Lecture 4

HRI, FALL '25

Andrea Bajcsy

Last Time:

- ☐ Bellman Eqn.
- ☐ Value Iteration + RL

This Time:

- ☐ RL summary
- ☐ POMDPs

→ HW #1 has been released on Canvas! Due: Sept. 18

→ Yilin's OHs: Mondays, 11am - 12pm in NSH 4306

Reinforcement learning (RL)

$\boxed{Q}$ In MDPs, we assumed we know $P(s'|s,a)$ and rewards $r(s,a)$ .... but what if we don't?

In reality, it's hard to know P and r explicitly, so how do we obtain π^*?

$\boxed{A}$ RL! It allows an agent to learn optimal policies by interacting with the environment.

⇒ agent tries out actions, observes rewards, and updates their policy to maximize rewards.

Model-based vs. Model-free RL

↳ $\boxed{\text{High-level diff}}$ if the agent has access to (or learns) a $\overbrace{\text{transition function}}^{\text{"model of the world"}}$ + rewards

(A) Model-based: 1) collect data from environment via some π

$$\mathcal{D} := \{(s_t, a_t, s_{t+1}, r_t)_{t=1}^{N}\}$$

2) use supervised learning to fit empirical MDP model:

↳ count s' for each s, a; normalize; get $\hat{P}(s'|s,a)$

↳ discover each $\hat{r}(s,a,s')$ when we experience (s,a,s')

3) solve MDP via techniques above (+more!)

(B) Model-free: 1) collect data from environment via some π

$$\mathcal{D} := \{(s_t, a_t, s_{t+1}, r_t)_{t=1}^{N}\}$$

2) Directly learn a $\underbrace{\text{Q-function}}_{\text{"Q-learning"}}$ or $\underbrace{\pi}_{\text{"direct policy search"}}$ via trial + error

⇒ for more depth on topic see:

- Sutton & Barto, "Reinforcement Learning."
- Russell & Norvig, "AI: A Modern Approach". $\boxed{\text{Ch. 21}}$

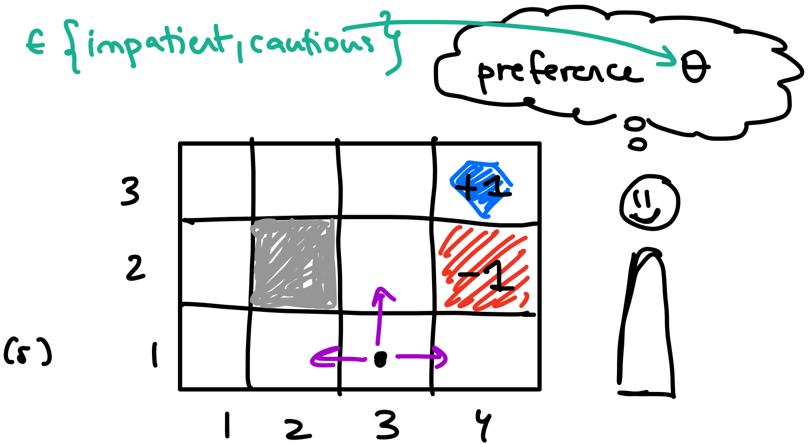
Partially-Observable Markov Decision Processes (POMDPs)

POMDPs extend MDPs to settings where the agent cannot directly observe the true state. Instead, the agent must infer the state from noisy observations.

Formally, a POMDP is a tuple

$$\mathcal{M} = < \overset{\checkmark}{S}, \overset{\checkmark}{A}, \overset{\checkmark}{T}, \overset{\checkmark}{r}, \underbrace{\Theta, Z}_{\text{new!}}, \overset{\checkmark}{\gamma} >$$

- states: $s \in S$ (partially observable)
- actions: $a \in A$



$s := (x, y, \theta)$

$b(s) = P(s)$

- transition function: $T: S \times A \to S$
  $T(s,a,s')$ or $P(s'|s,a)$
- reward function: $r: S \times A \times S \to \mathbb{R}$
  $r(s,a,s')$ or $r(s,a)$, $r(s)$
- observations: $o \in \Theta \rightarrow$ observation space
- observation function: $Z: S \times A \to \Theta$
  $Z(s',a,o')$ or
  $P(o'|s',a)$ or
  $z(s,o) = P(o|s')$ } simplification!

observation depends on (1) resulting state (since the world determines what can be observed)

(2) previous action (since actions affect what can be perceived)

- discount $\gamma \in [0,1]$

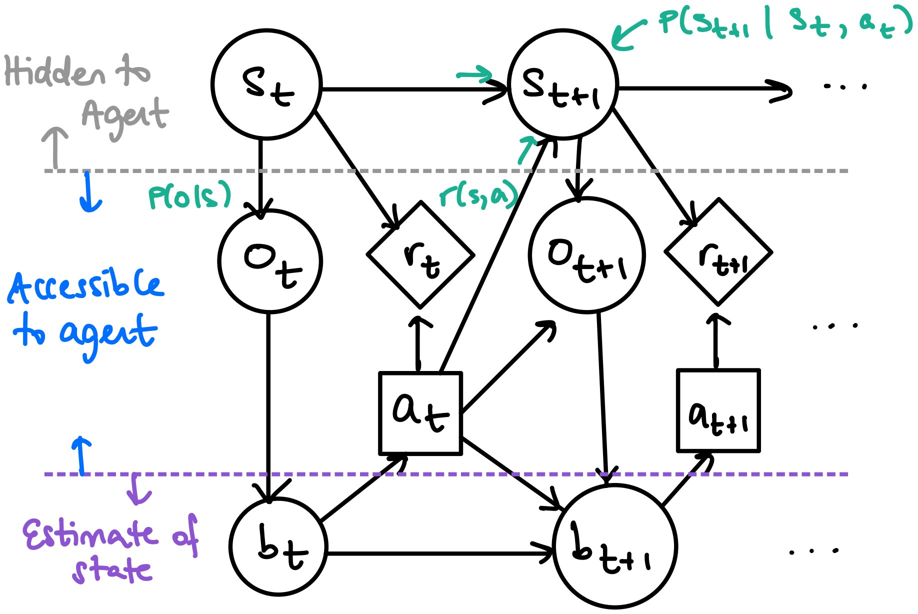
Why POMDPs? Many real-world problems have partial observability

→ ex. self-driving car doesn't know intentions of pedestrians & must infer them from observations $o \in \Theta$.

A key concept in POMDPs is the belief state. Since the R doesn't know s, it can maintain a belief state $b(s)$ as an estimate of state! It updates this belief over time via Baye's Rule after taking actions + getting observations.

Belief state: $b(s) = P(s)$

a probability distribution over world states



ASIDE: graphical notation from decision networks / influence diagrams

○ = chance node (i.e. rand var).

◇ = "utility" node

□ = decision node