Decision Making for Robots and Autonomous Systems

Fall 2015

DUKE COMPUTER SCIENCE

George Konidaris
gdk@cs.duke.edu

The Markov Assumption

The definition of a state:

- Sufficient statistic of past history,
- For predicting *s'* and *r*

$$T(s_{t+1}|s_t, a_t, s_{t-1}, a_{t-1}, ..., s_0) = T(s_{t+1}|s_t, a_t)$$

$$R(s_{t+1}, s_t, a_t, s_{t-1}, a_{t-1}, ..., s_0) = R(s_{t+1}, s_t, a_t)$$

That is what the state *means*.

Very strong assumption: the agent has access to state.

Markov and Robots

Does the robot see everything it needs to be able to predict the effects of its own actions?

Example

Example

Generally

Limited perception:

- Only get a *single view* of the world at a time
- Does not contain everything you need
- Comes from noisy sensors
- Might be aliasing

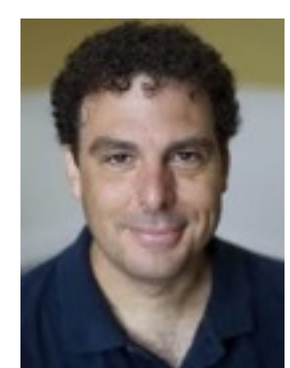
Important questions:

- How do we think about state?
- What do we really need?
- How can we estimate it?
- How can we plan without direct access to it?

POMDPs

Partially observable Markov decision processes:

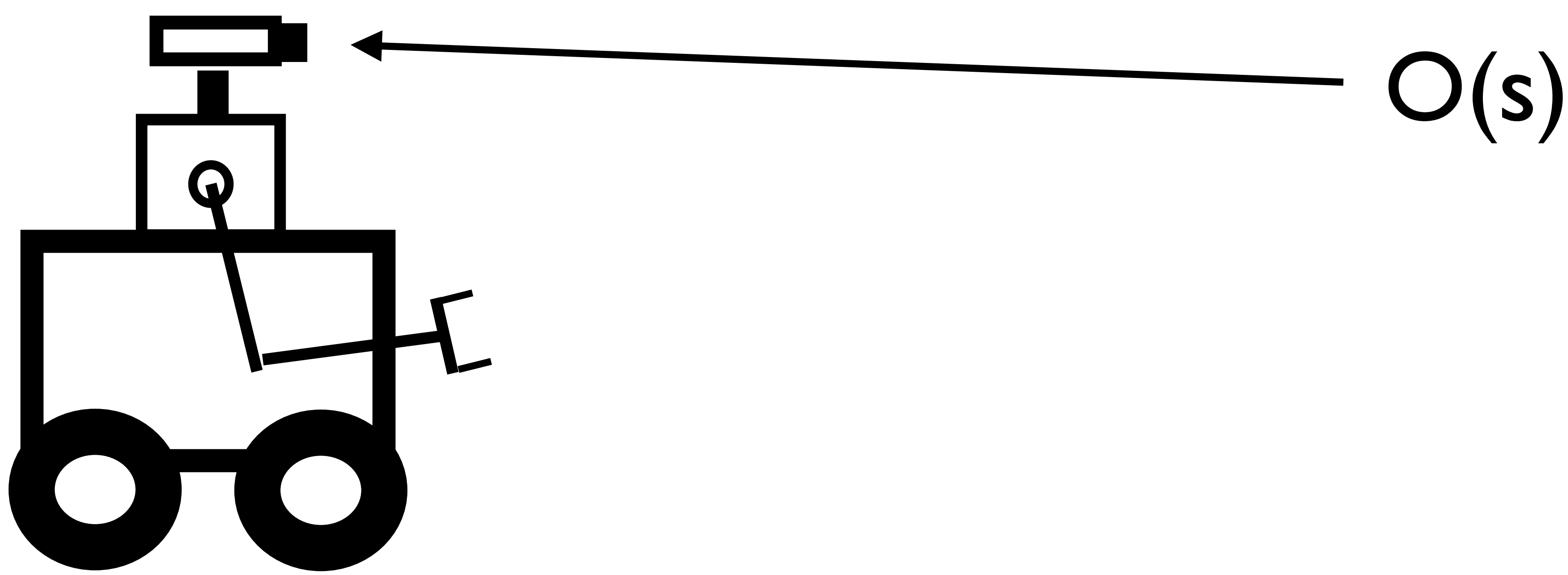
- Formalism for the non-Markov case
- Decision making *under state uncertainty*
- State uncertainty is unavoidable in real life
- *The* central theoretical objects for robotics


POMDPs

General idea:

- *There is an MDP.*
- Agent does not observe state directly
- Instead, observations!
- Observations probabilistically generated from state.



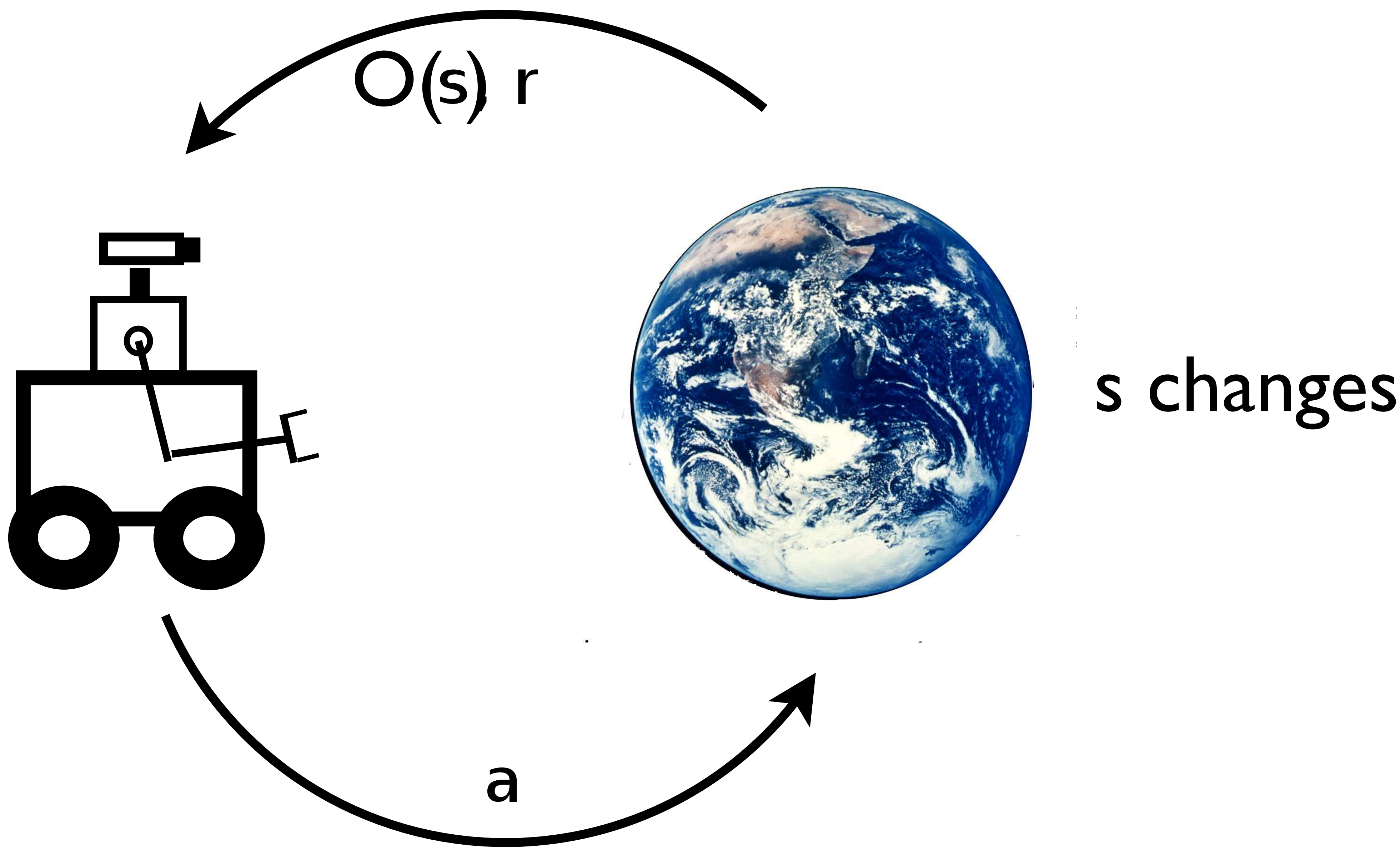
POMDPs

More formally, a POMDP is:

- S, a set of states
- A, a set of actions
- T, transition function
- R, reward function
- γ , discount factor
- Ω, set of observations
- O, observation function $O(\omega_t|s_t)$

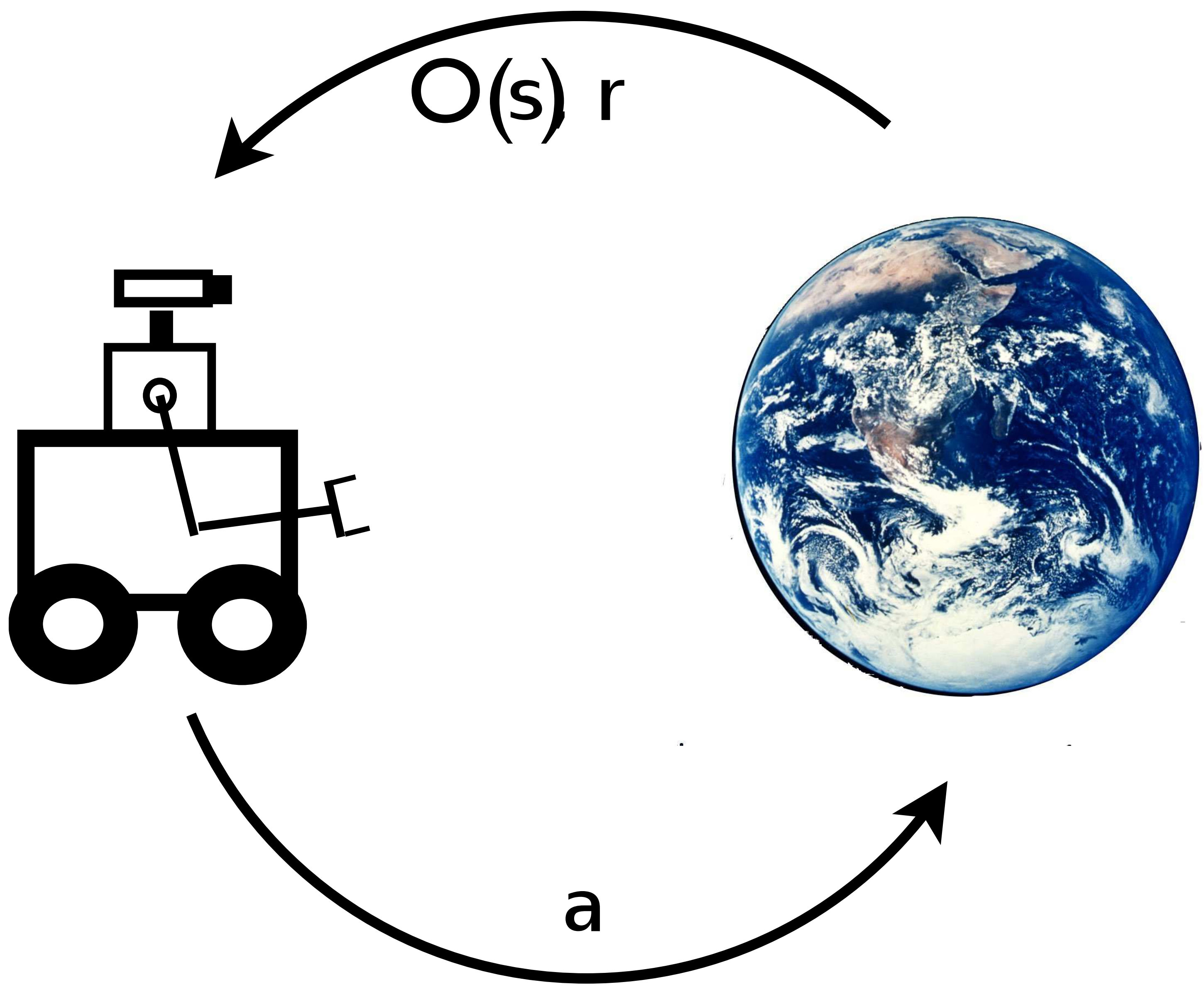
MDP

POMDPs



Robots

A robot is a device that induces a POMDP.



POMDPs

So:

- Environment is in state s_t
- Agent takes action a_t
- Environment transitions to state s_{t+1}
- Agent observes only $o_{t+1} = O(s_{t+1})$ and reward r.

So how to pick actions?

- Might need to take information seeking actions.

Objective is still to produce a **policy**, but now it cannot be a mapping from states to actions, because we do not have the state.

Policies Based on Histories

One approach is to write a policy as function of the agent's history:

- $\pi(a_t | o_t, o_{t-1}, a_{t-1}, ..., o_0, a_0)$

This is a little problematic because this is a function of an input that is of variable (and unboundedly growing) size.

Common approach: *kth* order Markov:

- $\pi(a_t | o_t, o_{t-1}, a_{t-1}, ..., o_{t-k}, a_{t-k})$

… this is like assuming that the last *k* observations are sufficient to specify the state (short term memory).

Belief State

Another approach:

- Estimate state using observations
- Belief state: distribution over states, *b(s)*
- Update based on observations
- Distribution represents state uncertainty
- Take action based on distribution

$$b(s_t) = P(s_t | o_t, o_{t-1}, a_{t-1}, ..., o_0, a_0)$$

Must implement a ***Bayes filter***.

Belief State Updates

We can update *b(s)* at each time step using Bayes' Rule.

$$\underbrace{b(s_{t+1})}_{\text{new belief}} \propto \underbrace{O(o_{t+1}|s_{t+1})}_{\text{observation function}} \sum_{s \in S} \underbrace{T(s_{t+1}|s, a_t)}_{\text{transition function}} \underbrace{b(s)}_{\text{old belief}}$$

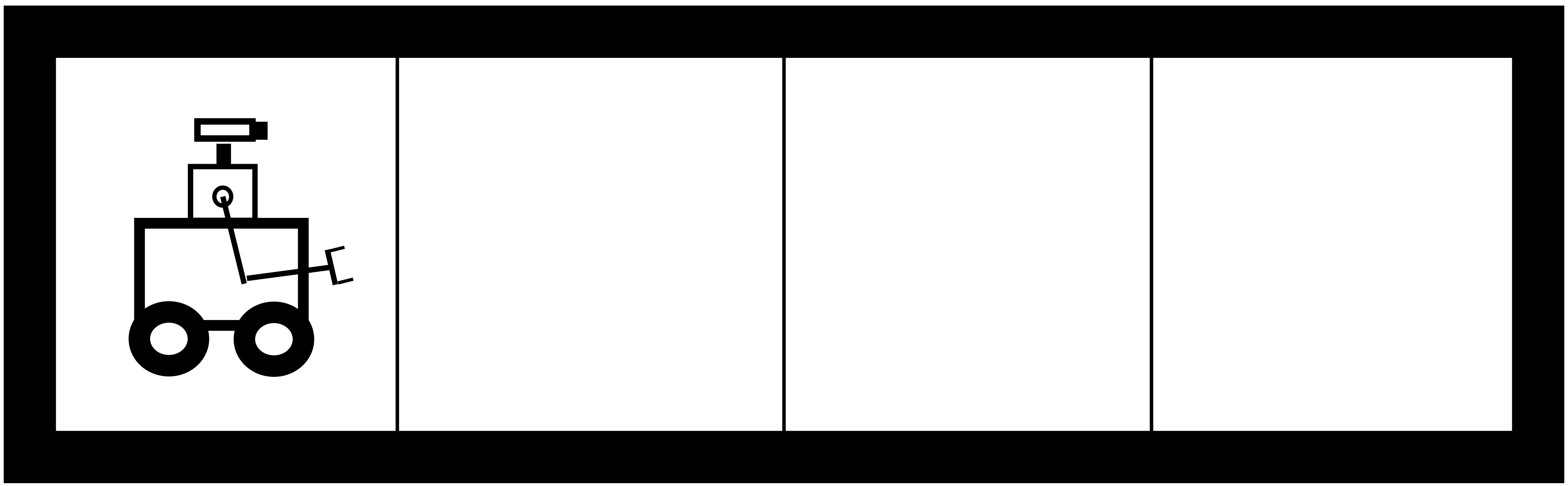
new belief

observation function

transition function

old belief

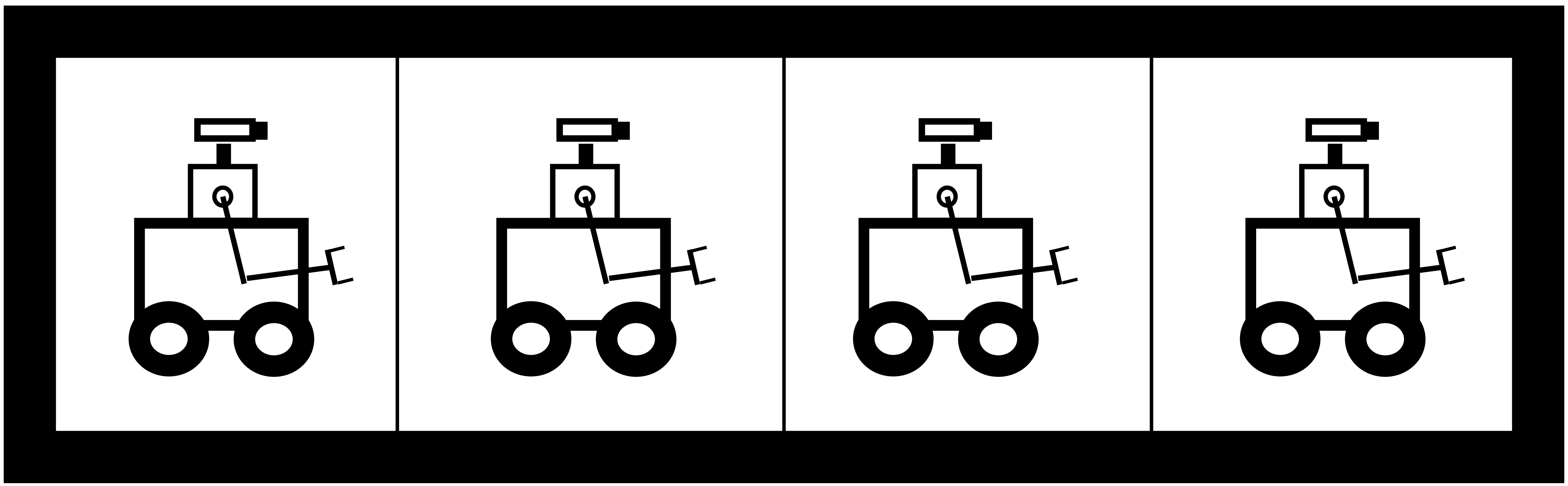
Example



observations:
walls each side?
(assume perfect sensing)

states:
position

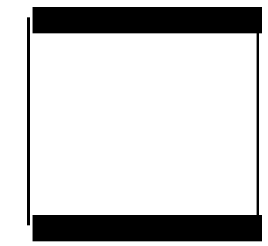
Example

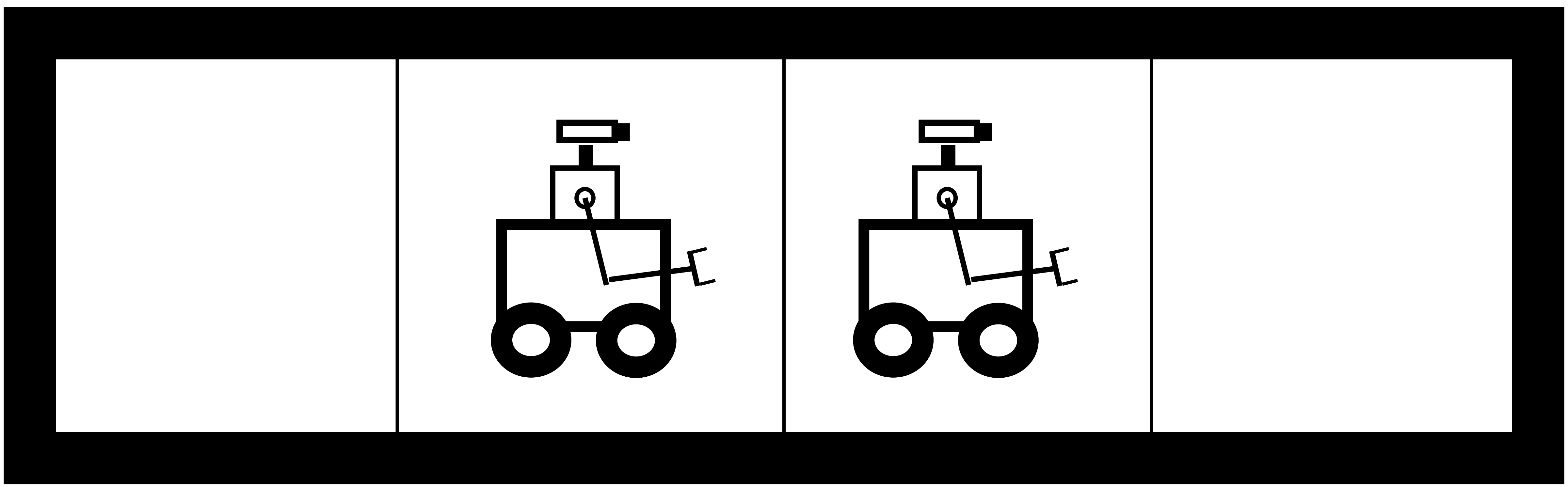


We start off not knowing where the robot is.

uniform distribution over positions

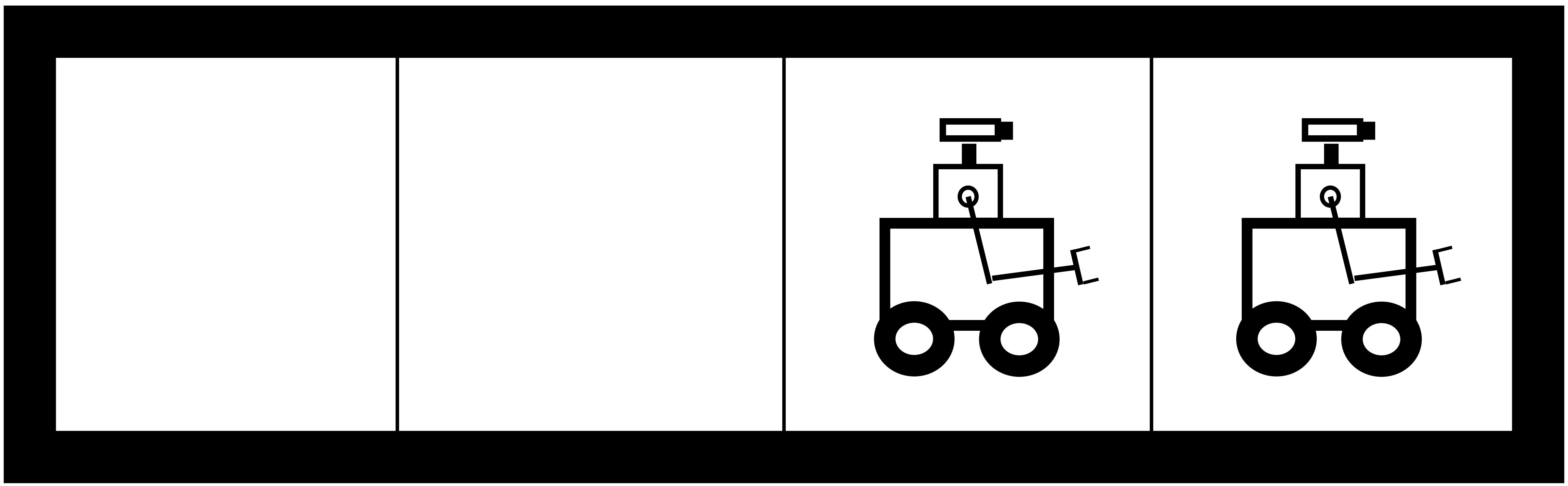
Example

first sensor reading: 



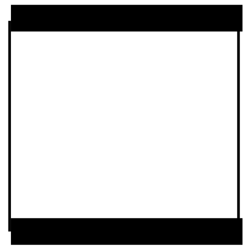
New distribution.

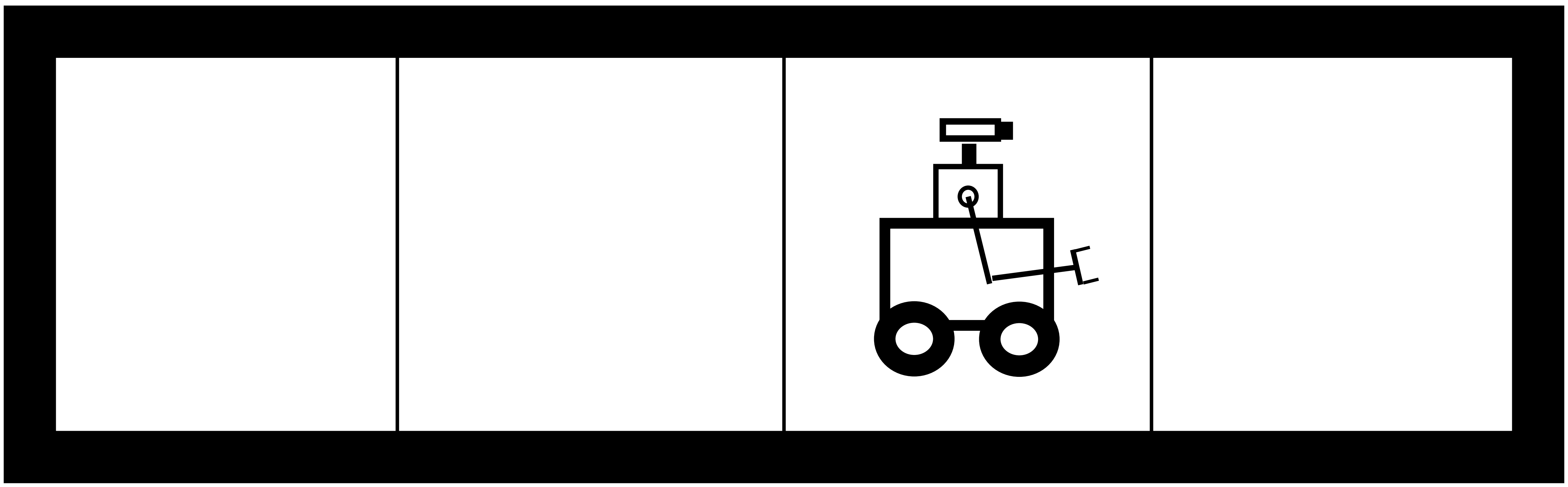
Example



Robot moves right
(pre-observation distribution)

Example

second sensor reading: 



Post-observation distribution.

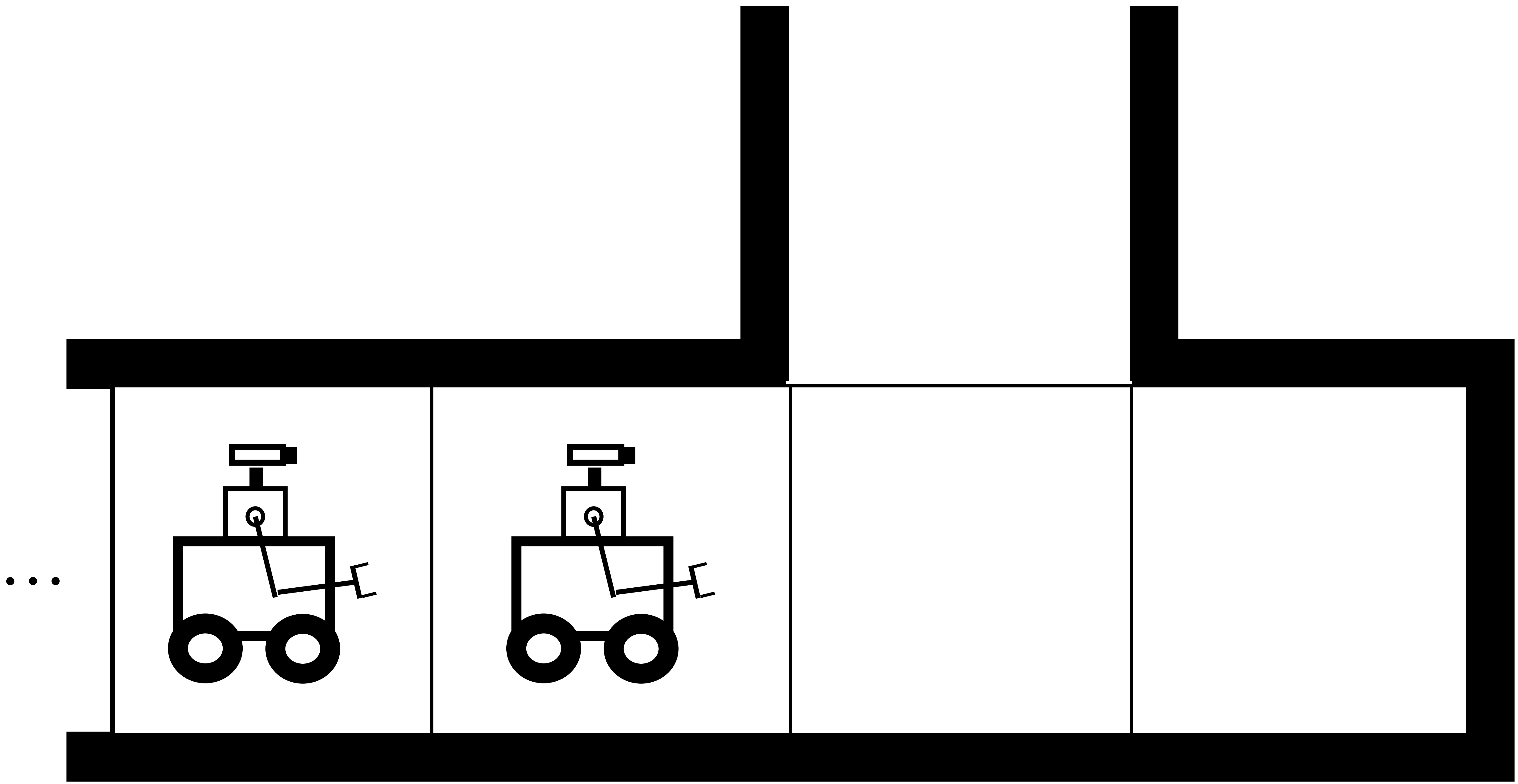
So

We can represent a belief about the world:

- Distribution over states
- Reflects best estimate given observations
- Formulation so far requires:
  - Knowledge of form of states
  - Knowledge of observation function
  - Knowledge of transition function

… even given these, **solving POMDPs is hard.**

Final Thought



What do you do?