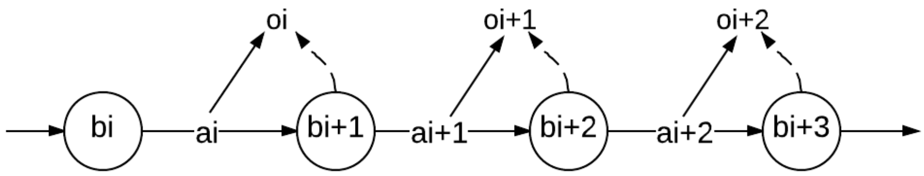
Derivation of PoMDP Bellman Equation

1 Notations

$\mathcal{S}$ state space
$\mathcal{A}$ action space
$\mathcal{O}$ observation space
$\mathcal{B}$ belief state space
N cardinality of $\mathcal{S}$
t number of time before termination
T state transition probability, $\mathcal{S} \times \mathcal{A} \to \mathbb{R}$, $T(s_i, a_i, s_{i+1}) = \mathbb{P}(s_{i+1}|s_i, a_i)$
R immediate reward, $\mathcal{S} \times \mathcal{A} \to \mathbb{R}$, $R(s_i, a_i)$
ρ average immediate reward, $\rho(b_i, a_i) = \sum_s b_i(s) R(s, a_i)$
O observation probability, $O(a_i, s_{i+1}, o_i) = \mathbb{P}(o_i|a_i, s_{i+1})$, $o_i \in \mathcal{O}$
b belief state, $b = \{b_1, \cdots, b_N\}$, $b_j = \mathbb{P}(s = j)$
SE state estimator, $b_{i+1} = SE(b_i, a_i, o_i)$
$V_t(b)$ value of being b with t time left
α α-vector, length N, $V_t(b) = \max_k \left(\alpha_t^k \cdot b\right)$. Policy is embedded in α.
τ belief transition probability, $\tau(b_i, a_i, b_{i+1}) = \mathbb{P}(b_{i+1}|b_i, a_i)$
Notice: s_i, s_{i+1} and s, s' may be used interchangably.



2 Derivation

$$V_{t+1}(b) = \max_{a \in \mathcal{A}} \left[\rho(b, a) + \gamma \sum_{b' \in \mathcal{B}} \tau(b, a, b') V_t(b') \right] \tag{1}$$

Notice, if b, a, o are given, then b' is fixed.

$$\sum_{b' \in \mathcal{B}} \tau(b, a, b') V_t(b') = \sum_{o \in \mathcal{O}} \mathbb{P}(o|a, b) V_t(SE(b, a, o)) \tag{2}$$

The s' entry of SE is

$$\begin{aligned} SE_{s'}(b, a, o) &= \mathbb{P}(s'|a, o, b) \\ &= \frac{\mathbb{P}(o|s', a, b)\mathbb{P}(s'|a, b)}{\sum_{s' \in \mathcal{S}} \mathbb{P}(o|s', a, b)\mathbb{P}(s'|a, b)} \\ &= \frac{O(a, s', o) \sum_s T(s, a, s') b(s)}{\sum_{s'} O(a, s', o) \sum_s T(s, a, s') b(s)} \end{aligned} \tag{3}$$

$$
\begin{aligned}
\mathbb{P}(o|a,b) &= \sum_s b(s)\mathbb{P}(o|a,s) \\
&= \sum_s b(s) \sum_{s'} \mathbb{P}(o, s'|a,s) \\
&= \sum_s b(s) \sum_{s'} \mathbb{P}(s'|a,s)\mathbb{P}(o|a,s') \\
&= \sum_s b(s) \sum_{s'} T(s,a,s')O(a,s',o)
\end{aligned}
\tag{4}
$$

Let $k = l(b') = l\left(SE(b,a,o)\right) \equiv l(b,a,o)$ be the α-vector index for $V_t(b') = \max_k(\alpha_t^k \cdot b')$. Therefore,

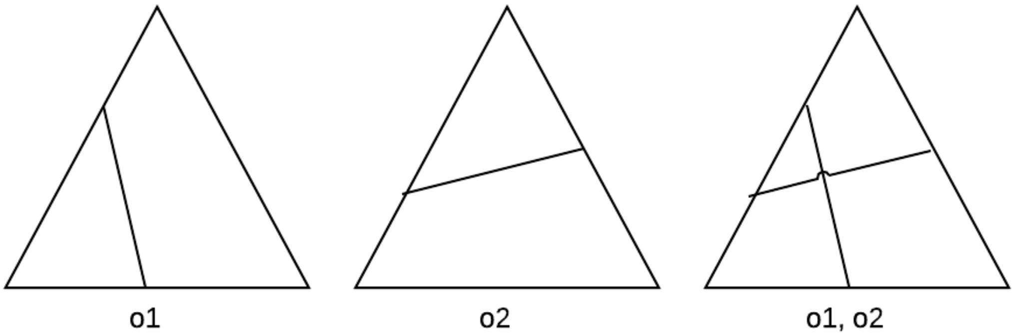
$$
V_{t+1}(b) = \max_a \left\{ \sum_s b(s)R(s,a) + \gamma \sum_o \left(\sum_s b(s) \sum_{s'} T(s,a,s')O(a,s',o) \right) \left(\alpha_t^{l(b,a,o)} \cdot SE(b,a,o) \right) \right\}
\tag{5}
$$

After simplification,

$$
V_{t+1}(b) = \max_{a\in\mathcal{A}} \left\{ \sum_{s\in\mathcal{S}} b(s) \underbrace{\left(R(s,a) + \gamma \sum_{j=1}^{N} T(s,a,s_j) \sum_{o\in\mathcal{O}} \alpha_{tj}^{l(b,a,o)} O(a,s_j,o) \right)}_{Y_s(a,b)} \right\}
\tag{6}
$$

3 Piecewise Linear Discussion

$Y_s(a,b)$ is piecewise constant on b. Prove as follows,

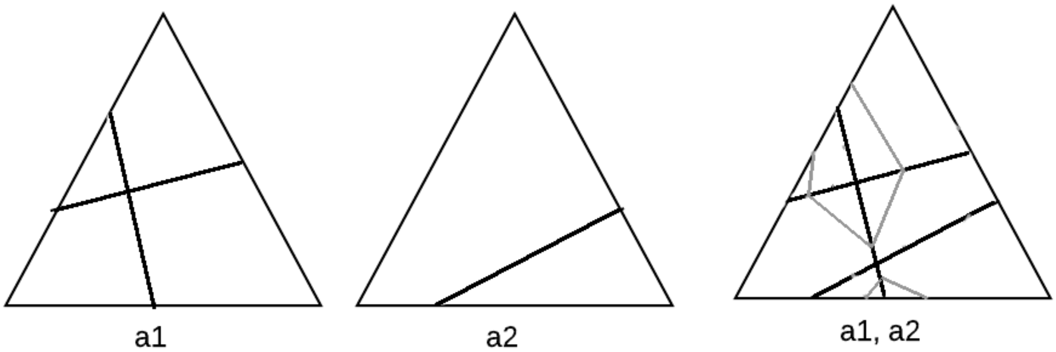


These triangles shows $\mathcal{B}$. Given a and o, $l(b,a,o)$ is piecewise constant. For example, given a, $\mathcal{B}$ is partitioned by the value of l into the first picture given o_1, and into the second picture given o_2. Given a, take the intersections (refinement) of all partitions of different $o \in \mathcal{O}$, get the third picture. For this given a, within this refined partition, $l(b,a,o)$ is constant for $\forall o$. So $Y_s(a,b)$ is piecewise constant in the refined partition. Therefore, $V_{t+1}(b)$ is piecewise linear. Convexity can also be shown, but the proof is neglected here.

For a given a, it is shown $Y_s(a,b)$ is piecewise constant. After taking maximum of Eqn(6), we have

$$
V_{t+1}(b) = \sum_{s\in\mathcal{S}} b(s) Y_s^*(b)
\tag{7}
$$

$Y_s^*(b)$ is piecewise constant. Its partition includes the intersection (refinement) of $Y_s(a,b)$ for $\forall a \in \mathcal{A}$ (accounts for observation change, black lines), and optimal action change (accouts for change of optimal action, grey lines).



3.1 Question

If we use mesh-based function approximation instead of piecewise linear (exact) $V(b)$, how to adaptively tweak the mesh (mesh-adaptive)?

4 Q-Learning in CoMDP

Q-learning is useful if the agent does not know the form of immediate reward and state transfer function in the environment.
Q-learning maintains a table of Q-values, $Q(s, a)$, that is the cumulative discounted reward of being in state s and take action a. In the begining the Q-value table is estimated, then we update Q-value by

$$Q_{t+1}(s, a) = r + \gamma \max_{a'} Q_t(s', a') \tag{8}$$

Here we assume the agent observes s' and r as soon as a is taken. Note t here means iteration number, not time.
The exploration probabilities of taking action a at state s is chosen by

$$\mathbb{P}(a|s) = \frac{e^{Q(s,a)\eta}}{\sum_j e^{Q(s,a_j)}\eta} \tag{9}$$

4.1 Question

How to facilitate Q-learning with a non-perfect environment model?

Sometimes, a relaxition can be added to Q-learning

$$Q_{t+1}(s, a) = Q_t(s, a) + \alpha \left(r + \gamma \max_{a'} Q_t(s', a') - Q_t(s, a) \right) \tag{10}$$

$0 \leq \alpha \leq 1$ is the learning rate.

5 Continuous-state PoMDP

In discrete state problem, the belief state is a *point* on a hyperplane (a line for two-state problem). In continous-state problem, the belief state is a distribution on the state space. For example, if we parameterize states by 2 parameters, then the state space is a 2D plane. And, a belief state is a probability distribution (or a normalized function) on the 2D plane. The $V(b)$ will be a functional that evaluates every possible functions. The convexity of $V(b)$ indicates: the average of V's for two probability distributions will be higher than the V for the averaged probability distribution.

For discrete state, α-vector assigns each state a value, and $V(b) = \max_i < \alpha_i, b >$. In continuous state, α_i is a function defined on state space, $V(b) = \max_i < \alpha_i, b >$