDPs compute a value function over belief space:

$$V_T(b) = \gamma \max_u \left[r(b,u) + \int V_{T-1}(b') p(b' \mid u, b)\, db' \right]$$

Refrences

- - POMDPs, Geoff Hollinger.
- Brief Intro to ML, Jingwei Zhang.
- Planning Under Uncertainty: POMDPs, Mcgill.
- POMDPs, Pieter Abbeel.
- Mr. Google.

Thank you