Reinforcement Learning May 29, 2019

Recitation 11

Lecturer:Yishay Mansour *TA: Lee Cohen*

POMDP

Definition 1. A partially observable Markov Decision Process (**POMDP**) can be described as a tuple $\langle S, A, T, R, \Omega, O \rangle$, where:

- S, A, T , and R describe a Markov decision process.
- Ω is a finite set of observations the agent can experience of its world.
- $O : S \times A \to \Pi(\Omega)$ is the observation function, which gives, for each action and resulting state, a probability distribution over possible observations (we write $O(o|a, s')$ for the probability of making observation o given that the agent took action a and landed in state s').

Table 1: Model Categories

	Known States	Unknown States
Controllable	MDP	POMDP
Uncontrollable	MC	HMM

Definition 2. Belief State
Belief state b is a distribution over state, so $b \in [0, 1]^{|S|}$, and $\sum_s b(s) = 1$. We compute the next belief state as follows:

$$b'(s') = \frac{O(o|s', a) \sum_s T(s'|a, s) b(s)}{\Pr(o|a, b)} \tag{1}$$

We decompose the problem of controlling a POMDP into two parts, as shown in Fig. 1. The agent makes observations and generates actions. It keeps an internal belief state, b, that summarizes its previous experience. The component labeled SE is the state estimator: it is responsible for updating the belief state based on the last action, the current observation, and the previous belief state. The component labeled π is the policy: as before, it is responsible for generating actions, but this time as a function of the agent's belief state rather than the state of the world.

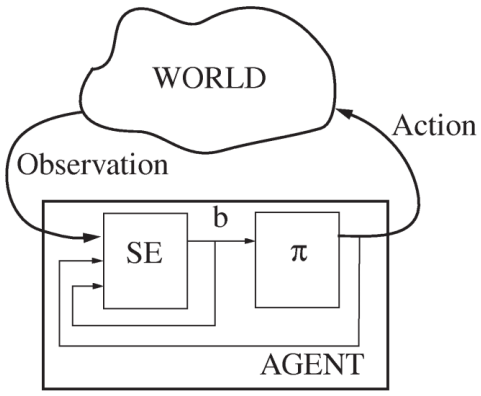


Figure 1: A POMDP agent can be decomposed into a state estimator (SE) and a policy (π)

Definition 3. "belief" MDP
The policy component of a POMDP agent must map the current belief state into action. Because the belief state is a sufficient statistic, the optimal policy is the solution of a continuous space "belief" MDP. It is defined as follows:

- B, the set of belief states, comprise the state space;
- A, the set of actions, remains the same;
- $\tau(b, a, b')$ is the state-transition function, which is defined as

$$\tau(b, a, b') = \Pr(b'|a, b) = \sum_{o \in \Omega} \Pr(b'|a, b, o) \Pr(o|a, b) \tag{2}$$

  where

$$\Pr(b'|b, a, o) = \begin{cases} 1 & \text{if} \quad SE(b, a, o) = b' \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

- $r(b, a)$ is the reward function on belief states, constructed from the original reward function on world states: $r(b, a) = \sum_{s \in S} b(s)R(s)$.

Definition 4. Value Iteration
We can run a value iteration in the POMDP (actually, on the belief states). We can write the update as follows:

$$\begin{aligned} V^*_{t+1}(b) &= \max_a [r(b, a) + \gamma \sum_o \Pr[o|b, a] V^*_t(T(b, a, o))] \\ V^*_{t+1}(b) &= \max_a V^a_t(b) \\ V^a_t(b) &= \sum_o V^{a,o}_t(b) \\ V^{a,o}_t(b) &= \frac{r(b, a)}{|O|} + \gamma \Pr[o|b, a] V_t(T(b, a, o)) \end{aligned}$$

Tiger Problem

Imagine an agent standing in front of two closed doors. Behind one of the doors is a tiger and behind the other is a large reward. If the agent opens the door with the tiger, then a large penalty is received (presumably in the form of some amount of bodily injury). Instead of opening one of the two doors, the agent can listen, in order to gain some information about the location of the tiger. Unfortunately, listening is not free; in addition, it is also not entirely accurate. There is a chance that the agent will hear a tiger behind the left-hand door when the tiger is really behind the right-hand door, and vice versa.

We refer to the state of the world when the tiger is on the left as s_l and when it is on the right as s_r. The actions are LEFT, RIGHT, and LISTEN. The reward for opening the correct door is +10 and the penalty for choosing the door with the tiger behind it is -100. The cost of listening is -1. There are only two possible observations: to hear the tiger on the left (TL) or to hear the tiger on the right (TR). Immediately after the agent opens a door and receives a reward or penalty, the problem resets, randomly relocating the tiger behind one of the two doors.

The transition and observation models can be described in detail as follows. The LISTEN action does not change the state of the world. The LEFT and RIGHT actions cause a transition to world state s_l with probability 0.5 and to state s_r with probability 0.5 (essentially resetting the problem). When the world is in state s_l, the LISTEN action results in observation TL with probability 0.85 and the observation TR with probability 0.15; conversely for world state s_r. No matter what state the world is in, the LEFT and RIGHT actions result in either observation with probability 0.5.

Finite-Horizon, $H = 1$

We first note that our value function receives as input the current belief state b. Therefore the belief interval is specified in terms of $b(s_l)$ only since $b(s_r) = 1 - b(s_l)$. We will first write down the value function of each action according to the action:

$$\begin{aligned} V_1^{\text{left}}(b) &= -100b(s_l) + 10b(s_r) = -100b(s_l) + 10(1 - b(s_l)) = -110b(s_l) + 10 \\ V_1^{\text{right}}(b) &= 110b(s_l) - 100 \\ V_1^{\text{listen}}(b) &= -1 \end{aligned}$$

We can plot the value function: The optimal policy for H=1 is given by:

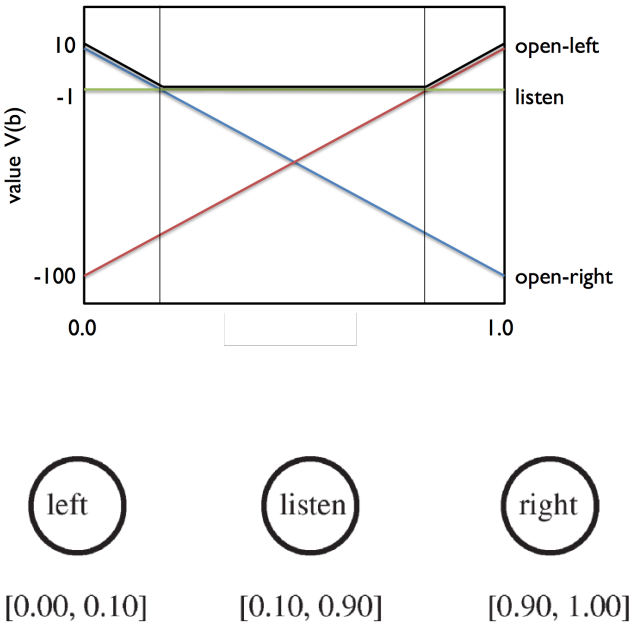


Figure 2: The optimal situation-action mapping for $t = 1$ for the tiger problem shows that each of the three actions is optimal for some belief state.

Finite-Horizon, $H = 2$

We will first compute probabilities that will be used by us later:

$$\begin{aligned}\Pr(TR|listen, b) =& \Pr(TR|s_l, listen)b(s_l) + \Pr(TR|s_r, listen)b(s_r) = 0.15b(s_l) + 0.85b(s_r) = \\ & 0.15b(s_l) + 0.85(1 - b(s_l)) = 0.85 - 0.7b(s_l) \\ \Pr(TL|listen, b) =& 0.15 + 0.6b(s_l)\end{aligned}$$

Now we can compute $V^{listen,TL}(b)$, since in V_1 we had a piece-wise linear function composed of 3 linear function - we will need to split the computation into 3 parts as well:

$$\begin{aligned}V_1^{listen,TL(\text{left})}(b) =& -\frac{1}{2} + \Pr(TL|listen, b)V_1^{\text{left}(b')} \\ =& -\frac{1}{2} + \Pr(TL|listen, b)[-110b'(s_l) + 10] \\ =& -\frac{1}{2} + \Pr(TL|listen, b)[-110\frac{\Pr(TL|listen, s_l)b(s_l)}{\Pr(TL|listen, b)} + 10] \\ =& -\frac{1}{2} + -110 * 0.85b(s_l) + 10(0.15 + 0.7b(s_l)) \\ =& 1 - 86.5b(s_l)\end{aligned}$$

In the same manner we can compute the following:

$$\begin{aligned}V_1^{listen,TL(\text{right})}(b) =& -15.5 + 23.5b(s_l) \\ V_1^{listen,TL(\text{listen})}(b) =& -0.65 - 0.7b(s_l)\end{aligned}$$

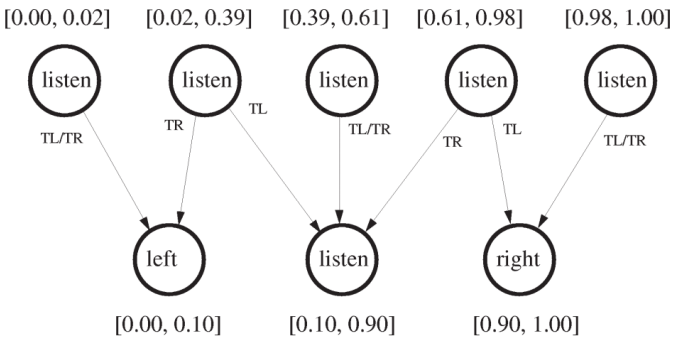
Using those function we can get the partition for $V_1^{listen,TL}(b)$:

$$\begin{aligned}0 \leq b(s_l) \leq 0.02 &: left \\ 0.02 < b(s_l) \leq 0.61 &: listen \\ 0.61 < b(s_l) \leq 1.0 &: right\end{aligned}$$

For $V_1^{listen,TR}(b)$ the results are symmetric:

$$\begin{aligned}0 \leq b(s_l) \leq 0.39 &: left \\ 0.39 < b(s_l) \leq 0.98 &: listen \\ 0.98 < b(s_l) \leq 1.0 &: right\end{aligned}$$

Combining the two to get $V_1^{listen}(b)$ partition:



This is also the partition for $V_2(b)$ since $V_2(b) = V_1^{listen}(b)$ - this can be verified by calculating $V_1^{left}(b)$ and $V_1^{right}(b)$ and taking the max over all actions.

An intuitive explanation is because if the agent were to open one of the doors at $t = 2$, then, on the next step, the tiger would be randomly placed behind one of the doors and the agent's belief state would be reset to $(0.5, 0.5)$. So after opening a door, the agent would be left with no information about the tiger's location and with one action remaining. We just saw that with one step to go and $b = (0.5, 0.5)$ the best thing to do is listen. Therefore, if the agent opens a door when $t = 2$, it will listen on the last step. It is a better strategy to listen when $t = 2$ in order to make a more informed decision on the last step

Plan Graph(Moore Automata)

One drawback of the POMDP approach is that the agent must maintain a belief state and use it to select an optimal action on every step; if the underlying state space or V is large, then this computation can be expensive. In many cases, it is possible to encode the policy in a graph that can be used to select actions without any explicit representation of the belief state; we refer to such graphs as plan graphs. The plan graph of the tiger problem is drawn below:

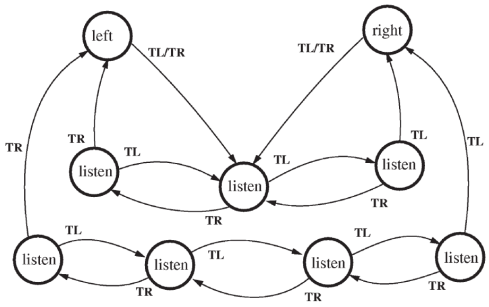


Figure 3: The optimal infinite-horizon policy for the tiger problem can be drawn as a plan graph. This structure counts the relative number of times the tiger was heard on the left as compared to the right.

Some of the nodes of the graph will never be visited once either door is opened and the belief state is reset to (0.5, 0.5). If the agent always starts in a state of complete uncertainty, then it will never be in a belief state that lies in the region of these nonreachable nodes. This results in a simpler version of the plan graph, shown below. The plan graph has a simple interpretation: keep listening until you have heard the tiger twice more on one side than the other.

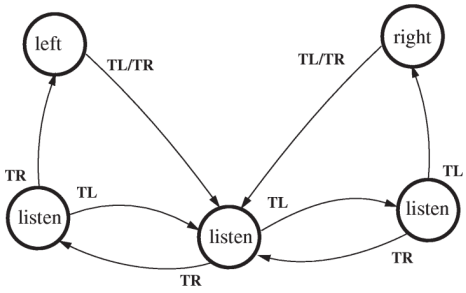


Figure 4: Given the initial belief state of (0.5, 0.5) for the tiger problem, some nodes of the plan graph can be trimmed.

Because the nodes represent a partition of the belief space and because all belief states

within a particular region will map to a single node on the next level, the plan graph representation does not require the agent to maintain an on-line representation of the belief state; the current node is a sufficient representation of the current belief. In order to execute a plan graph, the initial belief state is used to choose a starting node. After that, the agent need only maintain a pointer to a current node in the graph. On every step, it takes the action specified by the current node, receives an observation, then follows the arc associated with that observation to a new node. This process continues indefinitely.

Consider the following example: start at belief state $b_0 = (0.5, 0.5)$ and follow using the short automata when receiving the signals {TR,TR}.

$$\begin{aligned} b_1(s_l) =& \frac{Pr(TR|listen, s_l)b_0(s_l)}{Pr(TR|a, b_0)} = \frac{0.15 * 0.5}{0.15 * 0.5 + 0.85 * 0.5} = 0.15 \\ b_1(s_r) =& 0.85 \end{aligned}$$

And for b_2:

$$\begin{aligned} b_2(s_l) =& \frac{Pr(TR|listen, s_l)b_1(s_l)}{Pr(TR|a, b_1)} = \frac{0.15 * 0.15}{0.15 * 0.15 + 0.85 * 0.85} = 0.03 \\ b_2(s_r) =& 0.97 \end{aligned}$$

Furthermore we can check that indeed the automata is optimal by checking that in belief state b_2 it is best to open the left door:

$$\begin{aligned} V_{\text{left}}(b_2) =& 0.97 * 10 + 0.03 * (-100) = 6.7 \\ V_{\text{right}}(b_2) =& 0.03 * 10 + 0.97 * (-100) = -96.7 \\ V_{\text{listen}}(b_2) =& -1 \end{aligned}$$