POMDP(Belief-MDP) defined in the belief space $\mathbb{B}$

At time step t, the current belief b_t already contain all the information used for taking actions. So it is again a Markovian decision process

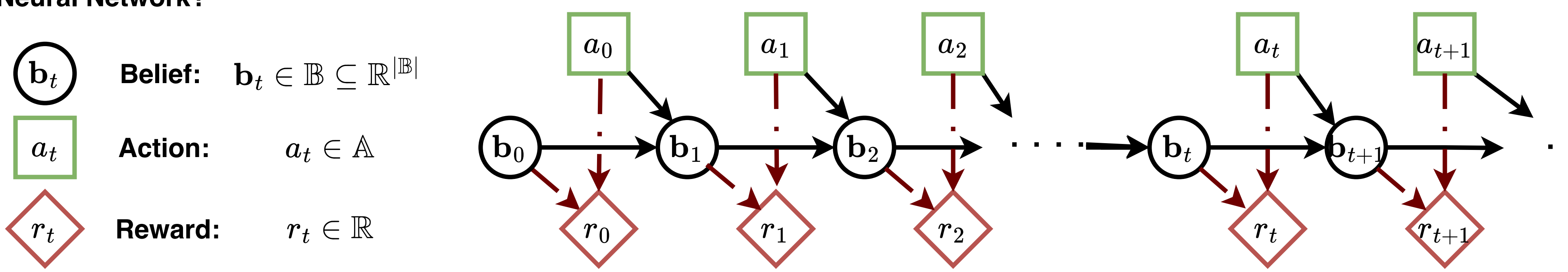
$$b_{t+1}(s) = P(S_{t+1} = s | o_{0:t+1}, a_{0:t}) = P(S_{t+1} = s | o_{t+1}, a_t, \mathbf{b}_t)$$

where the capital $\mathbf{b}_t$ is the believed Probability distribution of all state at time t,
if the states are discrete, the $\mathbf{b}_t$ is a Probability Mass Function containing the probability of all states;
if the states are continuous , the $\mathbf{b}_t$ is a Probability Density Function over the state space $\mathbb{S}$.
The dimension of the belief space $d_\mathbb{B}$ is dependent on the dimension of the state space $d_\mathbb{S}$. $d_\mathbb{S}$ is the number of features chosen as the state variable
These $d_\mathbb{S}$ number of different features could have discrete values or continuous values.
For the state with discrete values. If there are N possible values, the belief state $\mathbf{b}_t$ is a N-dimensional Probability Mass Function, which is a point in the $N-1)$-simplex
The set of all such belief states is the standard $(N-1)-$simplex, which is defined by

$$\Delta^{N-1} = \{(p_1, p_2, \cdots, p_N) \in \mathbb{R}^N | p_i > 0 \; \forall i \text{ and } \sum_{i=1}^{N} p_i = 1)\}$$

For continuous state space with ∞ number of possible values, the belief state $\mathbf{b}_t$ is a Probability Density Function.
But in real application, we could use a finite parametric representation to approximate the belief.
If we define the number of finite parameters are $d_\mathbb{B}$, then the belief state $\mathbf{b}_t$ is represented by a vector with $d_\mathbb{b}$, $\mathbf{b}_t \in \mathbb{R}^{d_\mathbb{B}}$

e.g.
Gaussian distribution
Gaussian Mixture distribution
Neural Network?



Transition Model:

$$P(\mathbf{b}_{t+1}|\mathbf{b}_t, a_t) = \int_{o_{t+1}\in\mathbb{O}} P(\mathbf{b}_{t+1}|\mathbf{b}_t, a_t, o_{t+1}) P(o_{t+1}|\mathbf{b}_t, a_t) do_{t+1}$$

$$P(\mathbf{b}_{t+1}|\mathbf{b}_t, a_t) = \sum_{o_{t+1}\in\mathbb{O}} P(\mathbf{b}_{t+1}|\mathbf{b}_t, a_t, o_{t+1}) P(o_{t+1}|\mathbf{b}_t, a_t)$$

Reward Model:

$$r(a_t, \mathbf{b}_t) = \int_{s_t\in\mathbb{S}} r(a_t, s_t) b_t(s_t)$$

$$r(a_t, \mathbf{b}_t) = \sum_{s_t\in\mathbb{S}} r(a_t, s_t) b_t(s_t)$$