Lecture 7: Planning with Partial Observability

Scribe: *Jessica Lupanow and Emily Meschke*

1 Review

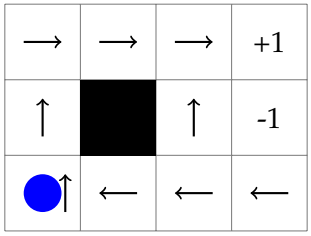
Previously we talked about Markov Decision Processes (MDPs) in which we had a state space, S, an action space, A, transition matrix, T, and reward, R.

$$MDP :< S, A, T, R >$$

We were looking for the optimal policy, π^*, such that:

$$\pi^* = argmax_\pi E[\sum_{t=0}^{T-1} r_t]$$

And we ended up with something like this:



2 Partially Observable Markov Decision Process

The difference between MDPs and Partially Observable Markov Decision Processes (POMDPs) is that the robot does not know its true state. For this reason, it instead maintains a belief, or probability distribution, over all possible states, $B : \pi(s)$. This means that we have some prior belief about our state, we take an action, we get a new observation, and then we update our belief.

Now that we are incorporating observations into our MDPs, we must consider our set of possible observations, Ω, as well as the observation function, M.

$$POMDP :< S, A, T, R, \Omega, M >$$

where M describes, given a state, the probability distribution of possible observations:

$$M : SxA \rightarrow \Pi(\Omega)$$

2.1 Belief

Our belief can be updated according to the following:

$$b'(s') = P(s'|o, a, b)$$

Applying Bayes' Rule:

$$\begin{aligned} b'(s') &= \frac{P(o|s', a, b)P(s'|a, b)}{P(o|a, b)} \\ &= \frac{P(o|s', a, b)\sum_{s\in S} P(s'|a, b, s)P(s|a, b)}{P(o|a, b)} \end{aligned}$$

Neither our observation nor our next state depend on where we think we are, so we can eliminate b in the first and second probabilities in the numerator. Our previous state also doesn't depend on our current action, so we can eliminate a in the third probability in the numerator. These cancellations are reflected below.

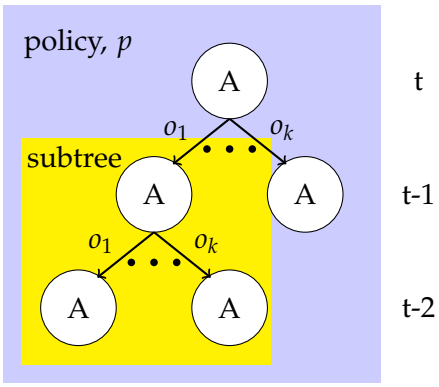
$$\begin{aligned} b'(s') &= \frac{P(o|s', a, \cancel{b})\sum_{s\in S} P(s'|a, \cancel{b}, s)P(s|\cancel{a}, b)}{P(o|a, b)} \\ &= \frac{P(o|s', a)\sum_{s\in S} P(s'|a, s)b(s)}{P(o|a, b)} \\ &= \eta M(o|s', a)\sum_{s\in S} T(s'|a, s)b(s) \end{aligned}$$

Now the optimal policy, π^*, maps beliefs to actions rather than states to action as with MDPs.

$$\pi^* : B \rightarrow A$$

2.2 Policy Trees

We can write our policy for a POMDP as a tree, where each node is an action and each edge is an observation:



To determine the optimal policy tree, p^*, we must define the values of possible trees, p. If we know what we state we're in, we can do the following:

$$V_t^p(s) = R(s, a(p)) + \text{Expected future value}$$

where the expected future value is given by the value of the selected subtree (one of which is highlighted in yellow in the example above):

$$V_t^p(s) = R(s, a(p)) + \sum_{s' \in S} P(s'|s, a(p)) \sum_{o_i \in \Omega} P(o_i|s', a(p)) V_{t-1}^{p,o_i}(s')$$

However, since we don't know the true state in a POMDP, we must incorporate our current belief into the value estimation.

$$V_t^p(b) = \sum_{s \in S} b(s) V_t^p(s)$$

Thus:

$$V_t(b) = max_{p \in P} \sum_{s \in S} b(s) V_t^p(s)$$

2.3 One Time Step Example: Assistive Wheelchair

Now as an example, let's say that we have an assitive wheelchair that can take a user left or right (the states). It's possible actions are to go left, go right, or ask the user what they would like to do (the action space). The human could want to move right or move left (the observation space). This scenario can be described by the following POMDP:

S : {left(L), right(R)}
A : {ask, go left (GL), go right (GR)}
$T : S \times A \to \pi(s')$
Ω : {move left (ML), move right (MR)}
$M : S \times A \to \pi(\Omega)$
$R : S \times A \to \mathbb{R}$

The transition function (T) and observation function (M) can be defined as look-up tables.

$T : (S, ask) \to S'$

		S′	
		R	L
S	R	1	0
	L	0	1

$T : (S, GL) \to S'$

		S′	
		R	L
S	R	0.5	0.5
	L	0.5	0.5

$T : (S, GR) \to S'$

		S′	
		R	L
S	R	0.5	0.5
	L	0.5	0.5

The action "ask" doesn't change the world state, and each movement, GL or GR, resets the world state.

$M : (S, ask) \to \Omega$

		Ω	
		MR	ML
S	R	0.9	0.1
	L	0.1	0.9

$M : (S, GL) \to \Omega$

		Ω	
		MR	ML
S	R	0.5	0.5
	L	0.5	0.5

$M : (S, GR) \to \Omega$

		Ω	
		MR	ML
S	R	0.5	0.5
	L	0.5	0.5

While the action "ask" doesn't change the world state, the observation function accounts for sensor noise.

Rewards are distributed as follows:

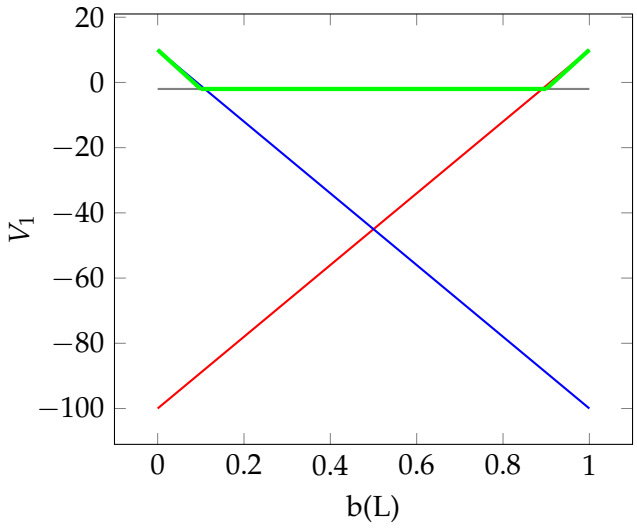
$$R(*, \text{ask}) = -2$$

$$R(\text{L}, \text{GL}) = R(\text{R}, \text{GR}) = 10$$
$$R(\text{L}, \text{GR}) = R(\text{R}, \text{GL}) = -100$$

The goal is to construct an optimal policy tree from the set of possible policy trees, where the maximum policy is chosen at each point in the belief state. For this example, we will calculate the value of a policy that chooses action GL.

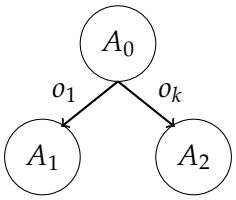
$$\begin{aligned} V_1^{\text{GL}} &= \sum_{s \in S} b(s) V_1^{\text{GL}}(s) \\ &= b(\text{L}) V_1^{\text{GL}}(\text{L}) + (1 - b(\text{L})) V_1^{\text{GL}}(\text{R}) \\ &= (b(\text{L}))(10) + (1 - b(\text{L}))(-100) \\ &= 110 b(\text{L}) - 100 \end{aligned}$$

Below, we plot the value function (V_1) for each policy (ask, GL, or GR) across the probability we are in state L (b(L)). The optimal policy is a piece-wise, convex shape, where the policy (in this case a single action) that maximizes the value function is chosen across the belief state. The value is always higher when the agent is more certain it is in a specific state, when b(L) approaches 1 (probably left), and b(L) approaches 0 (probably right). This reflects the value of having more information about the current state.



2.4 Two Time steps Example: Assistive Wheelchair Continued

In this case, two actions, and the possible subtrees, are considered.



The value for a policy is calculated by considering the immediate reward for the action given the policy $a(p)$ + the likelihood of potential future states and their values:

$$V_2^p(S) = R(s, a(p)) + \sum_{s' \in S} T(s'|s, a(p)) \sum_{o \in \Omega} M(o_i|s, a(p)) V_1^{a(p), o_i}(s)$$

First, we will calculate the value of one policy (p_1=GL).

$$\begin{aligned} V_2^{p_1}(\text{L}) = R(\text{L, GL}) &+ T(\text{L}|\text{L, GL}) M(\text{ML}|\text{L, GL}) V_1^{GL,ML}(L) \\ &+ T(\text{L}|\text{L, GL}) M(\text{MR}|\text{L, GL}) V_1^{GL,MR}(L) \\ &+ T(\text{R}|\text{L, GL}) M(\text{ML}|\text{L, GL}) V_1^{GL,ML}(R) \\ &+ T(\text{R}|\text{L, GL}) M(\text{MR}|\text{L, GL}) V_1^{GL,MR}(R) \\ = 10 &+ (0.5)(0.5)(10) \\ &+ (0.5)(0.5)(10) \\ &+ (0.5)(0.5)(-100) \\ &+ (0.5)(0.5)(-100) \\ = -35 & \end{aligned}$$

$$\begin{aligned} V_2^{p_1}(\text{R}) &= R(\text{R, GL}) + ... \\ &= -100 + ... \\ &= -145 \end{aligned}$$

So the value of the policy with root action GL is:

$$\begin{aligned} V_2^{p_1}(b) &= b(\text{L})(-35) + (1 - b(\text{L}))(-145) \\ &= 110 b(\text{L}) - 145 \end{aligned}$$

Next, we will consider another policy (p_2=ask).

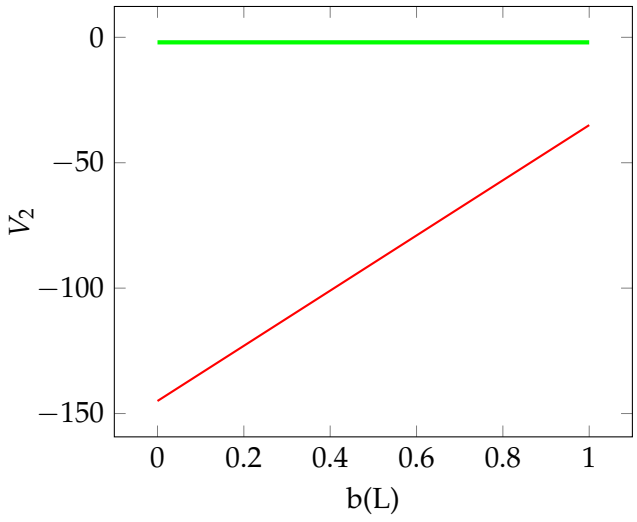
$$\begin{aligned}
V_2^{p_2}(\text{L}) &= R(\text{L,ask}) + T(\text{L}|\text{L, ask})M(\text{ML}|\text{L, ask})V_1^{\text{ask,ML}}(L) \\
&\quad + T(\text{L}|\text{L, ask})M(\text{MR}|\text{L, ask})V_1^{\text{ask,MR}}(L) \\
&\quad + T(\text{R}|\text{L, ask})M(\text{ML}|\text{L, ask})V_1^{\text{ask,ML}}(R) \\
&\quad + T(\text{R}|\text{L, ask})M(\text{MR}|\text{L, ask})V_1^{\text{ask, MR}}(R) \\
&= -1 + (1)(0.9)(10) \\
&\quad + (1)(0.1)(-100) \\
&\quad + (0) \\
&\quad + (0) \\
&= -2
\end{aligned}$$

$$\begin{aligned}
V_2^{p_2}(\text{R}) &= R(\text{R, ask}) + ... \\
&= -1 + ... \\
&= -2
\end{aligned}$$

So the value of the policy with root action ask is:

$$\begin{aligned}
V_2^{p_2}(b) &= b(\text{L})(-2) + (1 - b(\text{L}))(-2) \\
&= -2
\end{aligned}$$

In this case, the second policy (p_2 = ask) completely dominates the first policy (p_1 = GL). This is due to the fact that asking a question early 1) doesn't change the world state and 2) reduces the uncertainty of the belief state, increasing the reward for the subsequent value estimations. Dominated policies will never contribute to the optimal policy.



2.5 Finding the Optimal Policy: Summary

The equation to calculate the value of the policy p at time-step t is shown below:

$$V_t^p = \max_{a \in A}[\sum_{s \in S} b(s)R(s,a) + \sum_{o_i \in \Omega} \sum_{s' \in S} M(o_i|s',a)T(s'|s,a)V_{t-1}^p(s')]$$

To calculate the policy values more efficiently, one can construct two vectors: one mapping $S, A \rightarrow R$, and the other mapping $S', A \rightarrow O$, that can be combined to calculate the value of a policy p at time point t given every possible belief state.

$$\alpha^R = R(s,a)$$
$$\alpha^{o_1} = \sum_{s' \in S} M(o_i|s',a)T(s'|s,a)V_{t-1}^p(s')$$
$$V_t^b = \max_{a \in A}[\alpha^R \cdot b + \sum_{o_i \in \Omega} \max_{\alpha^{o_i} \in p_{t-1}} \alpha^{o_i} \cdot b]$$

Finally, one must decide at which points of the belief state to sample policies. A common strategy is to approximate the areas of the belief state that are reachable by an optimal policy and sample from there.