CSCI 561
Foundation for Artificial Intelligence

Reasoning over Time
Temporal Models

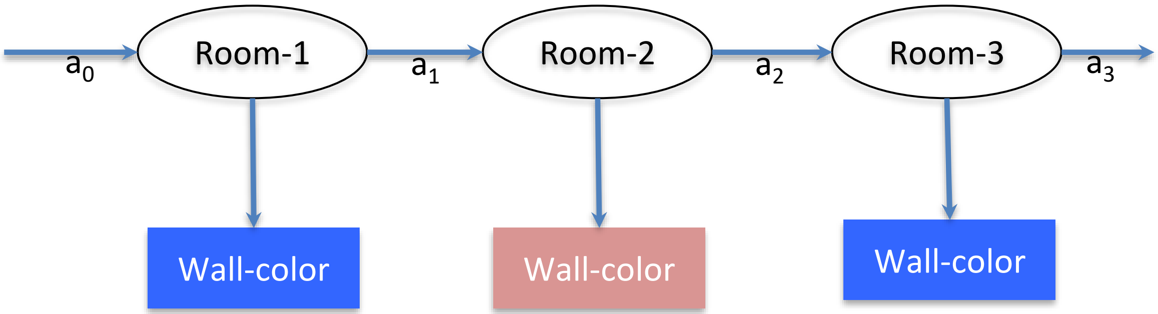
Professor Wei-Min Shen
University of Southern California

Outline: Temporal Models

- Models with Actions and Sensors (ALFE 4-5) (aka POMDP)
  - Slides 1-20 are essential for you to understand the concepts
  - The rest will follow naturally if you do
- Markov Chains
  - No observation, no explicit actions, transit randomly
- Hidden Markov Model
  - No explicit actions, state transit randomly
- Dynamic Bayesian Networks
  - No explicit actions, States are Bayesian Networks
- Continuous State Model
  - No explicit actions, States are continuous
- POMDP
  - Discrete states with probabilistic actions, sensors, & transitions

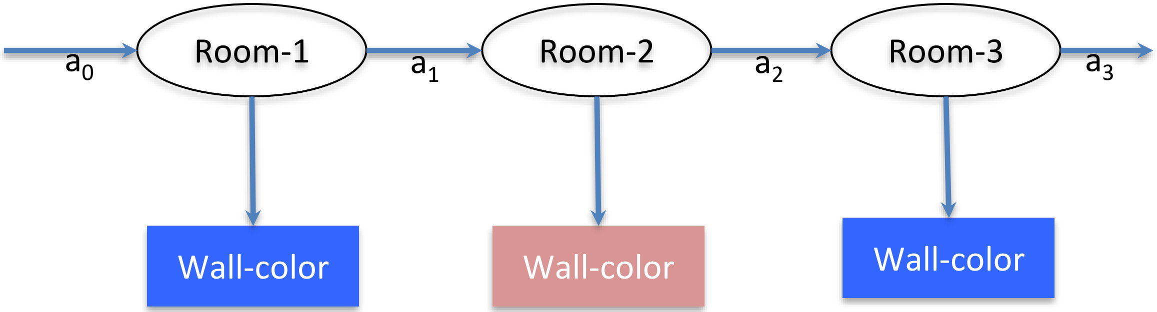
Example of Reasoning over Time

- Speech Recognition
  - “Listening is not always equal to hearing” ☺
- Traveling through rooms with colored walls
  - “Seeing does not always tell where you are” ☺



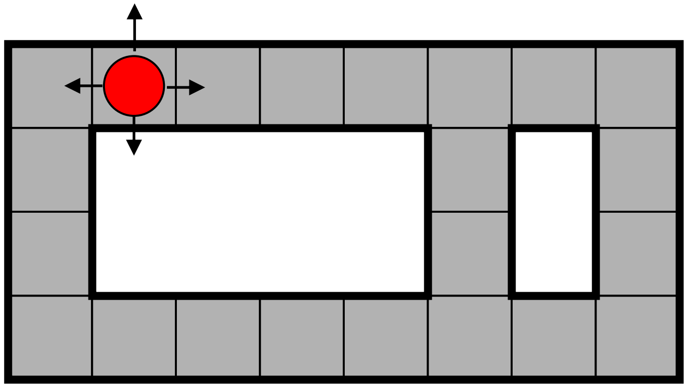
Problem of Reasoning over Time

- Given:
  - A model of the world (e.g., a map of rooms with colored walls)
  - An experience of observations and actions from time 1 to t
- Compute (among others):
  - Which states (e.g., rooms) the robot was/is/will-be in?



Example: Robot Localization

Example from Michael Pfeiffer



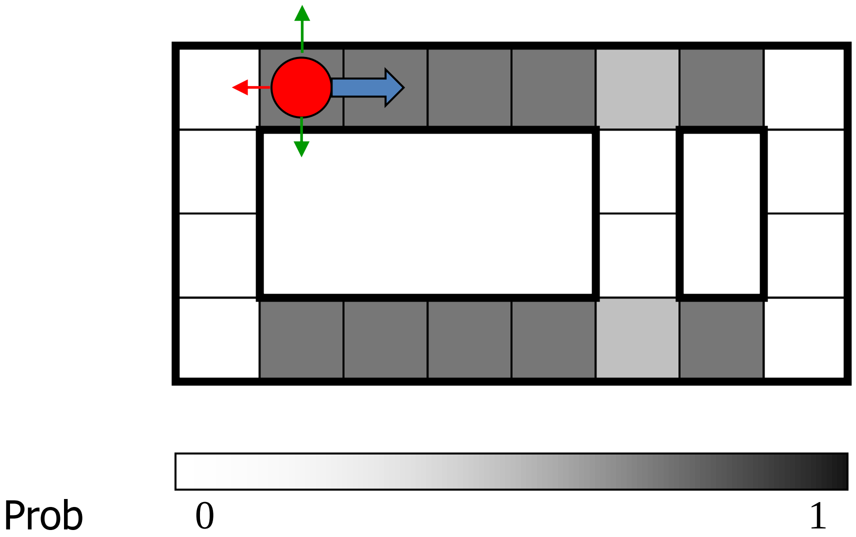
State X: Room

t=0 (before sensing, all states are equally likely)

Sensor model: never more than 1 mistake

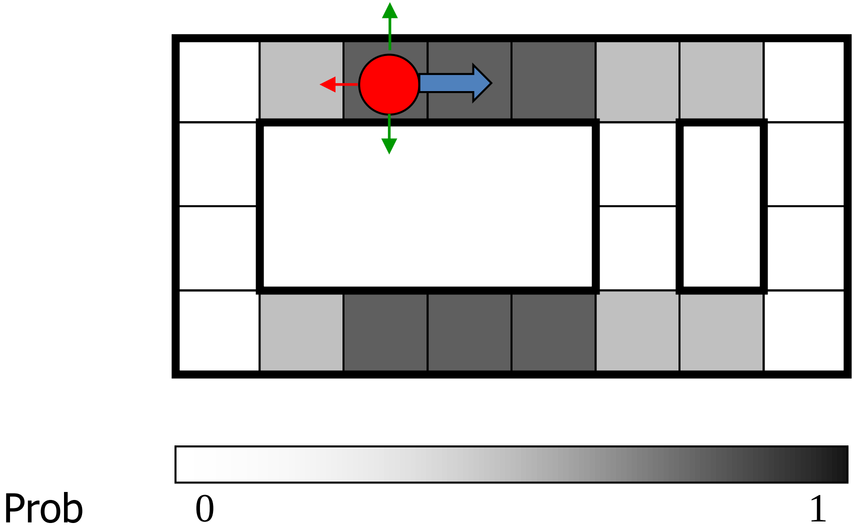
Motion model: may not execute action with small prob.

Example: Robot Localization



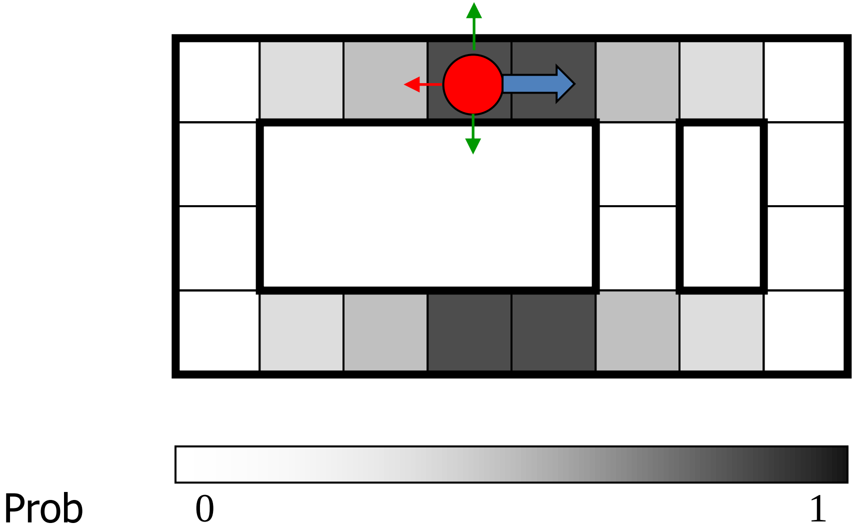
t=1 (sensing: no room up or down)

Example: Robot Localization



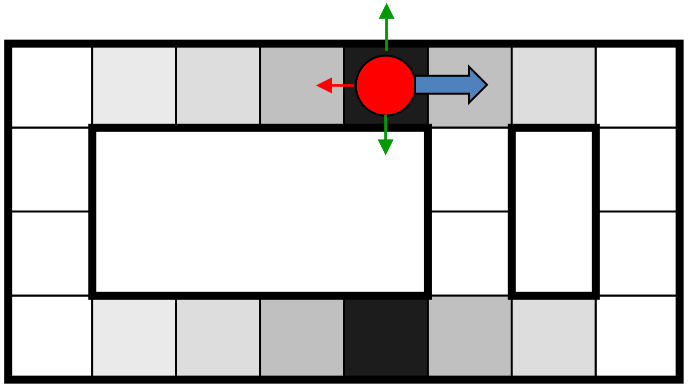
t=2

Example: Robot Localization



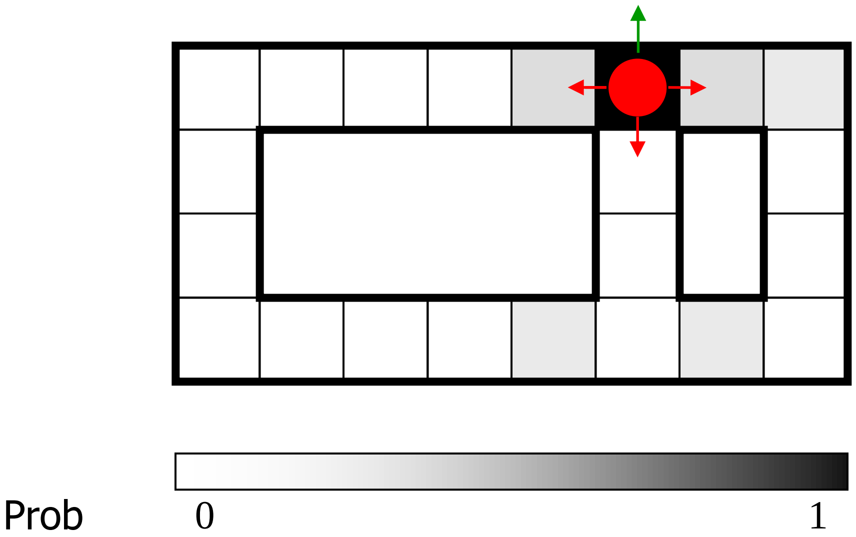
t=3

Example: Robot Localization

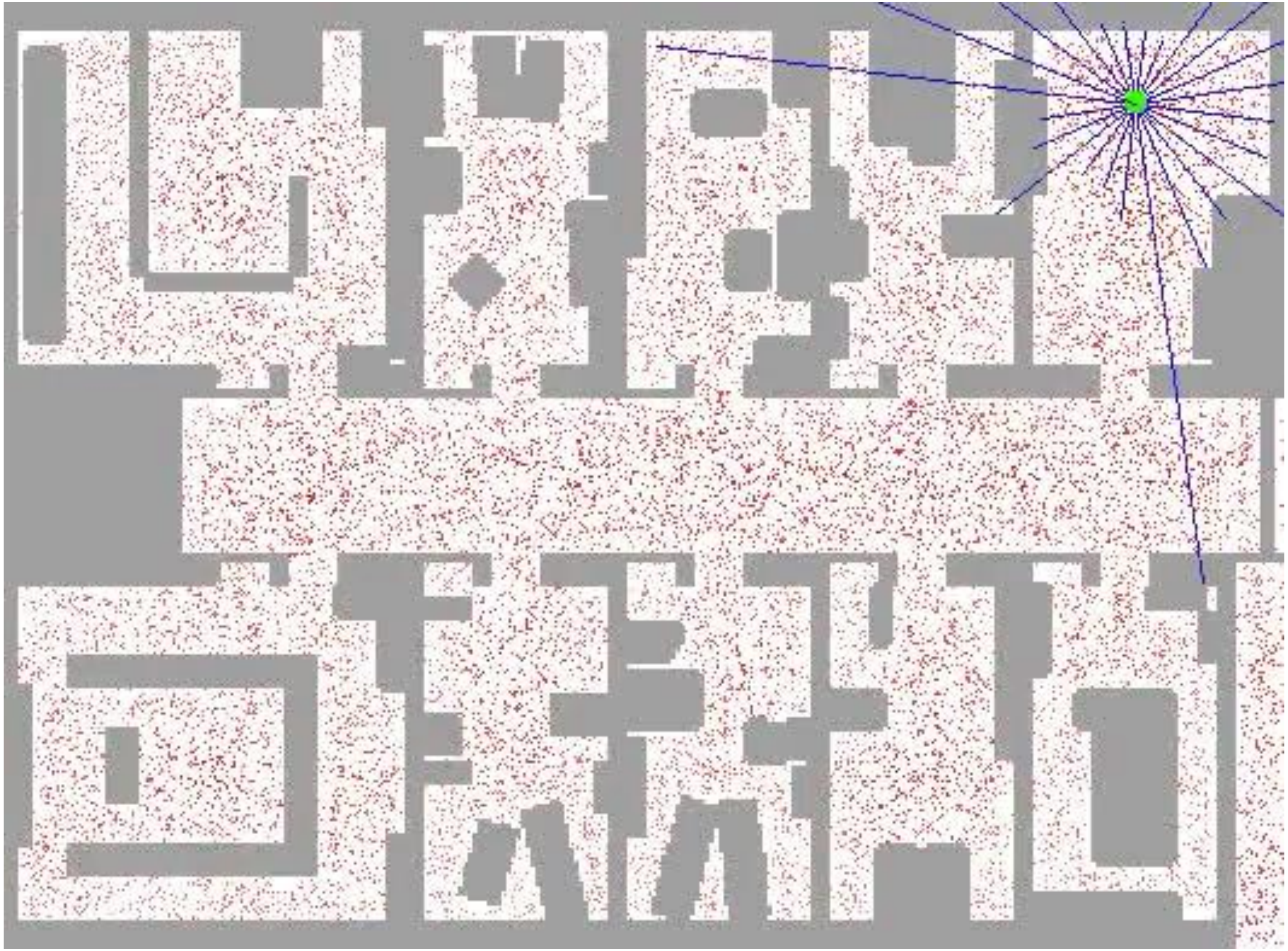


t=4

Example: Robot Localization



t=5



Little Prince Environment (S, A, Φ, θ, π)

- Transition Probabilities Φ (ask “Action Model”)

F	S0	S1	S2	S3
S0	0.1	0.1	0.1	0.7
S1	0.1	0.1	0.7	0.1
S2	0.7	0.1	0.1	0.1
S3	0.1	0.7	0.1	0.1

B	S0	S1	S2	S3
S0	0.1	0.1	0.7	0.1
S1	0.1	0.1	0.1	0.7
S2	0.1	0.7	0.1	0.1
S3	0.7	0.1	0.1	0.1

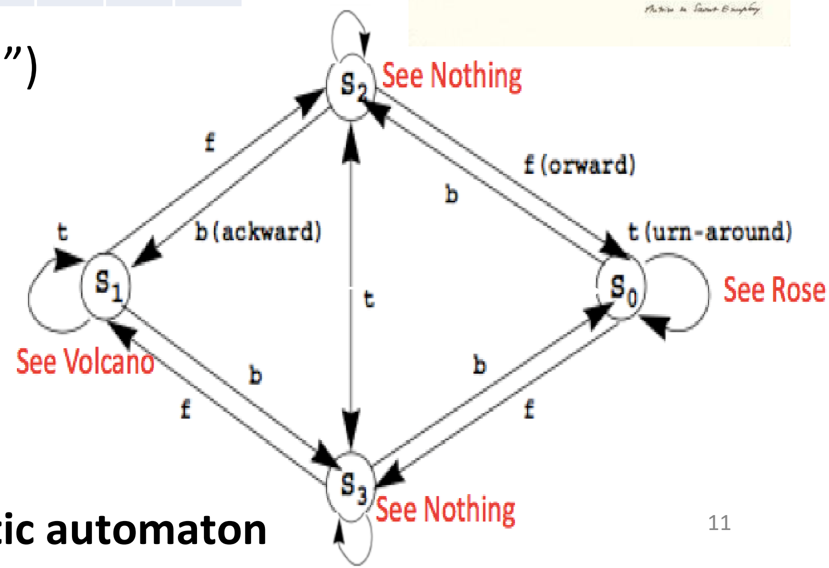
T	S0	S1	S2	S3
S0	0.7	0.1	0.1	0.1
S1	0.1	0.7	0.1	0.1
S2	0.1	0.1	0.1	0.7
S3	0.1	0.1	0.7	0.1

- Appearance Probabilities θ (aka “Sensor Model”)

θ	Rose	Volcano	Nothing
S0	0.8	0.1	0.1
S1	0.1	0.8	0.1
S2	0.1	0.1	0.8
S3	0.1	0.1	0.8

- Initial State Probabilities π (aka “Belief States”)

π	S0	S1	S2	S3
	.25	.25	.25	.25



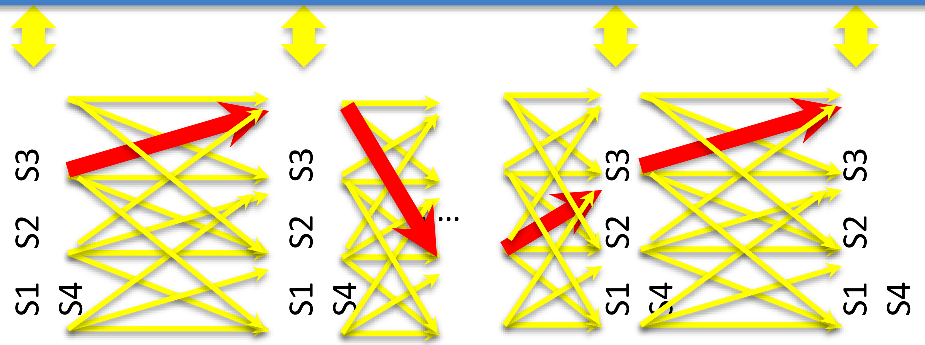
Stochastic automaton

Little Prince in Action

experience $E_{1:T}$ {rose}, fward, {none}, …, turn, {rose}, back, {volcano} time

state s

Hidden!



- Given: "*experience*" $\mathbf{E}_{1:T}$ (time 1 through T)
- You can Infer:
  - $\mathbf{P(X_t|E_{1:t})}$, where (which state) am I at now? (estimation, filtering, localization)
  - $\mathbf{P(X_{t+k}|E_{1:t})}$, where I will be at time t+k? (prediction)
  - $\mathbf{P(X_k|E_{1:t})}$, where I was at time k < t ? (smooth)
  - $\mathbf{P(X_{1:t}|E_{1:t})}$ the probability of every state sequence that I went through? (explanation)
  - $\mathbf{P(x_{1:t}|E_{1:t})}$ the most likely sequence of states I went through? (Viterbi algorithm)

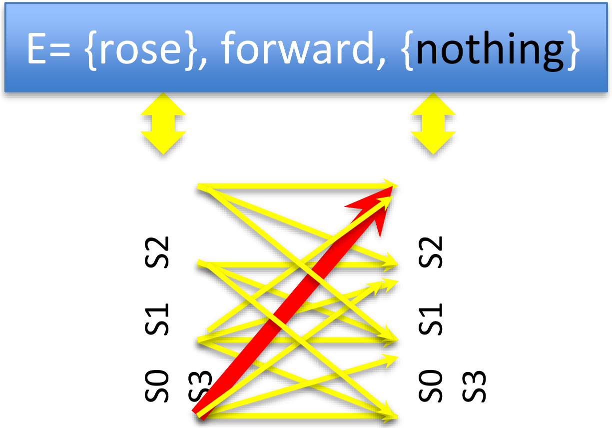
Some General Problems for Temporal Models

1 *k* *t* *t+k* time

- $\mathbf{P}(\mathbf{X}_{1:t}|\mathbf{E}_{1:t})$
  - compare all state sequences I might go through (”explanation”)
- $\mathbf{P}(\mathbf{X}_{t}|\mathbf{E}_{1:t})$
  - which state I am in now (“state estimation” “localization”)
- $\mathbf{P}(\mathbf{X}_{t+k}|\mathbf{E}_{1:t})$
  - which state I will be in at time t+k (“prediction”)
- $\mathbf{P}(\mathbf{X}_{k}|\mathbf{E}_{1:t})$
  - which state I was in at time k ($k<t$) (“smoothing”)
- $\mathbf{P}(\mathbf{x}_{1:t}|\mathbf{E}_{1:t})$
  - what is the best state sequence that I went through (”viterbi”)
- $\mathbf{P}(\mathbf{M}_{t}|\mathbf{E}_{1:t})$
  - How correct is my model at time t (“model learning”)

P(X|E) Example

- Given: "*experience*" $E_{1:2}$={rose, forward, nothing}
- Infer: the most likely "*state sequence*" $X_{1:2}$ *// by comparing all*



$E_{1:2}$={rose, forward, nothing}
$O_{1:2}$={rose, nothing} // observations
A_1={forward} // action sequence
$E_{1:2}=O_{1:2}A_1$

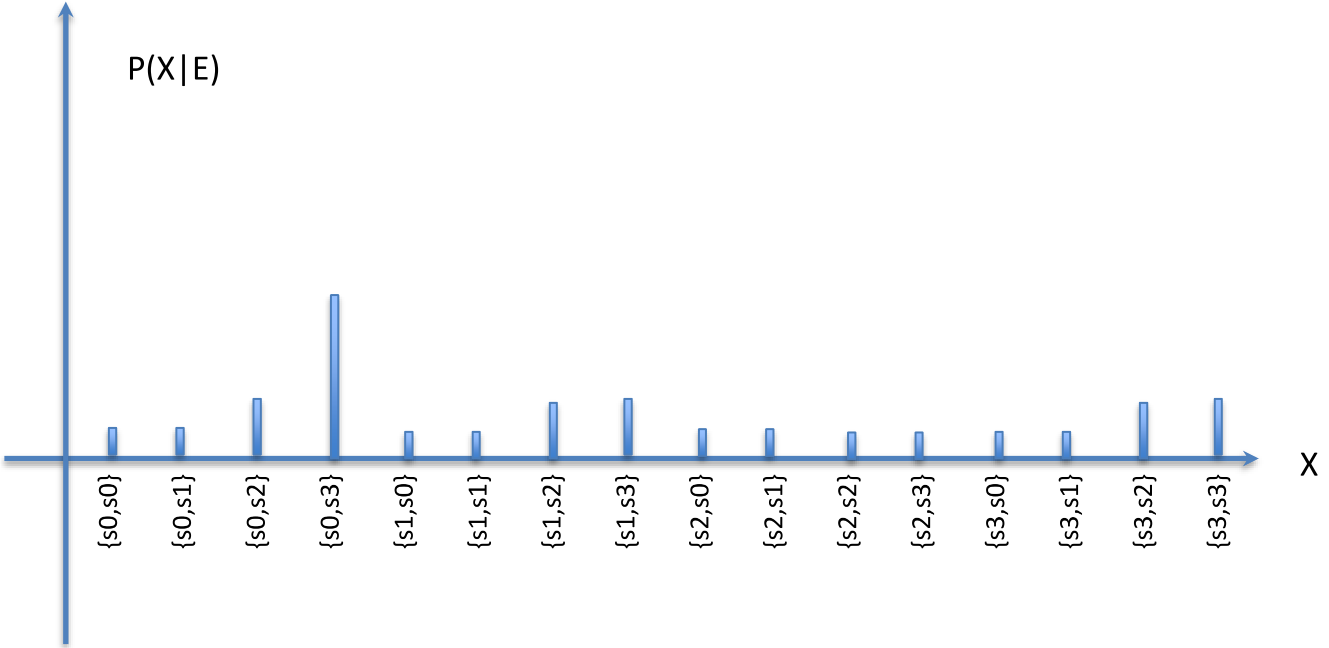
Compute P(X|E):

16 possible X, which one is most likely?

{s0,s0}, {s0,s1}, {s0,s2}, {s0,s3},
{s1,s0}, {s1,s1}, {s1,s2}, {s1,s3},
{s2,s0}, {s2,s1}, {s2,s2}, {s2,s3},
{s3,s0}, {s3,s1}, {s3,s2}, {s3,s3}

Find the "best sequence of states" that support the experience !

P(X|E) Example in a Graph



Compute $P(X_{1:T}|E_{1:T})$

- $P(X_{1:T}|E)$ where E=OA
- $X_{1:T}$ is all possible sequences of states
  - Let $x=\{i_1, i_2, \ldots, i_T\}$ be a sequence of states
- $P(x|E) = p(xE)/P(E) = P(xOA)/P(OA)$

  $= p(x|A)p(O|xA)/p(O|A)$
- $p(x|A)=$ $\pi_{i_1}(1)P_{i_1i_2}[b_1]P_{i_2i_3}[b_2]\cdots P_{i_{T-1}i_T}[b_{T-1}]$
- $p(O|xA)=$ $\theta_{i_1}(z_1)\theta_{i_2}(z_2)\cdots\theta_{i_{T-1}}(z_{T-1})\theta_{i_T}(z_T)$

E={rose, forward, nothing}
O={rose, nothing} // observations
A={forward} // action sequence

{s0,s0}, {s0,s1}, {s0,s2}, {s0,s3},
{s1,s0}, {s1,s1}, {s1,s2}, {s1,s3},
{s2,s0}, {s2,s1}, {s2,s2}, {s2,s3},
{s3,s0}, {s3,s1}, {s3,s2}, {s3,s3}

$O = \{z_1, z_2, \ldots, z_T\}$, $A = \{b_1, b_2, \ldots, b_{T-1}\}$, $x = \{i_1, i_2, \ldots, i_T\}$

In Our Example

- 16 possible x

{s0,s0}, {s0,s1}, {s0,s2}, {s0,s3},{s1,s0}, {s1,s1}, {s1,s2}, {s1,s3}, {s2,s0}, {s2,s1}, {s2,s2}, {s2,s3},{s3,s0}, {s3,s1}, {s3,s2}, {s3,s3}

- p(x|A)
  - $p(\{s_0,s_0\}|\{f\})=\pi(s_0)p_{s0,s0}(f)=.25*.1=.025$
  - $p(\{s_0,s_1\}|\{f\})=\pi(s_0)p_{s0,s1}(f)=.25*.1=.025$
  - $p(\{s_0,s_2\}|\{f\})=\pi(s_0)p_{s0,s2}(f)=.25*.1=.025$
  - $p(\{s_0,s_3\}|\{f\})=\pi(s_0)p_{s0,s3}(f)=.25*.7=.175$
  - $p(\{s_1,s_0\}|\{f\})=\pi(s_1)p_{s1,s0}(f)=.25*.1=.025$
  - $p(\{s_1,s_1\}|\{f\})=\pi(s_1)p_{s1,s1}(f)=.25*.1=.025$
  - $p(\{s_1,s_2\}|\{f\})=\pi(s_1)p_{s1,s2}(f)=.25*.1=.175$
  - $p(\{s_1,s_3\}|\{f\})=\pi(s_1)p_{s1,s3}(f)=.25*.7=.025$
  - $p(\{s_2,s_0\}|\{f\})=\pi(s_2)p_{s2,s0}(f)=.25*.1=.175$
  - $p(\{s_2,s_1\}|\{f\})=\pi(s_2)p_{s2,s1}(f)=.25*.1=.025$
  - $p(\{s_2,s_2\}|\{f\})=\pi(s_2)p_{s2,s2}(f)=.25*.1=.025$
  - $p(\{s_2,s_3\}|\{f\})=\pi(s_2)p_{s2,s3}(f)=.25*.7=.025$
  - $p(\{s_3,s_0\}|\{f\})=\pi(s_3)p_{s3,s0}(f)=.25*.1=.025$
  - $p(\{s_3,s_1\}|\{f\})=\pi(s_3)p_{s3,s1}(f)=.25*.1=.175$
  - $p(\{s_3,s_2\}|\{f\})=\pi(s_3)p_{s3,s2}(f)=.25*.1=.025$
  - $p(\{s_3,s_3\}|\{f\})=\pi(s_3)p_{s3,s3}(f)=.25*.7=.025$

- p(O|xA)
  - $\Theta s_0(r)\ \Theta s_0(n)=.8*.1=.08$
  - $\Theta s_0(r)\ \Theta s_1(n)=.8*.1=.08$
  - $\Theta s_0(r)\ \Theta s_2(n)=.8*.8=.64$
  - $\Theta s_0(r)\ \Theta s_3(n)=.8*.8=.64$
  - $\Theta s_1(r)\ \Theta s_0(n)=.1*.1=.01$
  - $\Theta s_1(r)\ \Theta s_1(n)=.1*.1=.01$
  - $\Theta s_1(r)\ \Theta s_2(n)=.1*.8=.08$
  - $\Theta s_1(r)\ \Theta s_3(n)=.1*.8=.08$
  - $\Theta s_2(r)\ \Theta s_0(n)=.1*.1=.01$
  - $\Theta s_2(r)\ \Theta s_1(n)=.1*.1=.01$
  - $\Theta s_2(r)\ \Theta s_2(n)=.1*.8=.08$
  - $\Theta s_2(r)\ \Theta s_3(n)=.1*.8=.08$
  - $\Theta s_3(r)\ \Theta s_0(n)=.1*.1=.01$
  - $\Theta s_3(r)\ \Theta s_1(n)=.1*.1=.08$
  - $\Theta s_3(r)\ \Theta s_2(n)=.1*.8=.08$
  - $\Theta s_3(r)\ \Theta s_3(n)=.1*.8=.08$

E={rose, forward, nothing}
O={rose, nothing},
A={forward}

Best Explanation is: {s0, s3}

Which Explanation is the Best?

- Among all possible sequences of states, the best "explanation" is the sequence of states that gives the maximal value for

$$\pi_{i_1}(1)\theta_{i_1}(z_1)P_{i_1 i_2}[b_1]\cdots\theta_{i_{T-1}}(z_{T-1})P_{i_{T-1} i_T}[b_{T-1}]\theta_{i_T}(z_T)$$

Experience:

Observations: $o_1,\ o_2, \ldots, o_{T-1},\ o_T$

Actions: $b_1, b_2, \ldots,\ b_{T-1}$

Sensor models: $\theta_i(k) = p(z_k|s_i),\ z_k \in Z$, and $s_i \in S$,

Action models: $P_{ij}[b] = P(s_j \mid s_i, b)$

Explanation: State sequence: $i_1, i_2, i_3, \ldots, i_T$

Compute Hidden State Sequence

$O = \{z_1, z_2, \ldots, z_T\}$, $A = \{b_1, b_2, \ldots, b_T\}$, $I = \{i_1, i_1, \ldots, i_T\}$

- *Experience consists of both O* and *A*
- O is observation sequence and *A* is action sequence in experience
- *M* is the model, *C* is the background information

$$
\begin{aligned}
p(I|AMC) &= \pi_{i_1}(1)P_{i_1 i_2}[b_1]P_{i_2 i_3}[b_2]\cdots P_{i_{T-1} i_T}[b_{T-1}] \\
p(O|IAMC) &= \theta_{i_1}(z_1)\theta_{i_2}(z_2)\cdots\theta_{i_{T-1}}(z_{T-1})\theta_{i_T}(z_T)
\end{aligned}
$$

On slide #16, we used *x* for *I, so* these two terms were written as p(x|A) and p(O|xA) there.

Compute Hidden State Sequence

$$p(O|AMC) \;=\; \sum_I p(O,I|AMC) = \sum_I p(I|AMC)p(O|IAMC)$$

Therefore, we have

$$p(O|AMC) = \sum_{I=i_1 i_2 \ldots i_T} \pi_{i_1}(1)\theta_{i_1}(z_1)P_{i_1 i_2}[b_1]\cdots\theta_{i_{T-1}}(z_{T-1})P_{i_{T-1}i_T}[b_{T-1}]\theta_{i_T}(z_T)$$

Among all possible sequences in ***I***, there is one with the maximal probability.

However, this straightforward way of computing $p(O|AMC)$ is very expensive. It requires on the order of $2TN^T$ calculations, since at every time $t = 1, 2, \ldots, T$, there are N possible states to go through and for each summand about $2T$ calculations are required. Clearly a more efficient procedure is required. One such procedure is the combination of the forward and backward procedures.

P(O|AMC) by Forward Procedure

Main idea: do not consider all possible state sequences, but every step in the experience, and compute P($O|AMC$) incrementally on the time t:

$t = 1, 2, \ldots, T$ as follows:

$$\begin{aligned}
\alpha_1(i) &= p(z_1, i_1 = s_i | AMC) \\
\alpha_2(i) &= p(z_1, z_2, i_2 = s_i | AMC) \\
&\vdots \\
\alpha_t(i) &= p(z_1, z_2, \ldots, z_t, i_t = s_i | AMC) \\
&\vdots \\
\alpha_{T-1}(i) &= p(z_1, z_2, \ldots, z_{T-1}, i_{T-1} = s_i | AMC) \\
\alpha_T(i) &= p(z_1, z_2, \ldots, z_{T-1}, z_T, i_T = s_i | AMC)
\end{aligned}$$

The term $\alpha_1(i)$ represents the probability of being in state s_i and seeing z_1 at time $t = 1$. The term $\alpha_t(i)$ represents the probability of seeing $\{z_1, \ldots, z_t\}$ and ending in state s_i at time t.

Forward Procedure

The computation of $\alpha_t(i)$ is straightforward. When $t = 1$, the probability of $\alpha_1(i)$ depends only on θ and π:

$$\alpha_1(i) = \pi_i \theta_i(z_1) \tag{5.9}$$

Furthermore, the value of $\alpha_{t+1}(j)$ can be computed recursively based on the values of $\alpha_t(i)$. In other words, in order to be in state s_j at time $t+1$, the system must have been in any previous state s_i at time t (with probability $\alpha_t(i)$) and then made a transition to state s_j with probability $P_{ij}[b_t]$. Thus the probability of reaching state s_j at time $t+1$ is the sum, for all $i \in S$, of the product of three probabilities: $\alpha_t(i)$, the probability of being at state s_i at time t; $P_{ij}[b_t]$, the probability of taking the transition from s_i to s_j under action b_t, and, $\theta_j(z_{t+1})$, the probability of seeing z_{t+1} at state s_j. That is,

$$\alpha_{t+1}(j) = \sum_{i \in S} \alpha_t(i) P_{ij}[b_t] \theta_j(z_{t+1}) \tag{5.10}$$

Forward Procedure

Finally, when $\alpha_T(i)$ is known for all i, we can calculate the value of $p(O|AMC)$ by summing up $\alpha_T(i)$ on all model states. That is,

Complexity is *O*(TN^2)

$$p(O|AMC) = \sum_{s_i \in S} \alpha_T(i) \tag{5.11}$$

Forward Procedure Example

Time: t_1------------------------------------t_2-- t_3------------
E = { rose, forward, none, turn, volcano, ...}

$\alpha_2(i)$:

$\alpha_2(s_0) = \alpha_1(s_0)\, p_{s0,s0}(f)\, \Theta s_0(n)$
$+ \alpha_1(s_1)\, p_{s1,s0}(f)\, \Theta s_0(n)$
$+ \alpha_1(s_2)\, p_{s2,s0}(f)\, \Theta s_0(n)$
$+ \alpha_1(s_3)\, p_{s3,s0}(f)\, \Theta s_0(n)$

$\alpha_2(s_1) = \alpha_1(s_0)\, p_{s0,s1}(f)\, \Theta s_1(n)$
$+ \alpha_1(s_1)\, p_{s1,s1}(f)\, \Theta s_1(n)$
$+ \alpha_1(s_2)\, p_{s2,s1}(f)\, \Theta s_1(n)$
$+ \alpha_1(s_3)\, p_{s3,s1}(f)\, \Theta s_1(n)$

$\alpha_2(s_2) = \alpha_1(s_0)\, p_{s0,s2}(f)\, \Theta s_2(n)$
$+ \alpha_1(s_1)\, p_{s1,s2}(f)\, \Theta s_2(n)$
$+ \alpha_1(s_2)\, p_{s2,s2}(f)\, \Theta s_2(n)$
$+ \alpha_1(s_3)\, p_{s3,s2}(f)\, \Theta s_2(n)$

$\alpha_2(s_3) = \alpha_1(s_0)\, p_{s0,s3}(f)\, \Theta s_3(n)$
$+ \alpha_1(s_1)\, p_{s1,s3}(f)\, \Theta s_3(n)$
$+ \alpha_1(s_2)\, p_{s2,s3}(f)\, \Theta s_3(n)$
$+ \alpha_1(s_3)\, p_{s3,s3}(f)\, \Theta s_3(n)$

$\alpha_3(i)$:

$\alpha_3(s_0) = \alpha_2(s_0)$
$p_{s0,s0}(t)\, \Theta s_0(v)$
$+ \alpha_2(s_1)\, p_{s1,s0}(t)\, \Theta s_0(v)$
$+ \alpha_2(s_2)\, p_{s2,s0}(t)\, \Theta s_0(v)$
$+ \alpha_2(s_3)\, p_{s3,s0}(t)\, \Theta s_0(v)$

$\alpha_3(s_1) = \alpha_2(s_0)\, p_{s0,s1}(t)\, \Theta s_1(v)$
$+ \alpha_2(s_1)\, p_{s1,s1}(t)\, \Theta s_1(v)$
$+ \alpha_2(s_2)\, p_{s2,s1}(t)\, \Theta s_1(v)$
$+ \alpha_2(s_3)\, p_{s3,s1}(t)\, \Theta s_1(v)$

$\alpha_3(s_2) = \alpha_2(s_0)\, p_{s0,s2}(t)\, \Theta s_2(v)$
$+ \alpha_2(s_1)\, p_{s1,s2}(t)\, \Theta s_2(v)$
$+ \alpha_2(s_2)\, p_{s2,s2}(t)\, \Theta s_2(v)$
$+ \alpha_2(s_3)\, p_{s3,s2}(t)\, \Theta s_2(v)$

$\alpha_3(s_3) = \alpha_2(s_0)\, p_{s0,s3}(t)\, \Theta s_3(v)$
$+ \alpha_2(s_1)\, p_{s1,s3}(t)\, \Theta s_3(v)$
$+ \alpha_2(s_2)\, p_{s2,s3}(t)\, \Theta s_3(v)$
$+ \alpha_2(s_3)\, p_{s3,s3}(t)\, \Theta s_3(v)$

$\alpha_1(i)$:

$\alpha_1(s_0) = \pi(s_0)\Theta s_0(r) = .25 * .8 = .20$
$\alpha_1(s_1) = \pi(s_1)\Theta s_1(r) = .25 * .1 = .025$
$\alpha_1(s_2) = \pi(s_2)\Theta s_2(r) = .25 * .1 = .025$
$\alpha_1(s_3) = \pi(s_3)\Theta s_3(r) = .25 * .1 = .025$

Backward Procedure

- Similar to forward
- But starting from the end and walking backwards

- See ALFE 5.10.1

The backward procedure is symmetric to the forward procedure. Instead of incrementing E from the beginning to the end, it increments from the end to the beginning. In particular, we have

$$\begin{aligned}
\beta_{T-1}(i) &= p(z_T|AMC, i_{T-1} = s_i) \\
\beta_{T-2}(i) &= p(z_{T-1}, z_T|AMC, i_{T-2} = s_i) \\
&\vdots \\
\beta_t(i) &= p(z_{t+1}, \ldots, z_{T-1}, z_T|AMC, i_t = s_i) \\
&\vdots \\
\beta_2(i) &= p(z_3, \cdots, z_{T-1}, z_T|AMC, i_2 = s_i) \\
\beta_1(i) &= p(z_2, z_3, \cdots, z_{T-1}, z_T|AMC, i_1 = s_i)
\end{aligned}$$

Each quantity $\beta_t(i)$ specifies the probability of seeing the sequence $z_{t+1}, \ldots, z_T$ if the state at time t is s_i.

As in the forward procedure, the value of $\beta_{t-1}(i)$ can be easily computed. When $t = T$, we define

$$\beta_T(i) = 1 \ \text{ for all states } s_i \tag{5.12}$$

Furthermore, the value of $\beta_{t-1}(i)$ can be computed based on the values of $\beta_t(j)$. In other words, in order to be in state s_i at time $t-1$, the system would have to be (with probability $\beta_t(i)$) in some next state s_j at time t with observation z_t, having made a transition from s_i to s_j with probability $P_{ij}[b_{t-1}]$. Thus, the probability of being at state i at time $t-1$ is the sum of the product of the following three probabilities: $P_{ij}[b_{t-1}]$, the probability of taking the transition from s_i to s_j under action b_{t-1}, $\theta_j(z_t)$, the probability of seeing z_t at s_j, and, $\beta_t(j)$, the probability of seeing the rest of observations after time t given state j at time t. Thus, our equation is

$$\beta_{t-1}(i) = \sum_{j \in S} P_{ij}[b_{t-1}]\theta_j(z_t)\beta_t(j) \tag{5.13}$$

Finally, when all $\beta_1(i)$ are known, $p(O|AMC)$ can be computed by summing up $\beta_1(i)$ on all model states. That is,

$$p(O|AMC) = \sum_{s_i \in S} \pi_i(1)\theta_i(z_1)\beta_1(i) \tag{5.14}$$

Thus, equations 5.12, 5.13, and 5.14 constitute the backward procedure.

Inference: State Estimation

1 *k* *t* *t+k* time

- $\mathbf{P}(\mathbf{X}_t | \mathbf{E}_{1:t})$
  - Which state I am most likely in now?
  - It is the state s_i such that $\alpha_T(i)$ is maximal
  - Viterbi algorithm could be used for this.

Inference: State Prediction

1 *k* *t* *t+k* time

- $\mathbf{P}(\mathbf{X}_{t+k} | \mathbf{E}_{1:t})$
  - which state I will be in at time *t+k*?
  - Depending on your action during (*t*, *t+k*]
  - If you have a non-action, use that information and continue computing the future α values.

  - Example/exercise: compute when k=1, k=2, …

Inference: State Smoothing

1 *k* *t* *t+k* time

- $\mathbf{P(X_k | E_{1:t})}$
  - which state I was in at time k (smoothing)
  - It is the state s_i such that
    $\alpha_k(i)\ P_{ij}(b_{k+1})\ \theta_j(z_{k+1})\ \beta_{k+1}(j)$. is maximal
  - This is to say, given time 1,…,k, k+1, ...,t

    (best forward from 1 to k)

    & (best transition from k to k+1)

    & (best backward to k+1 from t)

State Explanation

- $\mathbf{P}(\mathbf{X}_{1:t} | \mathbf{E}_{1:t})$
  - what states I have been through (explanation)
  - They are the following states:
    - t=1: the state s_i such that $\alpha_1(i)$ is the maximal
    - t=2: the state s_i such that $\alpha_2(i)$ is the maximal

    - t=T: the state s_i such that $\alpha_T(i)$ is the maximal

Model Learning

- $\mathbf{P}(\mathbf{M}_t | \mathbf{E}_{1:t})$
  - How correct is my model of the world (learning)
  - We will teach you this later.

Temporal Models

- Models with actions and sensors (ALFE 4-5) (aka POMDP)
- Markov Chains
  - No observation, no explicit actions, transit randomly
- Hidden Markov Model
  - No explicit actions, state transit randomly (e.g., predicting weather)
- Continuous State Model
  - No explicit actions, States are continuous
- Dynamic Bayesian Networks
  - No explicit actions, States are Bayesian Networks
- POMDP
  - Discrete states with probabilistic actions, sensors, & transitions

Hidden Markov Models

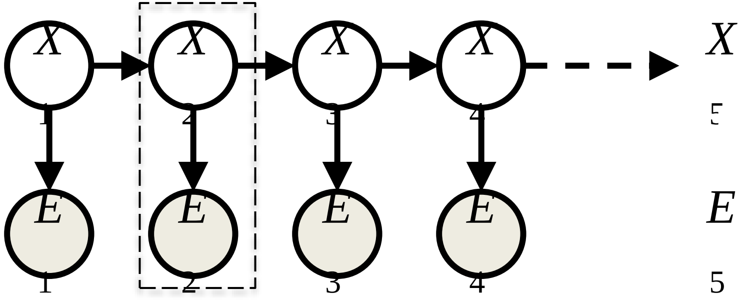
- Markov chains not so useful for most agents
  - Eventually you don't know anything anymore
  - Need observations to update your beliefs

- Hidden Markov models (HMMs)
  - Underlying Markov chain over states S
  - You observe outputs (effects) at each time step
  - As a Bayes' net:

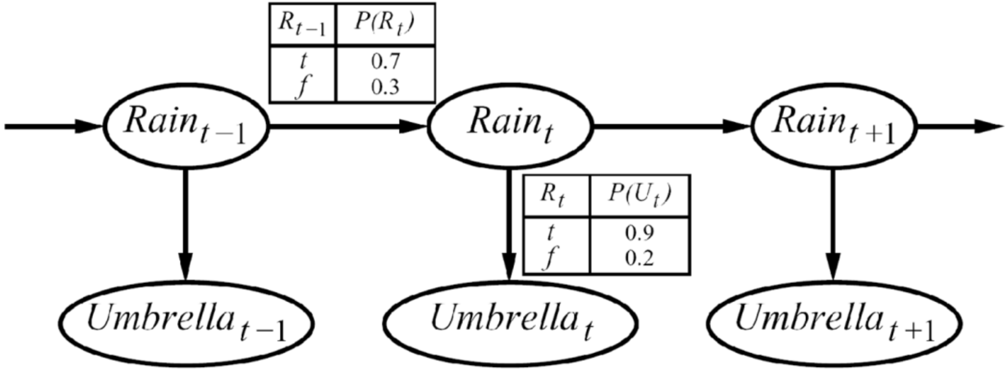
Compared to the general action/sensor models, HMM does not have an explicit action model.

Hidden States

observation Evidence



Rain/Umbrella Example in the Book

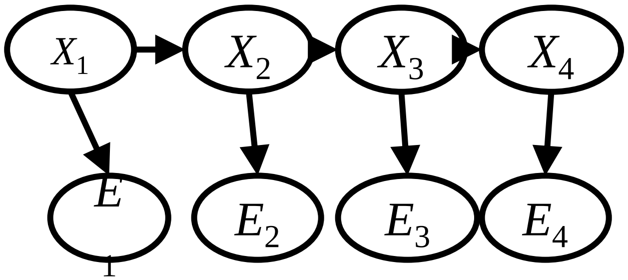


Story background:
Security guard at top-secret underground facility.
Is it raining? No direct observation.
But observe whether director has umbrella when coming to work.

- An HMM is defined by:
  - Initial distribution: $P(X_1)$
  - Transitions: $P(X|X_{-1})$
  - Emissions (sensor model): $P(E|X)$

Conditional Independence

- HMMs have two important independence properties:
  - Markov hidden process, future depends only on the present
  - Current observation is independent of all else, given the current state



- Quiz: does this mean that observations are independent?
  - [No, correlated by the hidden state]

Real HMM Examples

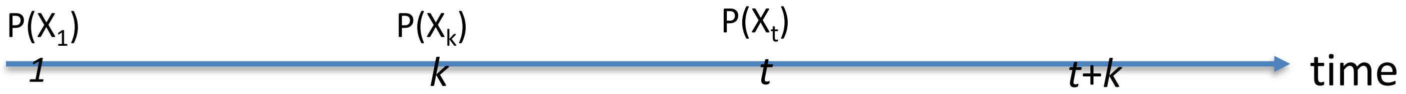
- Speech recognition HMMs:
  - Observations are acoustic signals (continuous valued)
  - States are specific positions in specific words (so, tens of thousands)

- Machine translation HMMs:
  - Observations are words (tens of thousands)
  - States are translation options

- Robot tracking:
  - Observations are range readings (continuous/discrete)
  - States are positions on a map (continuous/discrete)

Localization, Filtering, Monitoring

- Localization, Filtering, or monitoring, is the task of tracking the distribution P(X) (the belief state) over time
- We start with $P(X_1)$ in an initial setting, usually uniform
- As time passes, *1,…,t*, when we have observations, we update $P(X_t)$:
- $\mathbf{P}(\mathbf{X}_t|\mathbf{E}_{1:t})$



Filtering Example

<u>Day 0:</u>

$P(R_0)=\langle 0.5,0.5\rangle$

<u>Day 1:</u>

$P(R_1)=P(r_0)P(R_1|r_0)$

$= \alpha 0.5\langle 0.7,0.3\rangle + \alpha 0.5\langle 0.3,0.7\rangle$

$= \mathbf{\langle 0.5,0.5\rangle}$

<u>observe Umbrella appears ⍰ U_1=true</u>

updating with evidence for t=1 gives:

$P(R_1|u_1)=\alpha P(u_1|R_1)P(R_1)=\alpha\langle 0.9,0.2\rangle\langle 0.5,0.5\rangle$

$=\alpha\langle 0.45,0.1\rangle \mathbf{= \langle 0.818,0.182\rangle}$

<u>Day 2:</u>

$P(R_2|u_1)=P(R_2|r_1)\ P(r_1|u_1)$

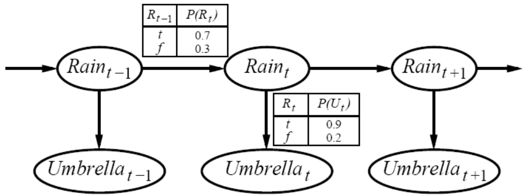
$=\alpha\langle 0.818\langle 0.7,0.3\rangle + 0.182\langle 0.3,0.7\rangle\rangle \mathbf{= \langle 0.627,0.373\rangle}$

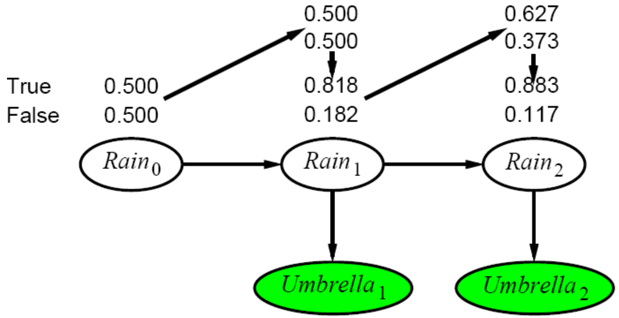
<u>observe Umbrella appears ⍰ U_2=true</u>

updating with evidence for t=2 gives:

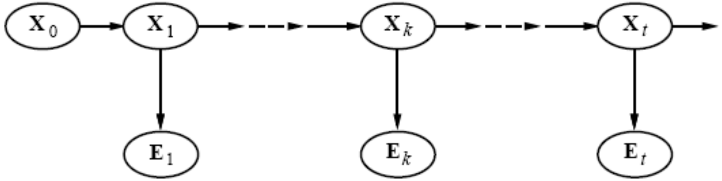
$P(R_2|u_1,u_2)=\alpha P(u_2|R_2)P(R_2|u_1)=\alpha\langle 0.9,0.2\rangle\langle 0.627,0.373\rangle$

$=\alpha\langle 0.565,0.075\rangle \mathbf{= \langle 0.883,0.117\rangle}$





Smoothing: $P(X_k | E_{1:T})$



- Divide evidence $e_{1:t}$ into $e_{1:k}$, $e_{k+1:t}$:

$$
\begin{aligned}
P(\mathbf{X}_k | \mathbf{e}_{1:t}) &= P(\mathbf{X}_k | \mathbf{e}_{1:k}, \mathbf{e}_{k+1:t})) \\
&= \alpha P(\mathbf{X}_k | \mathbf{e}_{1:k}) P(\mathbf{e}_{k+1:t} | \mathbf{X}_k, \mathbf{e}_{1:k}) && \text{Using Bayes rule} \\
&= \alpha P(\mathbf{X}_k | \mathbf{e}_{1:k}) P(\mathbf{e}_{k+1:t} | \mathbf{X}_k) && \text{Using conditional independence} \\
&= \alpha f_{1:k} b_{k+1:t}
\end{aligned}
$$

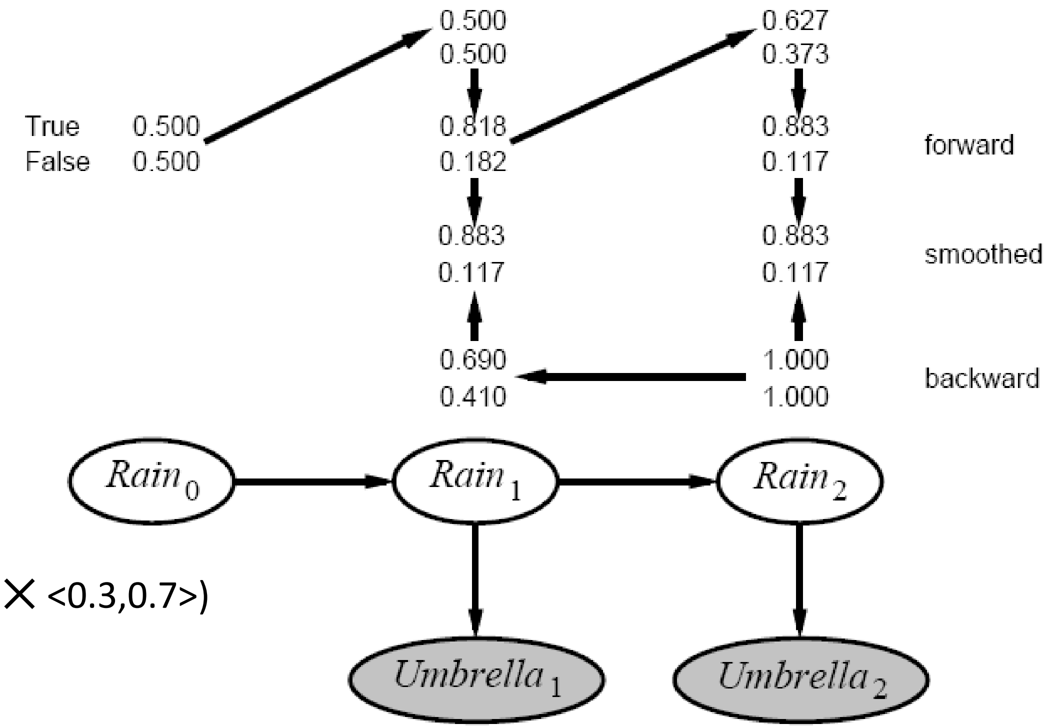
- Backward message computed by a backwards recursion:

$$
\begin{aligned}
P(\mathbf{e}_{k+1:t} | \mathbf{X}_k) &= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1:t} | \mathbf{X}_k, \mathbf{x}_{k+1}) P(\mathbf{x}_{k+1} | \mathbf{X}_k) && \text{Conditioning on } X_{k+1} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1:t} | \mathbf{x}_{k+1}) P(\mathbf{x}_{k+1} | \mathbf{X}_k) && \text{Using conditional independence} \\
&= \sum_{\mathbf{x}_{k+1}} P(\mathbf{e}_{k+1} | \mathbf{x}_{k+1}) \boxed{P(\mathbf{e}_{k+2:t} | \mathbf{x}_{k+1})} P(\mathbf{x}_{k+1} | \mathbf{X}_k)
\end{aligned}
$$

$= \text{BACKWARD}(b_{k+2:t}, e_{k+1:t})$

$$b_{k+1:t} = \Sigma_{Xt+1}(P(e_{k+1}|x_{k+1})P(x_{k+1}|X_k)\ b_{k+2:t}$$

Smoothing Example



<u>Compute estimate for rain at t=1</u>

$P(R_1|u_1,u_2)=\alpha P(R_1|u_1)P(u_2|R_1)$

$P(R_1|u_1) = <0.818,0.182>$

$P(u_2|R_1) =\Sigma_{r\mathbf{2}}P(u_2|r_2)\ P(|r_2)\ P(r_2|R_1)$

$=(0.9 \times 1 \times <0.7,0.3>) + (0.2 \times 1 \times <0.3,0.7>)$

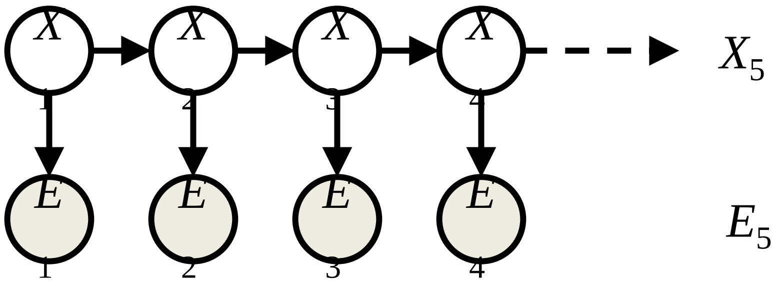
$= <0.69,0.41>$

<u>Smoothed estimate:</u>

$P(R_1|u_1,u_2)=\alpha<0.818,0.182> \times <0.69,0.41>$ **= <0.883,0.117>**

- Forward-backward algorithm: cache forward messages along the way
- Time linear in t (polytree inference), space $O(t_j f_j)$

Best Explanation Queries

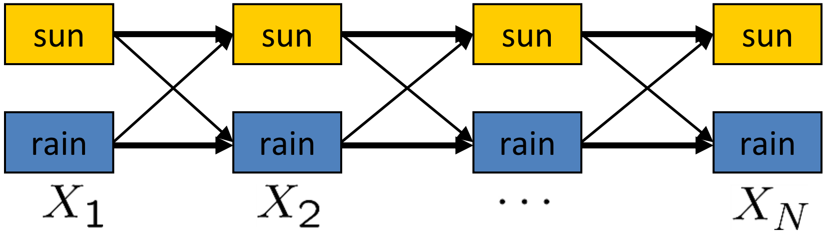


- Query: what is the most likely sequence of states?

$$\arg\max_{x_{1:t}} P(x_{1:t}|e_{1:t})$$

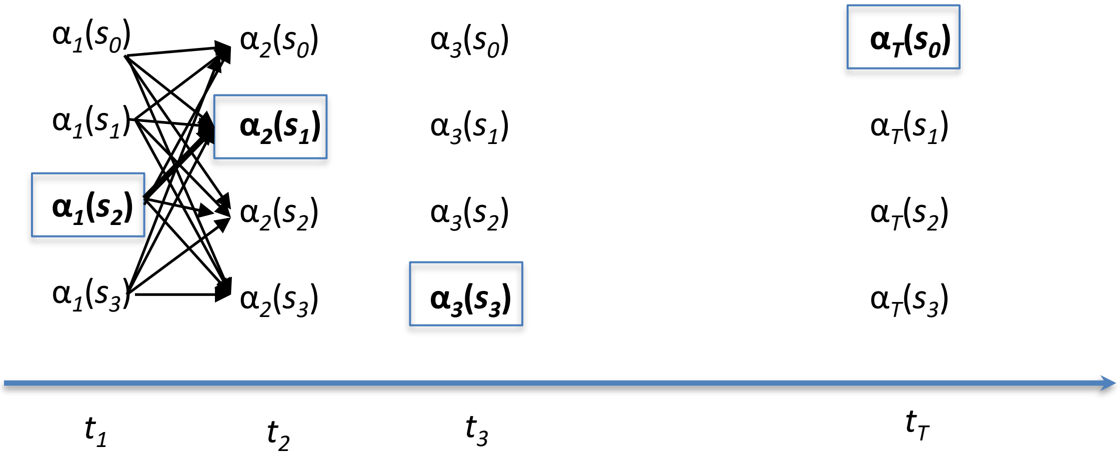
State Path Trellis

- State trellis: graph of states and transitions over time



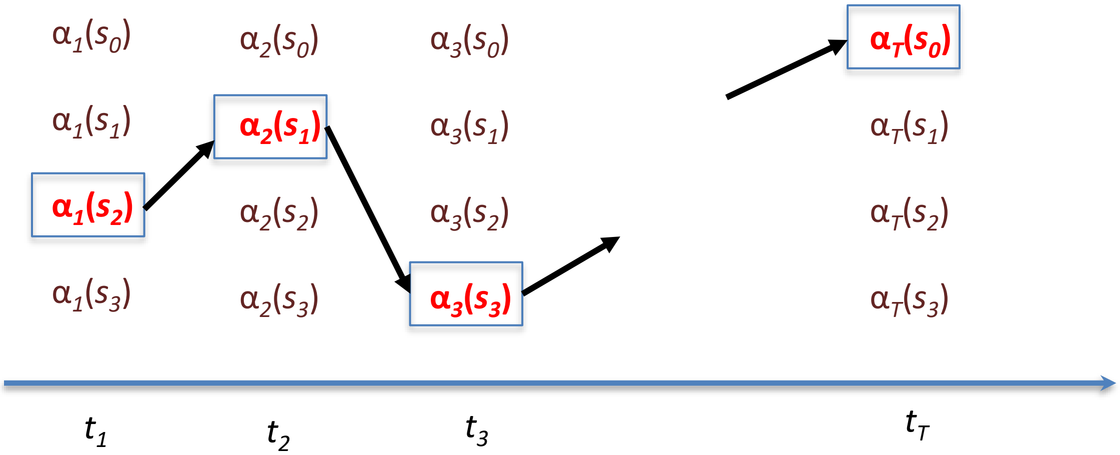
- Each arc represents some transition $x_{t-1} \rightarrow x_t$
- Each arc has weight $P(x_t|x_{t-1})P(e_t|x_t)$
- Each path is a sequence of states
- The product of weights on a path is the seq's probability
- Can think of the Forward (and now Viterbi) algorithms as computing sums of all paths (best paths) in this graph

Forward Procedure Computes <u>All</u> $\alpha_t(s_i)$ on State Trellis



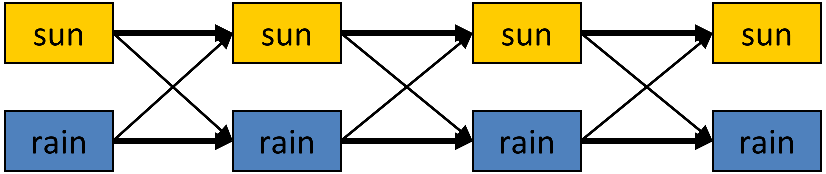
No need to compute all $\alpha_t(s_i)$, if we only need the best for each step
The Viterbi algorithm does this.

Viterbi Algorithm Computes Only the Best $\alpha_t(s_i)$ on State Trellis



No need to compute all $\alpha_t(s_i)$, if we only want to know the best for each step
The Viterbi algorithm does this.

Viterbi Algorithm: choose the <u>best</u> state seq.
Similar to before $\alpha_t(s_t)$, but use max not sum



$$x^*_{1:T} = \arg\max_{x_{1:T}} P(x_{1:T}|e_{1:T}) = \arg\max_{x_{1:T}} P(x_{1:T}, e_{1:T})$$

$\boxed{\alpha_t(s_t)}$

$$m_t[x_t] = \max_{x_{1:t-1}} P(x_{1:t-1}, x_t, e_{1:t})$$

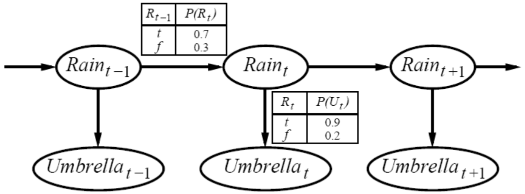
$$= \max_{x_{1:t-1}} P(x_{1:t-1}, e_{1:t-1}) P(x_t|x_{t-1}) P(e_t|x_t)$$

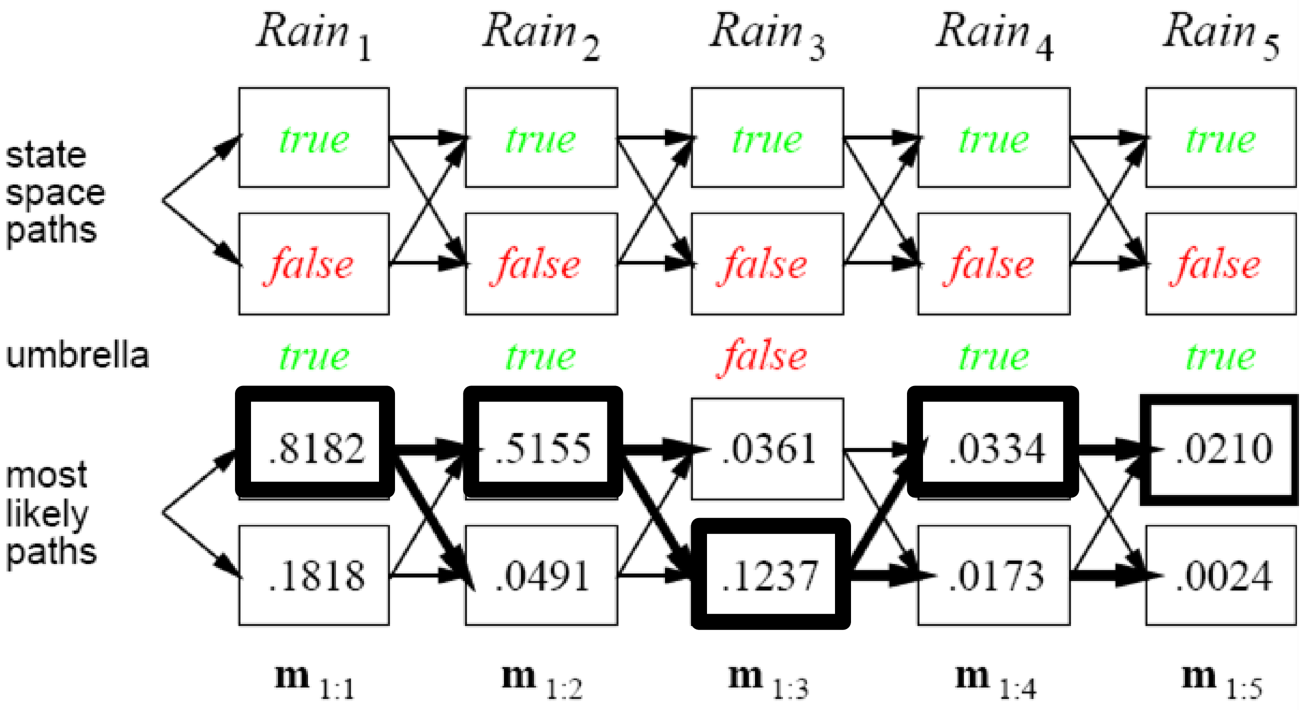
$$= P(e_t|x_t) \max_{x_{t-1}} P(x_t|x_{t-1}) \max_{x_{1:t-2}} P(x_{1:t-1}, e_{1:t-1})$$

$$= P(e_t|x_t) \max_{x_{t-1}} P(x_t|x_{t-1}) m_{t-1}[x_{t-1}]$$

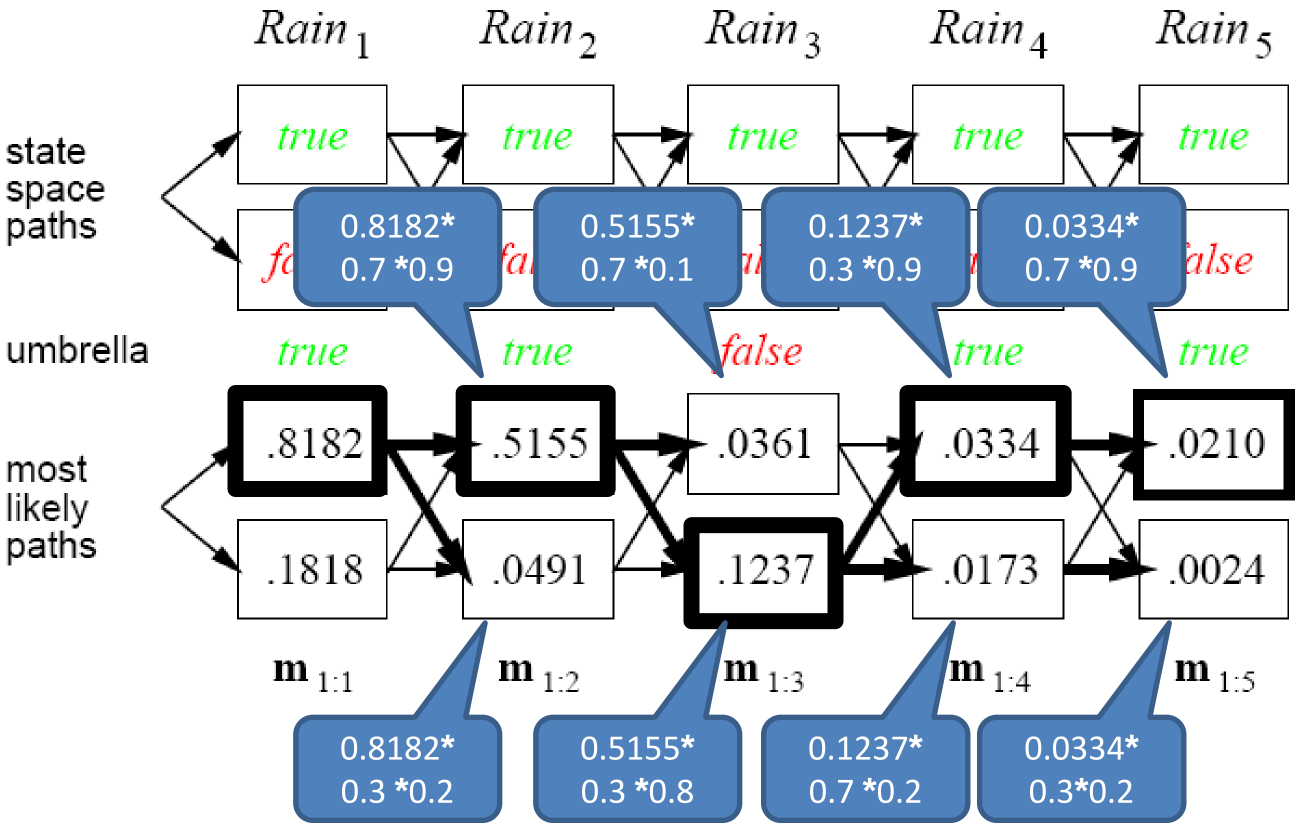
$\boxed{\alpha_{t-1}(s_{t-1})}$

Viterbi Example





Viterbi Example

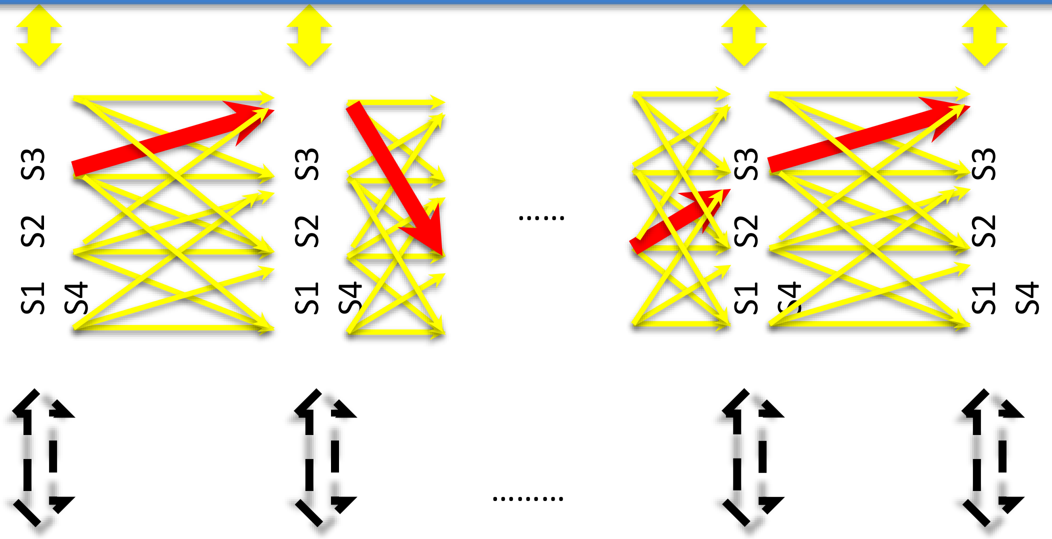


Temporal Models

- Models with actions and sensors (ALFE 4-5) (aka POMDP)
- Markov Chains
  - No observation, no explicit actions, transit randomly
- Hidden Markov Model
  - No explicit actions, state transit randomly
- Dynamic Bayesian Networks
  - No explicit actions, States are Bayesian Networks
- Continuous State Model
  - No explicit actions, States are continuous
- POMDP
  - Discrete states with probabilistic actions, sensors, & transitions

Dynamic Bayesian Networks

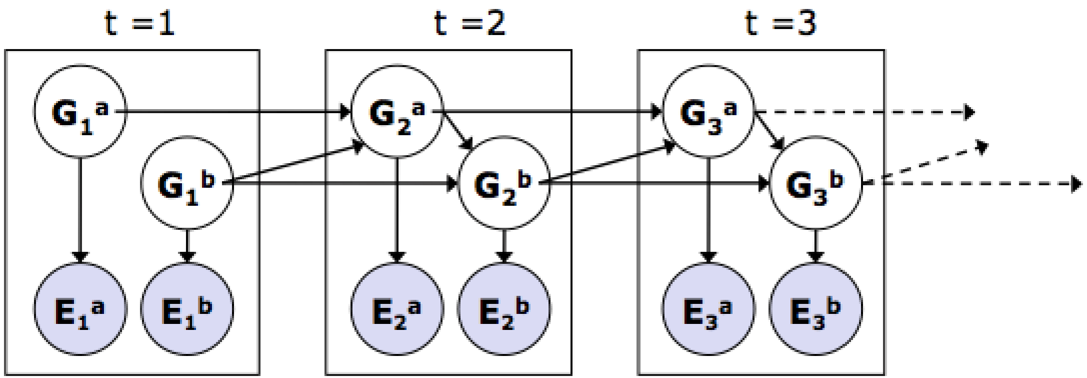
{evidence}, action, {evidence}, …, action, …, {evidence}, action, {evidence}



Each time "slice" is a Bayesian Network with variables and CPTs

Dynamic Bayes Nets (DBNs)

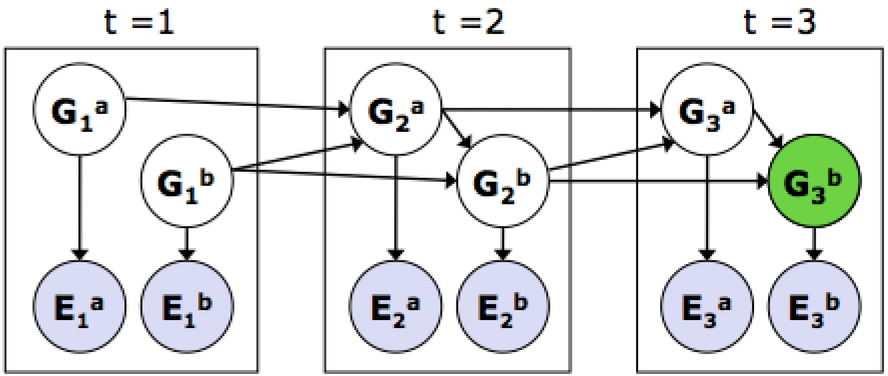
- We want to track multiple variables over time, using multiple sources of evidence
- Idea: Repeat a fixed Bayes net structure at each time
- Variables from time *t* can condition on those from *t-1*



- Discrete valued dynamic Bayes nets are also HMMs

Exact inference in DBNs

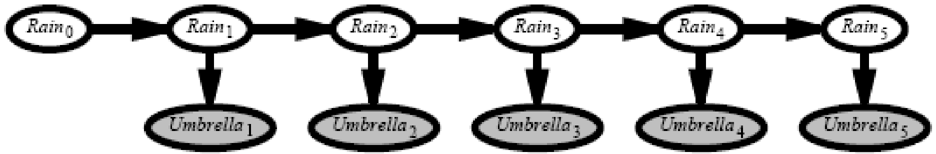
- Variable elimination applies to Dynamic Bayesian nets
- Procedure: "unroll" the network for T time steps, then eliminate variables until $P(X_T|e_{1:T})$ is computed



- Online belief updates: Eliminate all variables from the previous time step; store factors for current time only

Likelihood weighting for DBNs

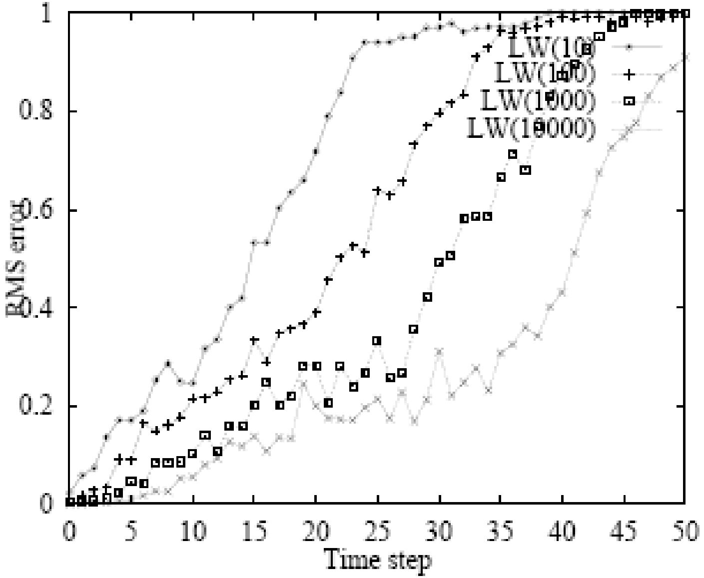
Set of weighted samples approximates the belief state



LW samples pay no attention to the evidence!

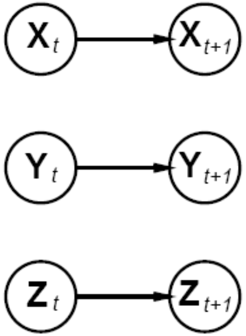
) fraction “agreeing” falls
exponentially with t

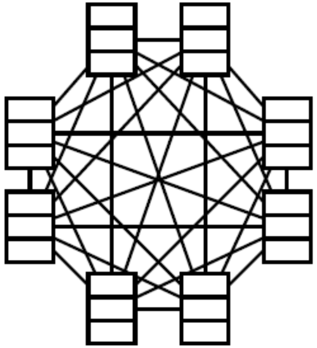
) num. samples required
grows exponentially with t



DBNs vs. HMMs

- Every HMM is a single-variable DBN; every discrete DBN is an HMM



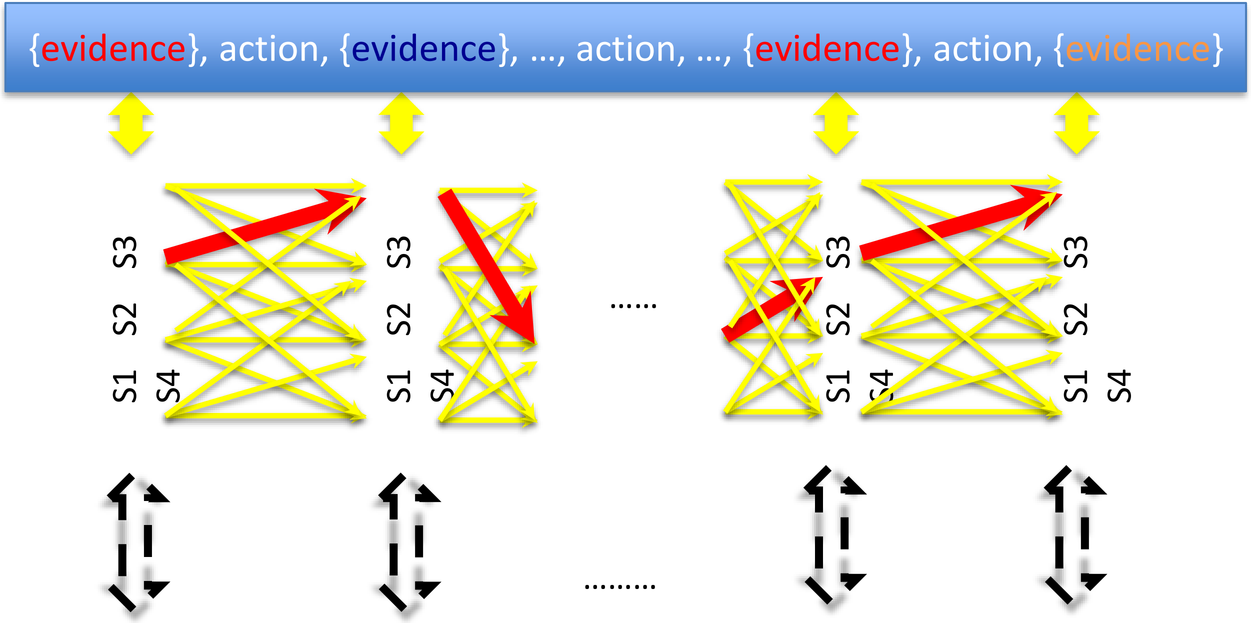


- DBN: Sparse dependencies, exponentially fewer parameters;
  e.g., 20 state variables, three parents each
  DBN has $20 \times 2^3 = 160$ parameters, HMM has $2^{20} \times 2^{20} = 10^{12}$

Temporal Models

- Models with actions and sensors (ALFE 4-5) (aka POMDP)
- Markov Chains
  - No observation, no explicit actions, transit randomly
- Hidden Markov Model
  - No explicit actions, state transit randomly
- Dynamic Bayesian Networks
  - No explicit actions, States are Bayesian Networks
- Continuous State Model
  - No explicit actions, States are continuous
- POMDP
  - Discrete states with probabilistic actions, sensors, & transitions

Kalman Filters



In each time slice, state variables are continuous (not discrete)

Summary

- Temporal models use states, transitions, sensors
  - Transitions may be related to agent's actions, or spontaneous
- Markov assumptions and stationarity assumption, so we need

  - transition model $P(X_t \mid X_{t-1})$ or $P(X_t \mid X_{t-1}, a_{t-1})$

  - sensor model $P(E_t \mid X_t)$
- Tasks: filtering, prediction, smoothing, most likely seq
  - all done recursively with constant cost per time step
- Types of models
  - HMM have a single discrete state variable
  - Dynamic Bayes nets subsume HMMs; exact update intractable
  - Other models may have internal structure driven by actions