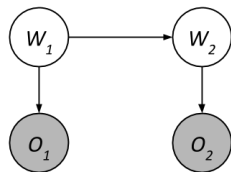


## 1 HMMs

Consider the following Hidden Markov Model.



$W_1$	$P(W_1)$
0	0.3
1	0.7

$W_t$	$W_{t+1}$	$P(W_{t+1} W_t)$
0	0	0.4
0	1	0.6
1	0	0.8
1	1	0.2

$W_t$	$O_t$	$P(O_t W_t)$
0	a	0.9
0	b	0.1
1	a	0.5
1	b	0.5

Suppose that we observe  $O_1 = a$  and  $O_2 = b$ .

Using the forward algorithm, compute the probability distribution  $P(W_2|O_1 = a, O_2 = b)$  one step at a time.

(a) Compute  $P(W_1, O_1 = a)$ .

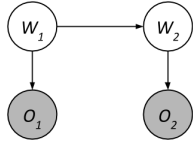
(b) Using the previous calculation, compute  $P(W_2, O_1 = a)$ .

(c) Using the previous calculation, compute  $P(W_2, O_1 = a, O_2 = b)$ .

(d) Finally, compute  $P(W_2|O_1 = a, O_2 = b)$ .

## 2 Particle Filtering

Let's use Particle Filtering to estimate the distribution of  $P(W_2|O_1 = a, O_2 = b)$ . Here's the HMM again.



$W_1$	$P(W_1)$
0	0.3
1	0.7

$W_t$	$W_{t+1}$	$P(W_{t+1} W_t)$
0	0	0.4
0	1	0.6
1	0	0.8
1	1	0.2

$W_t$	$O_t$	$P(O_t W_t)$
0	a	0.9
0	b	0.1
1	a	0.5
1	b	0.5

We start with two particles representing our distribution for  $W_1$ .

$P_1 : W_1 = 0$

$P_2 : W_1 = 1$

Use the following random numbers to run particle filtering:

[0.22, 0.05, 0.33, 0.20, 0.84, 0.54, 0.79, 0.66, 0.14, 0.96]

(a) **Observe:** Compute the weight of the two particles after evidence  $O_1 = a$ .

(b) **Resample:** Using the random numbers, resample  $P_1$  and  $P_2$  based on the weights.

(c) **Predict:** Sample  $P_1$  and  $P_2$  from applying the time update.

(d) **Update:** Compute the weight of the two particles after evidence  $O_2 = b$ .

(e) **Resample:** Using the random numbers, resample  $P_1$  and  $P_2$  based on the weights.

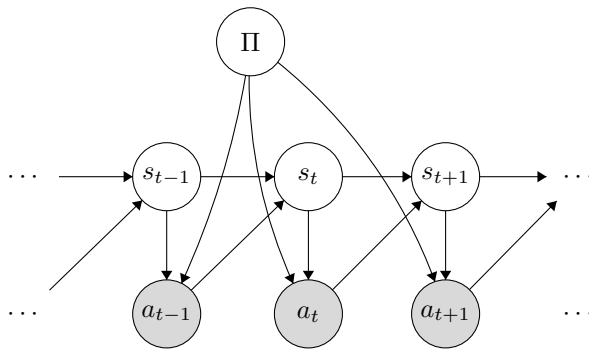
(f) What is our estimated distribution for  $P(W_2|O_1 = a, O_2 = b)$ ?

### 3 Particle Filtering Apprenticeship

We are observing an agent's actions in an MDP and are trying to determine which out of a set  $\{\pi_1, \dots, \pi_n\}$  the agent is following. Let the random variable  $\Pi$  take values in that set and represent the policy that the agent is acting under. We consider only *stochastic* policies, so that  $A_t$  is a random variable with a distribution conditioned on  $S_t$  and  $\Pi$ . As in a typical MDP,  $S_t$  is a random variable with a distribution conditioned on  $S_{t-1}$  and  $A_{t-1}$ . The full Bayes net is shown below.

The agent acting in the environment knows what state it is currently in (as is typical in the MDP setting). Unfortunately, however, we, the observer, cannot see the states  $S_t$ . Thus we are forced to use an adapted particle filtering algorithm to solve this problem. Concretely, we will develop an efficient algorithm to estimate  $P(\Pi | a_{1:t})$ .

(a) The Bayes net for part (a) is



(i) Select all of the following that are guaranteed to be true in this model for  $t > 3$ :

- $S_t \perp\!\!\!\perp S_{t-2} | S_{t-1}$
- $S_t \perp\!\!\!\perp S_{t-2} | S_{t-1}, A_{1:t-1}$
- $S_t \perp\!\!\!\perp S_{t-2} | \Pi$
- $S_t \perp\!\!\!\perp S_{t-2} | \Pi, A_{1:t-1}$
- $S_t \perp\!\!\!\perp S_{t-2} | \Pi, S_{t-1}$
- $S_t \perp\!\!\!\perp S_{t-2} | \Pi, S_{t-1}, A_{1:t-1}$
- None of the above

We will compute our estimate for  $P(\Pi | a_{1:t})$  by coming up with a recursive algorithm for computing  $P(\Pi, S_t | a_{1:t})$ . (We can then sum out  $S_t$  to get the desired distribution; in this problem we ignore that step.)

(ii) Write a recursive expression for  $P(\Pi, S_t | a_{1:t})$  in terms of the CPTs in the Bayes net above.

$$P(\Pi, S_t | a_{1:t}) \propto \underline{\hspace{15em}}$$

We now try to adapt particle filtering to approximate this value. Each particle will contain a single state  $s_t$  and a potential policy  $\pi_i$ .

(iii) The following is pseudocode for the body of the loop in our adapted particle filtering algorithm. Fill in the boxes with the correct values so that the algorithm will approximate  $P(\Pi, S_t | a_{1:t})$ .

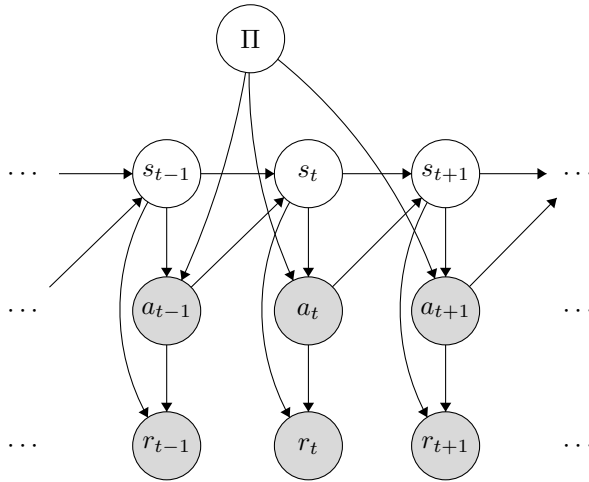
1. Elapse time: for each particle  $(s_t, \pi_i)$ , sample a successor  $s_{t+1}$  from  $\underline{\hspace{10em}}$ .

The policy  $\pi'$  in the new particle is  $\underline{\hspace{10em}}$ .

2. Incorporate evidence: To each new particle  $(s_{t+1}, \pi')$ , assign weight  $\underline{\hspace{10em}}$ .

3. Resample particles from the weighted particle distribution.

(b) We now observe the acting agent's actions *and* rewards at each time step (but we still don't know the states). Unlike the MDPs in lecture, here we use a stochastic reward function, so that  $R_t$  is a random variable with a distribution conditioned on  $S_t$  and  $A_t$ . The new Bayes net is given by



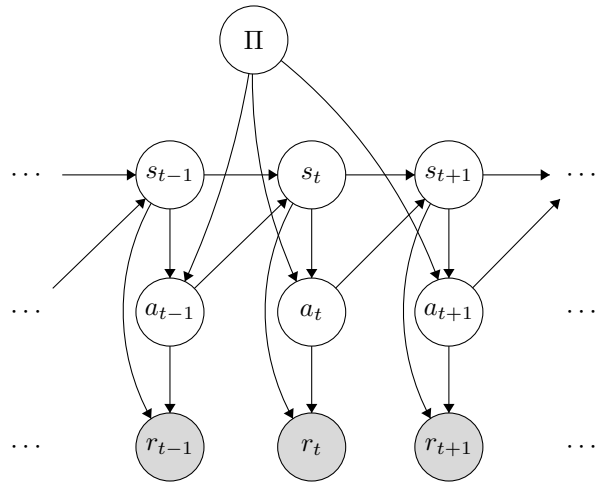
Notice that the observed rewards do in fact give useful information since d-separation does not give that  $R_t \perp\!\!\!\perp \Pi \mid A_{1:t}$ .

- (i) Give an active path connecting  $R_t$  and  $\Pi$  when  $A_{1:t}$  are observed. Your answer should be an ordered list of nodes in the graph, for example “ $S_t, S_{t+1}, A_t, \Pi, A_{t-1}, R_{t-1}$ ”.

- (ii) Write a recursive expression for  $P(\Pi, S_t \mid a_{1:t}, r_{1:t})$  in terms of the CPTs in the Bayes net above.

$$P(\Pi, S_t \mid a_{1:t}, r_{1:t}) \propto \underline{\hspace{15em}}$$

- (c) We now observe *only* the sequence of rewards and no longer observe the sequence of actions. The new Bayes net is shown on the right.



- (i) Write a recursive expression for  $P(\Pi, S_t, A_t \mid r_{1:t})$  in terms of the CPTs in the Bayes net above.

$$P(\Pi, S_t, A_t \mid r_{1:t}) \propto \underline{\hspace{15em}}$$

We now try to adapt particle filtering to approximate this value. Each particle will contain a single state  $s_t$ , a single action  $a_t$ , and a potential policy  $\pi_i$ .

- (ii) The following is pseudocode for the body of the loop in our adapted particle filtering algorithm. Fill in the boxes with the correct values so that the algorithm will approximate  $P(\Pi, S_t, A_t \mid r_{1:t})$ .

1. Elapse time: for each particle  $(s_t, a_t, \pi_i)$ , sample a successor state  $s_{t+1}$  from \_\_\_\_\_.

Then, sample a successor action  $a_{t+1}$  from \_\_\_\_\_.

The policy  $\pi'$  in the new particle is \_\_\_\_\_.

2. Incorporate evidence: To each new particle  $(s_{t+1}, a_{t+1}, \pi')$ , assign weight \_\_\_\_\_.

3. Resample particles from the weighted particle distribution.