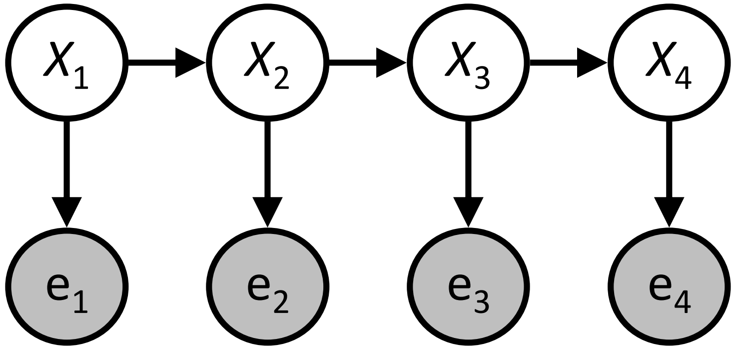
Warm-up as you walk in

- For the following Bayes net, write the query $P(X_4 \mid e_{1:4})$ in terms of the conditional probability tables associated with the Bayes net.



$P(X_4 \mid e_1, e_2, e_3, e_4) =$

Announcements

Assignments

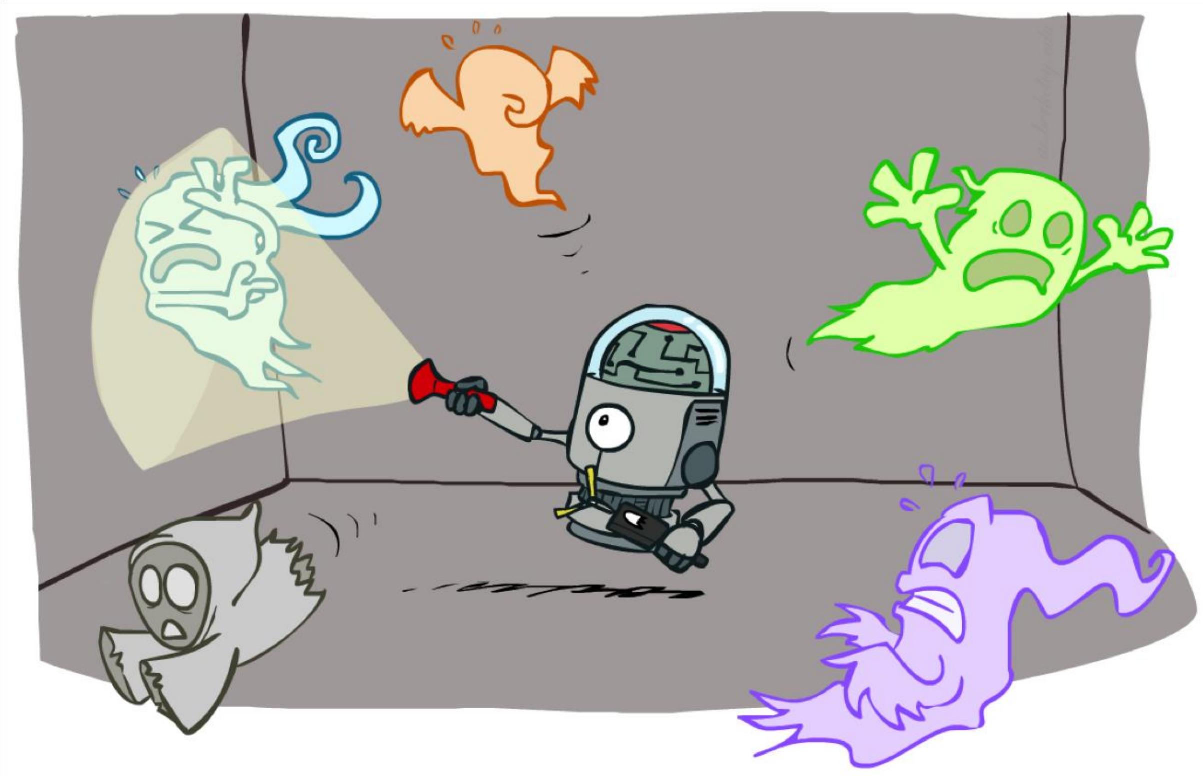
- HW11
  - Due Thur 4/25
- P5
  - Due Thur, 5/2

In-class Polls

- Denominator capped after midterm 2, 56 polls

AI: Representation and Problem Solving

HMMs and Particle Filters



Instructors: Pat Virtue & Stephanie Rosenthal

Slide credits: CMU AI and http://ai.berkeley.edu

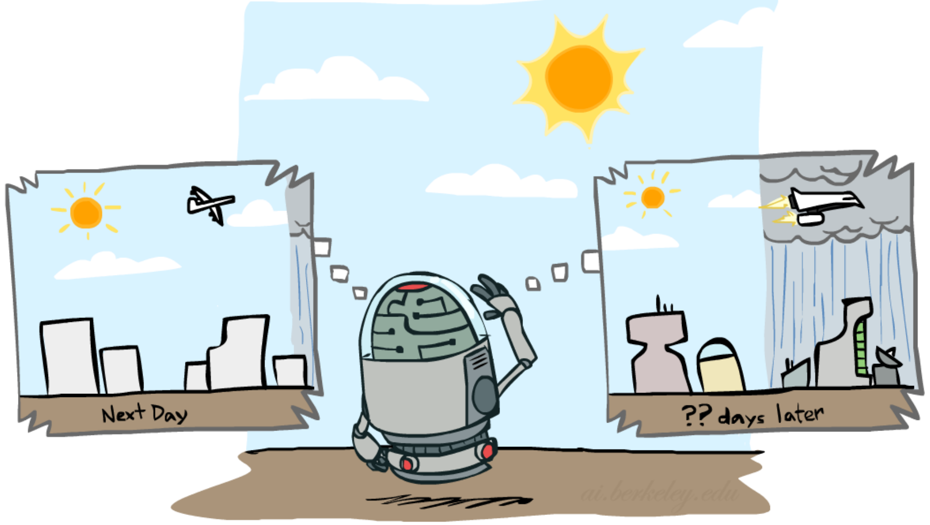
Markov chain warm-up

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

If you know the transition probabilities, $P(X_t \mid X_{t-1})$, and you know $P(X_4)$, write an equation to compute $P(X_5)$.

$$P(X_5) = \sum_{x_4} P(X_4 = x_4, X_5)$$

$$= \sum_{x_4} P(X_5 \mid X_4)\, P(X_4)$$



Markov chain warm-up

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

If you know the transition probabilities, $P(X_t \mid X_{t-1})$, and you know $P(X_4)$, write an equation to compute $P(X_5)$.

$$\begin{aligned} P(X_5) &= \sum_{x_4} P(x_4, X_5) \\ &= \sum_{x_4} P(X_5 \mid x_4) P(x_4) \end{aligned}$$

Markov chain warm-up

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

If you know the transition probabilities, $P(X_t \mid X_{t-1})$, and you know $P(X_4)$, write an equation to compute $P(X_5)$.

$$
\begin{aligned}
P(X_5) &= \textstyle\sum_{x_1,x_2,x_3,x_4} P(x_1, x_2, x_3, x_4, X_5) \\
&= \textstyle\sum_{x_1,x_2,x_3,x_4} P(X_5 \mid x_4) P(x_4 \mid x_3) P(x_3 \mid x_2) P(x_2 \mid x_1) P(x_1) \\
&= \textstyle\sum_{x_4} P(X_5 \mid x_4) \sum_{x_1,x_2,x_3} P(x_4 \mid x_3) P(x_3 \mid x_2) P(x_2 \mid x_1) P(x_1) \\
&= \textstyle\sum_{x_4} P(X_5 \mid x_4) \sum_{x_1,x_2,x_3} P(x_1, x_2, x_3, x_4) \\
&= \textstyle\sum_{x_4} P(X_5 \mid x_4) P(x_4)
\end{aligned}
$$

Weather prediction

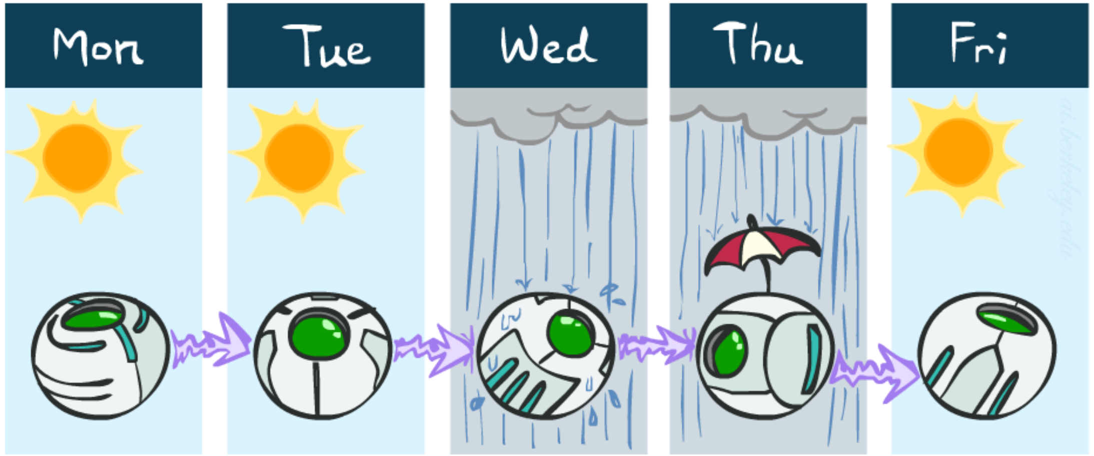
States {rain, sun}

- Initial distribution $P(X_0)$

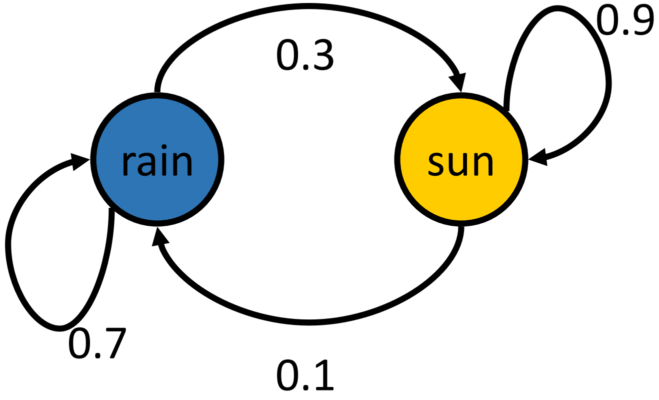
$P(X_0)$	
sun	rain
0.5	0.5

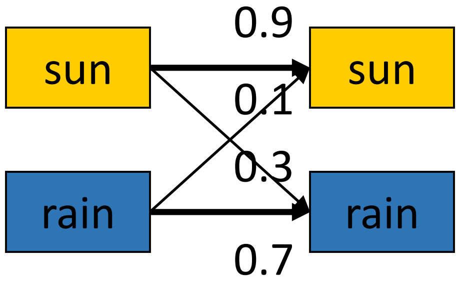
- Transition model $P(X_t \mid X_{t-1})$

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



Two new ways of representing the same CPT

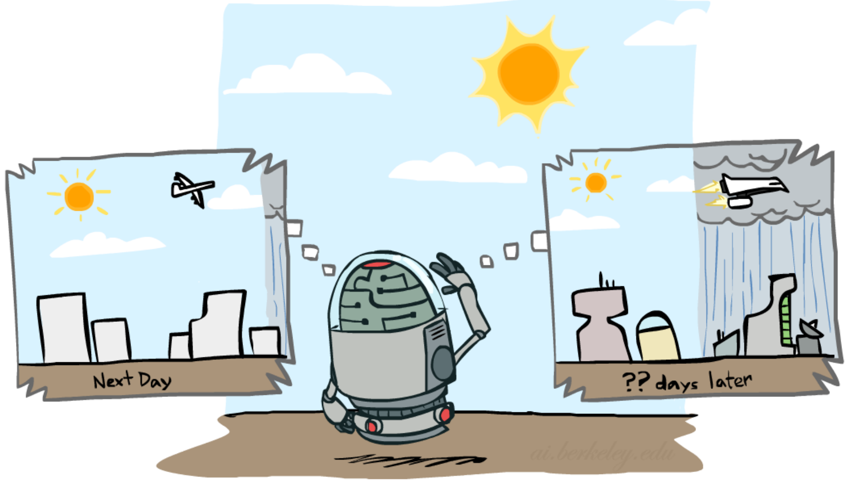




Weather prediction

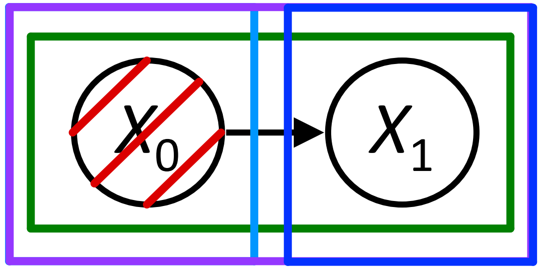
Time 0: $P(X_0)$ =<0.5,0.5>

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



What is the weather like at time 1?

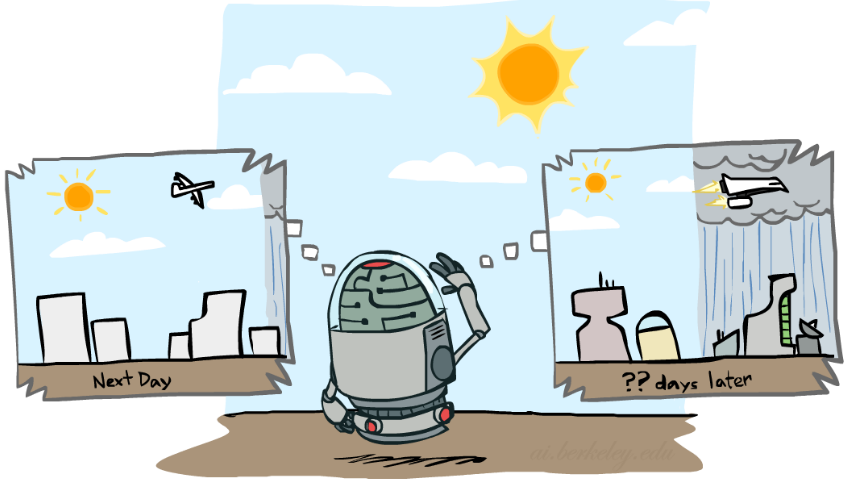
$$P(X_1) = \sum_{x_0} P(X_0=x_0, X_1)$$
$$= \sum_{x_0} P(X_1 \mid X_0=x_0)\, P(X_0=x_0)$$
$$= 0.5\langle 0.9,0.1\rangle + 0.5\langle 0.3,0.7\rangle$$
$$= \langle 0.6,0.4\rangle$$



Weather prediction, contd.

Time 1: $P(X_1) = <0.6, 0.4>$

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



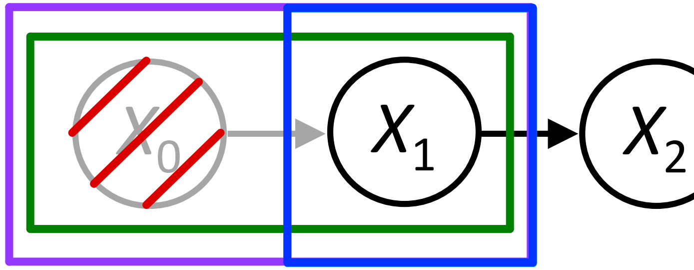
What is the weather like at time 2?

$$P(X_2) = \sum_{x_1} P(X_1 = x_1, X_2)$$

$$= \sum_{x_1} P(X_2 \mid X_1 = x_1)\, P(X_1 = x_1)$$

$$= 0.6<0.9, 0.1> + 0.4<0.3, 0.7>$$

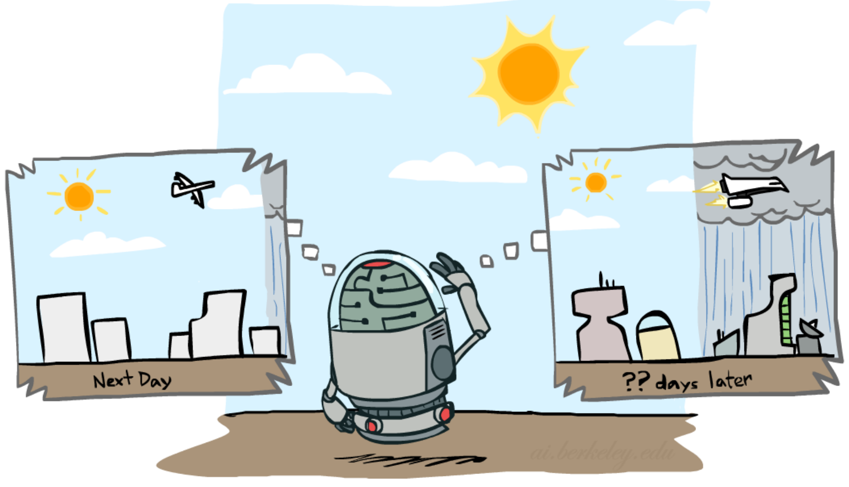
$$= <0.66, 0.34>$$



Weather prediction, contd.

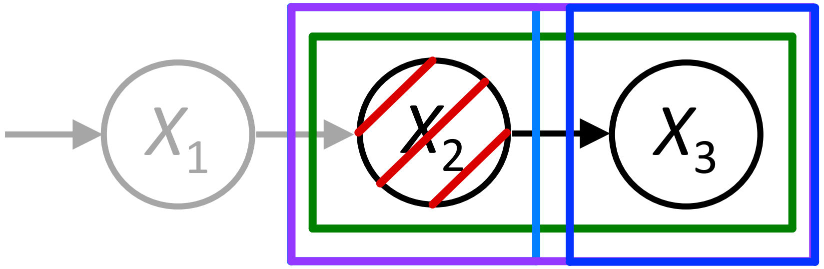
Time 2: $P(X_2) = <0.66,0.34>$

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



What is the weather like at time 3?

$$P(X_3) = \sum_{x_2} P(X_2=x_2, X_3)$$
$$= \sum_{x_2} P(X_3 \mid X_2=x_2)\, P(X_2=x_2)$$
$$= 0.66<0.9,0.1> + \; 0.34<0.3,0.7>$$
$$= <0.696,0.304>$$



Forward algorithm (simple form)

Transition model

Probability from previous iteration

What is the state at time *t*?

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

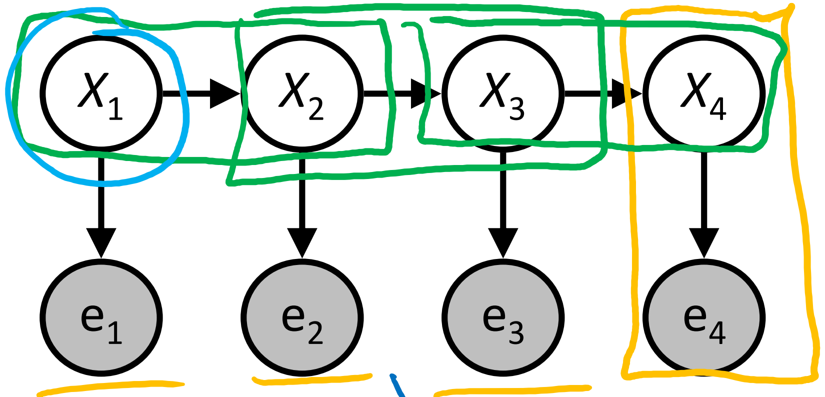
Iterate this update starting at *t*=0

Hidden Markov Models

HMM as a Bayes Net Warm-up

$P(X_3 | X_2)$

- For the following Bayes net, write the query $P(X_4 \mid e_{1:4})$ in terms of the conditional probability tables associated with the Bayes net.



$$P(X_4 \mid e_1, e_2, e_3, e_4) = \alpha P(X_4, e_1, e_2, e_3, e_4)$$

$$= \alpha \sum_{x_1} \sum_{x_2} \sum_{x_3} \underline{P(X_1, X_2, X_3, X_4, e_1, e_2, e_3, e_4)}$$

$P(e_4 | X_4)$

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

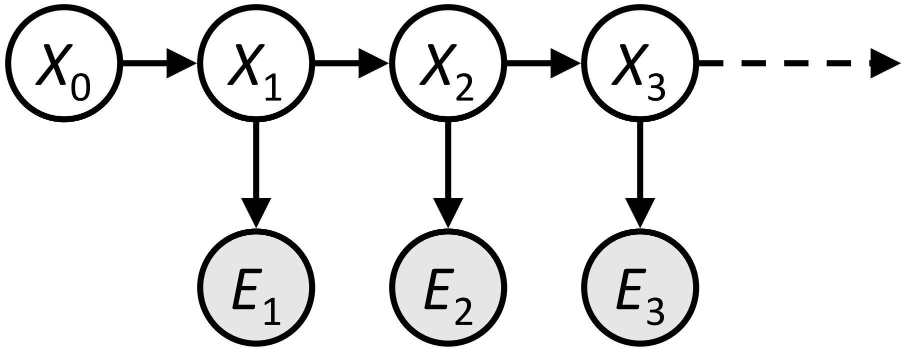
3 4 1

Hidden Markov Models

Usually the true state is not observed directly

Hidden Markov models (HMMs)

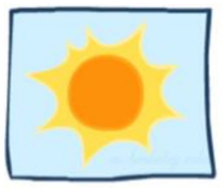
- Underlying Markov chain over states X
- You observe evidence E at each time step
- X_t is a single discrete variable; E_t may be continuous and may consist of several variables



Example: Weather HMM

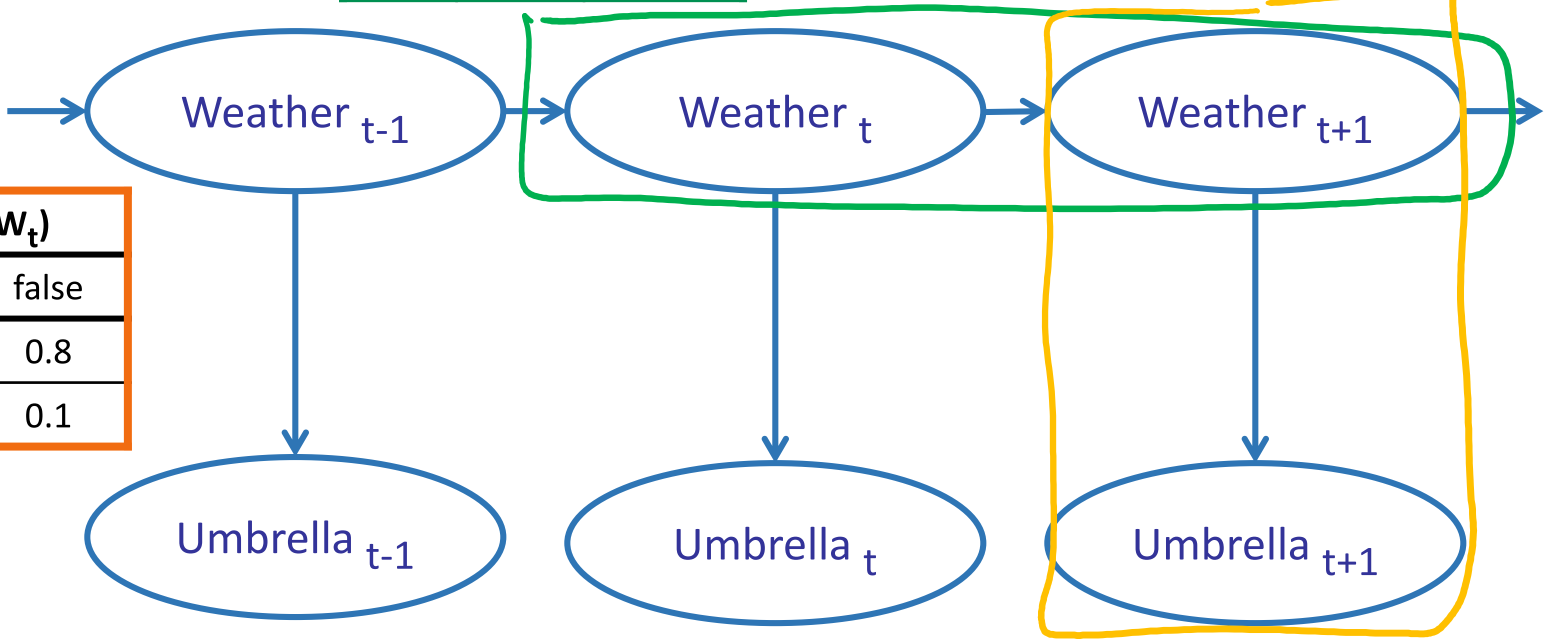
An HMM is defined by:

- Initial distribution: $P(X_0)$
- Transition model: $P(X_t \mid X_{t-1})$
- Sensor model: $P(E_t \mid X_t)$



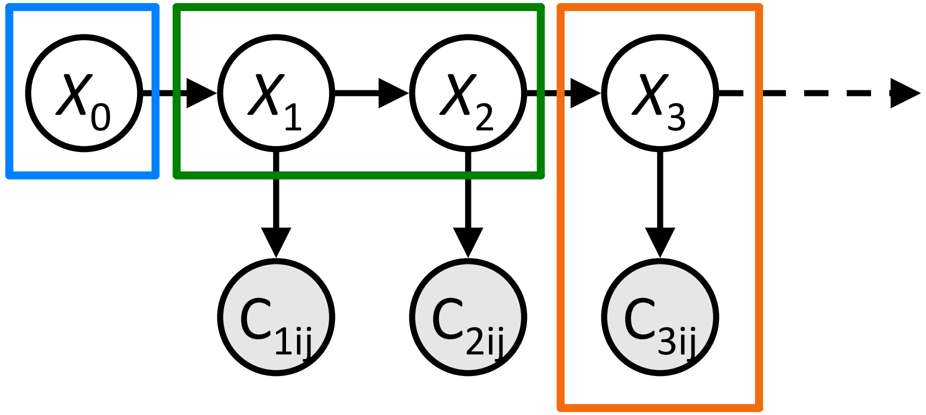
W_{t-1}	$P(W_t \mid W_{t-1})$	
	sun	rain
sun	0.9	0.1
rain	0.3	0.7

W_t	$P(U_t \mid W_t)$	
	true	false
sun	0.2	0.8
rain	0.9	0.1



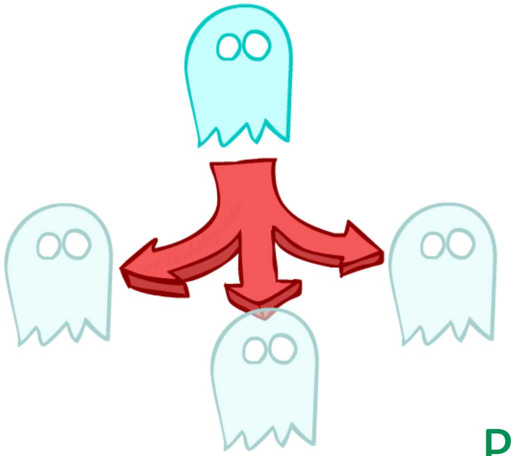
Example: Ghostbusters HMM

- State: location of moving ghost
- Observations: Color recorded by ghost sensor at clicked squares
- $P(X_0)$ = uniform
- $P(X_t \mid X_{t-1})$ = usually move clockwise, but sometimes move randomly or stay in place
- $P(C_{tij} \mid X_t)$ = same sensor model as before: red means close, green means far away.



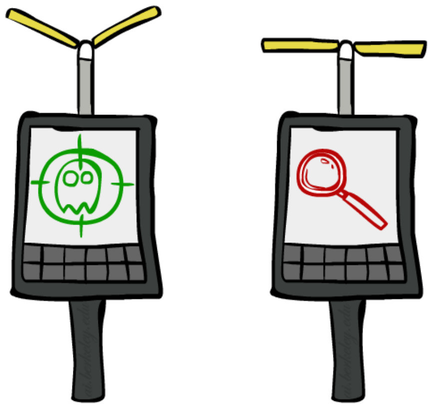
1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

$P(X_1)$



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

$P(X_2 \mid X_1=(2,3))$



[Demo: Ghostbusters – Circular Dynamics – HMM (L14D2)]

HMM as Probability Model

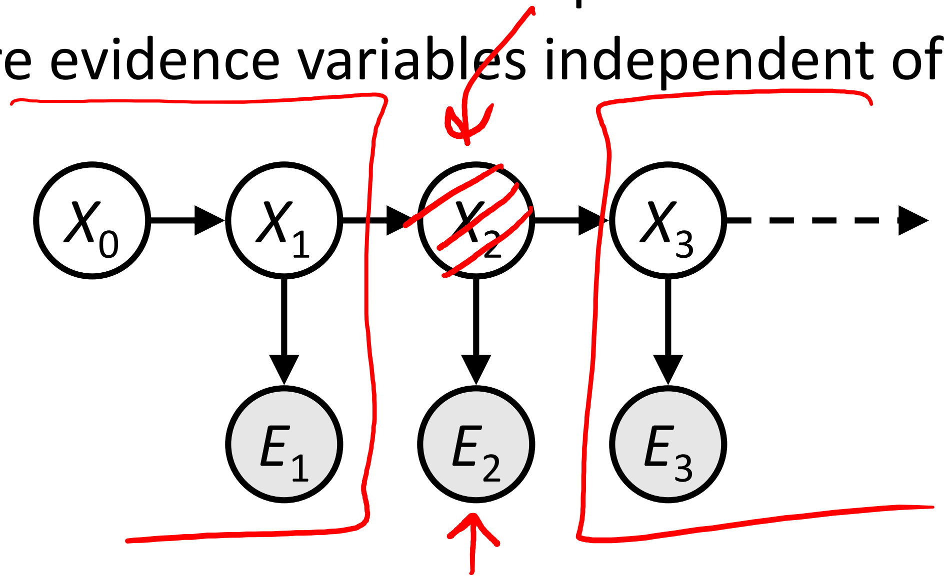
- Joint distribution for Markov model:

$$P(X_0,\dots, X_T) = P(X_0) \prod_{t=1:T} P(X_t \mid X_{t-1})$$

- Joint distribution for hidden Markov model:

$$P(X_0, \underline{X_1,E_1, \dots, X_T,E_T}) = P(X_0) \prod_{t=1:T} P(X_t \mid X_{t-1})\, P(E_t \mid X_t)$$

- Future states are independent of the past given the present
- Current evidence is independent of everything else given the current state
- Are evidence variables independent of each other?



Useful notation: $X_{a:b} = X_a, X_{a+1}, \dots, X_b$

For example: $P(X_{1:2} \mid e_{1:3}) = P(X_1, X_2, \mid e_1, e_2, e_3)$

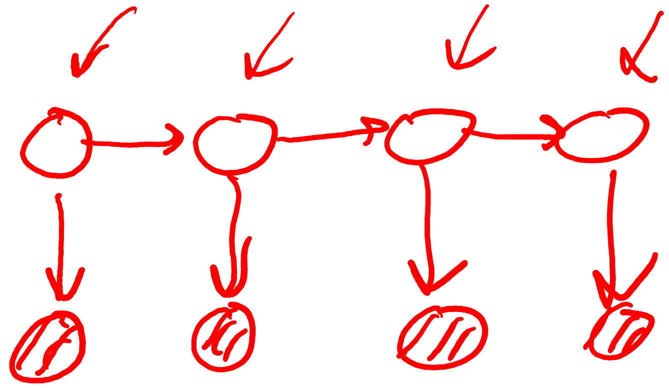
Real HMM Examples

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:

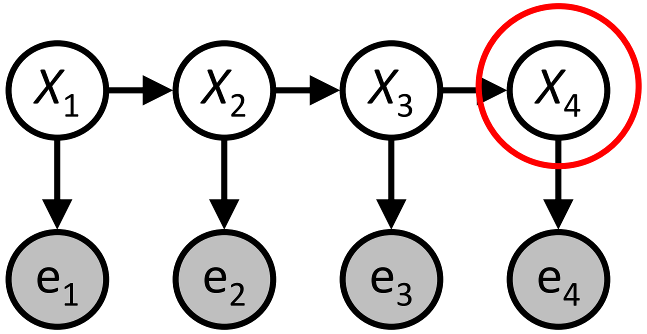
- Observations are range readings (continuous)
- States are positions on a map (continuous)

Molecular biology:

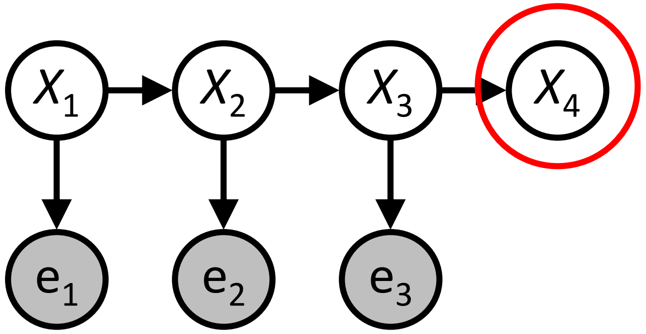
- Observations are nucleotides ACGT
- States are coding/non-coding/start/stop/splice-site etc.

Other HMM Queries

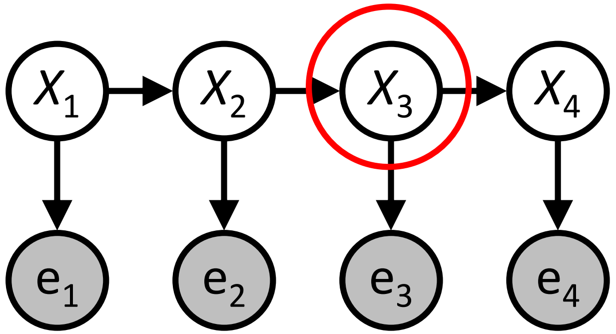
Filtering: $P(X_t|e_{1:t})$



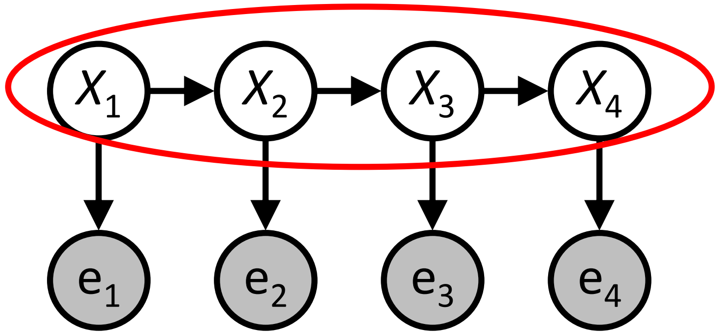
Prediction: $P(X_{t+k}|e_{1:t})$



Smoothing: $P(X_k|e_{1:t})$, $k<t$



Explanation: $P(X_{1:t}|e_{1:t})$



Inference Tasks

Filtering: $P(X_t | e_{1:t})$

- belief state—input to the decision process of a rational agent

Prediction: $P(X_{t+k} | e_{1:t})$ for $k > 0$

- evaluation of possible action sequences; like filtering without the evidence

Smoothing: $P(X_k | e_{1:t})$ for $0 \le k < t$

- better estimate of past states, essential for learning

Most likely explanation: $\text{argmax}_{x_{1:t}} P(x_{1:t} \mid e_{1:t})$

- speech recognition, decoding with a noisy channel

Pacman – Hunting Invisible Ghosts with Sonar

[Demo: Pacman – Sonar – No Beliefs(L14D1)]

Filtering Algorithm

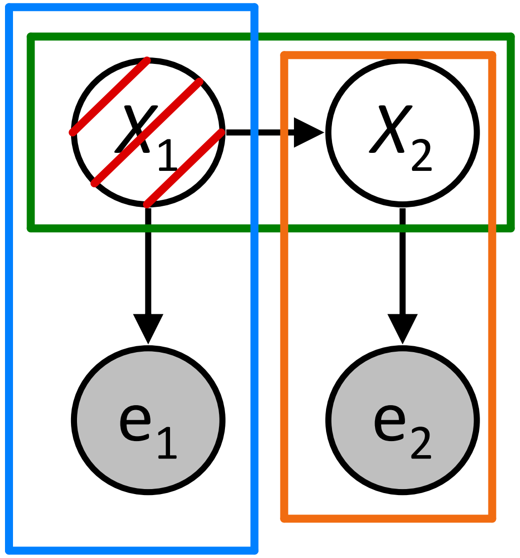
$$\underbrace{P(X_{t+1} \mid e_{1:t+1})} = \underbrace{\alpha}_{\text{Normalize}} \underbrace{P(e_{t+1} \mid X_{t+1})}_{\text{Update}} \underbrace{\sum_{x_t} P(X_{t+1} \mid x_t)\, P(x_t \mid e_{1:t})}_{\text{Predict}}$$

$$\mathbf{f}_{1:t+1} = \text{FORWARD}(\mathbf{f}_{1:t}, e_{t+1})$$

Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

Marching **forward** through the HMM network

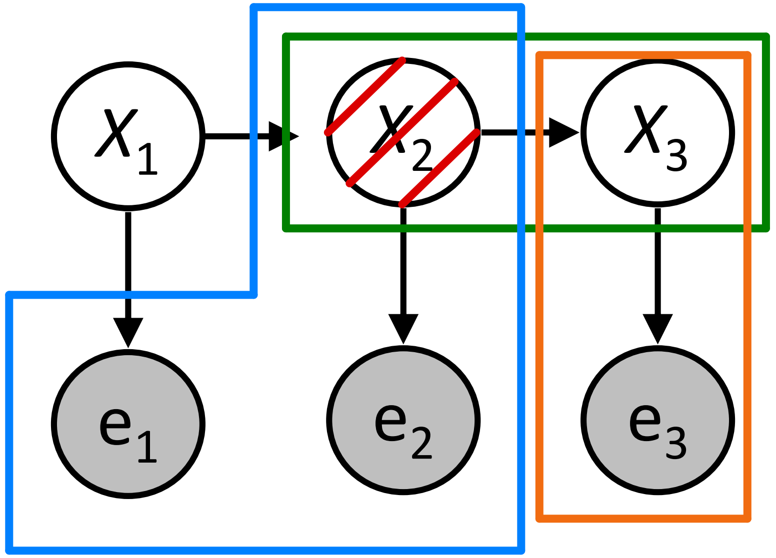


$P(X_1 | e_1) \rightarrow P(X_2 | e_1) \rightarrow P(X_2 | e_1, e_2) \rightarrow P(X_3 | e_1, e_2$

Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

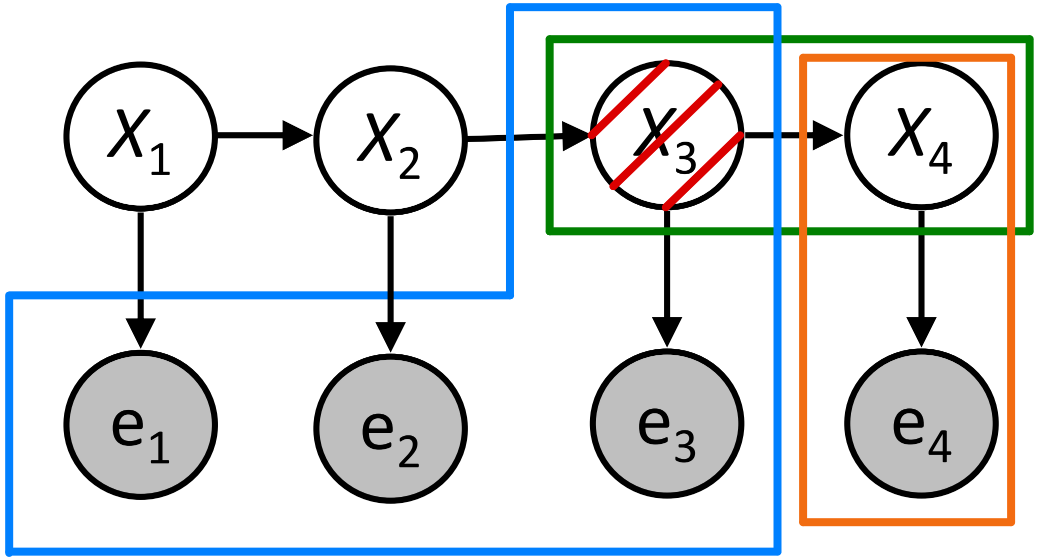
Marching **forward** through the HMM network



Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

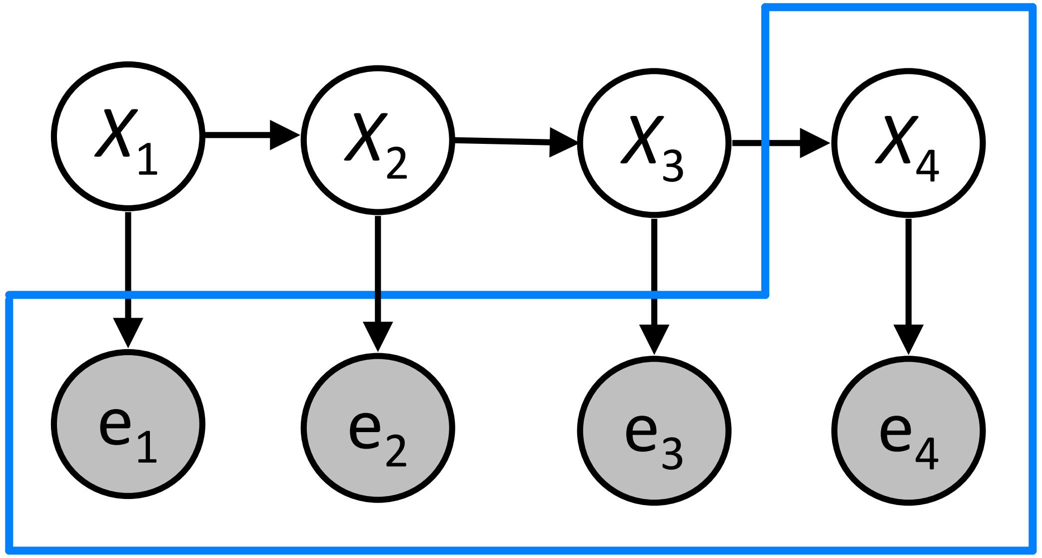
Marching **forward** through the HMM network



Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

Marching **forward** through the HMM network



Example: Prediction step

As time passes, uncertainty "accumulates"

(Transition model: ghosts usually go clockwise)

<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

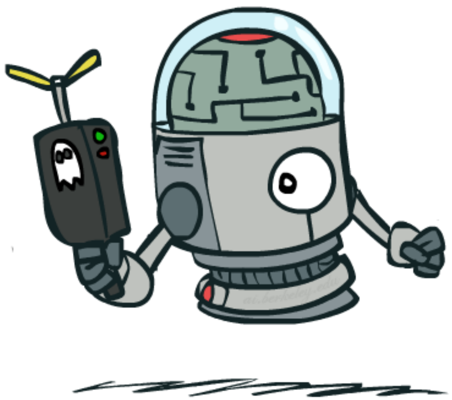
T = 1

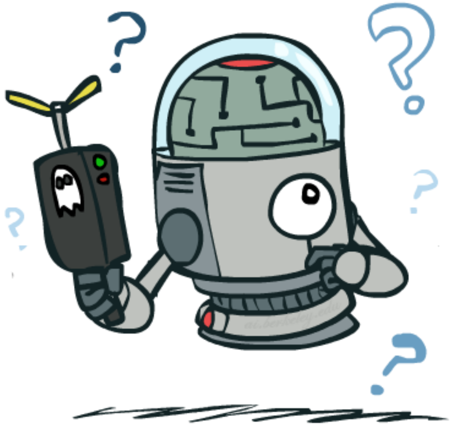
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	0.06	<0.01	<0.01	<0.01
<0.01	0.76	0.06	0.06	<0.01	<0.01
<0.01	<0.01	0.06	<0.01	<0.01	<0.01

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

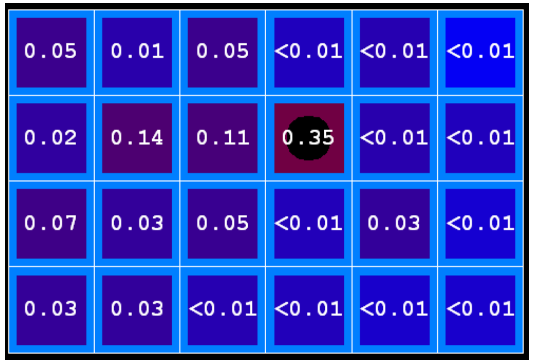
T = 5



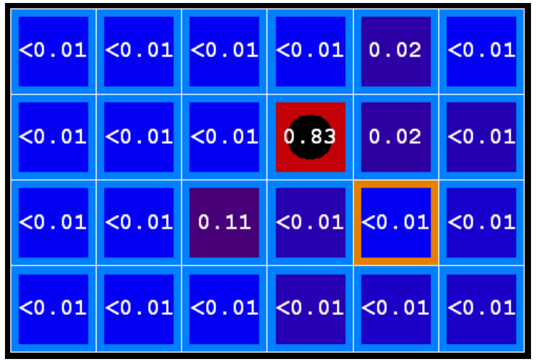


Example: Update step

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



Before observation



After observation

Demo Ghostbusters – Circular Dynamics -- HMM

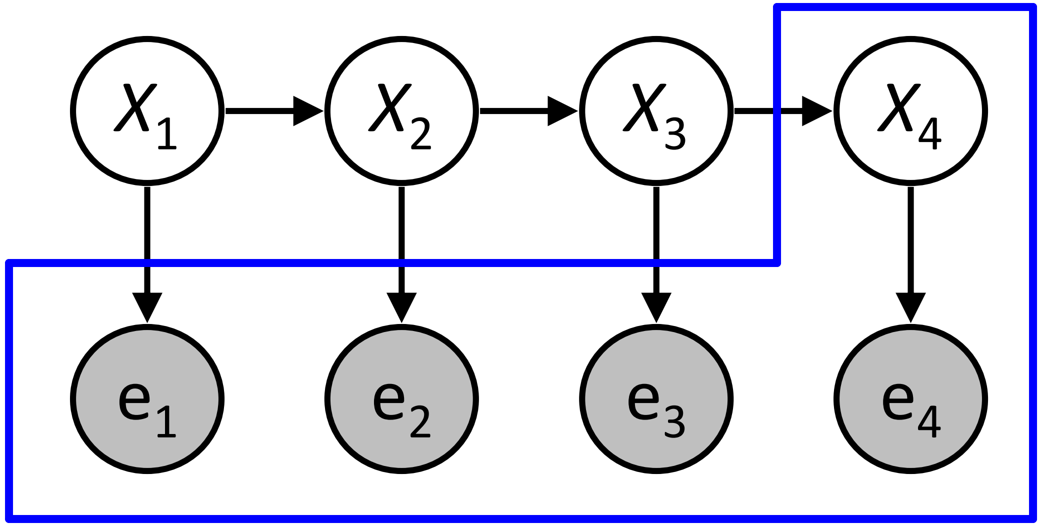
Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

$$P(X_t \mid e_{1:t}) = P(X_t \mid e_t, e_{1:t-1})$$

$$= \alpha\, P(X_t, e_t \mid e_{1:t-1})$$



Def. of cond. probability with X_t, e_t

Filtering **Algorithm**

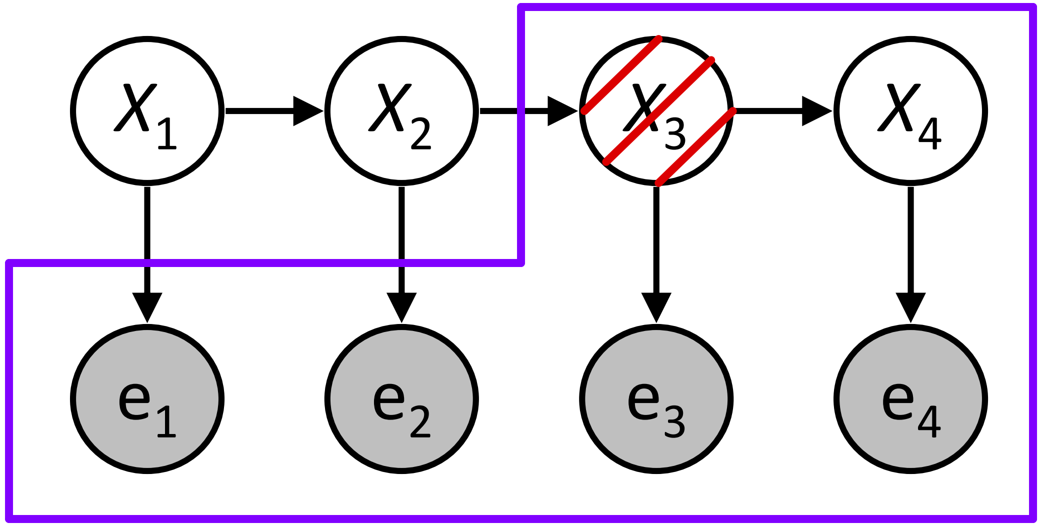
Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

$$P(X_t \mid e_{1:t}) = P(X_t \mid e_t, e_{1:t-1})$$

$$= \alpha\, P(X_t, \underline{e_t} \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} \underbrace{P(x_{t-1}, X_t, e_t \mid e_{1:t-1})}$$



Summation over variable X_{t-1}

Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

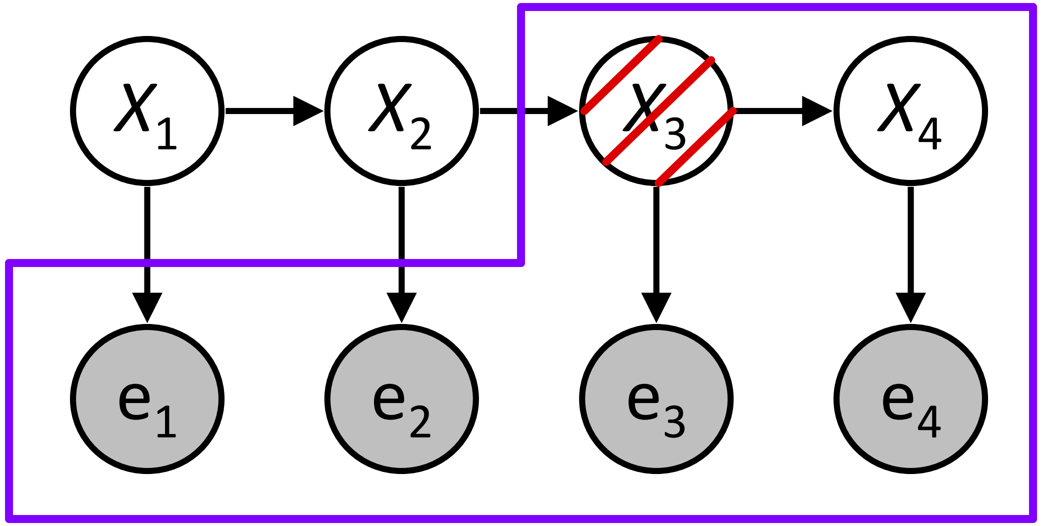
$$P(X_t \mid e_{1:t}) = P(X_t \mid e_t, e_{1:t-1})$$

$$= \alpha\, P(X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1}, X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1} \mid e_{1:t-1})\, P(X_t \mid x_{t-1}, e_{1:t-1})\, P(e_t \mid X_t, x_{t-1}, e_{1:t-1})$$

Chain rule with x_{t-1}, X_t, and e_t



Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

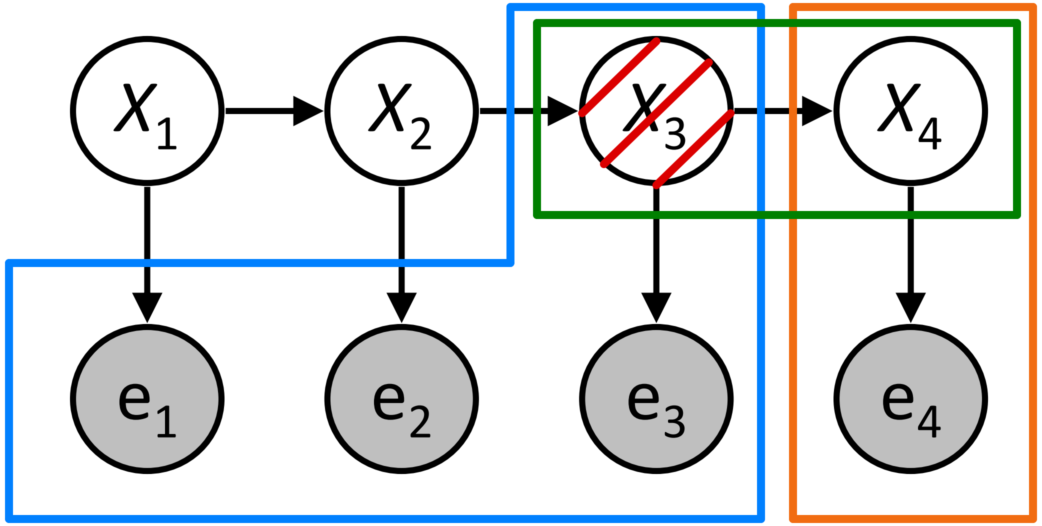
Chain rule with x_{t-1}, X_t, and e_t

$$P(X_t \mid e_{1:t}) = P(X_t \mid e_t, e_{1:t-1})$$

$$= \alpha\, P(X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1}, X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1} \mid e_{1:t-1})\, P(X_t \mid x_{t-1}, e_{1:t-1})\, P(e_t \mid X_t, x_{t-1}, e_{1:t-1})$$



Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

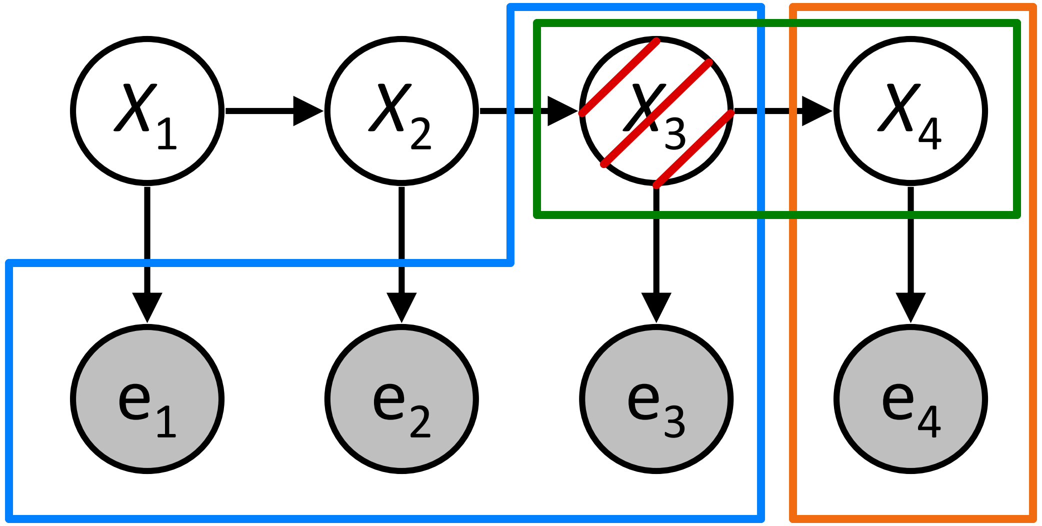
Matching math with Bayes net

$$P(X_t \mid e_{1:t}) = P(X_t \mid e_t, e_{1:t-1})$$

$$= \alpha\, P(X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1}, X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1} \mid e_{1:t-1})\, P(X_t \mid x_{t-1})\, P(e_t \mid X_t)$$



Bayes net conditional independence

Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

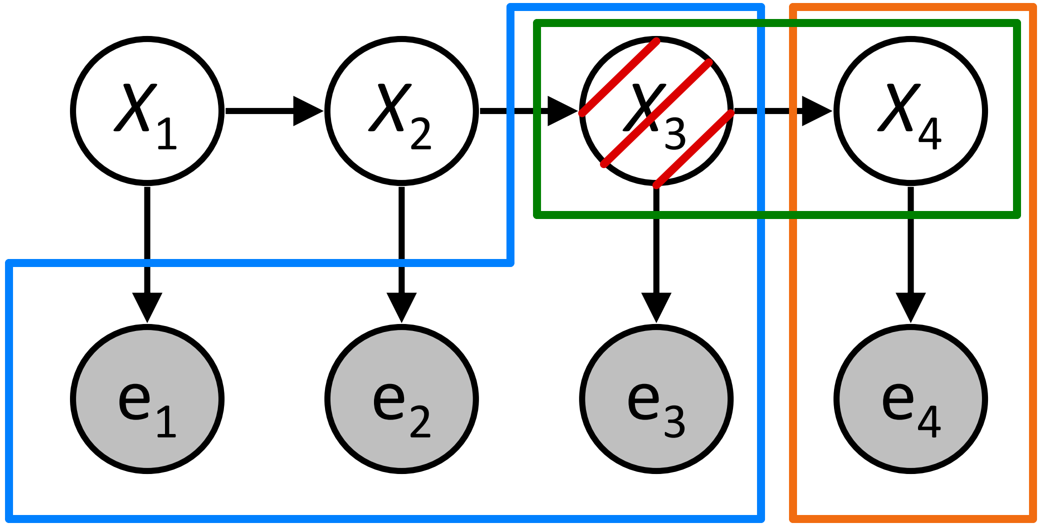
$$\boxed{P(X_t \mid e_{1:t})} = P(X_t \mid e_t, e_{1:t-1})$$

$$= \alpha\, P(X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1}, X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(\underline{x_{t-1}} \mid e_{1:t-1})\, P(X_t \mid \underline{x_{t-1}})\, P(e_t \mid X_t)$$

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



Pulling $P(e_t \mid X_t)$ out of the summation

Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net

$$P(X_t \mid e_{1:t}) = P(X_t \mid e_t, e_{1:t-1})$$

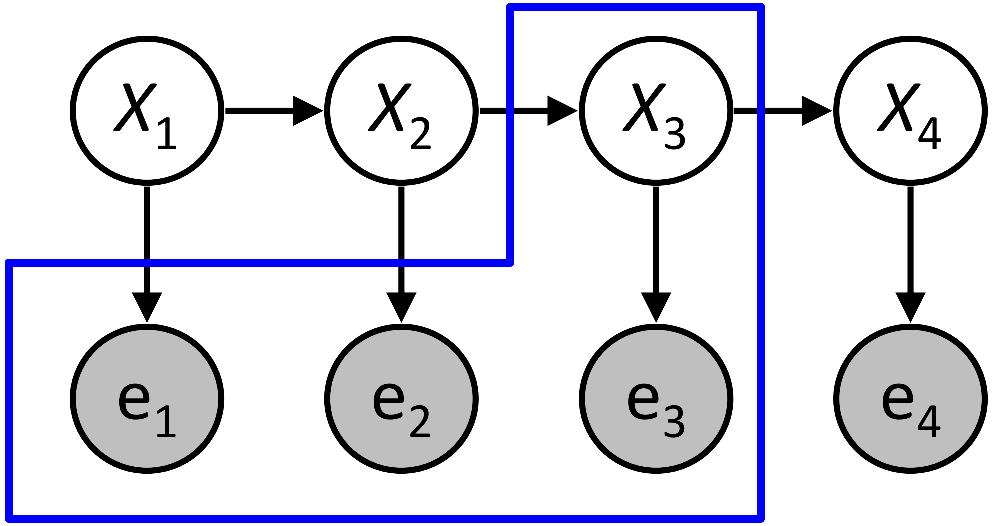
$$= \alpha\, P(X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1}, X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1} \mid e_{1:t-1})\, P(X_t \mid x_{t-1})\, P(e_t \mid X_t)$$

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

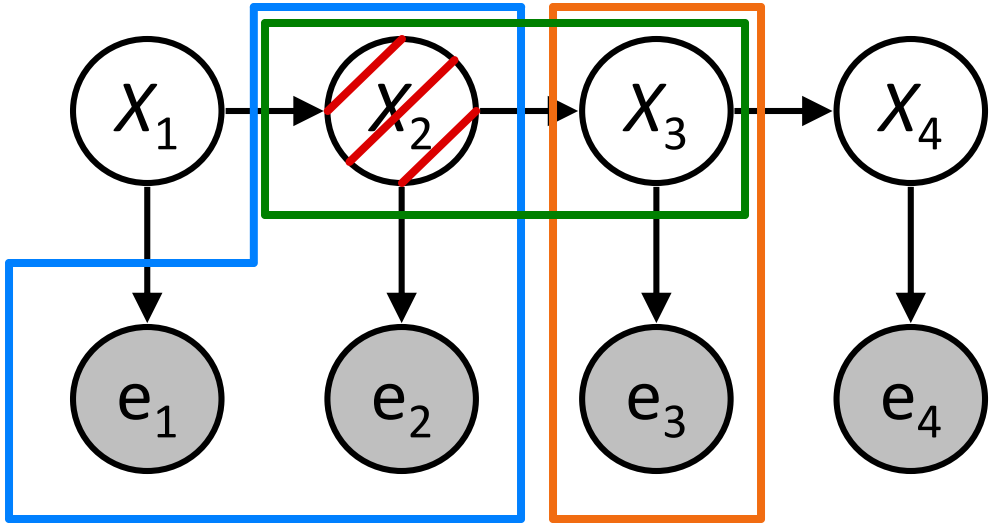
Recursion!



Filtering **Algorithm**

Query: What is the current state, given all of the current and past evidence?

Matching math with Bayes net



$$P(X_t \mid e_{1:t}) = P(X_t \mid e_t, e_{1:t-1})$$

$$= \alpha\, P(X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1}, X_t, e_t \mid e_{1:t-1})$$

$$= \alpha \sum_{x_{t-1}} P(x_{t-1} \mid e_{1:t-1})\, P(X_t \mid x_{t-1})\, P(e_t \mid X_t)$$

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

Recursion!

Filtering Algorithm

$$P(X_{t+1} \mid e_{1:t+1}) = \underbrace{\alpha}_{\text{Normalize}} \underbrace{P(e_{t+1} \mid X_{t+1})}_{\text{Update}} \underbrace{\textstyle\sum_{x_t} P(X_{t+1} \mid x_t)\, P(x_t \mid e_{1:t})}_{\text{Predict}}$$

$\boldsymbol{f}_{1:t+1} = \text{FORWARD}(\boldsymbol{f}_{1:t}, e_{t+1})$

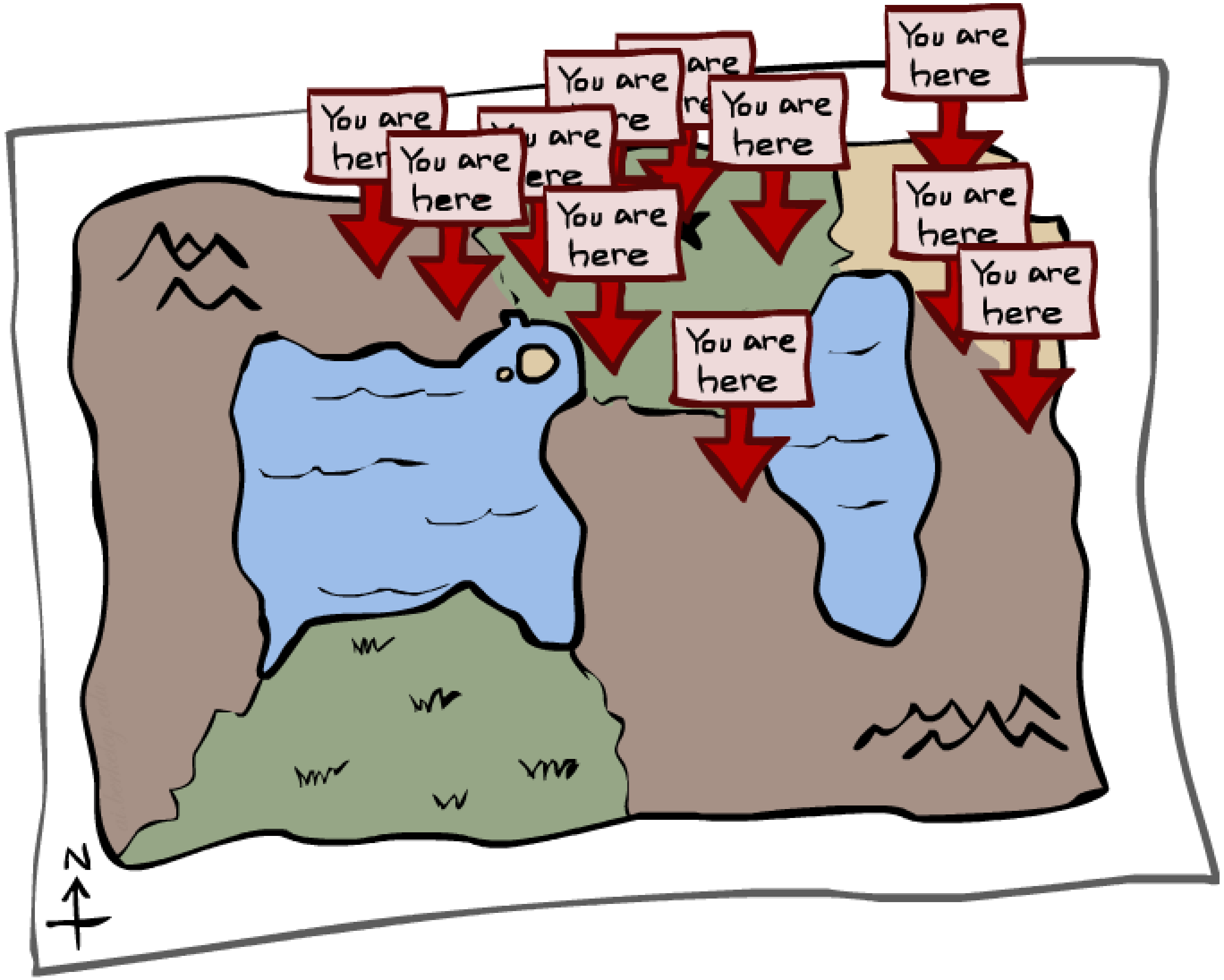
Cost per time step: $O(|X|^2)$ where $|X|$ is the number of states

Time and space costs are ***constant***, independent of t

$O(|X|^2)$ is infeasible for models with many state variables

We get to invent really cool approximate filtering algorithms

Particle Filtering

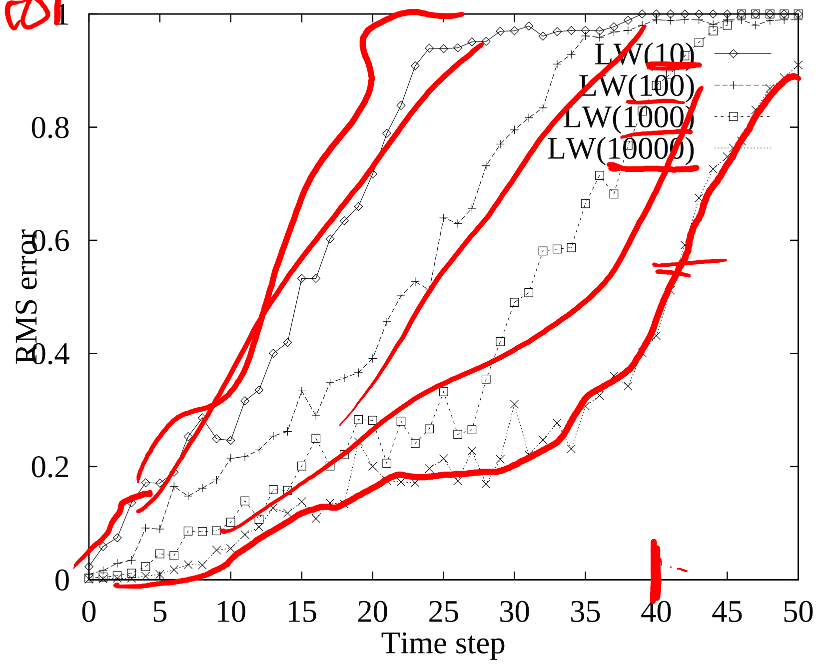


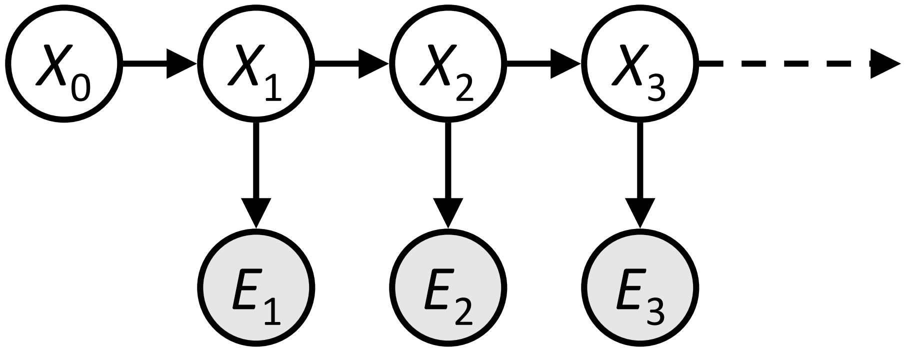
We need a new algorithm!

When $|X|$ is more than 10^6 or so (e.g., 3 ghosts in a 10x20 world), exact inference becomes infeasible

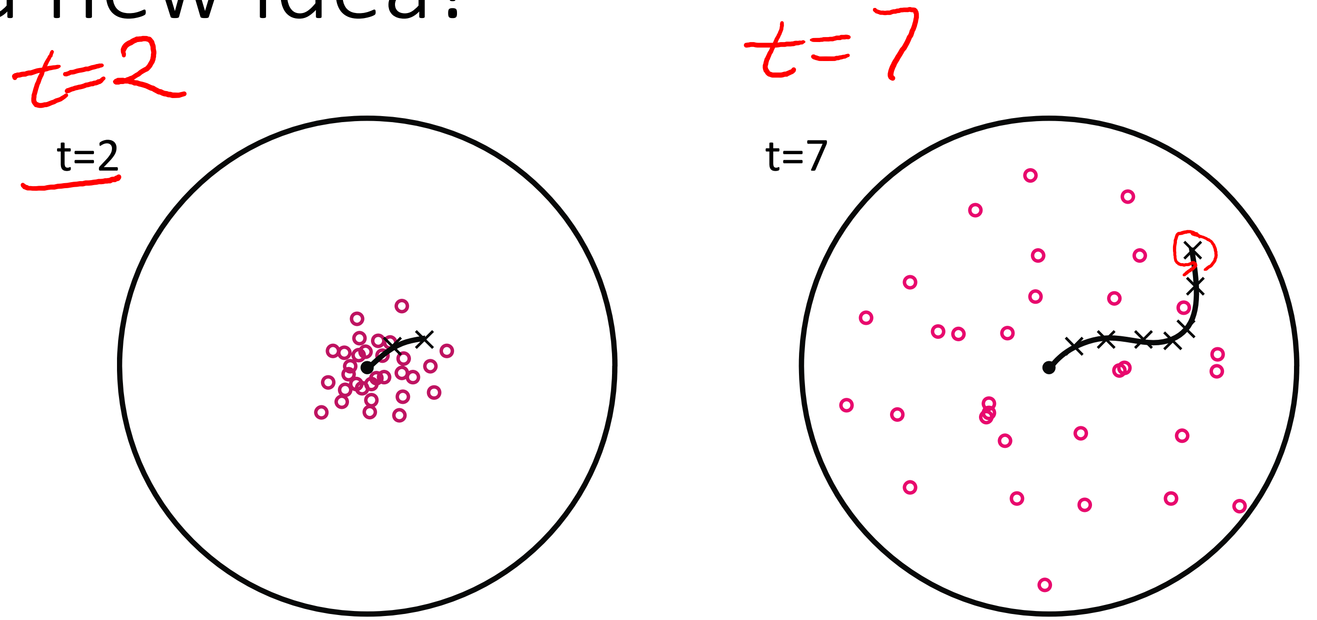
Likelihood weighting fails completely – number of samples needed grows ***exponentially*** with T

error





We need a new idea!



The problem: sample state trajectories go off into low-probability regions, ignoring the evidence; too few “reasonable” samples

Solution: kill the bad ones, make more of the good ones

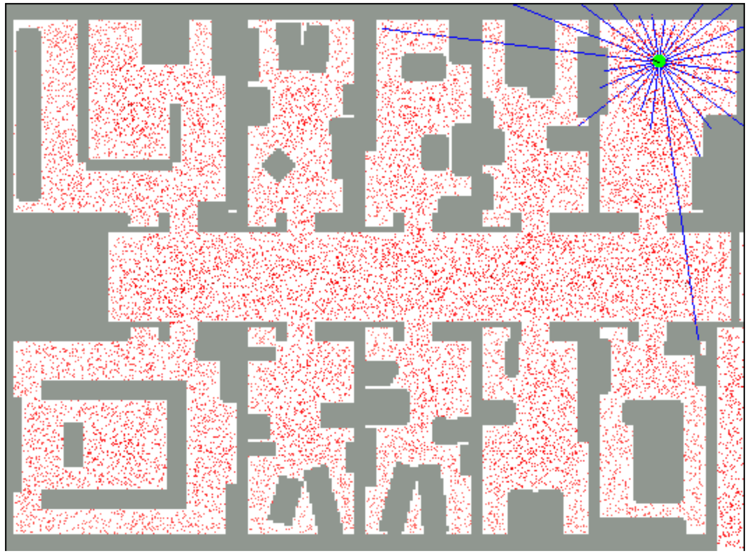
This way the population of samples stays in the high-probability region

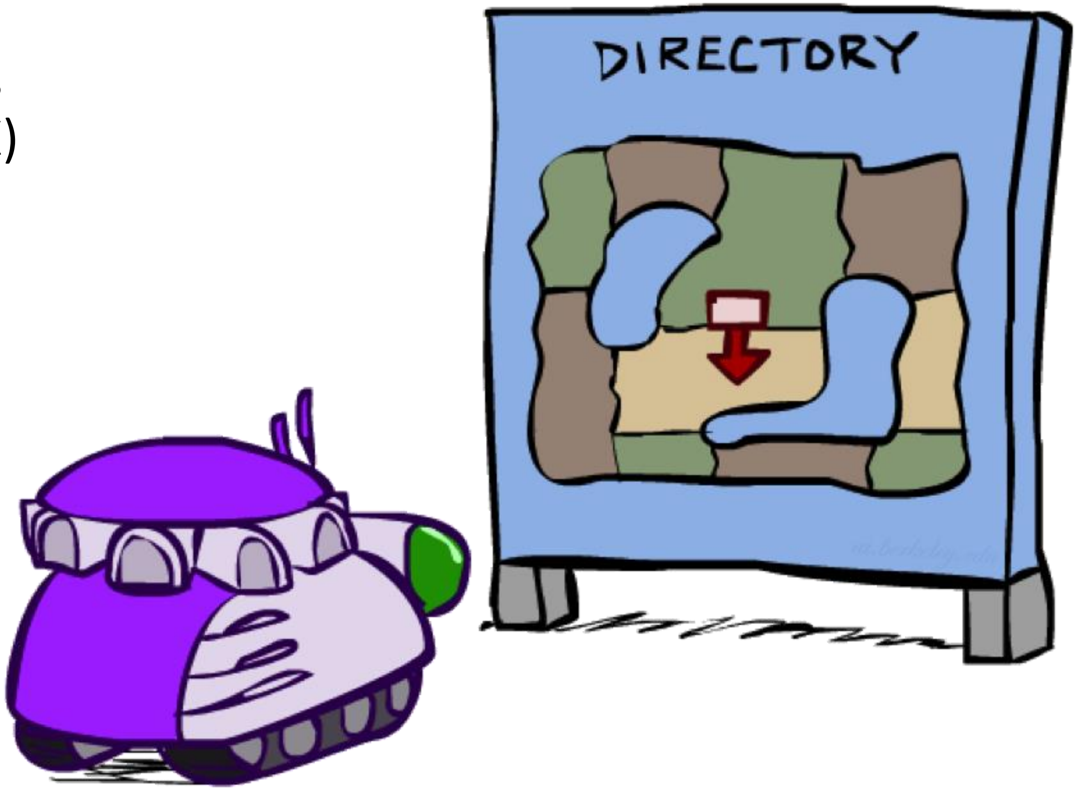
This is called resampling or survival of the fittest

Robot Localization

In robot localization:

- We know the map, but not the robot's position
- Observations may be vectors of range finder readings
- State space and readings are typically continuous (works basically like a very fine grid) and so we cannot store B(X)
- Particle filtering is a main technique





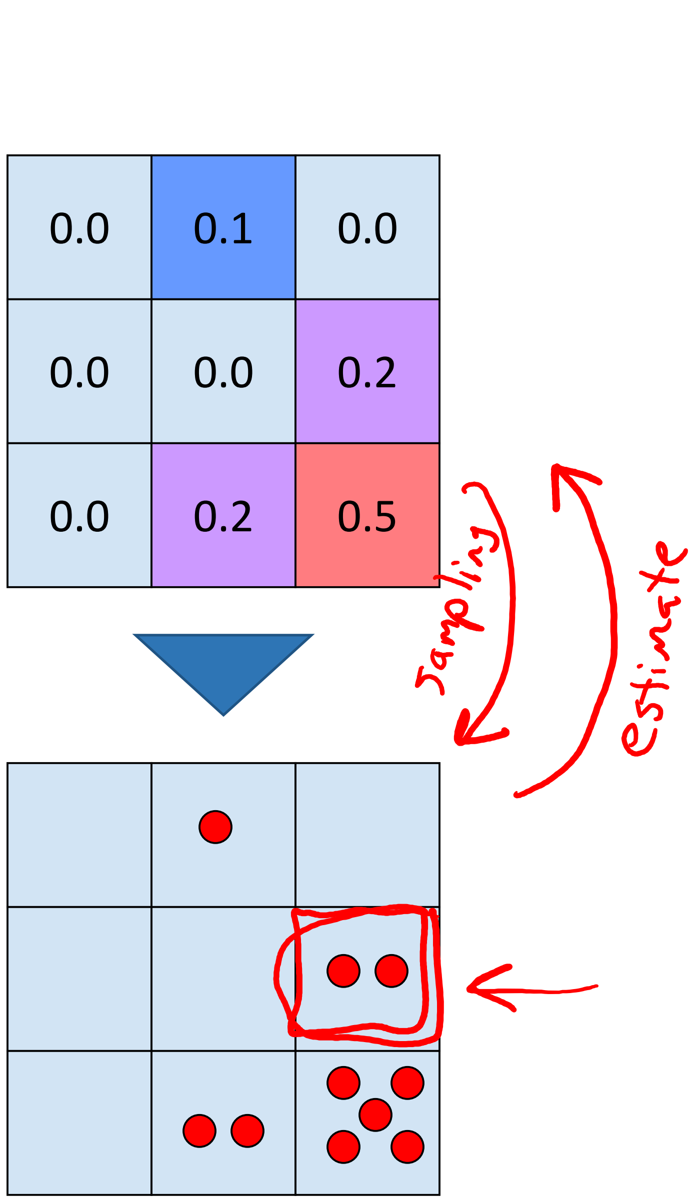
Particle Filter Localization (Sonar)

[Dieter Fox, et al.]

[Video: global-sonar-uw-annotated.avi]

Particle Filtering

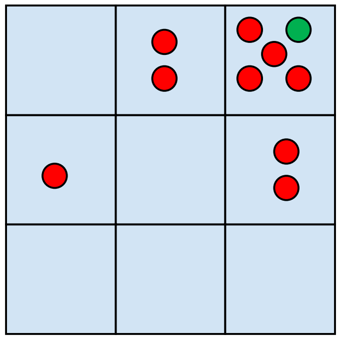
- Represent belief state by a set of samples
  - Samples are called ***particles***
  - Time per step is linear in the number of samples
  - But: number needed may be large
- This is how robot localization works in practice



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 may have P(x) = 0!
- More particles, more accuracy
- Usually we want a low-dimensional marginal
  - E.g., “Where is ghost 1?” rather than “Are ghosts 1,2,3 in {2,6], [5,6], and [8,11]?”

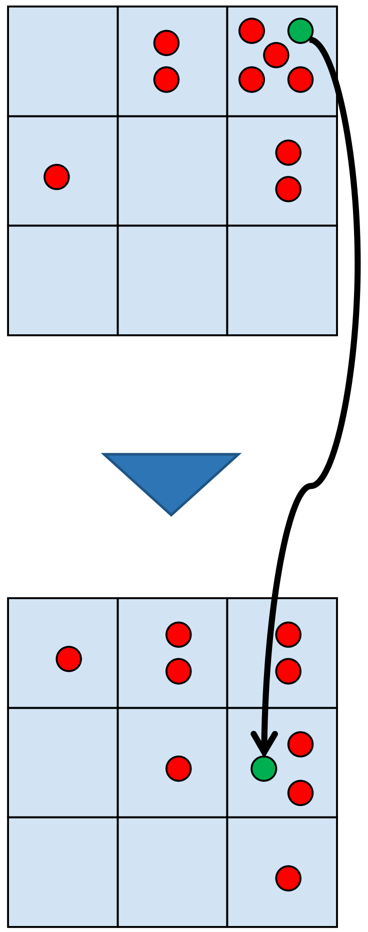
For now, all particles have a weight of 1

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

Particle Filtering: Propagate forward

- A particle in state x_t is moved by sampling its next position directly from the transition model:
  - $x_{t+1} \sim P(X_{t+1} \mid x_t)$
  - 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)

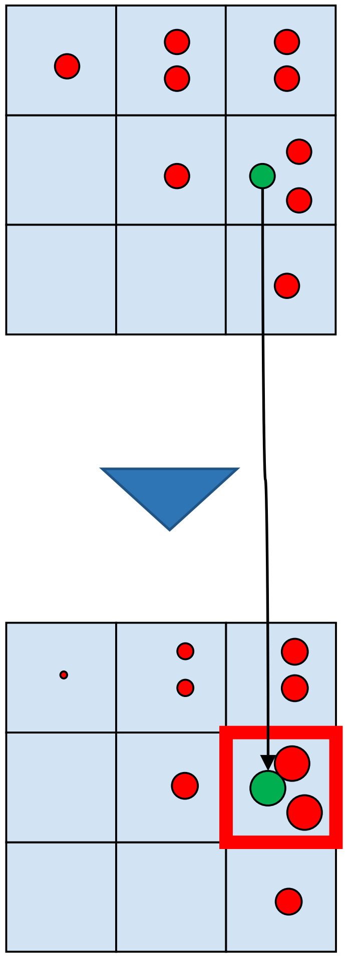


Particles:
(3,2)
(2,3)
(3,2)
(3,1)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(2,2)

Particle Filtering: Observe

- Slightly trickier:
  - Don't sample observation, fix it
  - Similar to likelihood weighting, weight samples based on the evidence
    - $W = P(e_t | x_t)$
  - Normalize the weights: particles that fit the data better get higher weights, others get lower weights

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

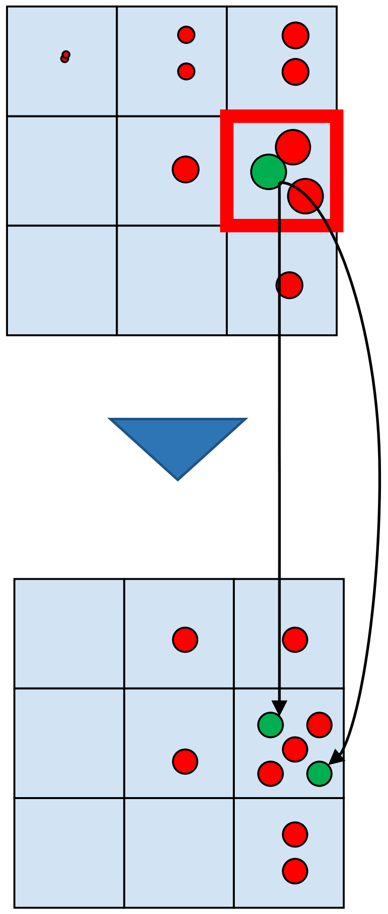
Particle Filtering: Resample

Rather than tracking weighted samples, we ***resample***

N times, we choose from our weighted sample distribution (i.e., draw with replacement)

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



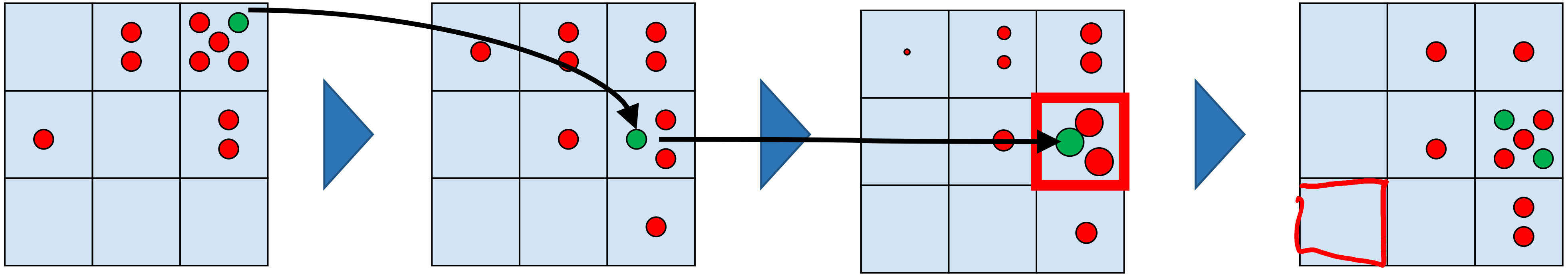
(New) Particles:
(3,2)
(2,2)
(3,2)
(2,3)
(3,3)
(3,2)
(1,3)
(2,3)
(3,2)
(3,2)

Summary: Particle Filtering

Particles: track samples of states rather than an explicit distribution

$P(X_{t+1})$

Propagate forward | Weight | Resample

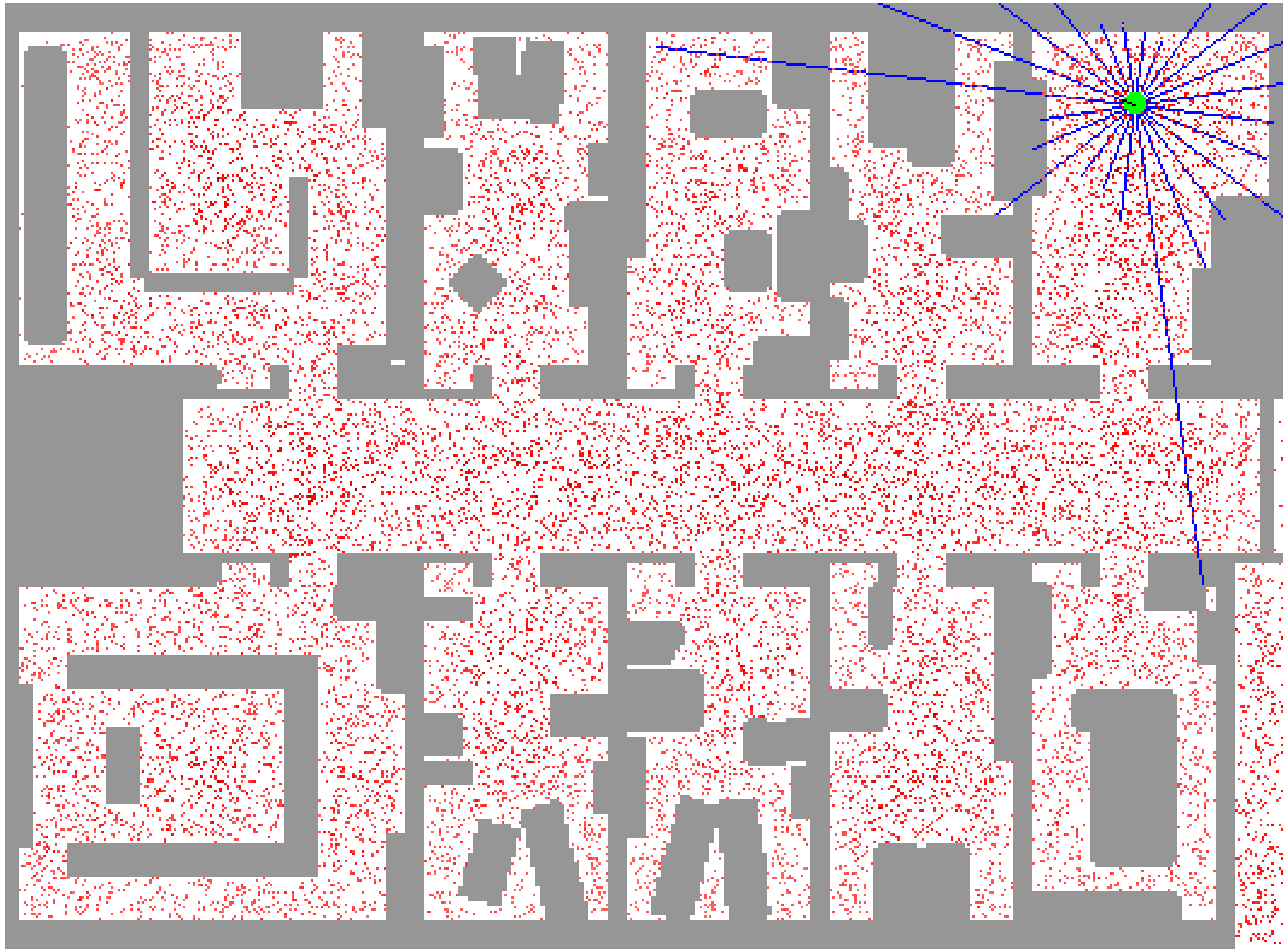


Particles:	Particles:	Particles:	(New) Particles:
(3,3)	(3,2)	(3,2) w=.9	(3,2)
(2,3)	(2,3)	(2,3) w=.2	(2,2)
(3,3)	(3,2)	(3,2) w=.9	(3,2)
(3,2)	(3,1)	(3,1) w=.4	(2,3)
(3,3)	(3,3)	(3,3) w=.4	(3,3)
(3,2)	(3,2)	(3,2) w=.9	(3,2)
(1,2)	(1,3)	(1,3) w=.1	(1,3)
(3,3)	(2,3)	(2,3) w=.2	(2,3)
(3,3)	(3,2)	(3,2) w=.9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

Consistency: see proof in AIMA Ch. 15

[Demos: ghostbusters particle filtering (L15D3,4,5)]

Particle Filter Localization (Laser)



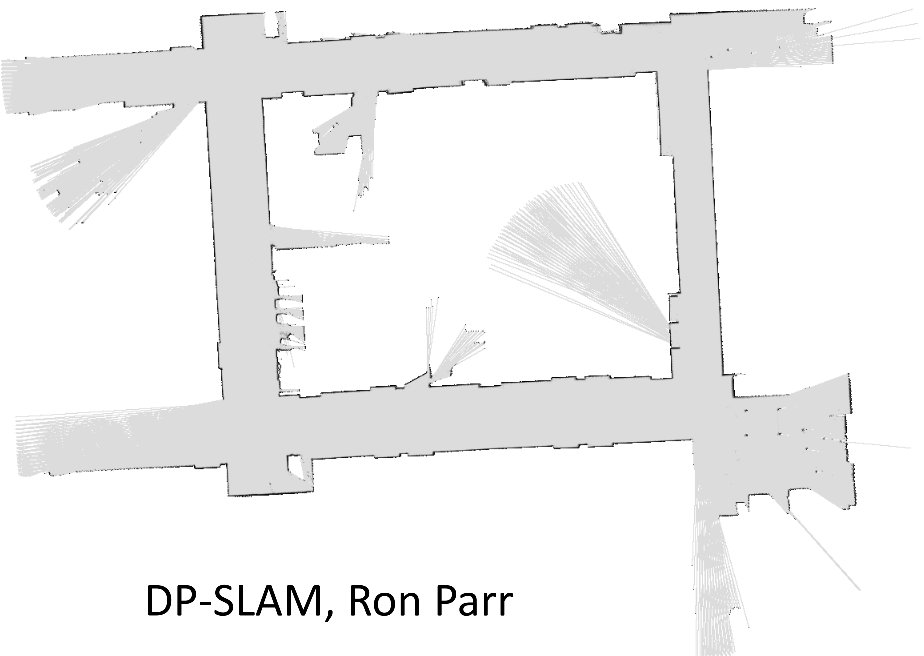
[Dieter Fox, et al.]

[Video: global-floor.gif]

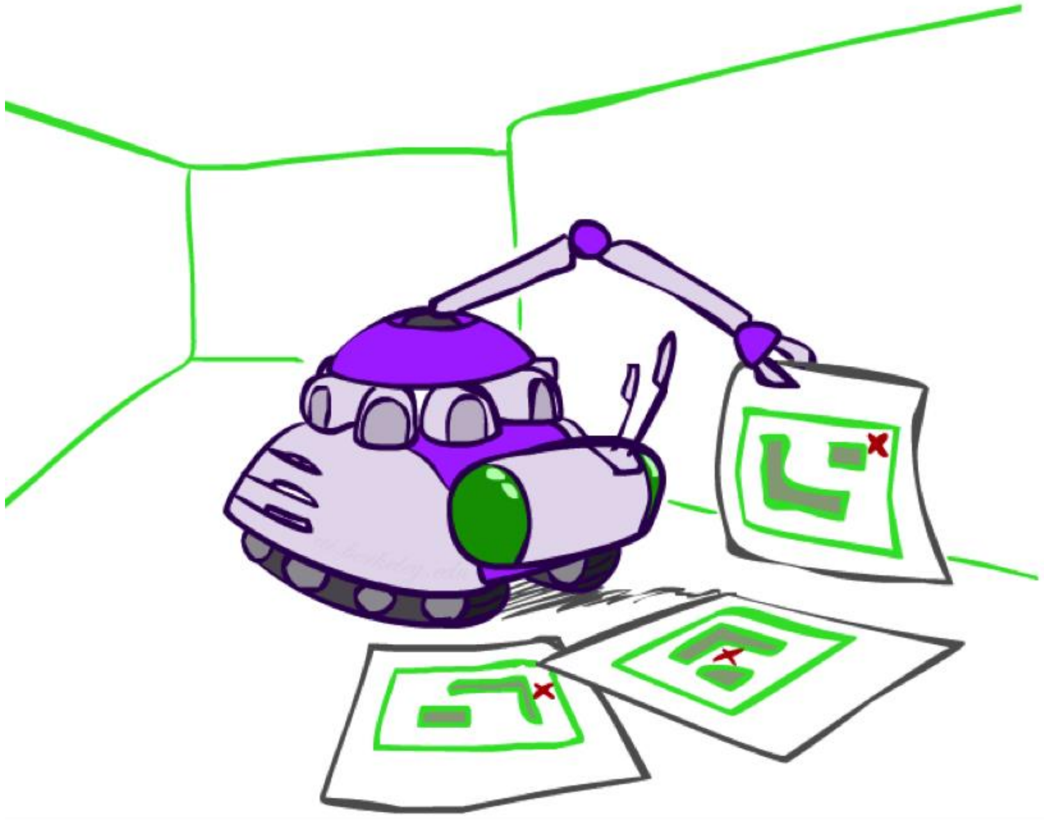
Robot Mapping

SLAM: Simultaneous Localization And Mapping

- We do not know the map or our location
- State consists of position AND map!
- Main techniques: Kalman filtering (Gaussian HMMs) and particle methods

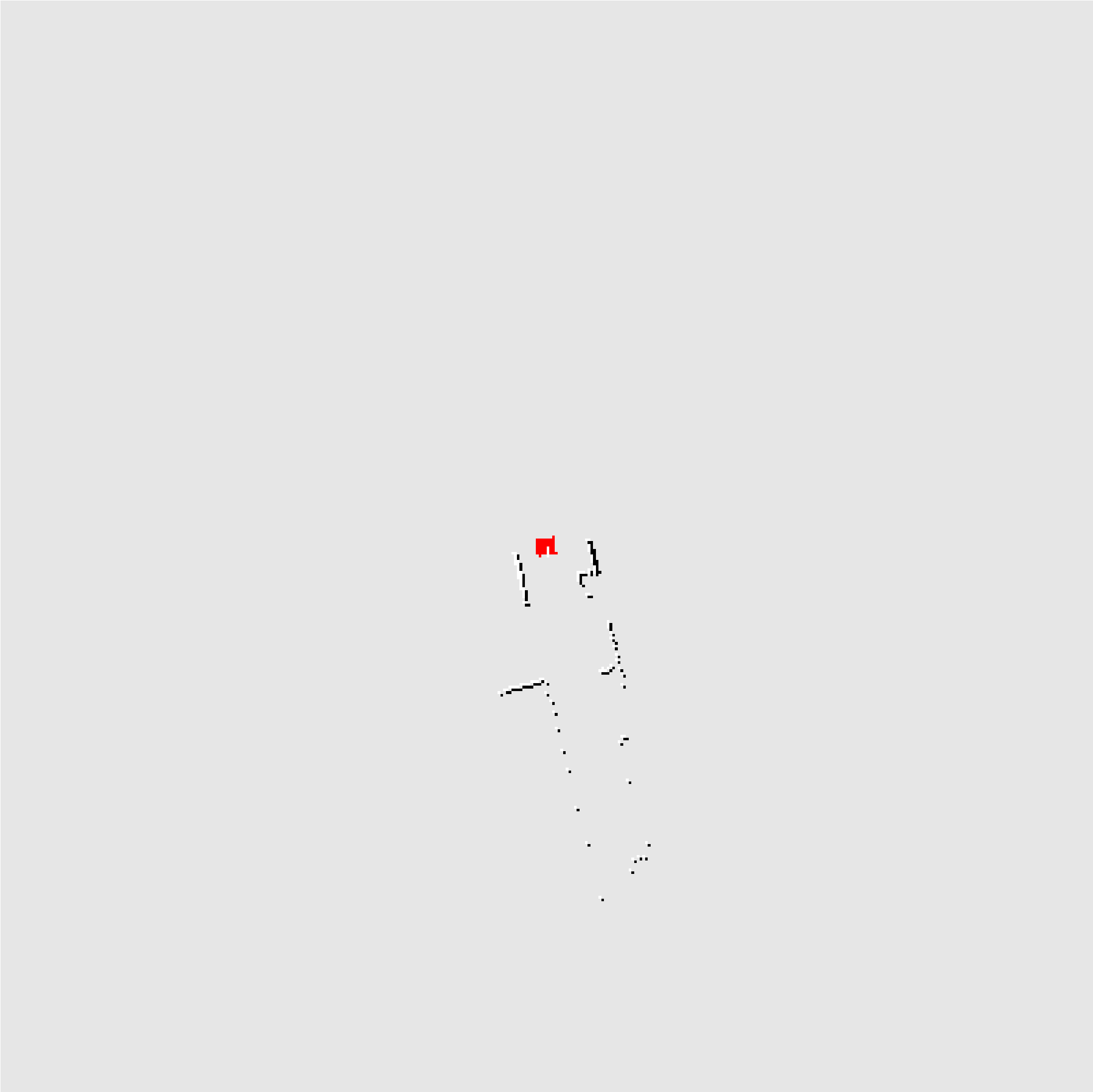


DP-SLAM, Ron Parr



[Demo: PARTICLES-SLAM-mapping1-new.avi]

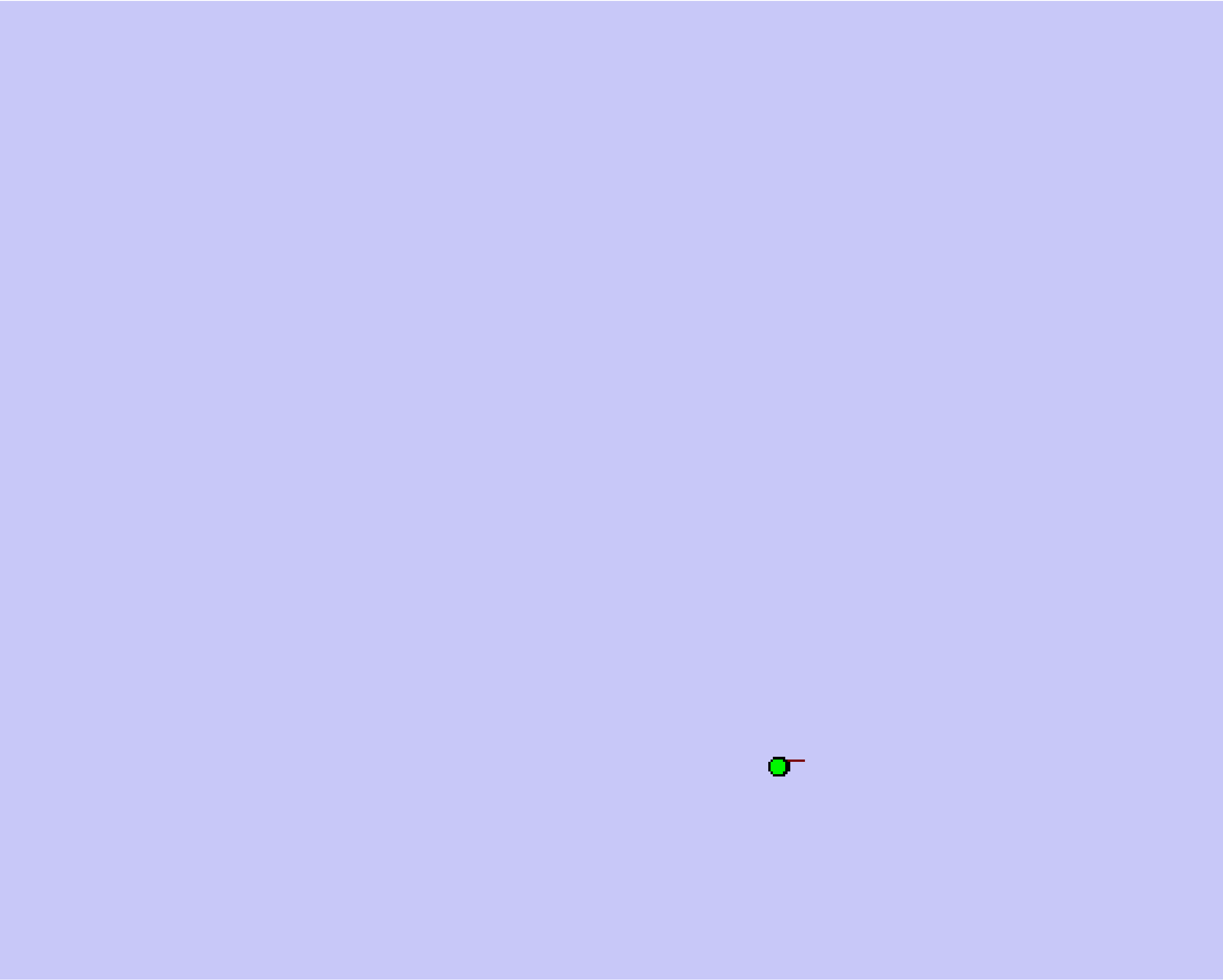
Particle Filter SLAM – Video 1



[Sebastian Thrun, et al.]

[Demo: PARTICLES-SLAM-mapping1-new.avi]

Particle Filter SLAM – Video 2



[Dirk Haehnel, et al.]

[Demo: PARTICLES-SLAM-fastslam.avi]