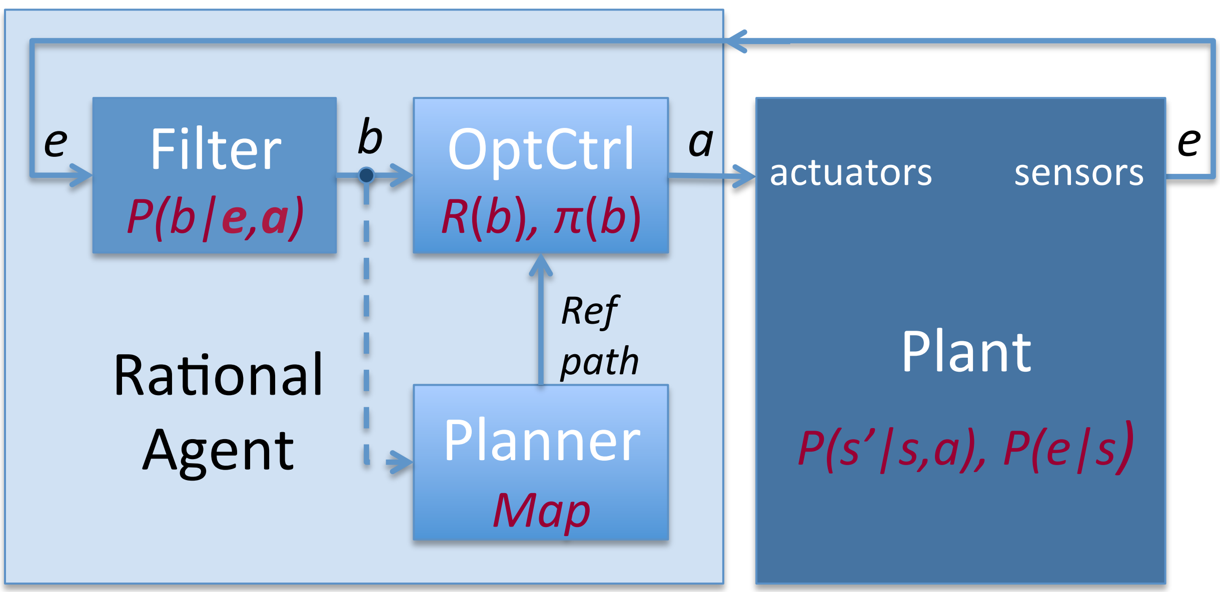
Probabilistic Reasoning Over Time

Chapter 15 (Models, Filters)



Reasoning over Time or Space

- Often, we want to reason about a sequence of observations

Reasoning over Time or Space

- Often, we want to reason about a sequence of observations
  - Speech recognition
  - Robot localization
  - User attention
  - Medical monitoring

Reasoning over Time or Space

- Often, we want to reason about a sequence of observations
  - Speech recognition
  - Robot localization
  - User attention
  - Medical monitoring

- Need to introduce time (or space) into our models

Outline

- Markov Models

  (= a particular Bayes net)

Outline

- Markov Models

  (= a particular Bayes net)

- Hidden Markov Models (HMMs)
  - Representation

    (= another particular Bayes net)

Outline

- Markov Models

  (= a particular Bayes net)
- Hidden Markov Models (HMMs)
  - Representation

    (= another particular Bayes net)
  - Inference
    - Forward algorithm (= variable elimination)
    - Particle filtering (= likelihood weighting with some tweaks)
    - Viterbi (= variable elimination, but replace sum by max

      = graph search)

Outline

- Markov Models

  (= a particular Bayes net)
- Hidden Markov Models (HMMs)
  - Representation

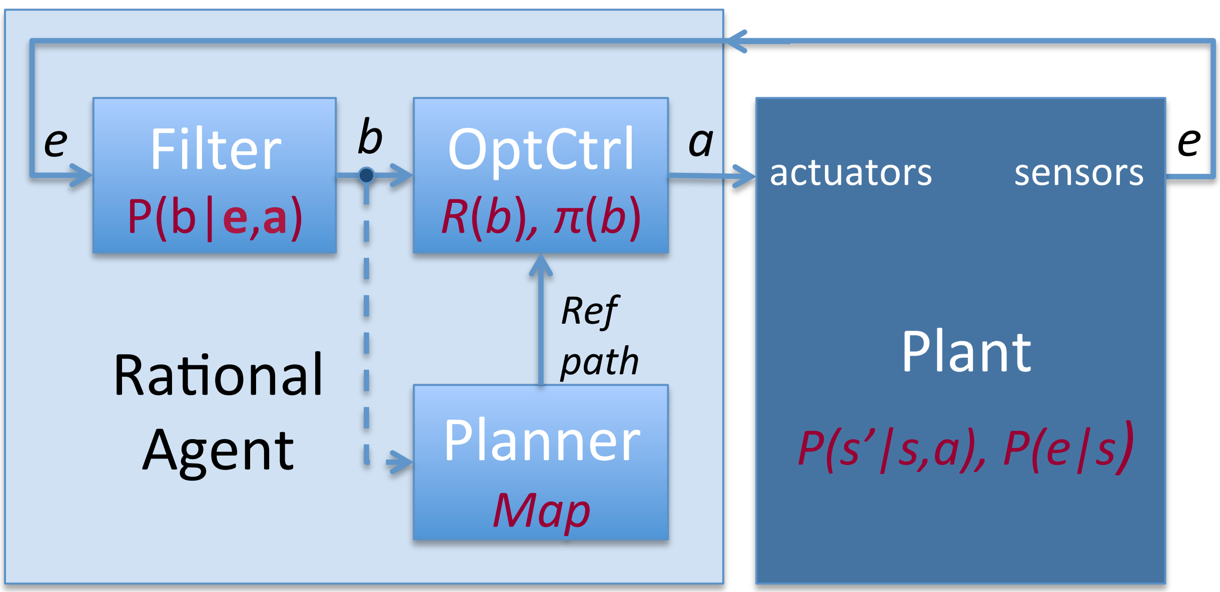
    (= another particular Bayes net)
  - Inference
    - Forward algorithm (= variable elimination)
    - Particle filtering (= likelihood weighting with some tweaks)
    - Viterbi (= variable elimination, but replace sum by max

      = graph search)
- Dynamic Bayes' Nets
  - Representation
    - (= yet another particular Bayes' net)

Outline

- Markov Models
  (= a particular Bayes net)
- Hidden Markov Models (HMMs)
  - Representation
    (= another particular Bayes net)
  - Inference
    - Forward algorithm (= variable elimination)
    - Particle filtering (= likelihood weighting with some tweaks)
    - Viterbi (= variable elimination, but replace sum by max
      = graph search)
- Dynamic Bayes' Nets
  - Representation
    - (= yet another particular Bayes' net)
  - Inference: forward algorithm and particle filtering

Markov Models

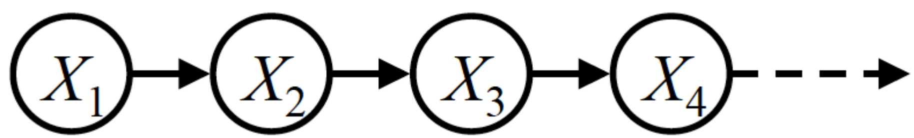


Markov Models

- A Markov model is a chain-structured BN
  - Each node is identically distributed (stationarity)
  - Value of X at a given time is called the state

Markov Models

- A Markov model is a chain-structured BN
  - Each node is identically distributed (stationarity)
  - Value of X at a given time is called the state
  - As a BN:

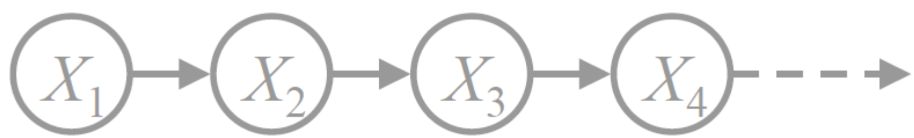


$$P(X_1) \qquad P(X_t|X_{t-1})$$

- Parameters: called transition probabilities or dynamics, specify how the state evolves over time (also, initial state probabilities)

Markov Models

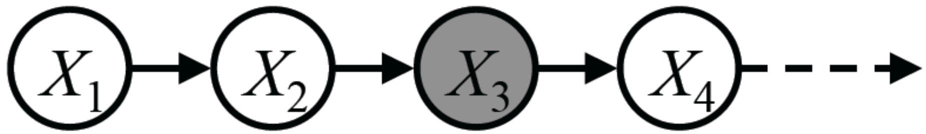
- A Markov model is a chain-structured BN
  - Each node is identically distributed (stationarity)
  - Value of X at a given time is called the state
  - As a BN:



$$P(X_1) \qquad P(X_t|X_{t-1})$$

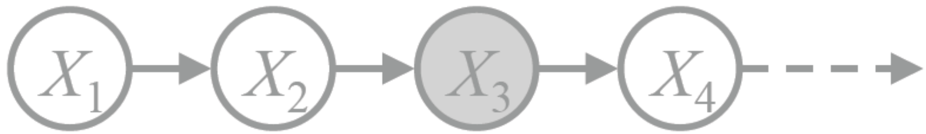
  - Parameters: called transition probabilities or dynamics, specify how the state evolves over time (also, initial state probabilities)
  - Same as MDP transition model, but no choice of action

Conditional Independence



- Basic conditional independence:
  - Past and future independent of the present
  - Each time step only depends on the previous
  - This is called the (first order) Markov property

Conditional Independence

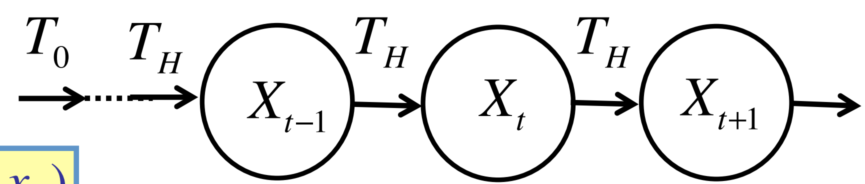


- Basic conditional independence:
  - Past and future independent of the present
  - Each time step only depends on the previous
  - This is called the (first order) Markov property
- Note that the chain is just a (growing) BN
  - We can always use generic BN reasoning on it if we truncate the chain at a fixed length

Markov Chain

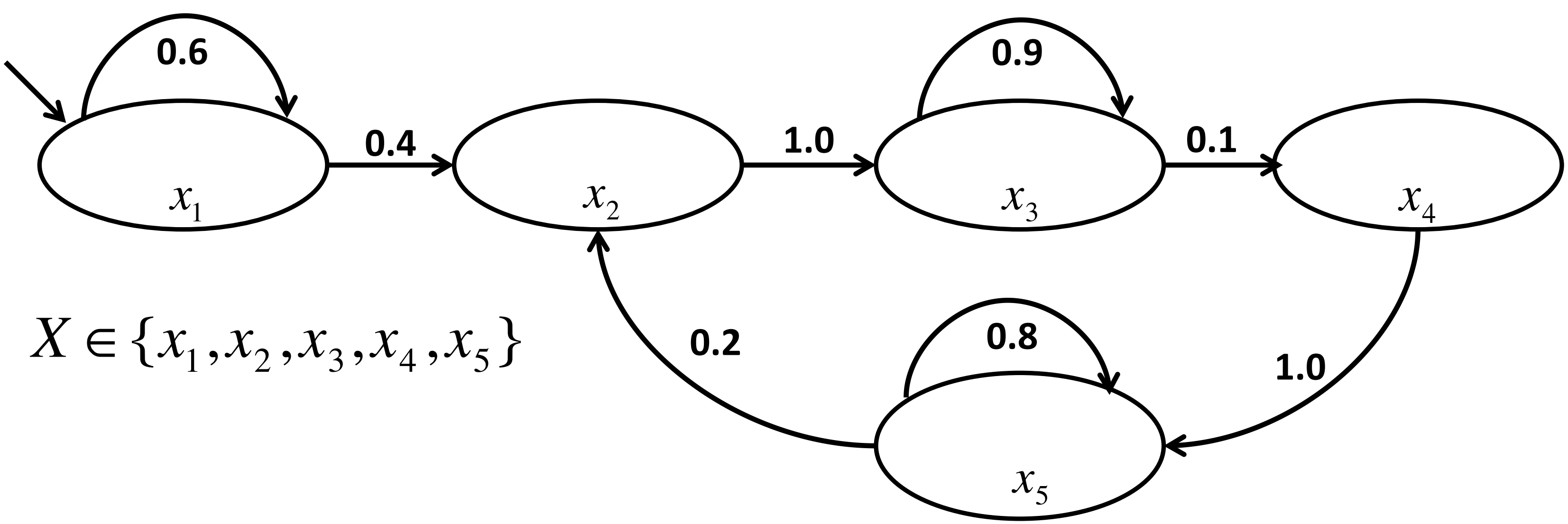
T_0	
x_1	1
x_2	
x_3	$P(x_3)$
x_4	
x_5	

T_H	x_1	x_2	x_3	x_4	x_5
x_1	**0.6**	**0.4**			
x_2			**1**		
x_3			**0.9**	**0.1**	$P(x_5 \mid x_3)$
x_4					**1**
x_5		**0.2**			**0.8**



$$\boldsymbol{P}(X_{t+1} \mid X_t) = T_H$$

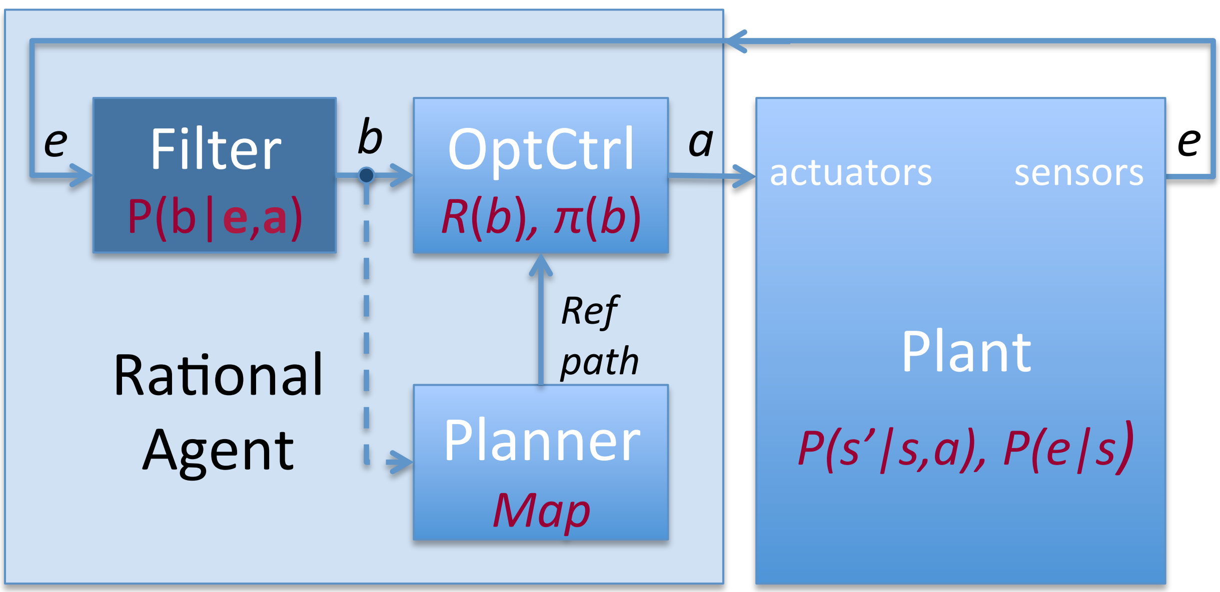
$$\boldsymbol{P}(X_0) = T_0$$

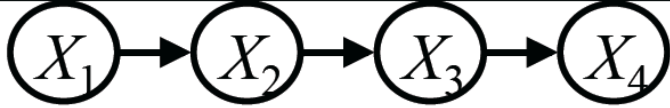


$X \in \{x_1, x_2, x_3, x_4, x_5\}$

Inference

Exact Algorithm

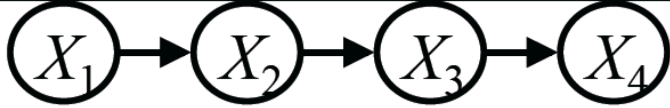




Query: P(X_4)

- Slow answer: inference by enumeration
  - Enumerate all sequences of length t which end in s
  - Add up their probabilities

$$P(X_4) = \sum_{x_1,x_2,x_3} P(x_1, x_2, x_3, X_4)$$



Query: P(X_4)

- Slow answer: inference by enumeration
  - Enumerate all sequences of length t which end in s
  - Add up their probabilities

$$P(X_4) = \sum_{x_1,x_2,x_3} P(x_1, x_2, x_3, X_4)$$

$$= P(X_1 = +x_1)P(X_2 = +x_2|X_1 = +x_1)P(X_3 = +x_3|X_2 = +x_2)P(X_4|X_3 = +x_3)$$

$X_1 \to X_2 \to X_3 \to X_4$

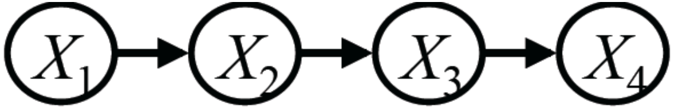
Query: P(X_4)

- Slow answer: inference by enumeration
  - Enumerate all sequences of length t which end in s
  - Add up their probabilities

$$P(X_4) = \sum_{x_1, x_2, x_3} P(x_1, x_2, x_3, X_4)$$

$$\begin{aligned}
&= P(X_1 = +x_1)P(X_2 = +x_2|X_1 = +x_1)P(X_3 = +x_3|X_2 = +x_2)P(X_4|X_3 = +x_3) \\
&+P(X_1 = +x_1)P(X_2 = +x_2|X_1 = +x_1)P(X_3 = -x_3|X_2 = +x_2)P(X_4|X_3 = -x_3) \\
&+P(X_1 = +x_1)P(X_2 = -x_2|X_1 = +x_1)P(X_3 = +x_3|X_2 = -x_2)P(X_4|X_3 = +x_3) \\
&+P(X_1 = +x_1)P(X_2 = -x_2|X_1 = +x_1)P(X_3 = -x_3|X_2 = -x_2)P(X_4|X_3 = -x_3) \\
&+P(X_1 = -x_1)P(X_2 = +x_2|X_1 = -x_1)P(X_3 = +x_3|X_2 = +x_2)P(X_4|X_3 = +x_3) \\
&+P(X_1 = -x_1)P(X_2 = +x_2|X_1 = -x_1)P(X_3 = -x_3|X_2 = +x_2)P(X_4|X_3 = -x_3) \\
&+P(X_1 = -x_1)P(X_2 = -x_2|X_1 = -x_1)P(X_3 = -x_3|X_2 = -x_2)P(X_4|X_3 = -x_3) \\
&+P(X_1 = -x_1)P(X_2 = -x_2|X_1 = -x_1)P(X_3 = -x_3|X_2 = -x_2)P(X_4|X_3 = -x_3)
\end{aligned}$$

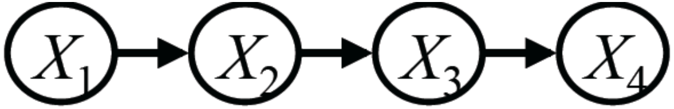
- = join on X_1, X_2, X_3, then sum over x_1, x_2, x_3



Query: $P(X_4)$

- Fast answer: variable elimination
  - Order: X_1, X_2, X_3

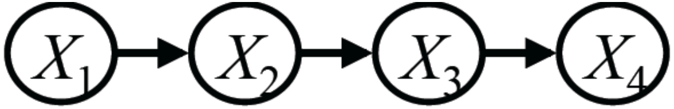
$$P(X_4) = \sum_{x_1,x_2,x_3} P(x_1, x_2, x_3, X_4)$$



Query: $P(X_4)$

- Fast answer: variable elimination
  - Order: X_1, X_2, X_3

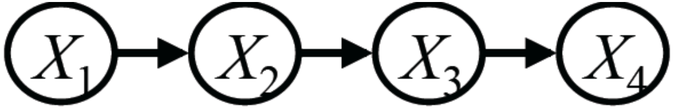
$$
\begin{aligned}
P(X_4) &= \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(X_4|x_3)P(x_3|x_2)P(x_2|x_1)P(x_1)
\end{aligned}
$$



Query: $P(X_4)$

- Fast answer: variable elimination
  - Order: X_1, X_2, X_3

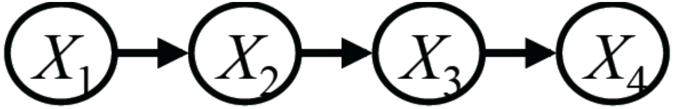
$$
\begin{aligned}
P(X_4) &= \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(X_4|x_3)P(x_3|x_2)P(x_2|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2} P(X_4|x_3)P(x_3|x_2)\sum_{x_1} P(x_2|x_1)P(x_1)
\end{aligned}
$$



Query: $P(X_4)$

- Fast answer: variable elimination
  - Order: X_1, X_2, X_3

$$
\begin{aligned}
P(X_4) &= \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(X_4|x_3)P(x_3|x_2)P(x_2|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2} P(X_4|x_3)P(x_3|x_2)\sum_{x_1} P(x_2|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2} P(X_4|x_3)P(x_3|x_2)P(x_2)
\end{aligned}
$$



Query: P(X$_4$)

- Fast answer: variable elimination
  - Order: X_1, X_2, X_3

$$
\begin{aligned}
P(X_4) &= \sum_{x_1,x_2,x_3} P(x_1, x_2, x_3, X_4) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(X_4|x_3)P(x_3|x_2)P(x_2|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2} P(X_4|x_3)P(x_3|x_2)\sum_{x_1} P(x_2|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2} P(X_4|x_3)P(x_3|x_2)P(x_2) \\
&= \sum_{x_3} P(X_4|x_3)\sum_{x_2} P(x_3|x_2)P(x_2)
\end{aligned}
$$

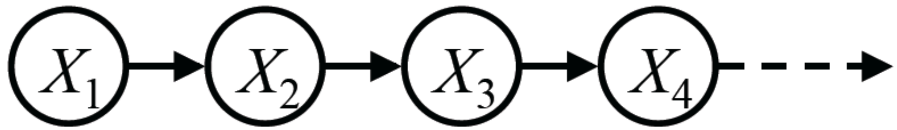
$$X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow X_4$$

Query: $P(X_4)$

- Fast answer: variable elimination
  - Order: X_1, X_2, X_3

$$\begin{aligned}
P(X_4) &= \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(X_4|x_3)P(x_3|x_2)P(x_2|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2} P(X_4|x_3)P(x_3|x_2)\sum_{x_1} P(x_2|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2} P(X_4|x_3)P(x_3|x_2)P(x_2) \\
&= \sum_{x_3} P(X_4|x_3)\sum_{x_2} P(x_3|x_2)P(x_2) \\
&= \sum_{x_3} P(X_4|x_3)P(x_3) \\
&= P(X_4)
\end{aligned}$$

Query P(X_t)



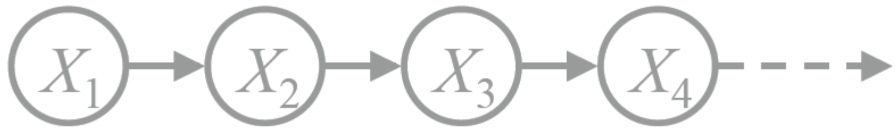
$$P(X_1) \qquad P(X_t|X_{t-1})$$

- Variable elimination in order X_1, X_2, …, X_{t-1} computes for k = 2, 3, …, t

$$P(x_k) = \sum_{x_{k-1}} P(x_k|x_{k-1}) P(x_{k-1})$$

Forward simulation

Query P(X_t)



$$P(X_1) \qquad P(X_t|X_{t-1})$$

- Variable elimination in order X_1, X_2, …, X_{t-1} computes for k = 2, 3, …, t

$$P(x_k) = \sum_{x_{k-1}} P(x_k|x_{k-1})P(x_{k-1})$$

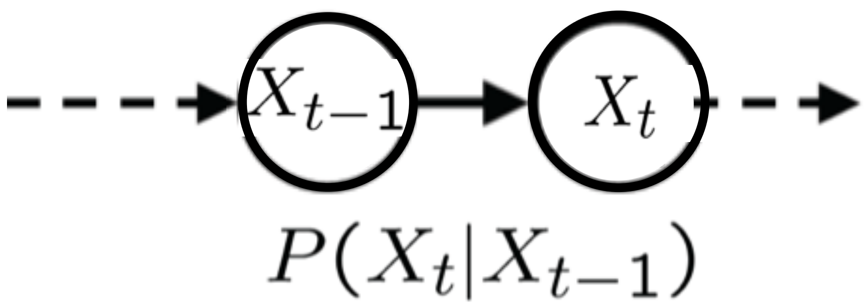
Forward simulation

= “mini-forward algorithm”

Note: common thread in this lecture: special cases of algorithms we already know, and they have a special name in the context of HMMs for historical reasons.

Example Markov Chain: Weather

- States: X = {rain, sun}
- CPT $P(X_t \mid X_{t-1})$:

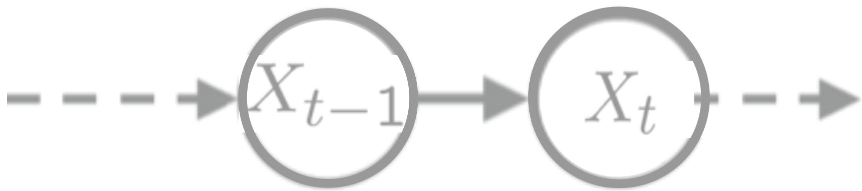


X_{t-1}	X_t	$P(X_t \mid X_{t-1})$
sun	sun	0.9
sun	rain	0.1
rain	sun	0.3
rain	rain	0.7

Example Markov Chain: Weather

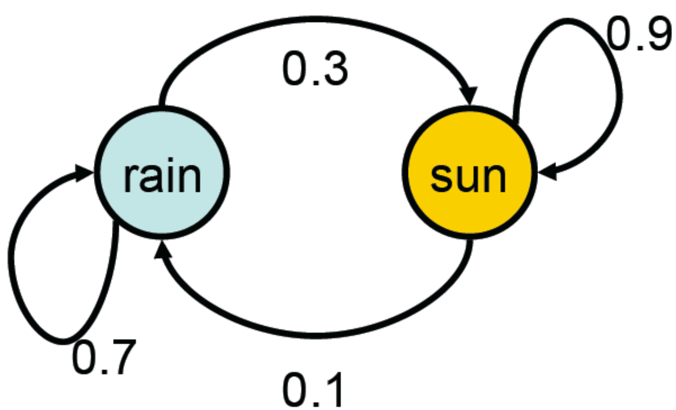
- States: X = {rain, sun}
- CPT $P(X_t \mid X_{t-1})$:

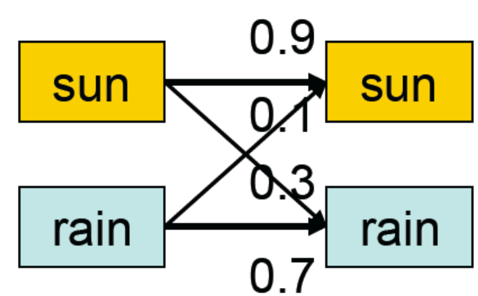
X_{t-1}	X_t	$P(X_t \mid X_{t-1})$
sun	sun	0.9
sun	rain	0.1
rain	sun	0.3
rain	rain	0.7



$$P(X_t|X_{t-1})$$

Two new ways of representing the same CPT, that are often used for Markov models (These are not BNs!)



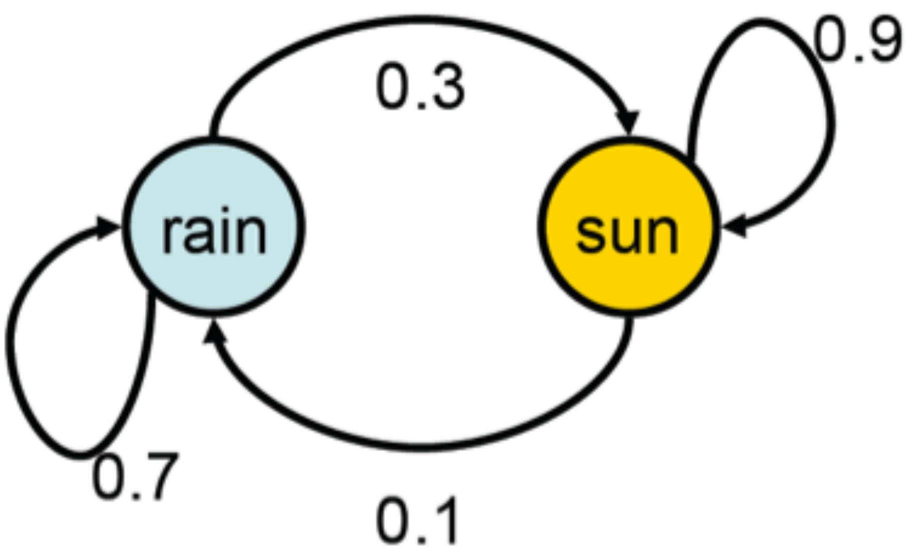


Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\left\langle \begin{matrix} 1.0 \\ 0.0 \end{matrix} \right\rangle$$

$P(X_1)$



$$P(X_t) = P(X_{t-1})\, P(X_t \mid X_{t-1})$$

$$\begin{array}{cc} \text{sun} & \text{rain} \\ [\,1.0 & 0.0\,] \end{array} \quad \begin{array}{cc} \text{sun} & \text{rain} \\ \end{array}$$

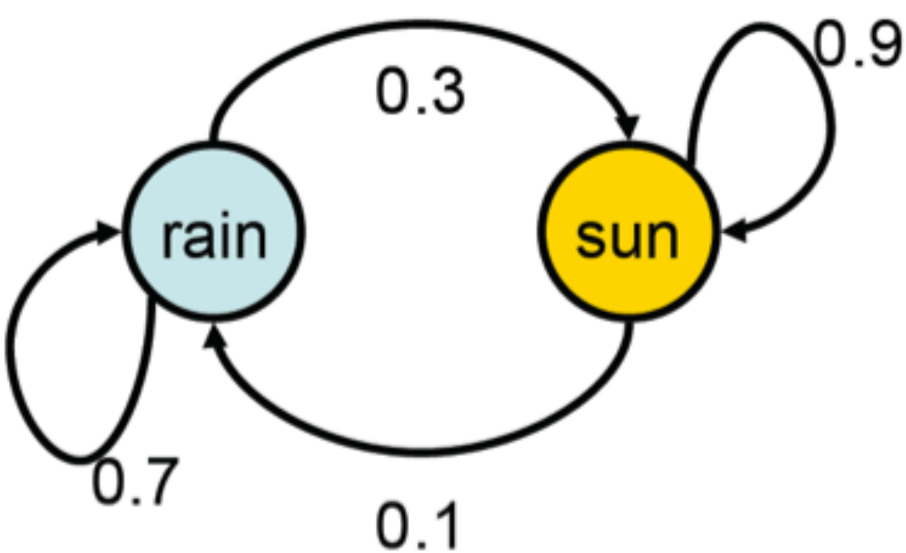
$$\begin{array}{c|c} \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix} & \begin{array}{c} \text{sun} \\ \text{rain} \end{array} \end{array}$$

Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\left\langle \begin{array}{c} 1.0 \\ 0.0 \end{array} \right\rangle \quad \left\langle \begin{array}{c} 0.9 \\ 0.1 \end{array} \right\rangle$$

$$P(X_1) \qquad P(X_2)$$



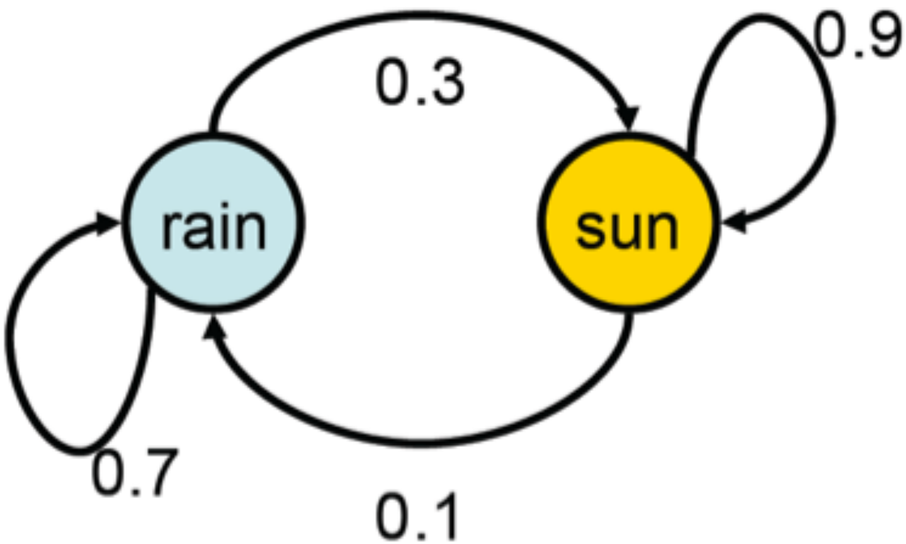
$$P(X_t) = P(X_{t-1})\, P(X_t \mid X_{t-1})$$

$$\begin{array}{cc} \text{sun} & \text{rain} \\ \left[\, 1.0 \right. & \left. 0.0 \,\right] \end{array} \quad \begin{array}{c} \begin{array}{cc} \text{sun} & \text{rain} \end{array} \\ \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix} \end{array} \begin{array}{l} \text{sun} \\ \text{rain} \end{array}$$

Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\underbrace{\left\langle \begin{matrix} 1.0 \\ 0.0 \end{matrix} \right\rangle}_{P(X_1)} \qquad \underbrace{\left\langle \begin{matrix} 0.9 \\ 0.1 \end{matrix} \right\rangle}_{P(X_2)} \qquad \underbrace{\left\langle \begin{matrix} 0.84 \\ 0.16 \end{matrix} \right\rangle}_{P(X_3)}$$



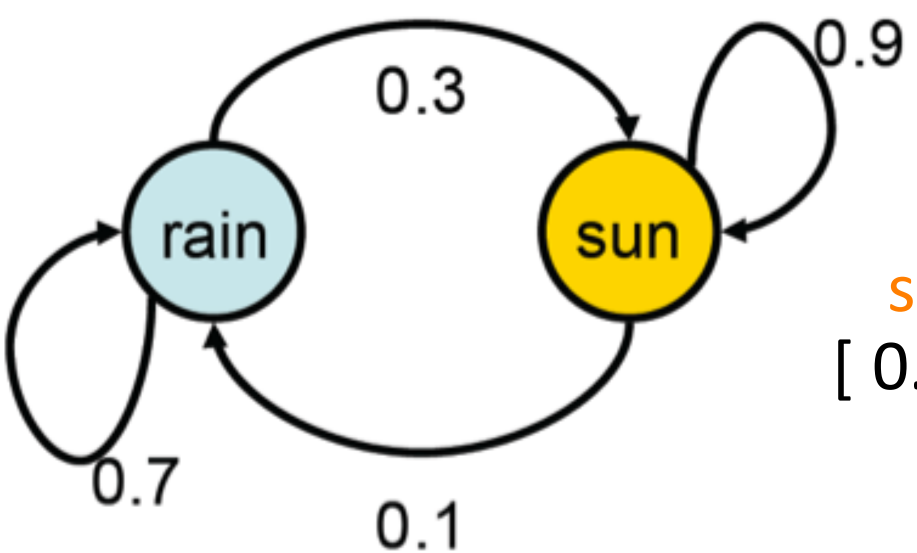
$$P(X_t) = P(X_{t-1})\, P(X_t \mid X_{t-1})$$

$$\begin{array}{cc} \text{sun} & \text{rain} \\ \multicolumn{2}{c}{[\ 0.9 \quad 0.1\ ]} \end{array} \quad \begin{array}{c} \begin{array}{cc} \text{sun} & \text{rain} \end{array} \\ \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix} \end{array} \begin{array}{l} \\ \text{sun} \\ \text{rain} \end{array}$$

Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\underbrace{\left\langle \begin{matrix} 1.0 \\ 0.0 \end{matrix} \right\rangle}_{P(X_1)} \quad \underbrace{\left\langle \begin{matrix} 0.9 \\ 0.1 \end{matrix} \right\rangle}_{P(X_2)} \quad \underbrace{\left\langle \begin{matrix} 0.84 \\ 0.16 \end{matrix} \right\rangle}_{P(X_3)} \quad \underbrace{\left\langle \begin{matrix} 0.804 \\ 0.196 \end{matrix} \right\rangle}_{P(X_4)}$$



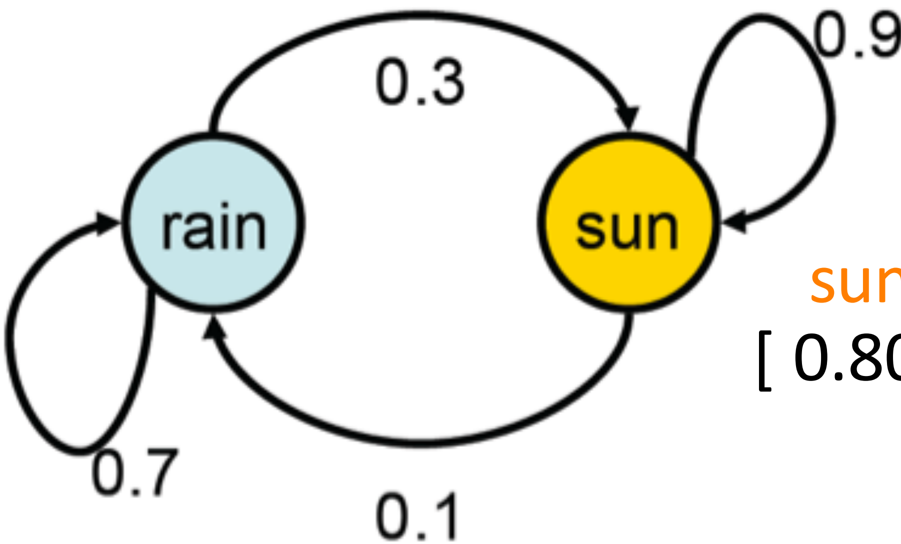
$$P(X_t) = P(X_{t-1})\, P(X_t \mid X_{t-1})$$

$$\begin{array}{cc} \text{sun} & \text{rain} \\ \left[\, 0.84 \right. & \left. 0.16 \,\right] \end{array} \quad \begin{array}{c} \begin{array}{cc} \text{sun} & \text{rain} \end{array} \\ \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix} \end{array} \begin{array}{l} \\ \text{sun} \\ \text{rain} \end{array}$$

Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\underbrace{\left\langle \begin{matrix} 1.0 \\ 0.0 \end{matrix} \right\rangle}_{P(X_1)} \quad \underbrace{\left\langle \begin{matrix} 0.9 \\ 0.1 \end{matrix} \right\rangle}_{P(X_2)} \quad \underbrace{\left\langle \begin{matrix} 0.84 \\ 0.16 \end{matrix} \right\rangle}_{P(X_3)} \quad \underbrace{\left\langle \begin{matrix} 0.804 \\ 0.196 \end{matrix} \right\rangle}_{P(X_4)} \longrightarrow \underbrace{\left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle}_{P(X_\infty)}$$



$$P(X_t) = P(X_{t-1})\, P(X_t \mid X_{t-1})$$

$$\begin{array}{cc} \text{sun} & \text{rain} \\ [\,0.804 & 0.196\,] \end{array} \quad \begin{array}{c} \begin{array}{cc} \text{sun} & \text{rain} \end{array} \\ \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix} \end{array} \begin{array}{l} \\ \text{sun} \\ \text{rain} \end{array}$$

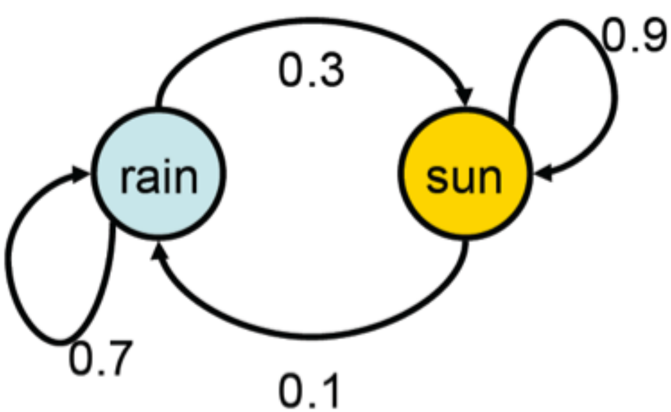
Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\begin{array}{ccccc} \left\langle \begin{array}{c} 1.0 \\ 0.0 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.9 \\ 0.1 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.84 \\ 0.16 \end{array} \right\rangle & \left\langle \begin{array}{c} 0.804 \\ 0.196 \end{array} \right\rangle & \Rightarrow \left\langle \begin{array}{c} 0.75 \\ 0.25 \end{array} \right\rangle \\ P(X_1) & P(X_2) & P(X_3) & P(X_4) & P(X_\infty) \end{array}$$

- From initial observation of rain

$$\begin{array}{c} \left\langle \begin{array}{c} 0.0 \\ 1.0 \end{array} \right\rangle \\ P(X_1) \end{array}$$



$$\begin{array}{cc} \text{sun} & \text{rain} \\ [\ 0.0 & 1.0\ ] \end{array} \quad \begin{array}{c} \begin{array}{cc} \text{sun} & \text{rain} \end{array} \\ \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix} \end{array} \begin{array}{c} \\ \text{sun} \\ \text{rain} \end{array}$$

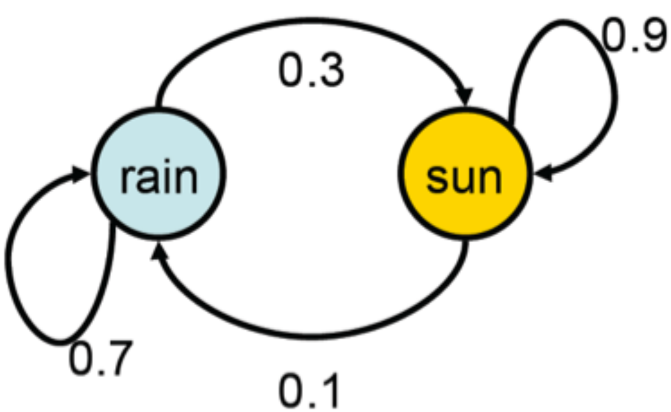
Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\underset{P(X_1)}{\left\langle \begin{matrix} 1.0 \\ 0.0 \end{matrix} \right\rangle} \quad \underset{P(X_2)}{\left\langle \begin{matrix} 0.9 \\ 0.1 \end{matrix} \right\rangle} \quad \underset{P(X_3)}{\left\langle \begin{matrix} 0.84 \\ 0.16 \end{matrix} \right\rangle} \quad \underset{P(X_4)}{\left\langle \begin{matrix} 0.804 \\ 0.196 \end{matrix} \right\rangle} \Rightarrow \underset{P(X_\infty)}{\left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle}$$

- From initial observation of rain

$$\underset{P(X_1)}{\left\langle \begin{matrix} 0.0 \\ 1.0 \end{matrix} \right\rangle} \quad \underset{P(X_2)}{\left\langle \begin{matrix} 0.3 \\ 0.7 \end{matrix} \right\rangle}$$



$$\begin{array}{cc} \text{sun} & \text{rain} \\ [\,0.0 & 1.0\,] \end{array} \quad \begin{array}{c} \begin{array}{cc} \text{sun} & \text{rain} \end{array} \\ \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix} \end{array} \begin{array}{c} \text{sun} \\ \text{rain} \end{array}$$

Example Run of Mini-Forward Algorithm

- From initial observation of sun

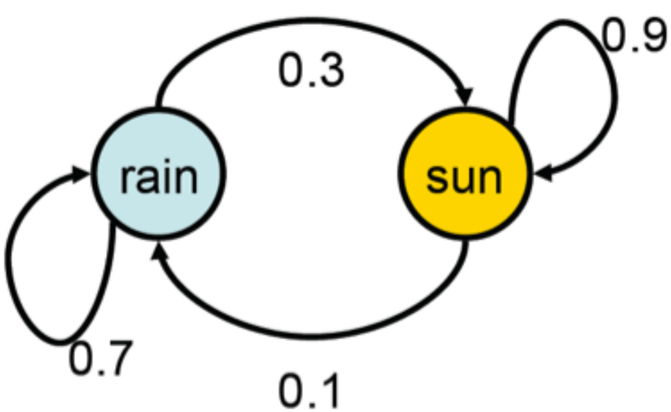
$$\left\langle \begin{matrix} 1.0 \\ 0.0 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.9 \\ 0.1 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.84 \\ 0.16 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.804 \\ 0.196 \end{matrix} \right\rangle \longrightarrow \left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle$$

$$\mathrm{P}(X_1) \quad \mathrm{P}(X_2) \quad \mathrm{P}(X_3) \quad \mathrm{P}(X_4) \qquad \mathrm{P}(X_\infty)$$

- From initial observation of rain

$$\left\langle \begin{matrix} 0.0 \\ 1.0 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.3 \\ 0.7 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.48 \\ 0.52 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.588 \\ 0.412 \end{matrix} \right\rangle \longrightarrow \left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle$$

$$\mathrm{P}(X_1) \quad \mathrm{P}(X_2) \quad \mathrm{P}(X_3) \quad \mathrm{P}(X_4) \qquad \mathrm{P}(X_\infty)$$



$$\begin{array}{cc} \text{sun} & \text{rain} \\ \multicolumn{2}{c}{[\; 0.588 \quad 0.412 \;]} \end{array} \qquad \begin{array}{cc} \text{sun} & \text{rain} \\ \end{array}$$

$$\begin{array}{c|c} & \begin{array}{cc} \text{sun} & \text{rain} \end{array} \\ \begin{array}{c} \text{sun} \\ \text{rain} \end{array} & \begin{bmatrix} 0.9 & 0.1 \\ 0.3 & 0.7 \end{bmatrix} \end{array}$$

Example Run of Mini-Forward Algorithm

- From initial observation of sun

$$\left\langle \begin{matrix} 1.0 \\ 0.0 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.9 \\ 0.1 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.84 \\ 0.16 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.804 \\ 0.196 \end{matrix} \right\rangle \Rightarrow \left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle$$

$$\mathrm{P}(X_1) \quad \mathrm{P}(X_2) \quad \mathrm{P}(X_3) \quad \mathrm{P}(X_4) \quad \mathrm{P}(X_\infty)$$

- From initial observation of rain

$$\left\langle \begin{matrix} 0.0 \\ 1.0 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.3 \\ 0.7 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.48 \\ 0.52 \end{matrix} \right\rangle \quad \left\langle \begin{matrix} 0.588 \\ 0.412 \end{matrix} \right\rangle \Rightarrow \left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle$$

$$\mathrm{P}(X_1) \quad \mathrm{P}(X_2) \quad \mathrm{P}(X_3) \quad \mathrm{P}(X_4) \quad \mathrm{P}(X_\infty)$$

- From yet another initial distribution $\mathrm{P}(X_1)$:

$$\left\langle \begin{matrix} p \\ 1-p \end{matrix} \right\rangle \quad \cdots \quad \Rightarrow \left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle$$

$$\mathrm{P}(X_1) \qquad \mathrm{P}(X_\infty)$$

Stationary Distributions

- For most chains:
  - influence of initial distribution gets less and less over time.
  - the distribution we end up in is independent of the initial distribution

Stationary Distributions

- For most chains:
  - influence of initial distribution gets less and less over time.
  - the distribution we end up in is independent of the initial distribution
- Stationary distribution:
  - Distribution we end up with is called the stationary distribution P_∞ of the chain
  - It satisfies

$$P_\infty(X) = P_{\infty+1}(X) = \sum_x P_{t+1|t}(X|x)P_\infty(x)$$

Application of Markov Chain Stationary Distribution: Gibbs Sampling*

- Each joint instantiation over all hidden and query variables is a state. Let $X = H \cup Q$

- Transitions:
  - With probability 1/n resample variable X_j according to $P(X_j \mid x_1, x_2, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n, e_1, \ldots, e_m)$

Application of Markov Chain Stationary Distribution: Gibbs Sampling*

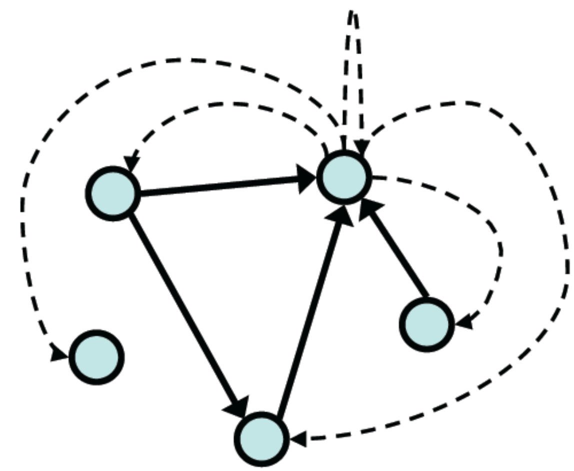
- Each joint instantiation over all hidden and query variables is a state. Let $X = H \cup Q$
- Transitions:
  - With probability 1/n resample variable X_j according to $P(X_j \mid x_1, x_2, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n, e_1, \ldots, e_m)$
- Stationary distribution:
  - = conditional distribution $P(X_1, X_2, \ldots, X_n | e_1, \ldots, e_m)$

  → When running Gibbs sampling long enough we get a sample from the desired distribution!

We did not prove this, all we did is stating this result.

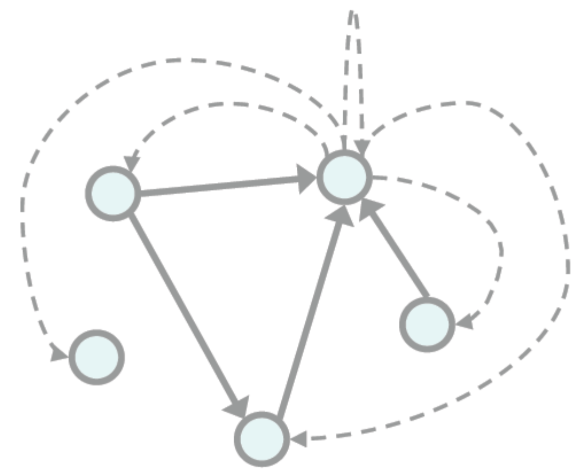
Application of Markov Chain Stationary Distribution: Web Link Analysis

- PageRank over a web graph
  - Each web page is a state
  - Initial distribution: uniform over pages
  - Transitions:
    - With prob. c, uniform jump to a random page (dotted lines, not all shown)
    - With prob. 1-c, follow a random outlink (solid lines)



Application of Markov Chain Stationary Distribution: Web Link Analysis

- PageRank over a web graph
  - Each web page is a state
  - Initial distribution: uniform over pages
  - Transitions:
    - With prob. c, uniform jump to a random page (dotted lines, not all shown)
    - With prob. 1-c, follow a random outlink (solid lines)



- Stationary distribution
  - Will spend more time on highly reachable pages
  - E.g. many ways to get to the Acrobat Reader download page
  - Somewhat robust to link spam
  - Google 1.0 returned the set of pages containing all your keywords in decreasing rank, now all search engines use link analysis along with many other factors (rank actually getting less important over time)

Outline

- Markov Models
  (= a particular Bayes net)

Outline

- Markov Models
  (= a particular Bayes net)
- Hidden Markov Models (HMMs)
  - Representation
    (= another particular Bayes net)
  - Inference
    - Forward algorithm (= variable elimination)
    - Particle filtering (= likelihood weighting with some tweaks)
    - Viterbi (= variable elimination, but replace sum by max
      = graph search)

Outline

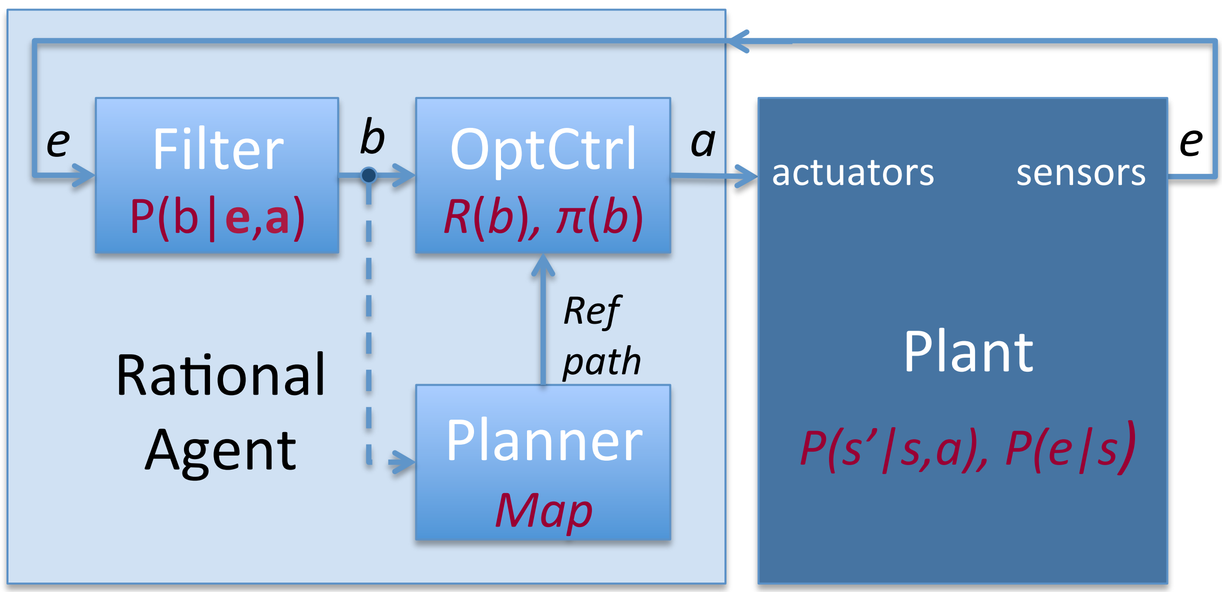
- Markov Models

  (= a particular Bayes net)
- Hidden Markov Models (HMMs)
  - Representation

    (= another particular Bayes net)
  - Inference
    - Forward algorithm (= variable elimination)
    - Particle filtering (= likelihood weighting with some tweaks)
    - Viterbi (= variable elimination, but replace sum by max

      = graph search)
- Dynamic Bayes' Nets
  - Representation
    - (= yet another particular Bayes' net)
  - Inference: forward algorithm and particle filtering

Hidden Markov Models



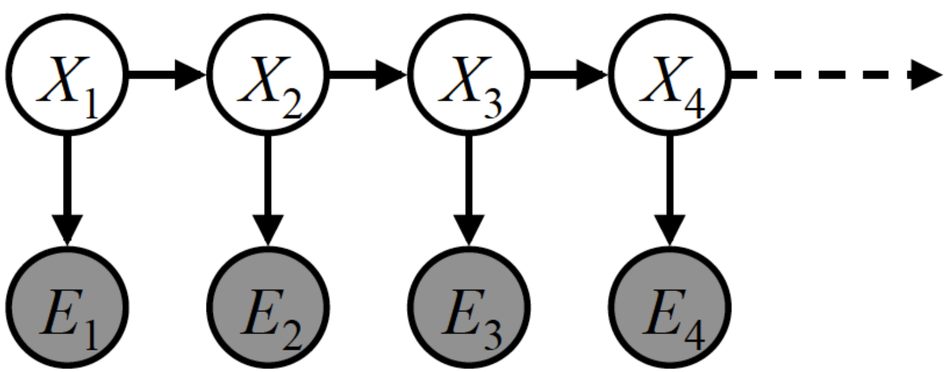
Hidden Markov Models

- Markov chains not so useful for most agents
  - Need observations to update your beliefs

Hidden Markov Models

- Markov chains not so useful for most agents
  - Need observations to update your beliefs

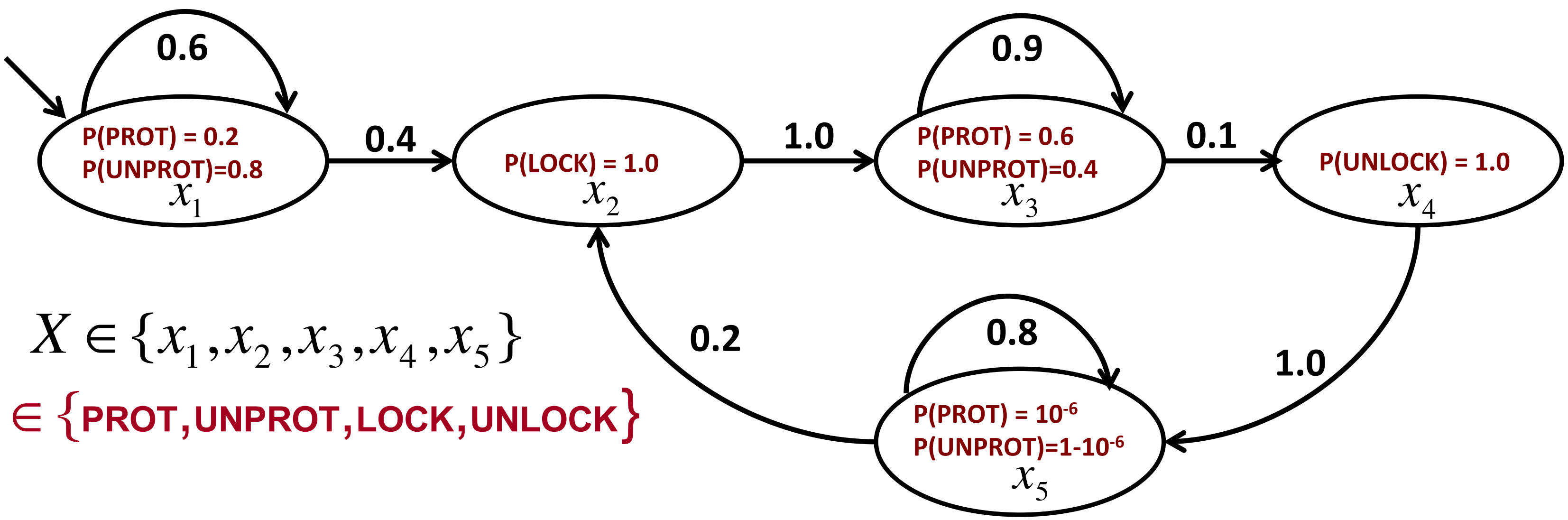
- Hidden Markov models (HMMs)
  - Underlying Markov chain over states S
  - You observe outputs (effects) at each time step
  - As a Bayes' net:



Hidden Markov Model

T_H	x_1	x_2	x_3	x_4	x_5
x_1	0.6	0.4			
x_2			1		
x_3			0.9	0.1	$P(x_5 \mid x_3)$
x_4					1
x_5		0.2			0.8

O_H	PROT	UNPROT	LOCK	UNLOCK
x_1	0.2	0.8		
x_2			1	
x_3	0.6	0.4		$P(\text{UNLOCK} \mid x_3)$
x_4				1
x_5	10^{-6}	$1-10^{-6}$		



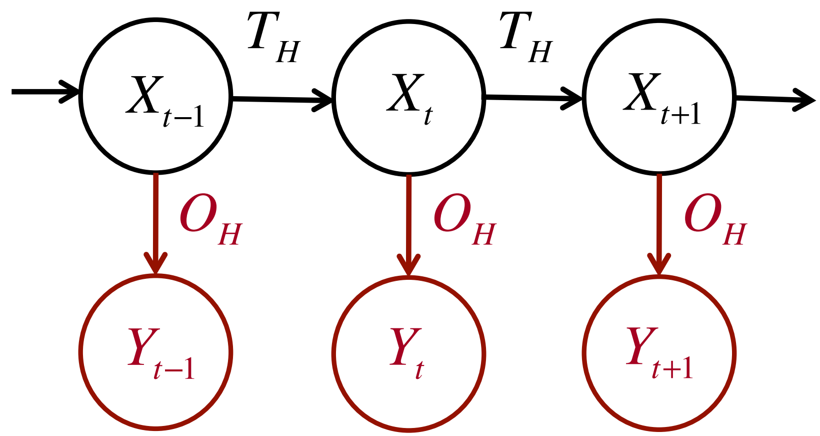
$X \in \{x_1, x_2, x_3, x_4, x_5\}$

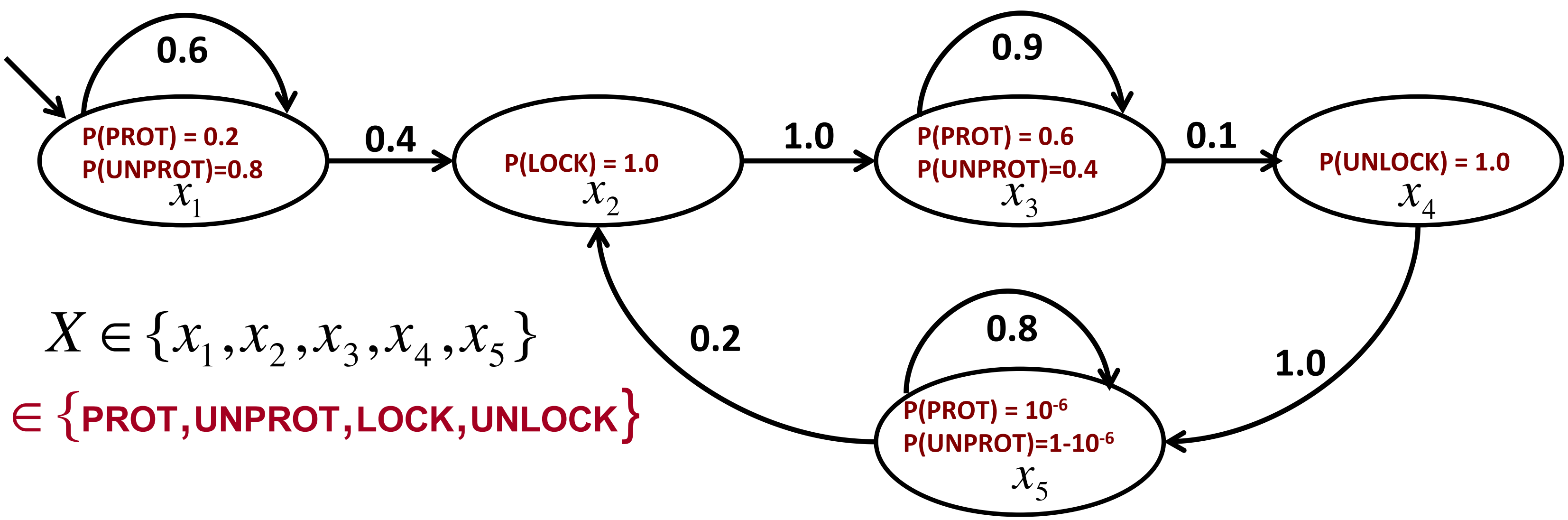
$Y \in \{\text{PROT}, \text{UNPROT}, \text{LOCK}, \text{UNLOCK}\}$

Hidden Markov Model

$$\boldsymbol{P}(X_{t+1} \mid X_t) = T_H$$

$$\boldsymbol{P}(Y_t \mid X_t) = O_H$$

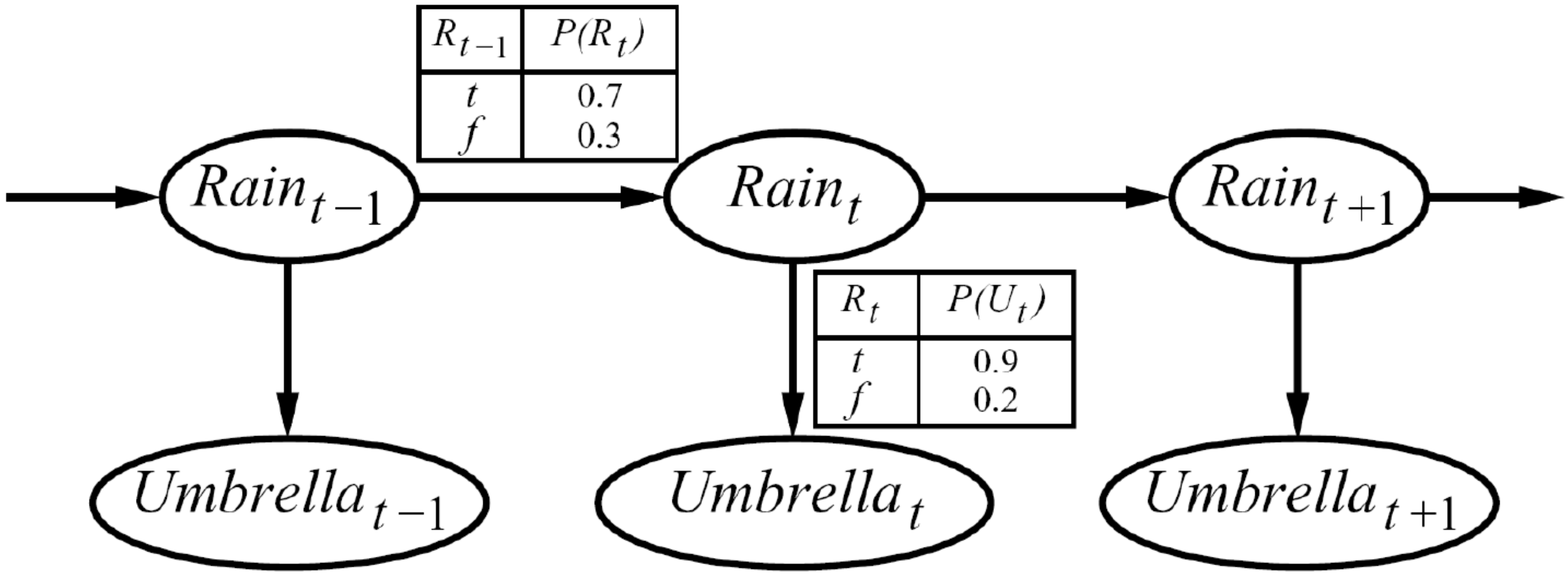




$$X \in \{x_1, x_2, x_3, x_4, x_5\}$$

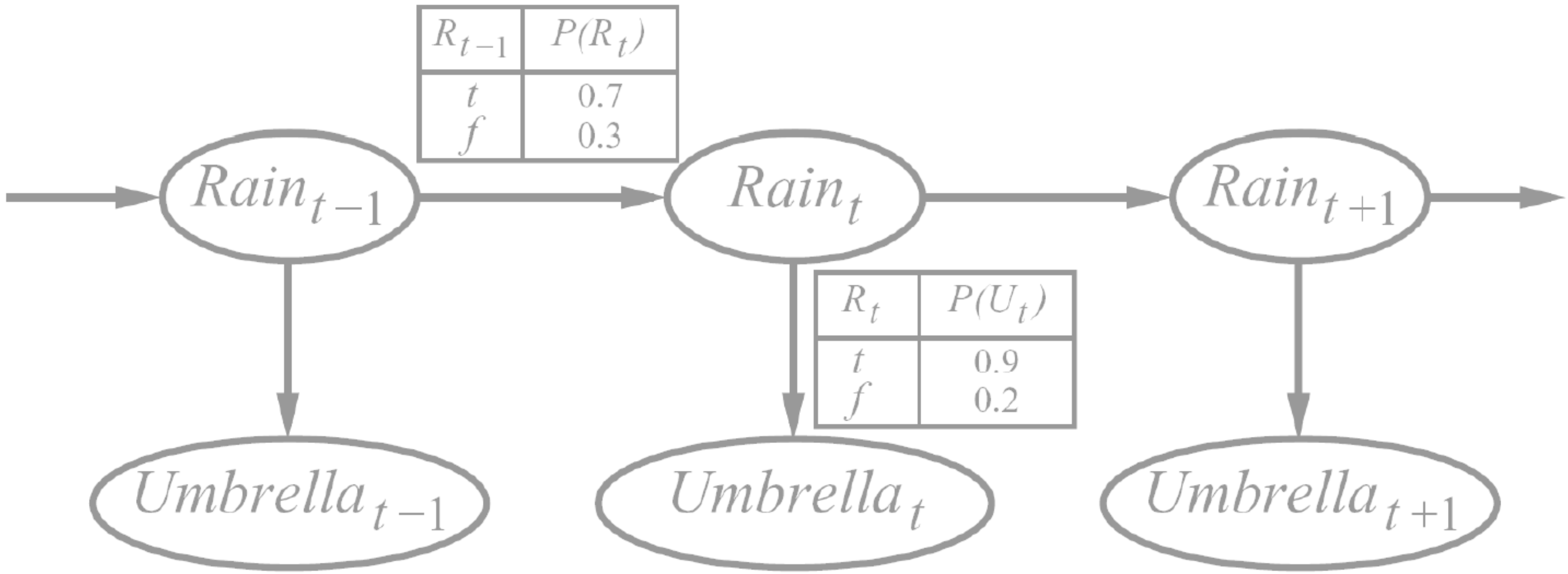
$$Y \in \{\text{PROT}, \text{UNPROT}, \text{LOCK}, \text{UNLOCK}\}$$

Example



R_{t-1} | $P(R_t)$
t 0.7
f 0.3
$Rain_{t-1}$
$Rain_t$
$Rain_{t+1}$
R_t | $P(U_t)$
t 0.9
f 0.2
$Umbrella_{t-1}$
$Umbrella_t$
$Umbrella_{t+1}$

Example



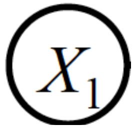
- An HMM is defined by:
  - Initial distribution: $P(X_1)$
  - Transitions: $P(X|X_{-1})$
  - Emissions: $P(E|X)$

Ghostbusters HMM

- $P(X_1)$ = uniform

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

$P(X_1)$

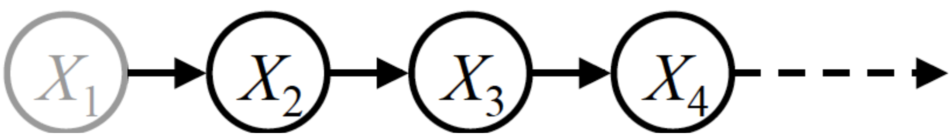


Ghostbusters HMM

- $P(X_1)$ = uniform
- $P(X|X')$ = usually move clockwise, but sometimes move in a random direction or stay in place

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

$P(X_1)$



1/6	1/6	1/2
0	1/6	0
0	0	0

$P(X|X' = <1,2>)$

Ghostbusters HMM

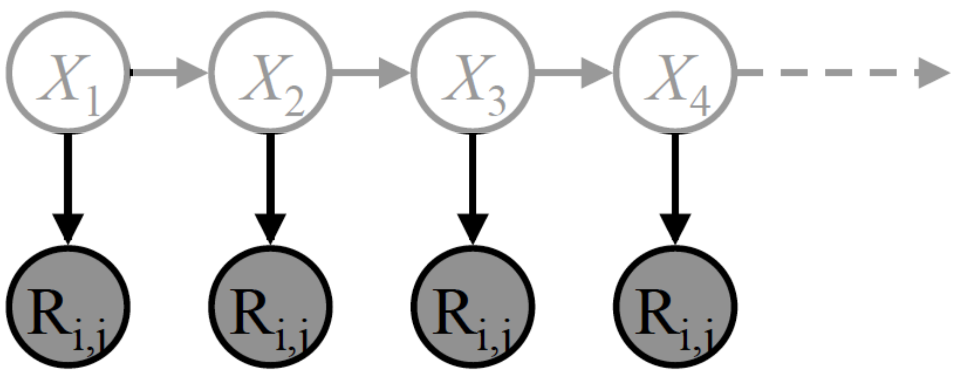
- $P(X_1)$ = uniform
- $P(X|X')$ = usually move clockwise, but sometimes move in a random direction or stay in place
- $P(R_{ij}|X)$ = sensor model:
  red means close, green means far away.

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

$P(X_1)$

1/6	1/6	1/2
0	1/6	0
0	0	0

$P(X|X' = <1,2>)$

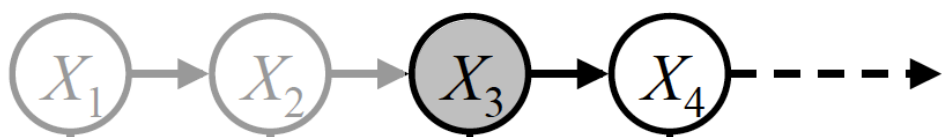


Conditional Independence

- HMMs have two important independence properties:

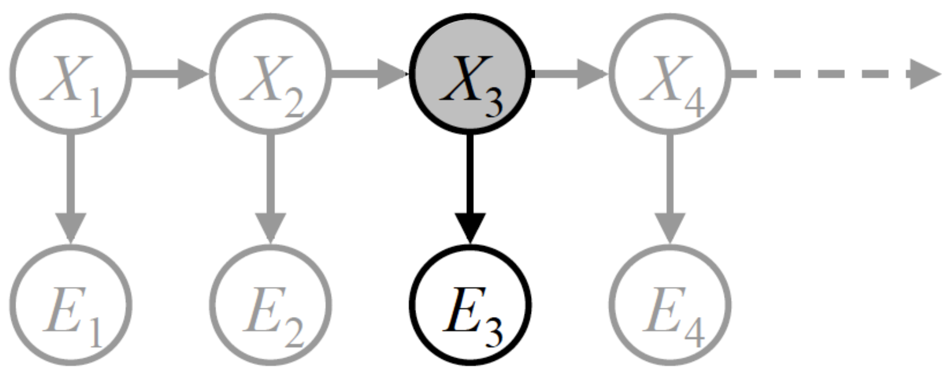
Conditional Independence

- HMMs have two important independence properties:
  - Markov hidden process, future depends on past via the present



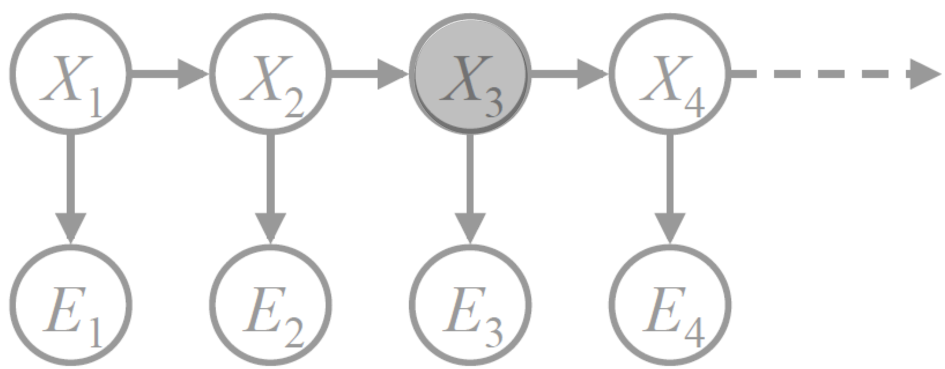
Conditional Independence

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



Conditional Independence

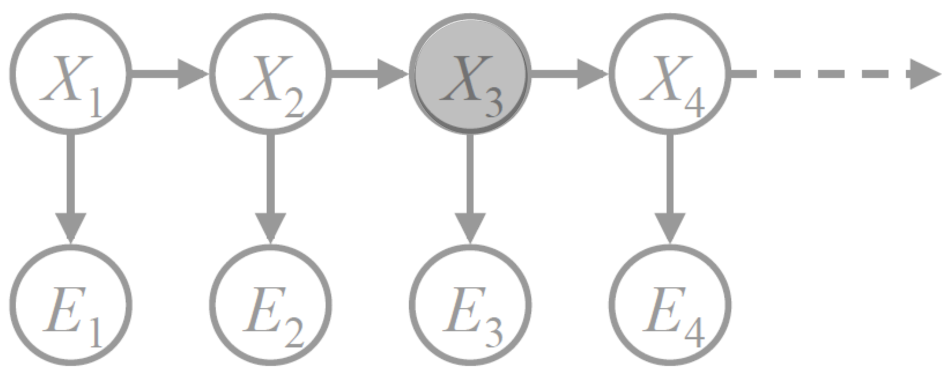
- HMMs have two important independence properties:
  - Markov hidden process, future depends on past via the present
  - Current observation independent of all else given current state



- Quiz: does this mean that evidence variables are guaranteed to be independent?

Conditional Independence

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



- Quiz: does this mean that evidence variables are guaranteed to be independent?
  - [No, they tend to 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)

Real HMM Examples

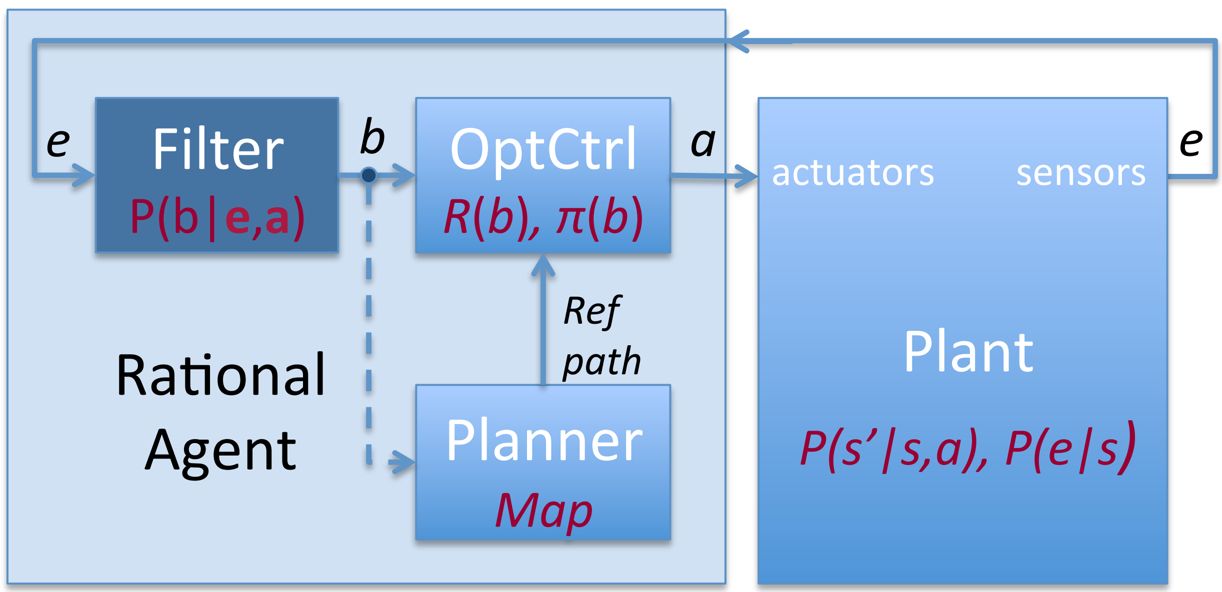
- Speech recognition HMMs:
  - Observations are acoustic signals (continuous valued)
  - States are specific positions in specific words (so, tens of thousands)

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

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)
  - States are positions on a map (continuous)

Filtering (State Estimation)
Exact Algorithm



Filtering / Monitoring

- Filtering, or monitoring, is the task of tracking the distribution $B_t(X) = P_t(X_t \mid e_1, \ldots, e_t)$ (the belief state) over time

Filtering / Monitoring

- Filtering, or monitoring, is the task of tracking the distribution $B_t(X) = P_t(X_t \mid e_1, \ldots, e_t)$ (the belief state) over time

- We start with $B_1(X)$ in an initial setting, usually uniform

Filtering / Monitoring

- Filtering, or monitoring, is the task of tracking the distribution $B_t(X) = P_t(X_t \mid e_1, \ldots, e_t)$ (the belief state) over time

- We start with $B_1(X)$ in an initial setting, usually uniform

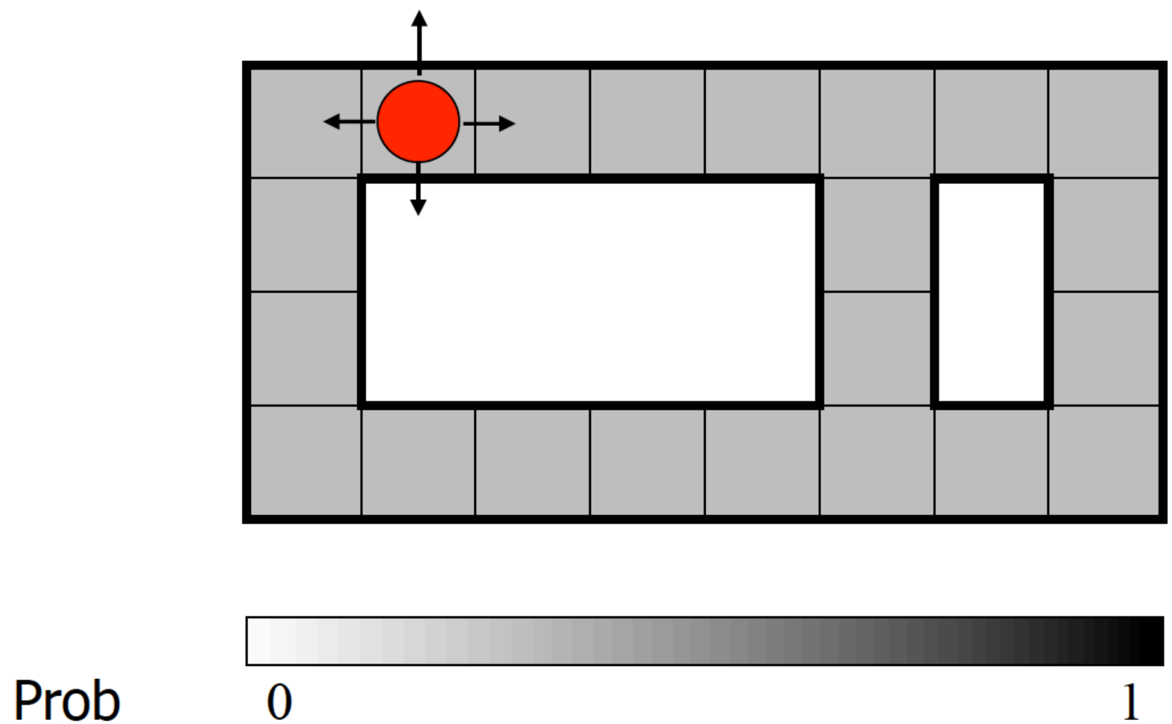
- As time passes, or we get observations, we update $B(X)$

Filtering / Monitoring

- Filtering, or monitoring, is the task of tracking the distribution $B_t(X) = P_t(X_t \mid e_1, \ldots, e_t)$ (the belief state) over time
- We start with $B_1(X)$ in an initial setting, usually uniform
- As time passes, or we get observations, we update $B(X)$
- The Kalman filter was invented in the 60's and first implemented as a method of trajectory estimation for the Apollo program

Example: Robot Localization

Example from Michael Pfeiffer

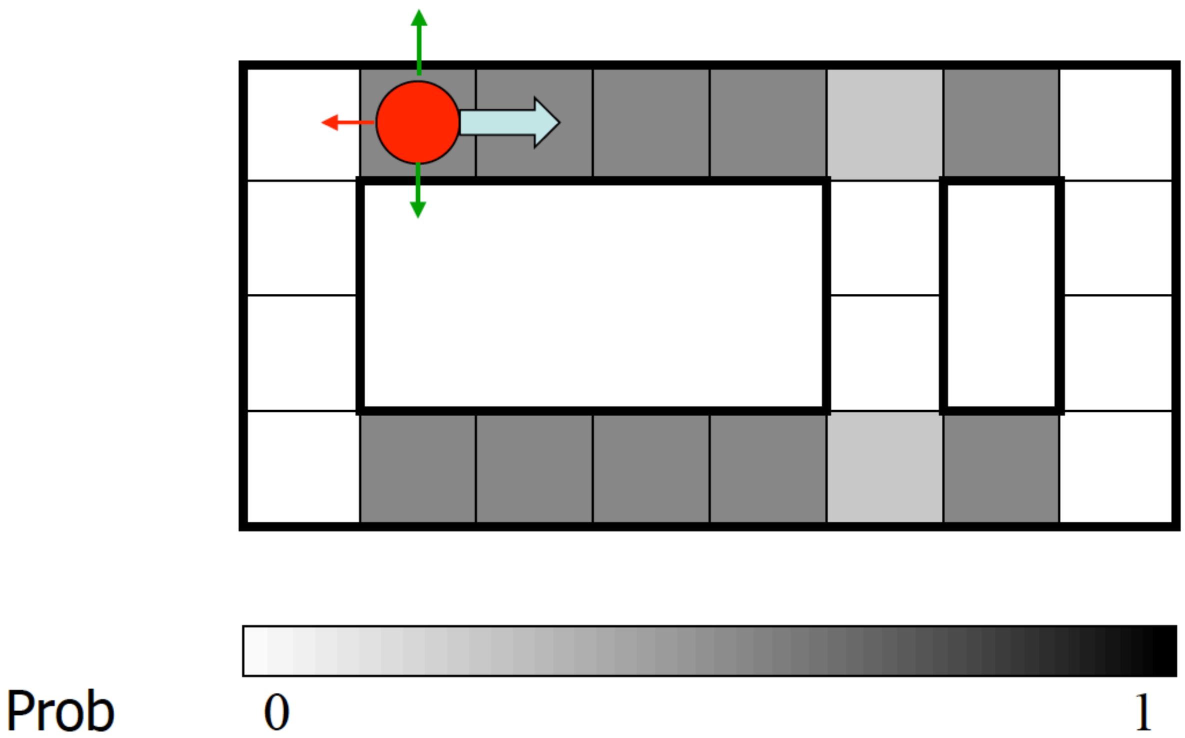


t=0

Sensor model: can read in which directions there is a wall, never more than 1 mistake

Motion model: may not execute action with small prob.

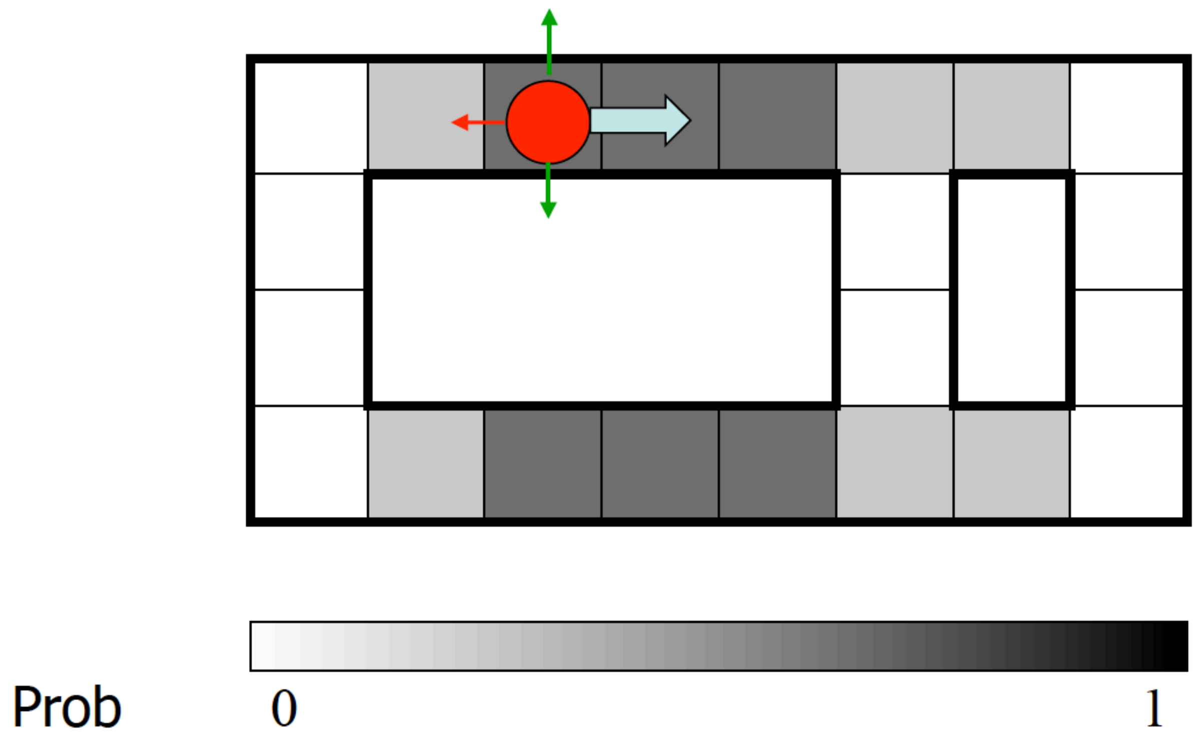
Example: Robot Localization



t=1

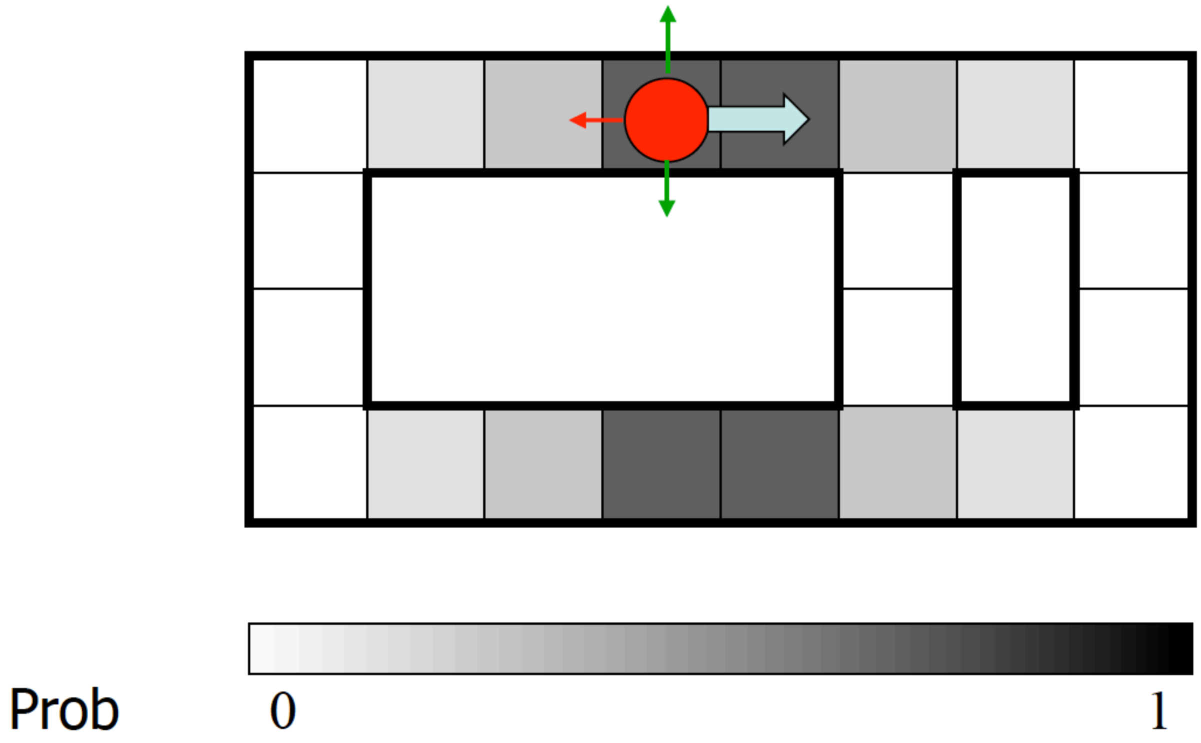
Lighter grey: was possible to get the reading, but less likely b/c required 1 mistake

Example: Robot Localization



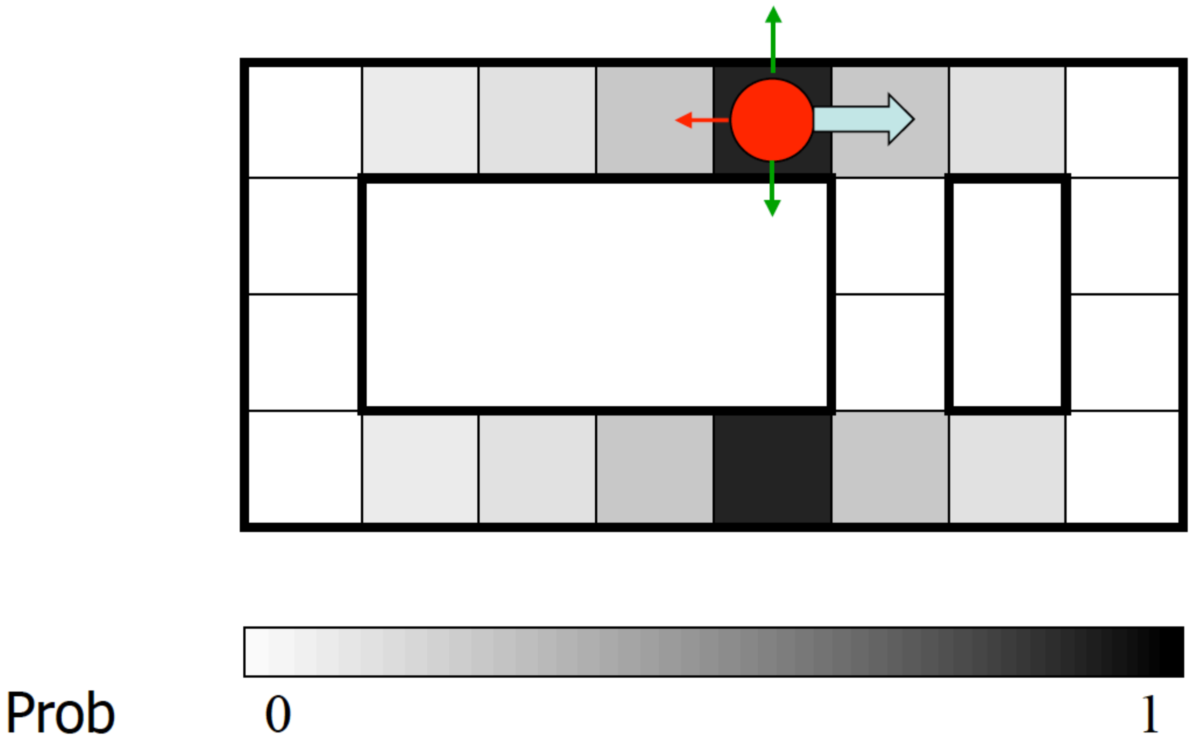
t=2

Example: Robot Localization



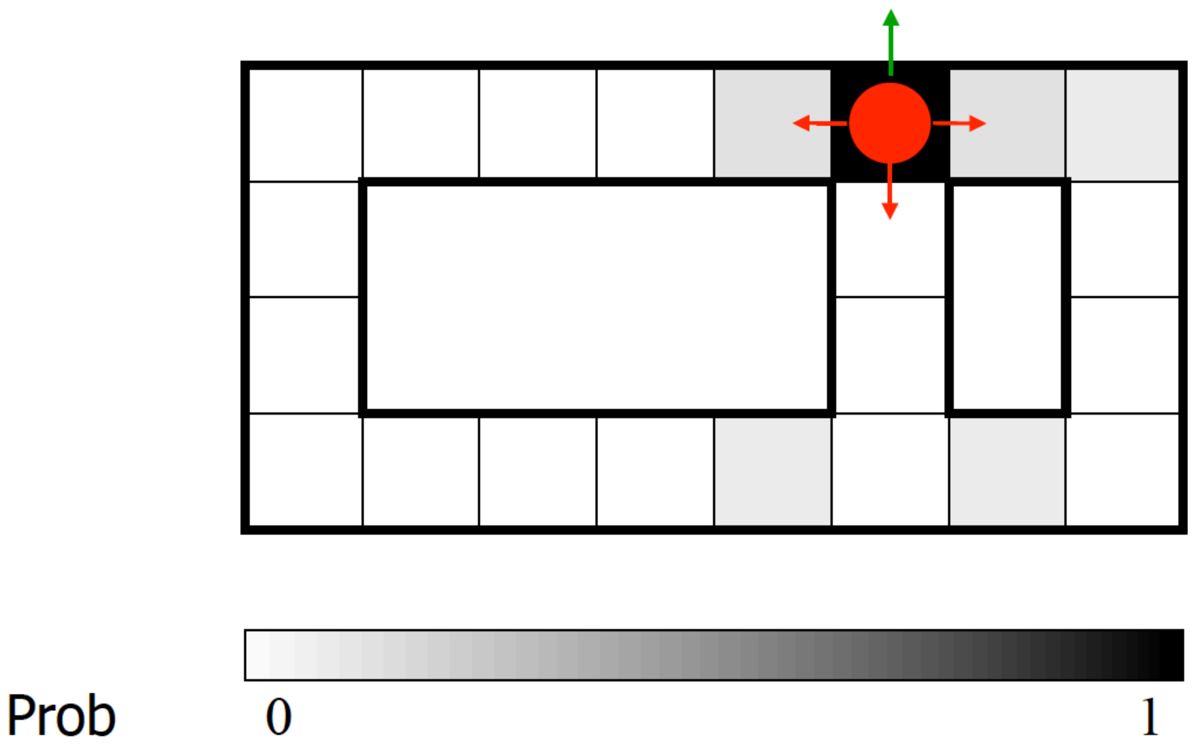
t=3

Example: Robot Localization



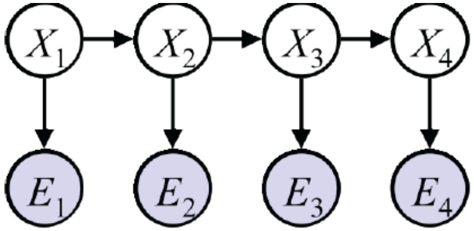
t=4

Example: Robot Localization



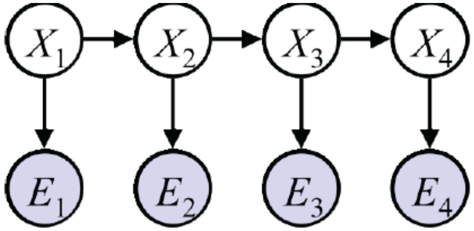
t=5

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

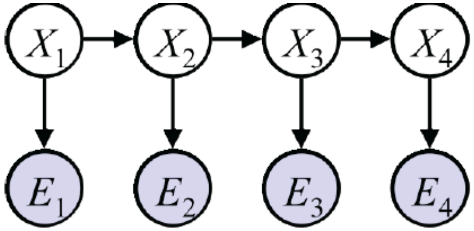
Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3

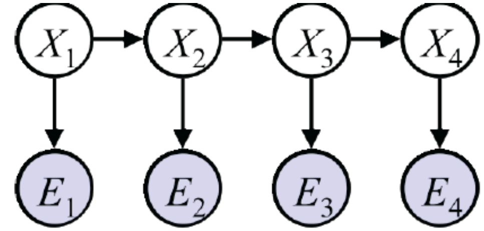


$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



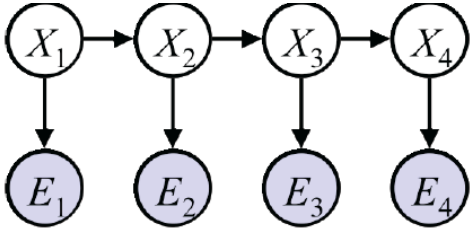
$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

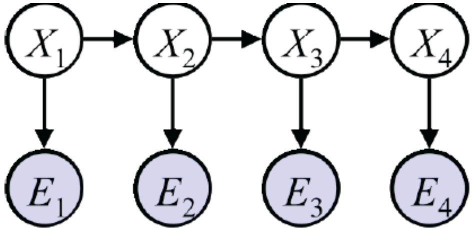
$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

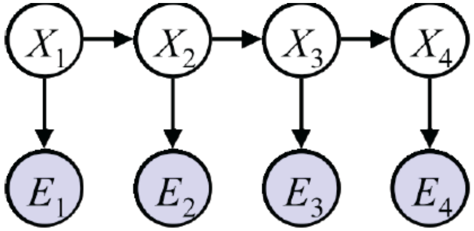
$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(x_2,e_1,e_2)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

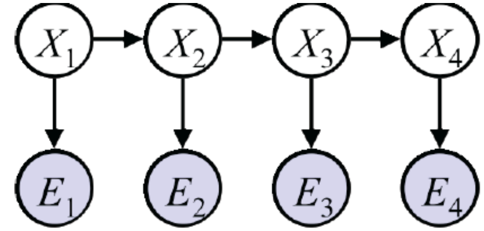
$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(x_2,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)\sum_{x_2} P(x_3|x_2)P(x_2,e_1,e_2)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1)$$

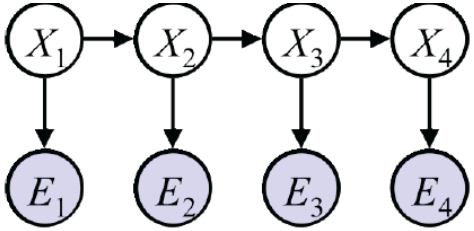
$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(x_2,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)\sum_{x_2} P(x_3|x_2)P(x_2,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3,e_1,e_2)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1)$$

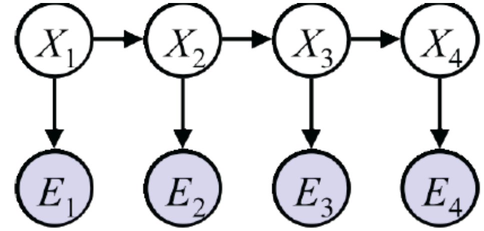
$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(x_2,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)\sum_{x_2} P(x_3|x_2)P(x_2,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(x_3,e_1,e_2,e_3)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(x_2,e_1,e_2)$$

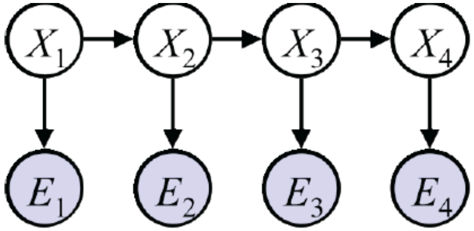
$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)\sum_{x_2} P(x_3|x_2)P(x_2,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(x_3,e_1,e_2,e_3)$$

$$= P(e_4|X_4)\sum_{x_3} P(X_4|x_3)P(x_3,e_1,e_2,e_3)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) = \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1)$$

$$= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1)$$

$$= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(x_2,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)\sum_{x_2} P(x_3|x_2)P(x_2,e_1,e_2)$$

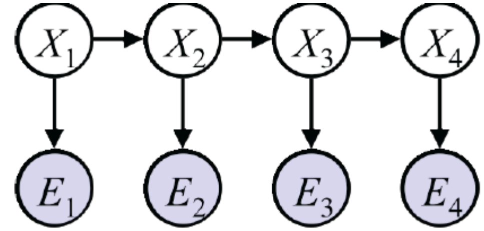
$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3,e_1,e_2)$$

$$= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(x_3,e_1,e_2,e_3)$$

$$= P(e_4|X_4)\sum_{x_3} P(X_4|x_3)P(x_3,e_1,e_2,e_3)$$

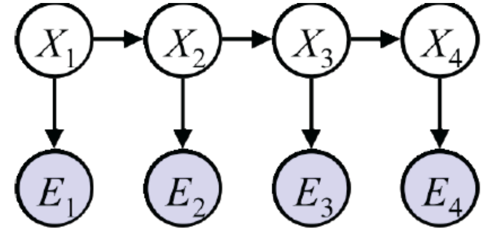
$$= P(e_4|X_4)P(X_4,e_1,e_2,e_3)$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$
\begin{aligned}
P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) &= \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1) \\
&= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1) \\
&= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1) \\
&= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(x_2,e_1,e_2) \\
&= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)\sum_{x_2} P(x_3|x_2)P(x_2,e_1,e_2) \\
&= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3,e_1,e_2) \\
&= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(x_3,e_1,e_2,e_3) \\
&= P(e_4|X_4)\sum_{x_3} P(X_4|x_3)P(x_3,e_1,e_2,e_3) \\
&= P(e_4|X_4)P(x_4,e_1,e_2,e_3) \\
&= P(X_4,e_1,e_2,e_3,e_4)
\end{aligned}
$$

Query: $P(X_4|e_1,e_2,e_3,e_4)$ --- Variable Elimination, X_1, X_2, X_3



$$
\begin{aligned}
P(X_4|e_1,e_2,e_3,e_4) \propto P(X_4,e_1,e_2,e_3,e_4) &= \sum_{x_1,x_2,x_3} P(x_1,x_2,x_3,X_4,e_1,e_2,e_3,e_4) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(e_1|x_1)P(x_1) \\
&= \sum_{x_3}\sum_{x_2}\sum_{x_1} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2|x_1)P(x_1,e_1) \\
&= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)\sum_{x_1} P(x_2|x_1)P(x_1,e_1) \\
&= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(e_2|x_2)P(x_2,e_1) \\
&= \sum_{x_3}\sum_{x_2} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3|x_2)P(x_2,e_1,e_2) \\
&= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)\sum_{x_2} P(x_3|x_2)P(x_2,e_1,e_2) \\
&= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(e_3|x_3)P(x_3,e_1,e_2) \\
&= \sum_{x_3} P(e_4|X_4)P(X_4|x_3)P(x_3,e_1,e_2,e_3) \\
&= P(e_4|X_4)\sum_{x_3} P(X_4|x_3)P(x_3,e_1,e_2,e_3) \\
&= P(e_4|X_4)P(x_4,e_1,e_2,e_3) \\
&= P(X_4,e_1,e_2,e_3,e_4)
\end{aligned}
$$

Re-occurring computation:

$$
\begin{aligned}
& P(x_t,e_1,e_2,\ldots,e_t) \\
= \; & P(e_t|x_t)\sum_{x_{t-1}} P(x_t|x_{t-1})P(x_{t-1},e_1,e_2,\ldots,e_{t-1})
\end{aligned}
$$

The Forward Algorithm

- We are given evidence at each time and want to know

$$B_t(X) = P(X_t|e_{1:t})$$

The Forward Algorithm

- We are given evidence at each time and want to know

$$B_t(X) = P(X_t|e_{1:t})$$

- We can derive the following updates

$$P(x_t|e_{1:t}) \propto_X P(x_t, e_{1:t})$$

We can normalize as we go if we want to have P(x|e) at each time step, or just once at the end…

The Forward Algorithm

- We are given evidence at each time and want to know

$$B_t(X) = P(X_t|e_{1:t})$$

- We can derive the following updates

$$P(x_t|e_{1:t}) \propto_X P(x_t, e_{1:t})$$

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

We can normalize as we go if we want to have P(x|e) at each time step, or just once at the end…

The Forward Algorithm

- We are given evidence at each time and want to know

$$B_t(X) = P(X_t|e_{1:t})$$

- We can derive the following updates

$$P(x_t|e_{1:t}) \propto_X P(x_t, e_{1:t})$$

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

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

We can normalize as we go if we want to have P(x|e) at each time step, or just once at the end…

The Forward Algorithm

- We are given evidence at each time and want to know

$$B_t(X) = P(X_t | e_{1:t})$$

- We can derive the following updates

$$\begin{aligned}
P(x_t | e_{1:t}) &\propto_X P(x_t, e_{1:t}) \\
&= \sum_{x_{t-1}} P(x_{t-1}, x_t, e_{1:t}) \\
&= \sum_{x_{t-1}} P(x_{t-1}, e_{1:t-1}) P(x_t | x_{t-1}) P(e_t | x_t) \\
&= P(e_t | x_t) \sum_{x_{t-1}} P(x_t | x_{t-1}) P(x_{t-1}, e_{1:t-1})
\end{aligned}$$

We can normalize as we go if we want to have P(x|e) at each time step, or just once at the end…

The Forward Algorithm

- We are given evidence at each time and want to know

$$B_t(X) = P(X_t|e_{1:t})$$

- We can derive the following updates

$$P(x_t|e_{1:t}) \propto_X P(x_t, e_{1:t})$$

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

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

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

We can normalize as we go if we want to have P(x|e) at each time step, or just once at the end…

- = exactly variable elimination in order X_1, X_2, …

Belief Updating = the forward algorithm broken down into two steps and with normalization

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

Belief Updating = the forward algorithm broken down into two steps and with normalization

- Forward algorithm: $P(x_t, e_{1:t}) = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
- Can break this down into:

Belief Updating = the forward algorithm broken down into two steps and with normalization

- Forward algorithm: $P(x_t, e_{1:t}) = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
- Can break this down into:
  - Time update: $P(x_t, e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$

Belief Updating = the forward algorithm broken down into two steps and with normalization

- Forward algorithm: $P(x_t, e_{1:t}) = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
- Can break this down into:
  - Time update: $P(x_t, e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
  - Observation update: $P(x_t, e_{1:t}) = P(e_t|x_t) P(x_t, e_{1:t-1})$

Belief Updating = the forward algorithm broken down into two steps and with normalization

- Forward algorithm: $P(x_t, e_{1:t}) = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
- Can break this down into:
  - Time update: $P(x_t, e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
  - Observation update: $P(x_t, e_{1:t}) = P(e_t|x_t) P(x_t, e_{1:t-1})$
- Normalizing in the observation update gives:

Belief Updating = the forward algorithm broken down into two steps and with normalization

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

- Can break this down into:
  - Time update: $P(x_t, e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
  - Observation update: $P(x_t, e_{1:t}) = P(e_t|x_t) P(x_t, e_{1:t-1})$

- Normalizing in the observation update gives:
  - Time update: $P(x_t|e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}|e_{1:t-1})$

Belief Updating = the forward algorithm broken down into two steps and with normalization

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

- Can break this down into:
  - Time update: $P(x_t, e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
  - Observation update: $P(x_t, e_{1:t}) = P(e_t|x_t) P(x_t, e_{1:t-1})$

- Normalizing in the observation update gives:
  - Time update: $P(x_t|e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}|e_{1:t-1})$
  - Observation update: $P(x_t|e_{1:t}) \propto P(e_t|x_t) P(x_t|e_{1:t-1})$

Belief Updating = the forward algorithm broken down into two steps and with normalization

- Forward algorithm: $P(x_t, e_{1:t}) = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
- Can break this down into:
  - Time update: $P(x_t, e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
  - Observation update: $P(x_t, e_{1:t}) = P(e_t|x_t) P(x_t, e_{1:t-1})$
- Normalizing in the observation update gives:
  - Time update: $P(x_t|e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}|e_{1:t-1})$
  - Observation update: $P(x_t|e_{1:t}) \propto P(e_t|x_t) P(x_t|e_{1:t-1})$
- Notation: $B_t(x_t) = P(x_t|e_{1:t}), \quad B'_t(x_t) = P(x_t|e_{1:t-1})$

Belief Updating = the forward algorithm broken down into two steps and with normalization

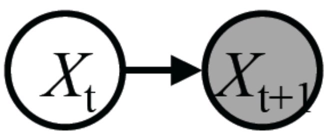
- Forward algorithm: $P(x_t, e_{1:t}) = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
- Can break this down into:
  - Time update: $P(x_t, e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
  - Observation update: $P(x_t, e_{1:t}) = P(e_t|x_t) P(x_t, e_{1:t-1})$
- Normalizing in the observation update gives:
  - Time update: $P(x_t|e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}|e_{1:t-1})$
  - Observation update: $P(x_t|e_{1:t}) \propto P(e_t|x_t) P(x_t|e_{1:t-1})$
- Notation: $B_t(x_t) = P(x_t|e_{1:t}), \quad B'_t(x_t) = P(x_t|e_{1:t-1})$
  - Time update: $B'_t(x_t) = \sum_{x_{t-1}} P(x_t|x_{t-1}) B_{t-1}(x_{t-1})$

Belief Updating = the forward algorithm broken down into two steps and with normalization

- Forward algorithm: $P(x_t, e_{1:t}) = P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
- Can break this down into:
  - Time update: $P(x_t, e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$
  - Observation update: $P(x_t, e_{1:t}) = P(e_t|x_t) P(x_t, e_{1:t-1})$
- Normalizing in the observation update gives:
  - Time update: $P(x_t|e_{1:t-1}) = \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}|e_{1:t-1})$
  - Observation update: $P(x_t|e_{1:t}) \propto P(e_t|x_t) P(x_t|e_{1:t-1})$
- Notation: $B_t(x_t) = P(x_t|e_{1:t}), \quad B'_t(x_t) = P(x_t|e_{1:t-1})$
  - Time update: $B'_t(x_t) = \sum_{x_{t-1}} P(x_t|x_{t-1}) B_{t-1}(x_{t-1})$
  - Observation update: $B_t(x_t) = P(e_t|x_t) B'_t(x_t)$

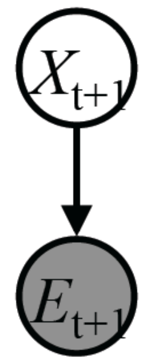
Belief updates can also easily be derived from basic probability

- **Passage of Time**
  - Given: $P(X_t)$, $P(X_{t+1} \mid X_t)$
  - Query: $P(x_{t+1})\ \forall\ x_{t+1}$



$$P(x_{t+1}) = \sum_{x_t} P(x_t, x_{t+1})$$
$$= \sum_{x_t} P(x_t) P(x_{t+1}|x_t)$$

- **Observation**
  - Given: $P(X_{t+1})$, $P(e_{t+1} \mid X_{t+1})$
  - Query: $P(x_{t+1} \mid e_{t+1})\ \forall\ x_{t+1}$



$$P(x_{t+1}|e_{t+1}) = P(x_{t+1}, e_{t+1})/P(e_{t+1})$$
$$\propto P(x_{t+1}, e_{t+1})$$
$$= P(x_{t+1}) P(e_{t+1}|x_{t+1})$$

Example: Passage of Time

- As time passes, uncertainty "accumulates"

<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	1.00	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01

T = 1

<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	0.06	<0.01	<0.01	<0.01
<0.01	0.76	0.06	0.06	<0.01	<0.01
<0.01	<0.01	0.06	<0.01	<0.01	<0.01

T = 2

0.05	0.01	0.05	<0.01	<0.01	<0.01
0.02	0.14	0.11	0.35	<0.01	<0.01
0.07	0.03	0.05	<0.01	0.03	<0.01
0.03	0.03	<0.01	<0.01	<0.01	<0.01

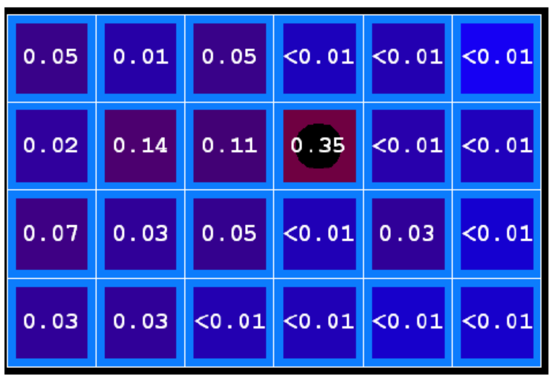
T = 5

$$B'(X') = \sum_x P(X'|x)B(x)$$

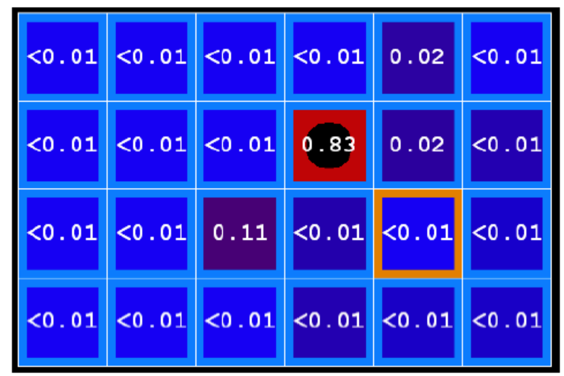
Transition model: ghosts usually go clockwise

Example: Observation

- As we get observations, beliefs get reweighted, uncertainty “decreases”



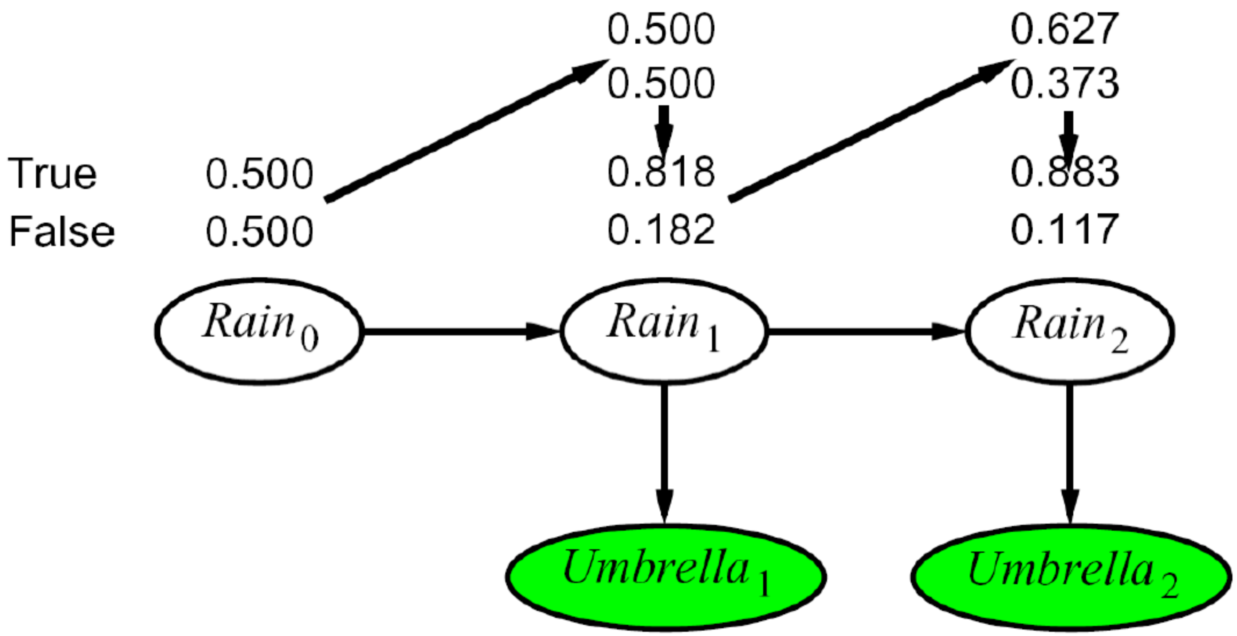
Before observation



After observation

$$B(X) \propto P(e|X)B'(X)$$

Example HMM



0.500
0.500
0.627
0.373
True
False
0.500
0.500
0.818
0.182
0.883
0.117
Rain 0
Rain 1
Rain 2
Umbrella 1
Umbrella 2

Outline

- ✓ Markov Models

  (= a particular Bayes net)

- Hidden Markov Models (HMMs)
  - ✓ Representation

    (= another particular Bayes net)

  - Inference
    - ✓ Forward algorithm (= variable elimination)
    - Particle filtering (= likelihood weighting with some tweaks)
    - Viterbi (= variable elimination, but replace sum by max

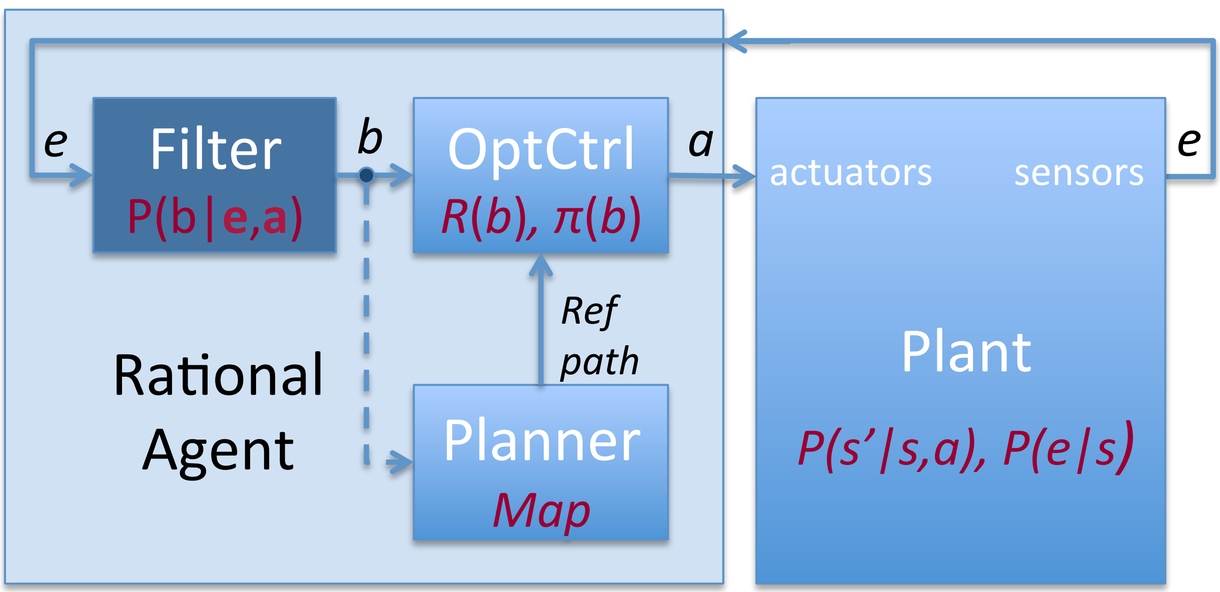
      = graph search)

Outline

- ✓ Markov Models
  (= a particular Bayes net)
- Hidden Markov Models (HMMs)
  - ✓ Representation
    (= another particular Bayes net)
  - Inference
    - ✓ Forward algorithm (= variable elimination)
    - Particle filtering (= likelihood weighting with some tweaks)
    - Viterbi (= variable elimination, but replace sum by max
      = graph search)
- Dynamic Bayes' Nets
  - Representation
    - (= yet another particular Bayes' net)
  - Inference: forward algorithm and particle filtering

Filtering (State Estimation)

Approximate Algorithm



Particle Filtering

- Filtering: approximate solution

Particle Filtering

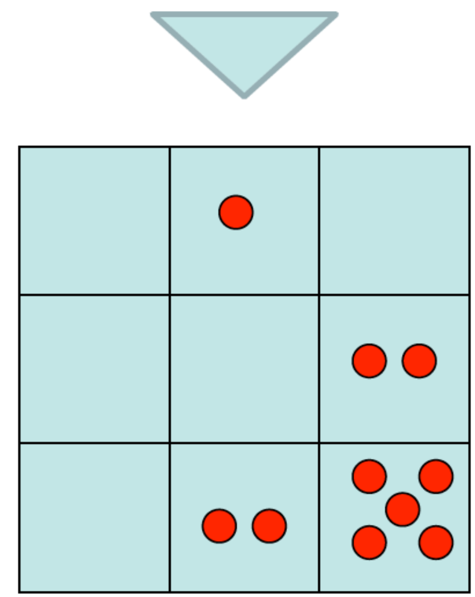
- Filtering: approximate solution
- Sometimes |X| is too big to use exact inference
  - |X| may be too big to even store B(X)
  - E.g. X is continuous

0.0	0.1	0.0
0.0	0.0	0.2
0.0	0.2	0.5

Particle Filtering

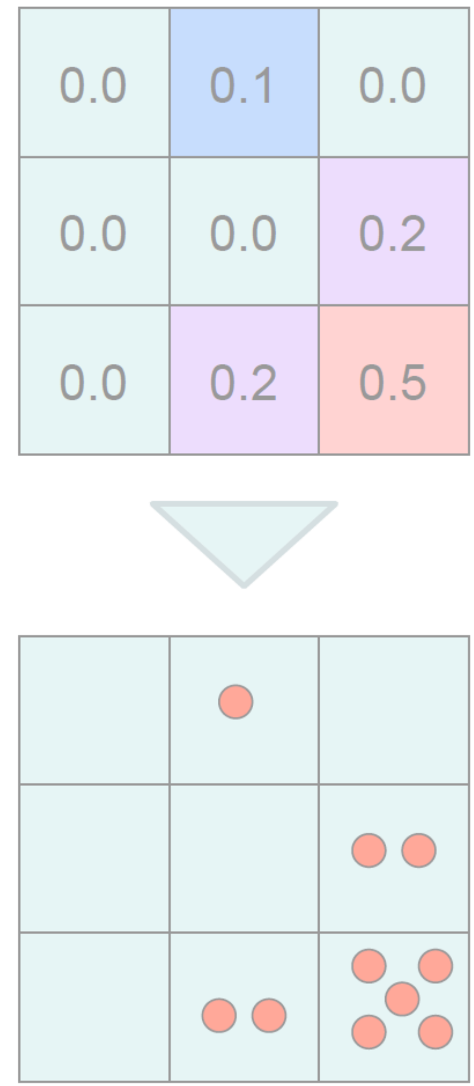
- Filtering: approximate solution
- Sometimes |X| is too big to use exact inference
  - |X| may be too big to even store B(X)
  - E.g. X is continuous
- Solution: approximate inference
  - Track samples of X, not all values
  - Samples are called particles

0.0	0.1	0.0
0.0	0.0	0.2
0.0	0.2	0.5



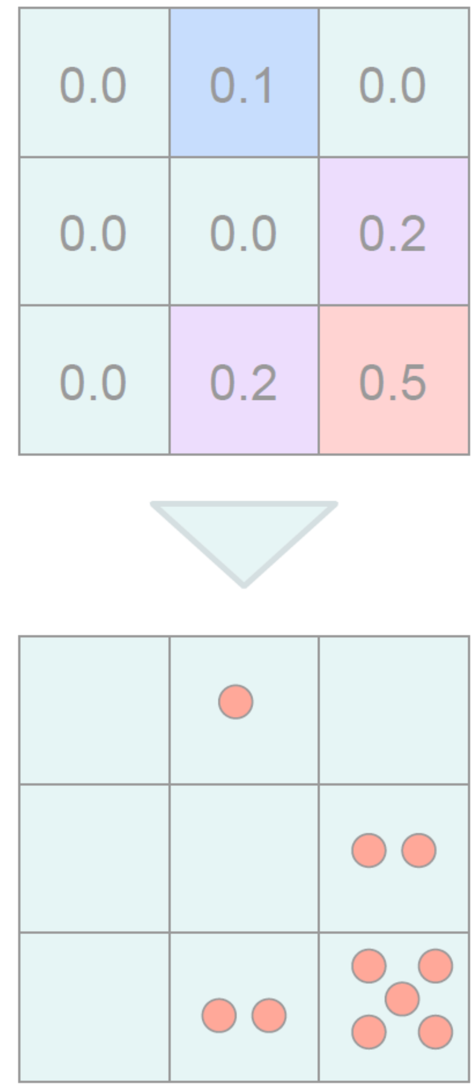
Particle Filtering

- Filtering: approximate solution
- Sometimes |X| is too big to use exact inference
  - |X| may be too big to even store B(X)
  - E.g. X is continuous
- Solution: approximate inference
  - Track samples of X, not all values
  - Samples are called particles
  - Time per step is linear in the number of samples
  - But: number needed may be large
  - In memory: list of particles, not states



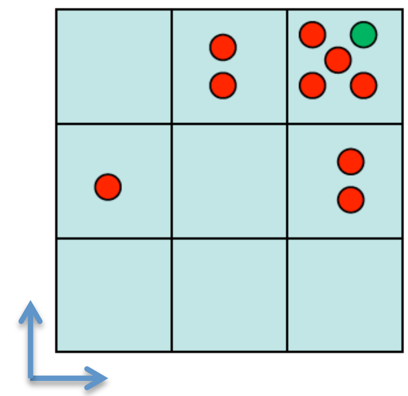
Particle Filtering

- Filtering: approximate solution
- Sometimes |X| is too big to use exact inference
  - |X| may be too big to even store B(X)
  - E.g. X is continuous
- Solution: approximate inference
  - Track samples of X, not all values
  - Samples are called particles
  - Time per step is linear in the number of samples
  - But: number needed may be large
  - In memory: list of particles, not states
- This is how robot localization works in practice
- Particle is just new name for sample



Representation: Particles

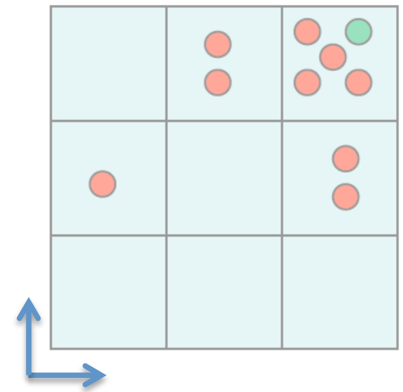
- Our representation of P(X) is now a list of N particles (samples)
  - Generally, N << |X|
  - Storing map from X to counts would defeat the point



Representation: Particles

- Our representation of P(X) is now a list of N particles (samples)
  - Generally, N << |X|
  - Storing map from X to counts would defeat the point

- P(x) approximated by number of particles with value x
  - So, many x will have P(x) = 0!
  - More particles, more accuracy



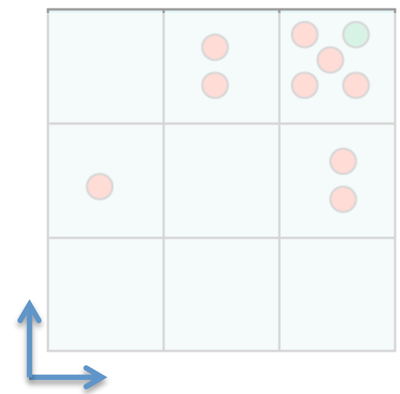
Particles:
(3,3)
(2,3)
(3,3)
(3,2)
(3,3)
(3,2)
(1,2)
(3,3)
(3,3)
(2,3)

Representation: Particles

- Our representation of P(X) is now a list of N particles (samples)
  - Generally, N << |X|
  - Storing map from X to counts would defeat the point

- P(x) approximated by number of particles with value x
  - So, many x will have P(x) = 0!
  - More particles, more accuracy

- For now, all particles have a weight of 1

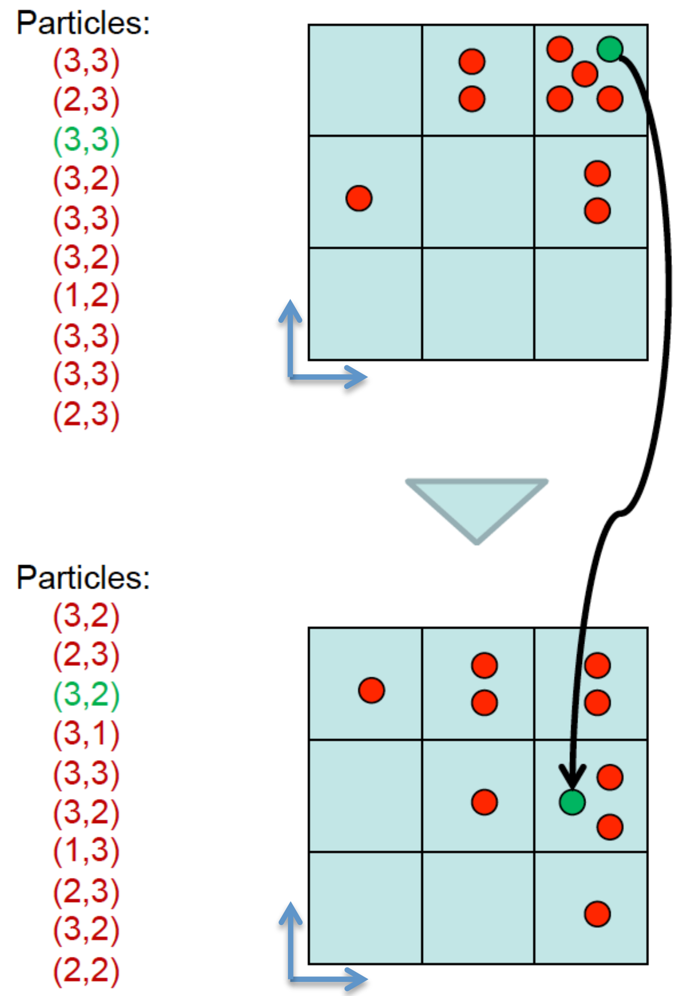


Particles:
(3,3)
(2,3)
(3,3)
(3,2)
(3,3)
(3,2)
(1,2)
(3,3)
(3,3)
(2,3)

Particle Filtering: Elapse Time

- Each particle is moved by sampling its next position from the transition model

$$x' = \text{sample}(P(X'|x))$$

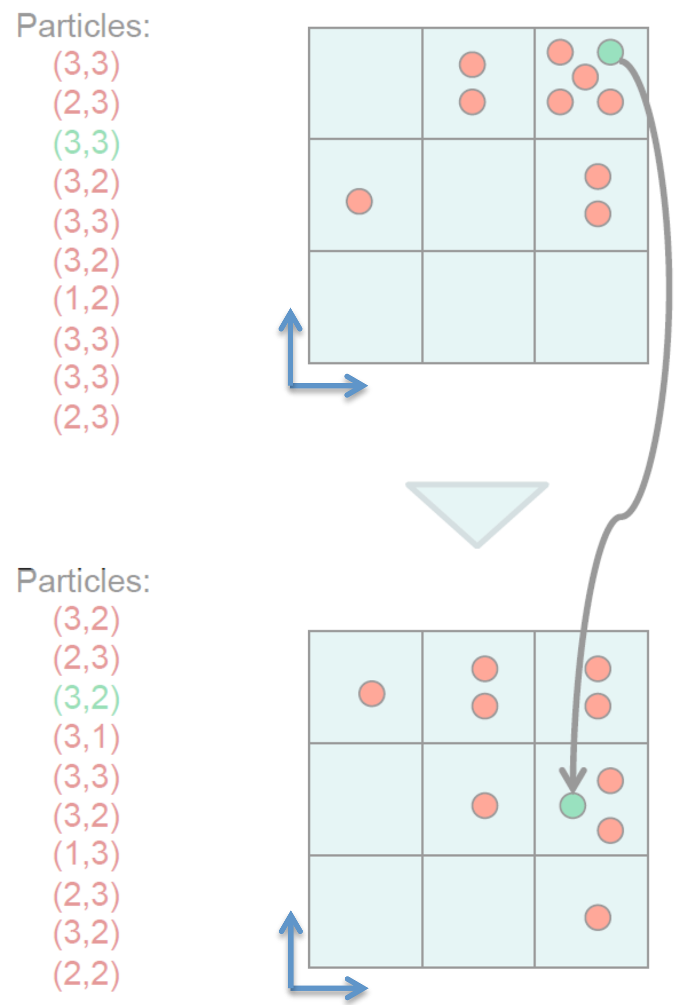


Particle Filtering: Elapse Time

- Each particle is moved by sampling its next position from the transition model

$$x' = \text{sample}(P(X'|x))$$

  - This is like prior sampling – samples' frequencies reflect the transition probs
  - Here, most samples move clockwise, but some move in another direction or stay in place



Particle Filtering: Elapse Time

- Each particle is moved by sampling its next position from the transition model

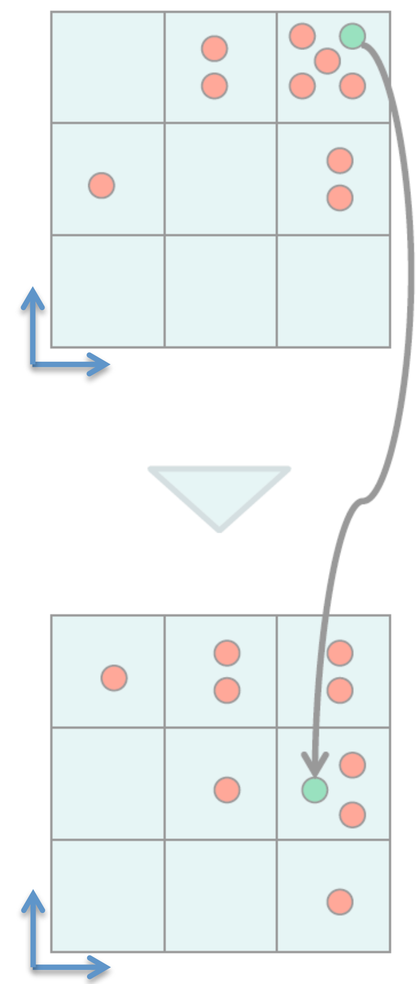
$$x' = \text{sample}(P(X'|x))$$

  - This is like prior sampling – samples' frequencies reflect the transition probs
  - Here, most samples move clockwise, but some move in another direction or stay in place

- This captures the passage of time
  - If enough samples, close to exact values before and after (consistent)

Particles:
(3,3)
(2,3)
(3,3)
(3,2)
(3,3)
(3,2)
(1,2)
(3,3)
(3,3)
(2,3)

Particles:
(3,2)
(2,3)
(3,2)
(3,1)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(2,2)



Particle Filtering: Observe

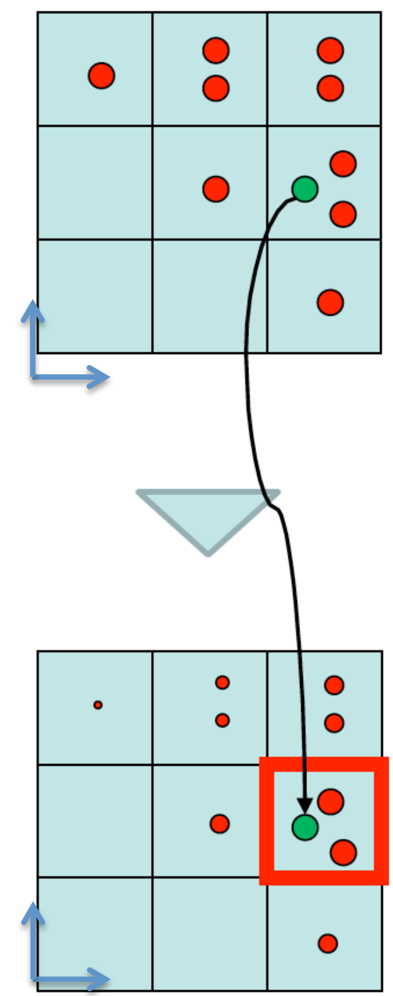
- Slightly trickier:
  - Don't sample observation, fix it
  - Similar to likelihood weighting, downweight samples based on the evidence

$$w(x) = P(e|x)$$

$$B(X) \propto P(e|X)B'(X)$$

Particles:
(3,2)
(2,3)
(3,2)
(3,1)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(2,2)

Particles:
(3,2) w=.9
(2,3) w=.2
(3,2) w=.9
(3,1) w=.4
(3,3) w=.4
(3,2) w=.9
(1,3) w=.1
(2,3) w=.2
(3,2) w=.9
(2,2) w=.4



Particle Filtering: Observe

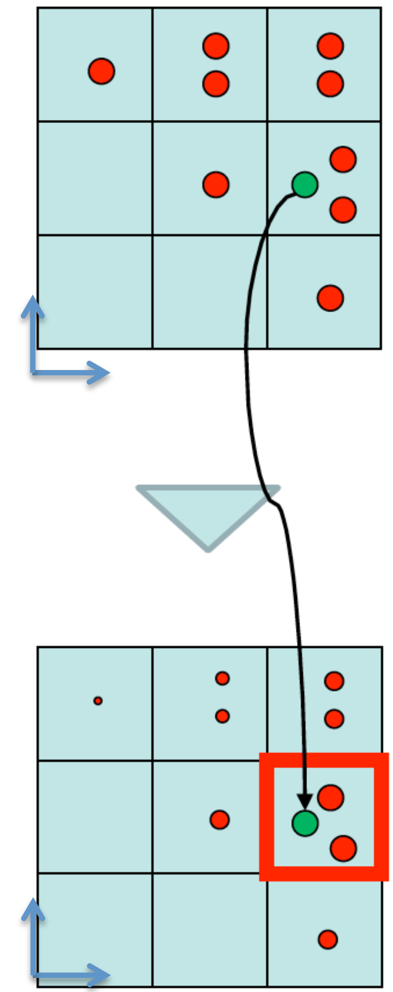
- Slightly trickier:
  - Don't sample observation, fix it
  - Similar to likelihood weighting, downweight samples based on the evidence

$$w(x) = P(e|x)$$

$$B(X) \propto P(e|X)B'(X)$$

- As before, the probabilities don't sum to one, since most have been downweighted (in fact they sum to an approximation of P(e))

Particles:
(3,2)
(2,3)
(3,2)
(3,1)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(2,2)

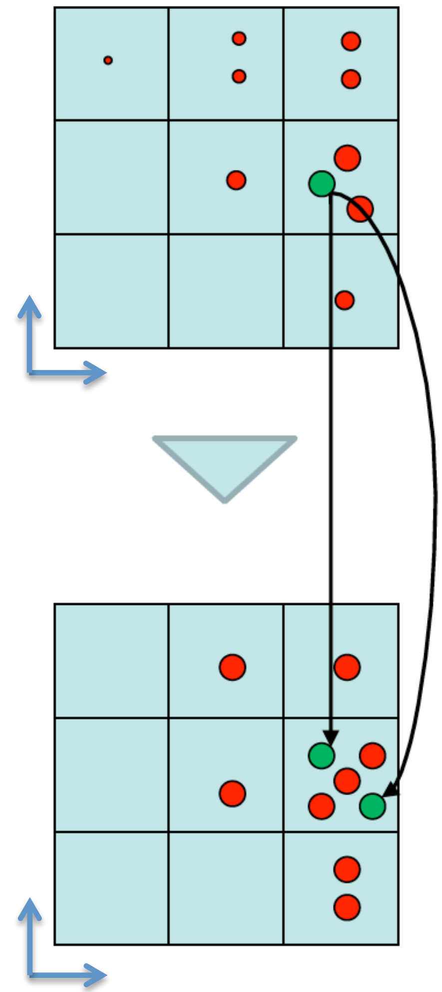


Particles:
(3,2) w=.9
(2,3) w=.2
(3,2) w=.9
(3,1) w=.4
(3,3) w=.4
(3,2) w=.9
(1,3) w=.1
(2,3) w=.2
(3,2) w=.9
(2,2) w=.4

Particle Filtering: Resample

- Rather than tracking weighted samples, we resample

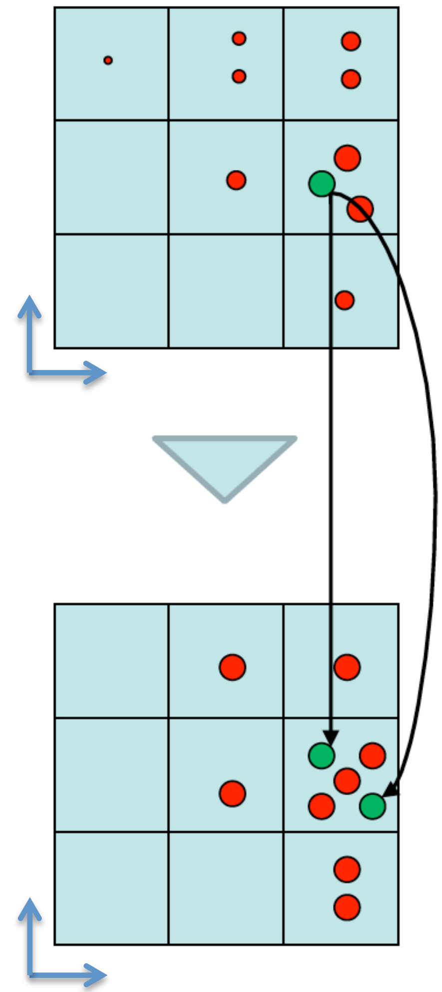
Particles:
(3,2) w=.9
(2,3) w=.2
(3,2) w=.9
(3,1) w=.4
(3,3) w=.4
(3,2) w=.9
(1,3) w=.1
(2,3) w=.2
(3,2) w=.9
(2,2) w=.4



Particle Filtering: Resample

- Rather than tracking weighted samples, we resample
- N times, we choose from our weighted sample distribution (i.e. draw with replacement)

Particles:
(3,2) w=.9
(2,3) w=.2
(3,2) w=.9
(3,1) w=.4
(3,3) w=.4
(3,2) w=.9
(1,3) w=.1
(2,3) w=.2
(3,2) w=.9
(2,2) w=.4

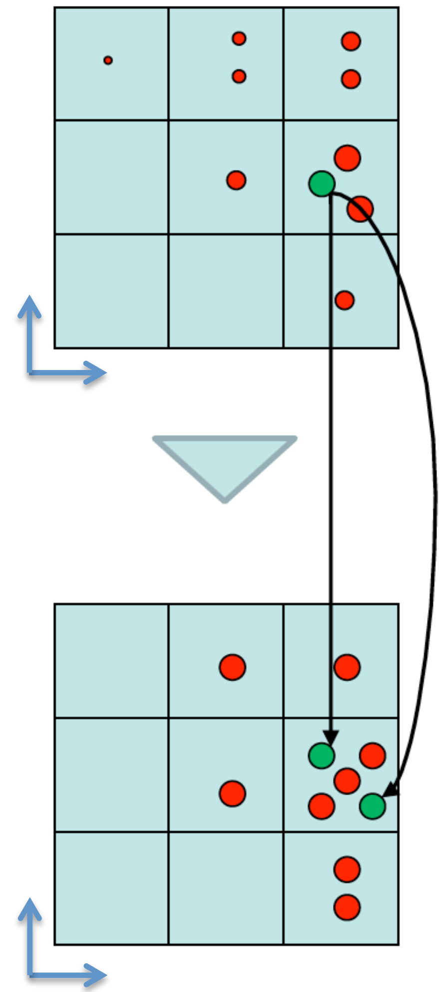


Particle Filtering: Resample

- Rather than tracking weighted samples, we resample
- N times, we choose from our weighted sample distribution (i.e. draw with replacement)
- This is equivalent to renormalizing the distribution

Particles:
(3,2) w=.9
(2,3) w=.2
(3,2) w=.9
(3,1) w=.4
(3,3) w=.4
(3,2) w=.9
(1,3) w=.1
(2,3) w=.2
(3,2) w=.9
(2,2) w=.4

(New) Particles:
(3,2)
(2,2)
(3,2)
(2,3)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(3,2)

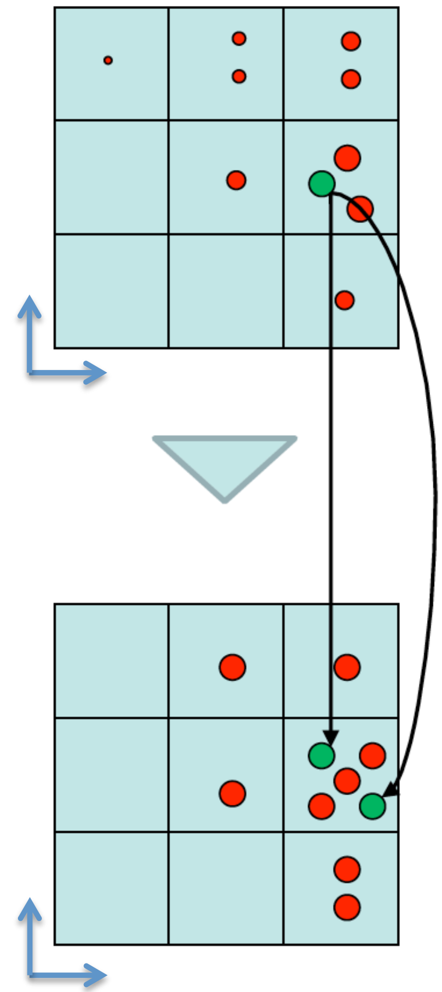


Particle Filtering: Resample

- Rather than tracking weighted samples, we resample
- N times, we choose from our weighted sample distribution (i.e. draw with replacement)
- This is equivalent to renormalizing the distribution
- Now the update is complete for this time step, continue with the next one

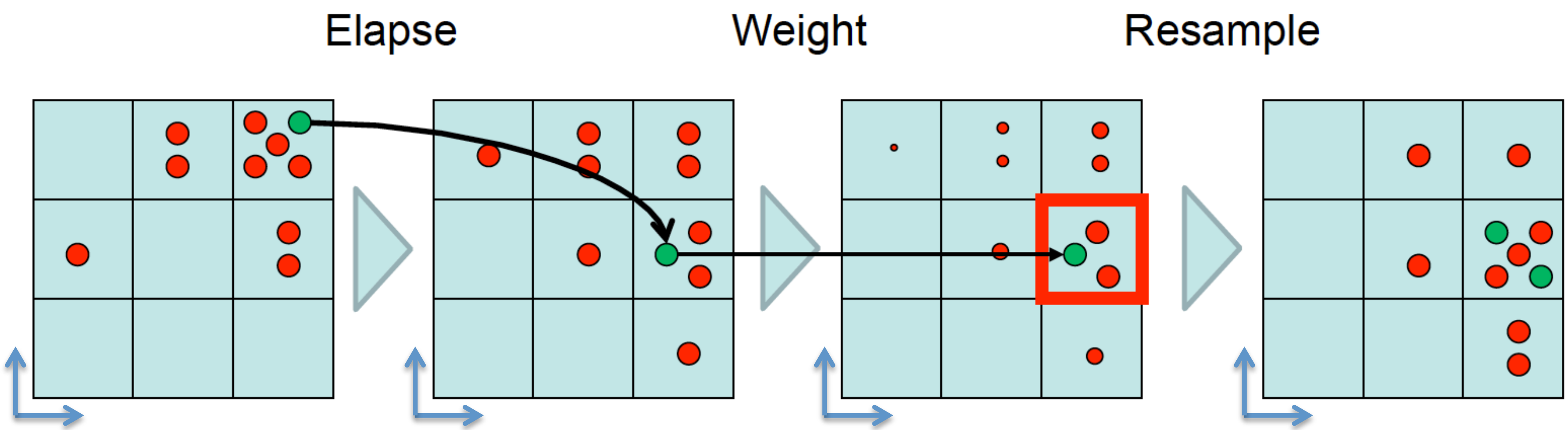
Particles:
(3,2) w=.9
(2,3) w=.2
(3,2) w=.9
(3,1) w=.4
(3,3) w=.4
(3,2) w=.9
(1,3) w=.1
(2,3) w=.2
(3,2) w=.9
(2,2) w=.4

(New) Particles:
(3,2)
(2,2)
(3,2)
(2,3)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(3,2)



Recap: Particle Filtering

- Particles: track samples of states rather than an explicit distribution

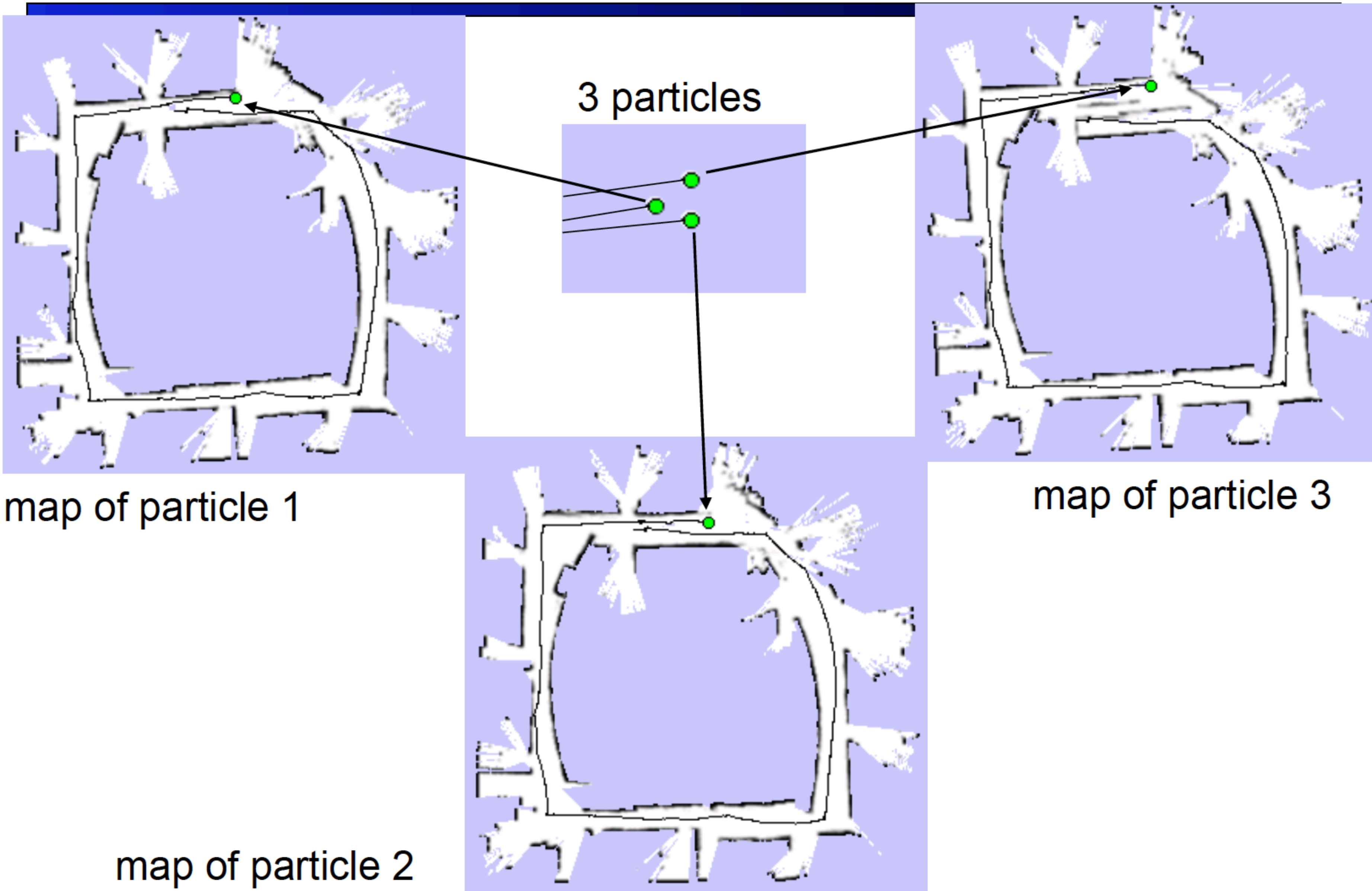


Particles:	Particles:	Particles:	(New) Particles:
(3,3)	(3,2)	(3,2) w=.9	(3,2)
(2,3)	(2,3)	(2,3) w=.2	(2,2)
(3,3)	(3,2)	(3,2) w=.9	(3,2)
(3,2)	(3,1)	(3,1) w=.4	(2,3)
(3,3)	(3,3)	(3,3) w=.4	(3,3)
(3,2)	(3,2)	(3,2) w=.9	(3,2)
(1,2)	(1,3)	(1,3) w=.1	(1,3)
(3,3)	(2,3)	(2,3) w=.2	(2,3)
(3,3)	(3,2)	(3,2) w=.9	(3,2)
(2,3)	(2,2)	(2,2) w=.4	(3,2)

Particle-Filtering Algorithm

```
function Particle-Filtering (e,dbn,N) returns P̂(X|e)
   local  S = GenerateFrom(P(X_0))     // vector of samples of size N
          W = 0                        // vector of weights of size N
   for i = 1:N {
      S[i] = Sample(P(X_1 | X_0 = S[i]))          // Step 1
      W[i] = P(e | X_1 = S[i]) }                  // Step 2
    S = Weighted-Sample-With-Replacement(N,S,W)   // Step 3
   return S
```

Particle Filter Example



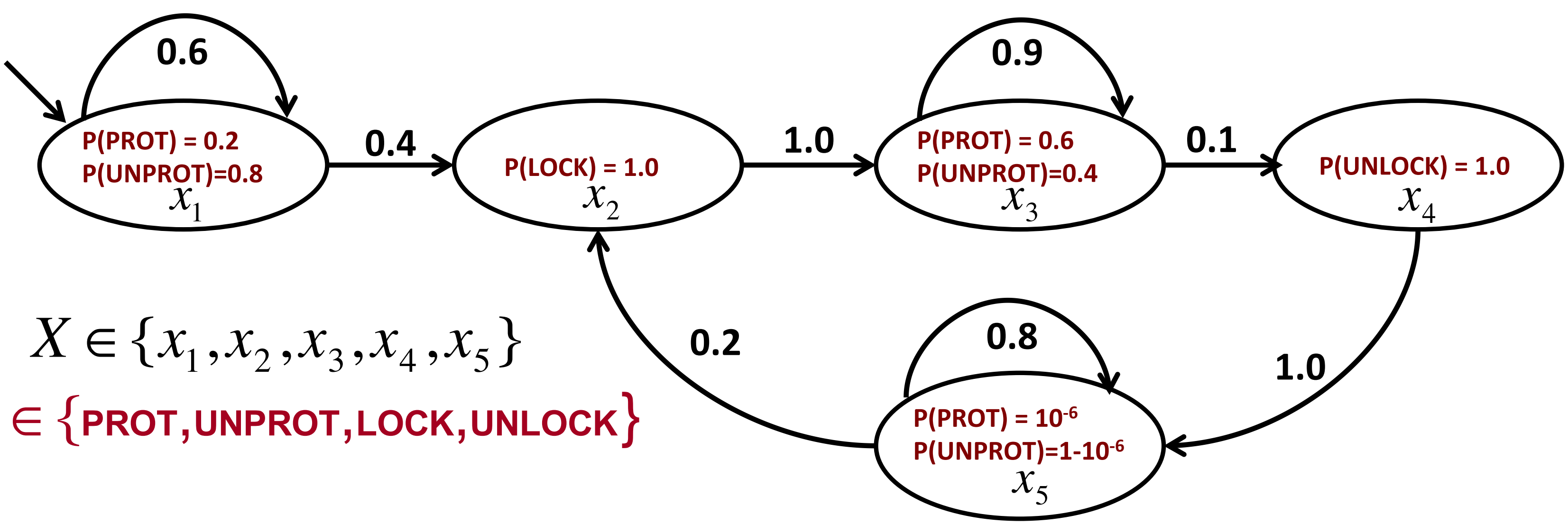
Hidden Markov Model

T_H	x_1	x_2	x_3	x_4	x_5
x_1	0.6	0.4			
x_2			1		
x_3			0.9	0.1	
x_4					1
x_5		0.2			0.8

$P(x_5 \mid x_3)$

O_H	PROT	UNPROT	LOCK	UNLOCK
x_1	0.2	0.8		
x_2			1	
x_3	0.6	0.4		
x_4				1
x_5	10^{-6}	$1-10^{-6}$		

$P(\text{UNLOCK} \mid x_3)$



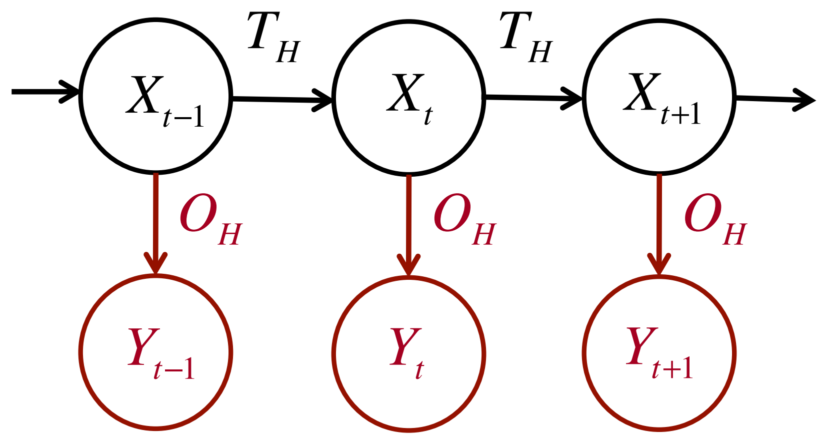
$X \in \{x_1, x_2, x_3, x_4, x_5\}$

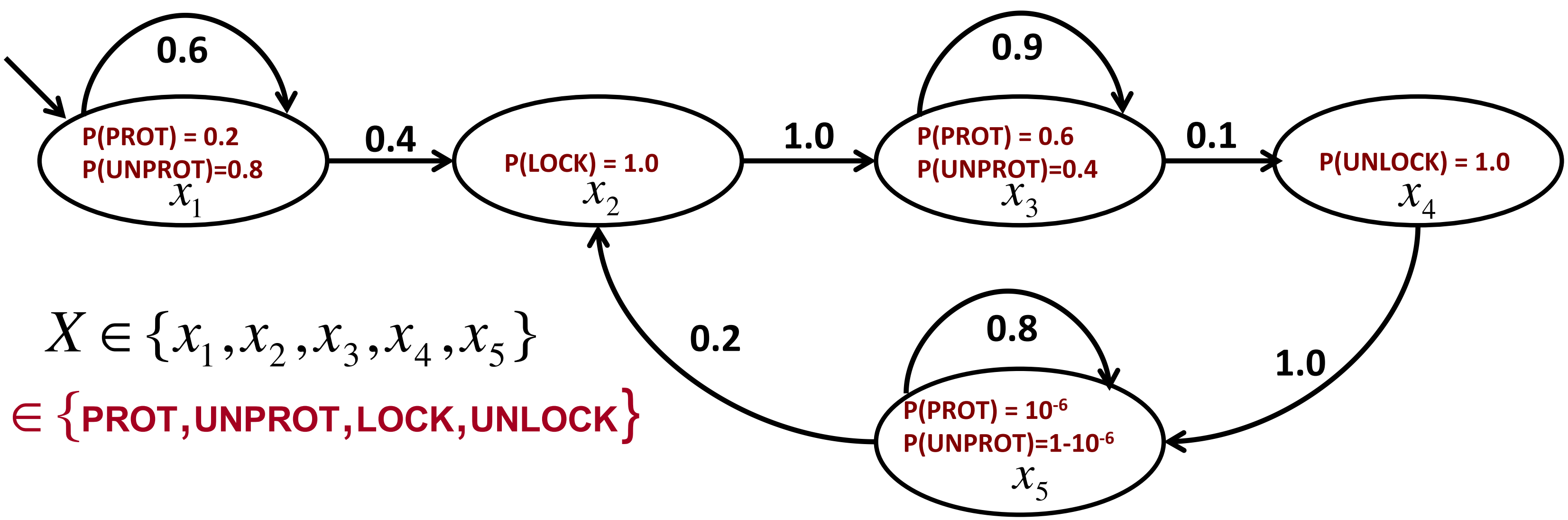
$Y \in \{\text{PROT}, \text{UNPROT}, \text{LOCK}, \text{UNLOCK}\}$

Hidden Markov Model

$$P(X_{t+1} \mid X_t) = T_H$$

$$P(Y_t \mid X_t) = O_H$$

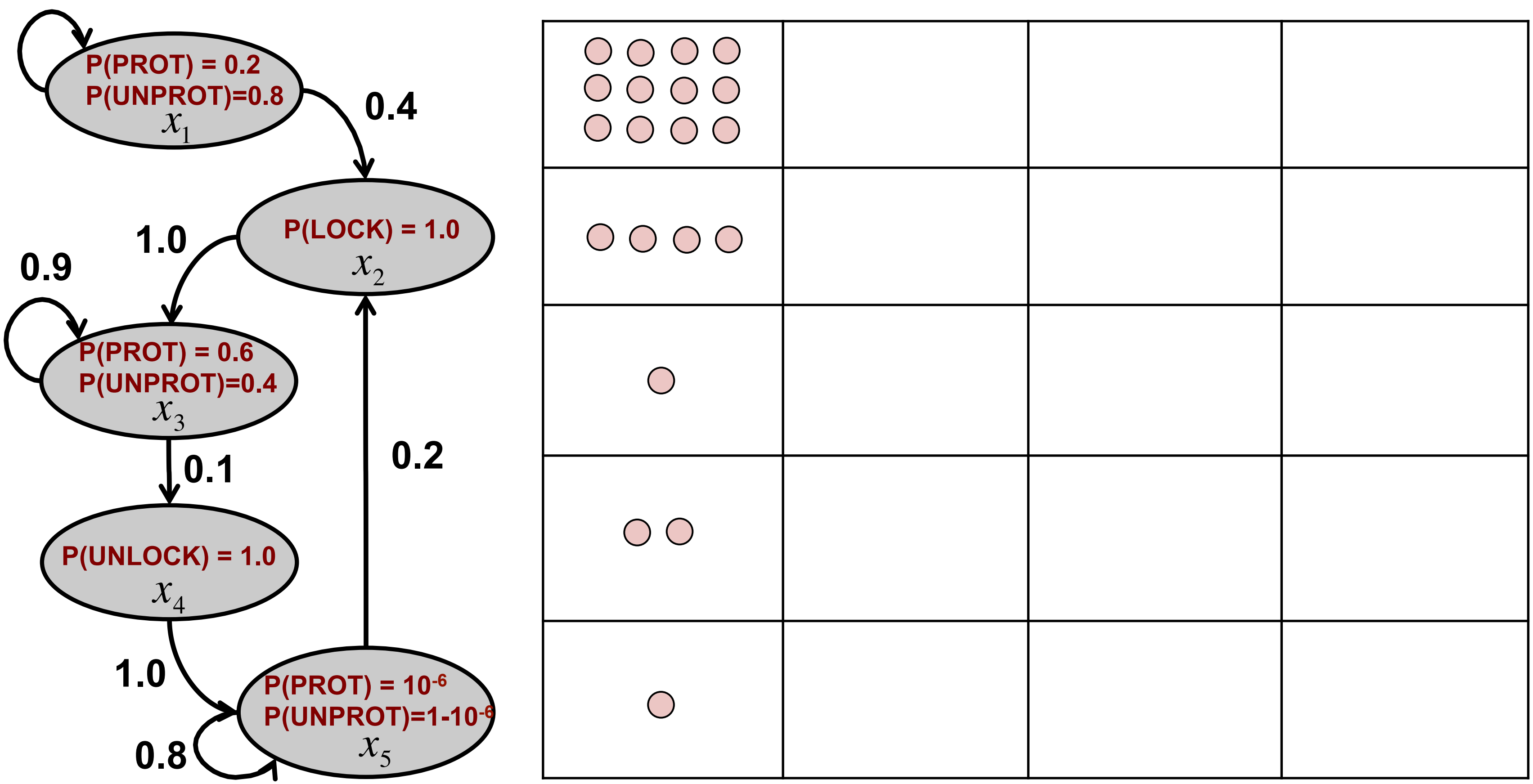




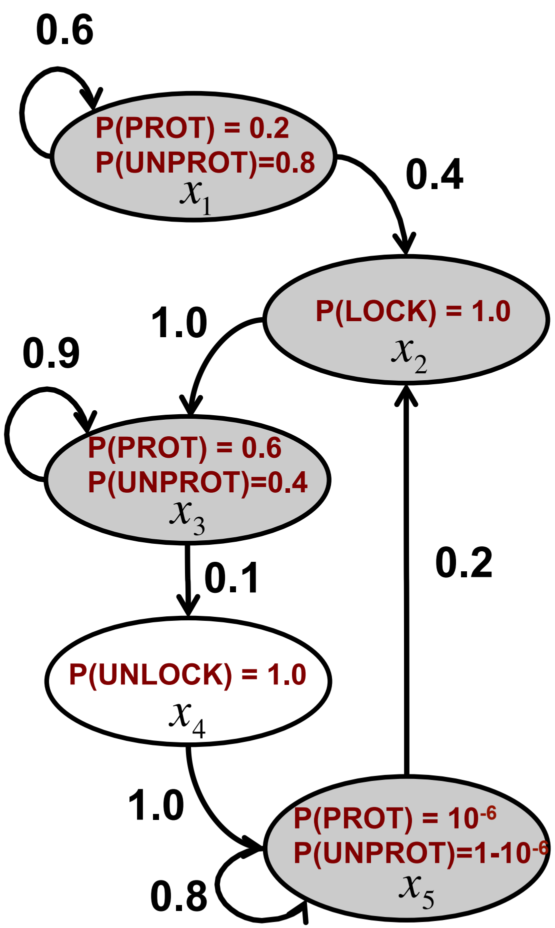
$$X \in \{x_1, x_2, x_3, x_4, x_5\}$$

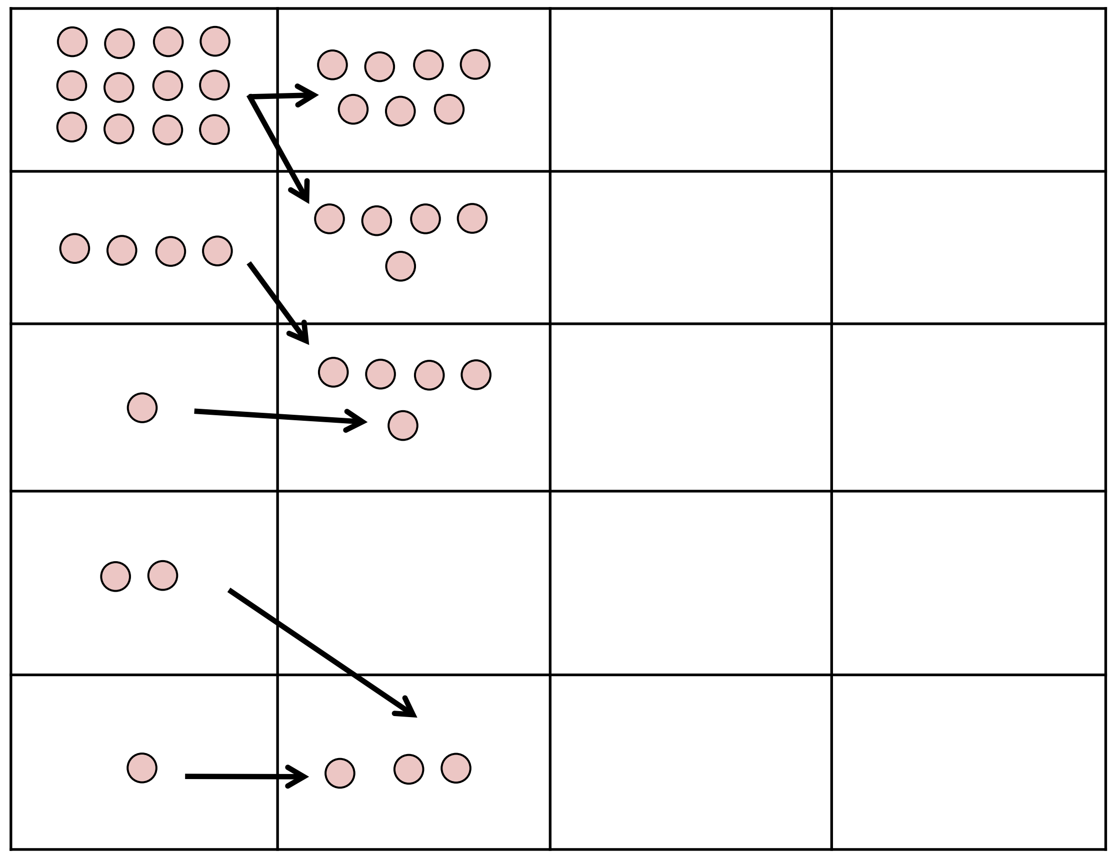
$$Y \in \{\text{PROT}, \text{UNPROT}, \text{LOCK}, \text{UNLOCK}\}$$

Initial Distribution of Particles

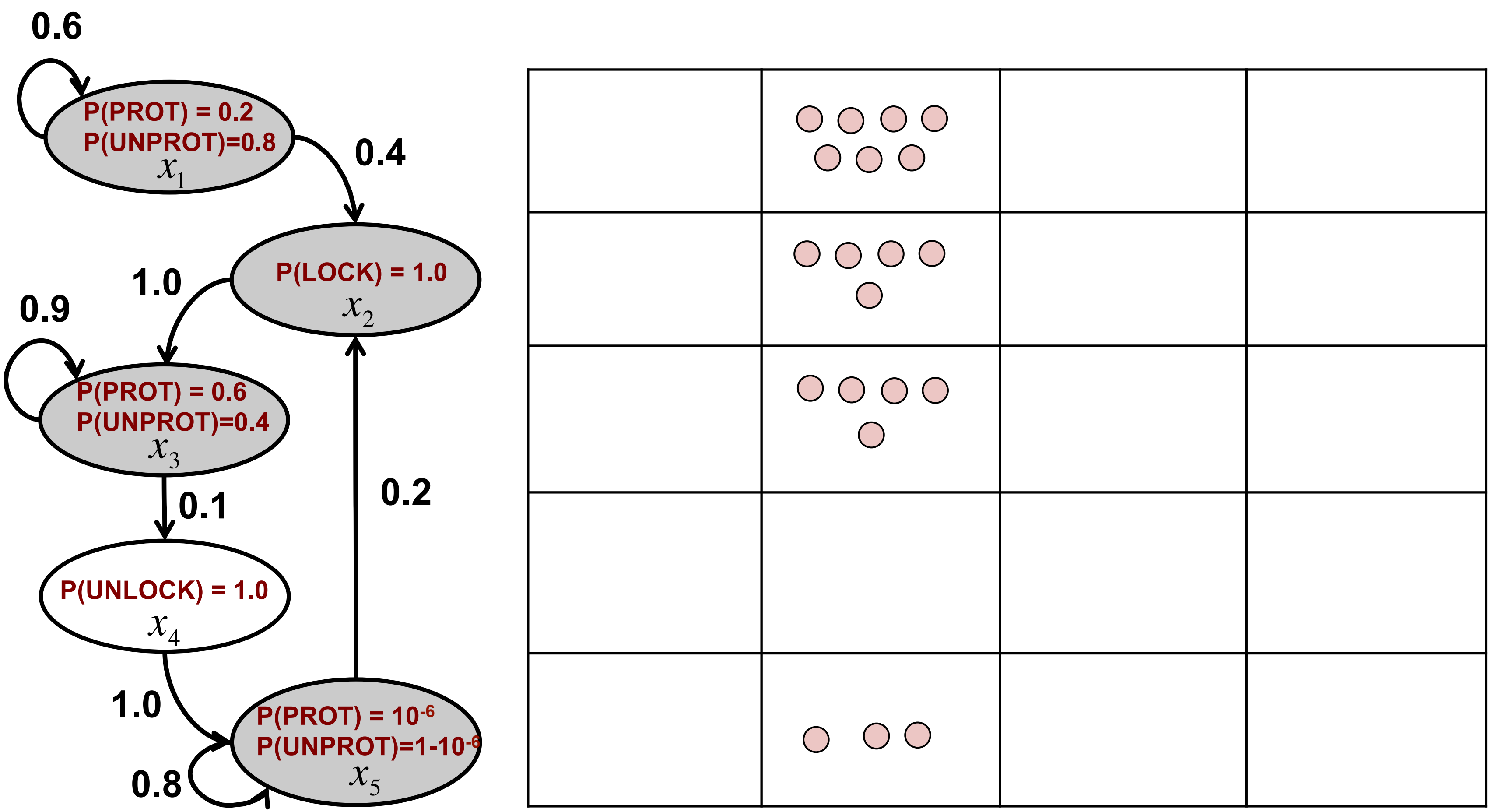


Take Next Step

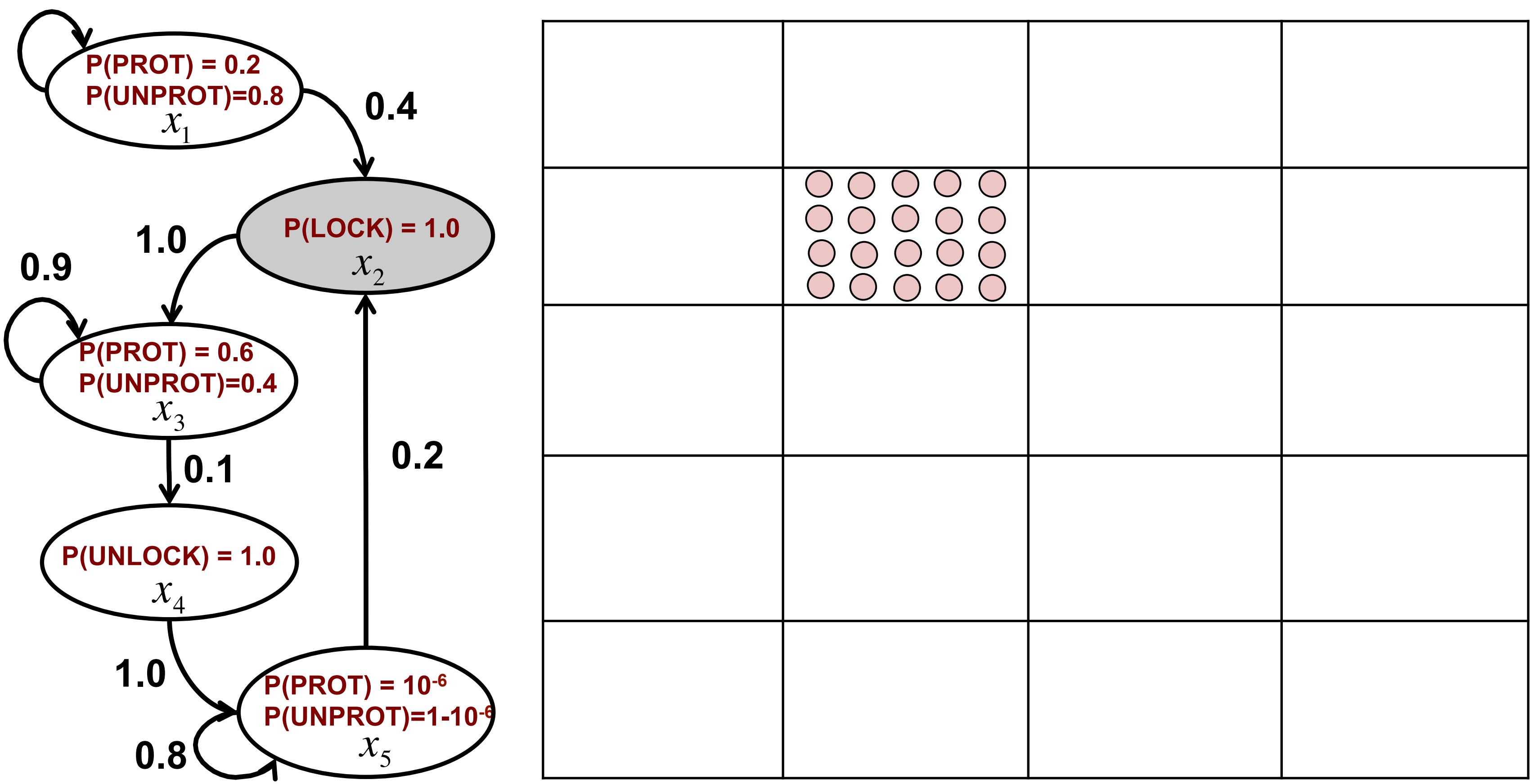




New Configuration of Particles



Observe LOCK and Resample



Further readings

- We are done with Probabilistic Reasoning

- To learn more:
  - Koller and Friedman, Probabilistic Graphical Models
  - Thrun, Burgard and Fox, Probabilistic Robotics