Markov Models, Hidden Markov Models, and Filtering

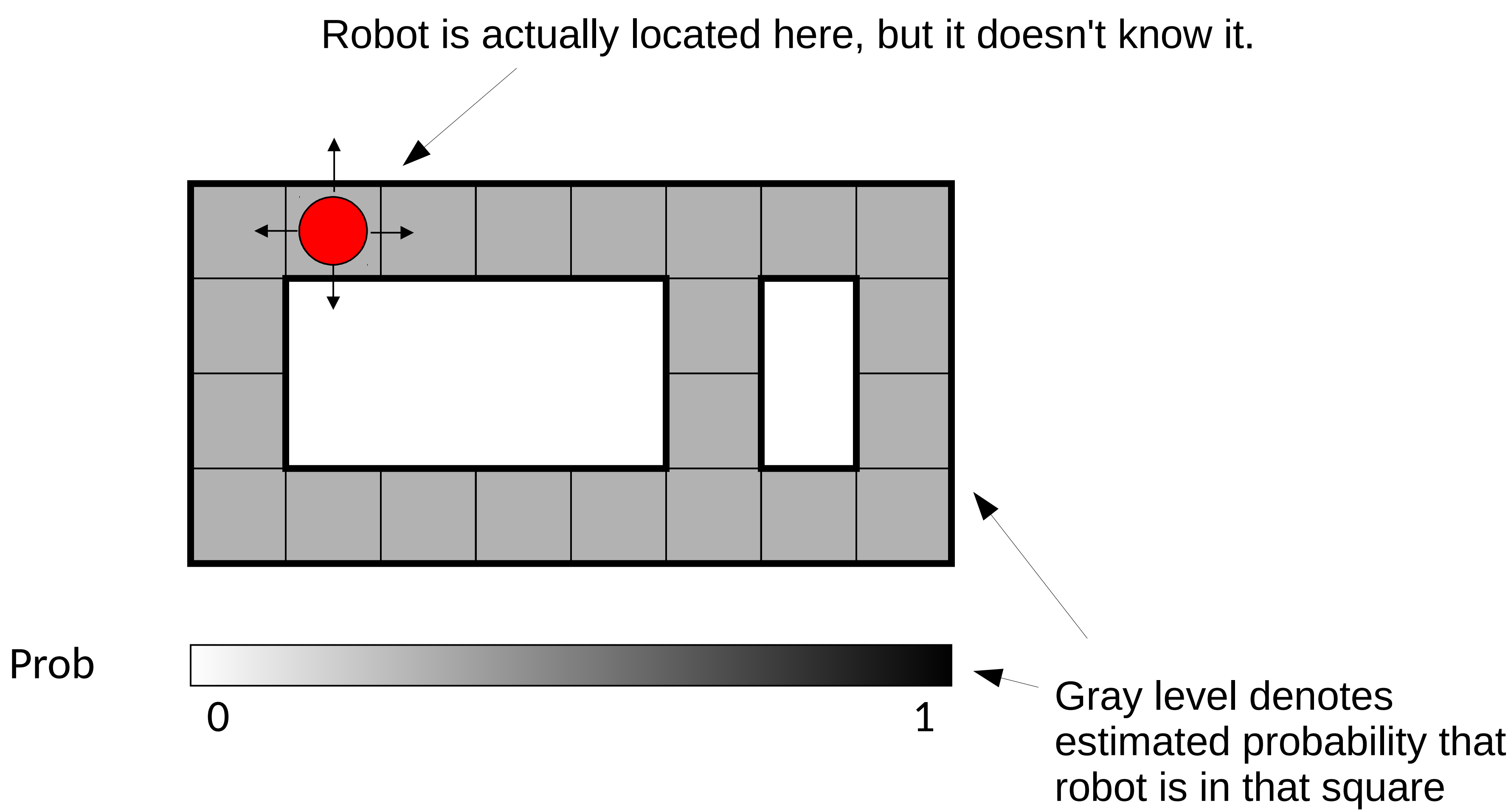
Robert Platt
Northeastern University

Some images and slides are used from:
1. CS188 UC Berkeley
2. RN, AIMA
3. Chris Amato

Motivation: robot localization

<u>Lost Robot!</u>
– robot cannot observe its location directly, i.e. no GPS
– can only make observations of the local environment...

Motivation: robot localization

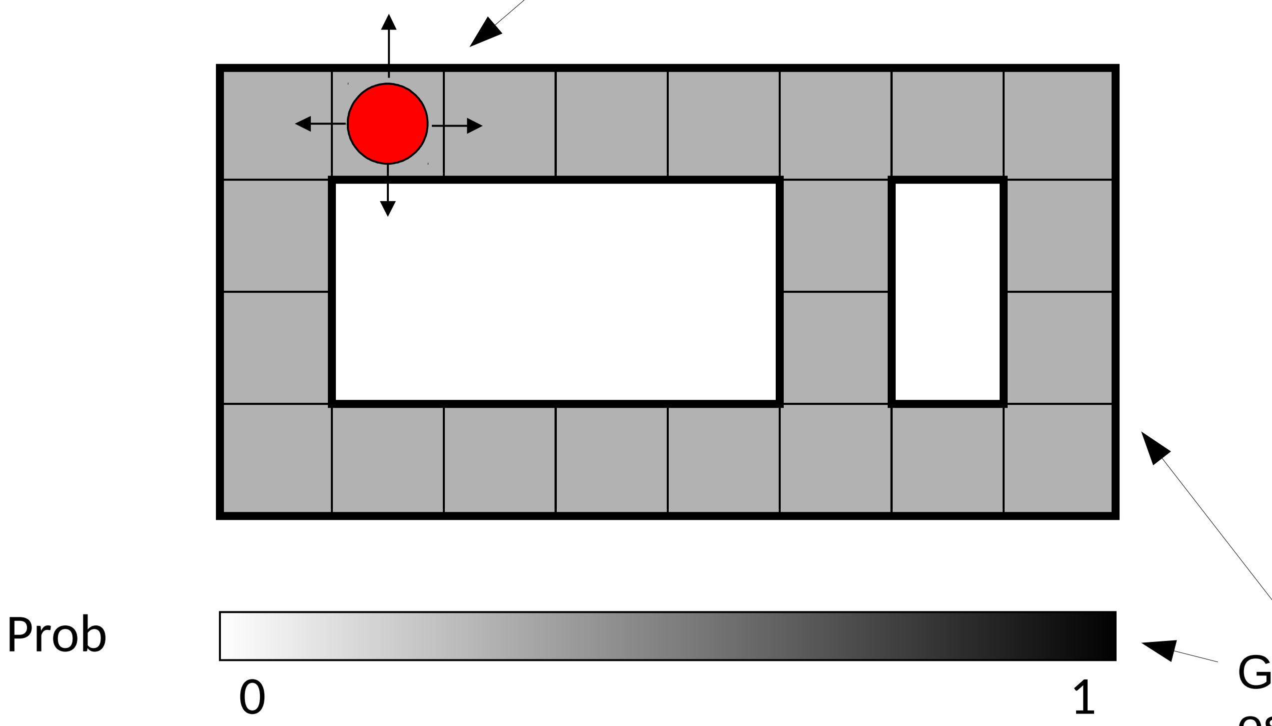


<u>Goal:</u> localize the robot based on sequential observations
– robot is given a map of the world; robot could be in any square
– initially, robot doesn't know which square it's in

Image: Berkeley CS188 course notes (downloaded Summer 2015)

Motivation: robot localization

Robot is actually located here, but it doesn't know it.



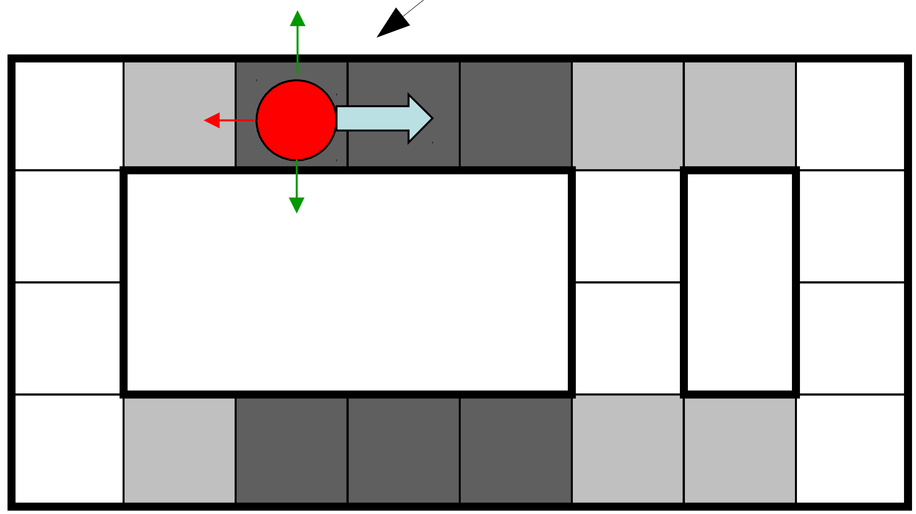
Gray level denotes estimated probability that robot is in that square

Goal: localize the robot based on sequential observations
– robot is given a map of the world; robot could be in any square
– initially, robot doesn't know which square it's in

Image: Berkeley CS188 course notes (downloaded Summer 2015)

Motivation: robot localization

Robot perceives that there are walls above and below, but no walls either left or right



Prob

0 1

Gray level denotes estimated probability that robot is in that square

On each time step, the robot moves, and then observes the directions in which there are walls.

– observes a four-bit binary number

– observations are noisy: there is a small chance that each bit will be flipped.

Image: Berkeley CS188 course notes (downloaded Summer 2015)

Motivation: robot localization

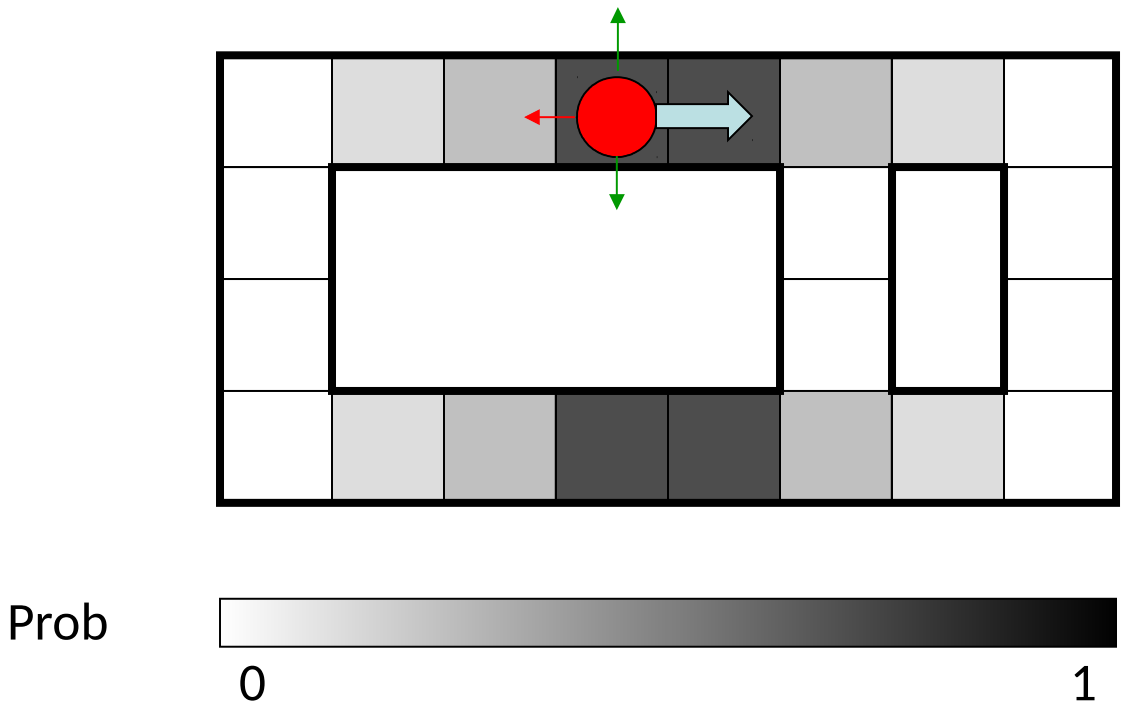


Image: Berkeley CS188 course notes (downloaded Summer 2015)

Motivation: robot localization

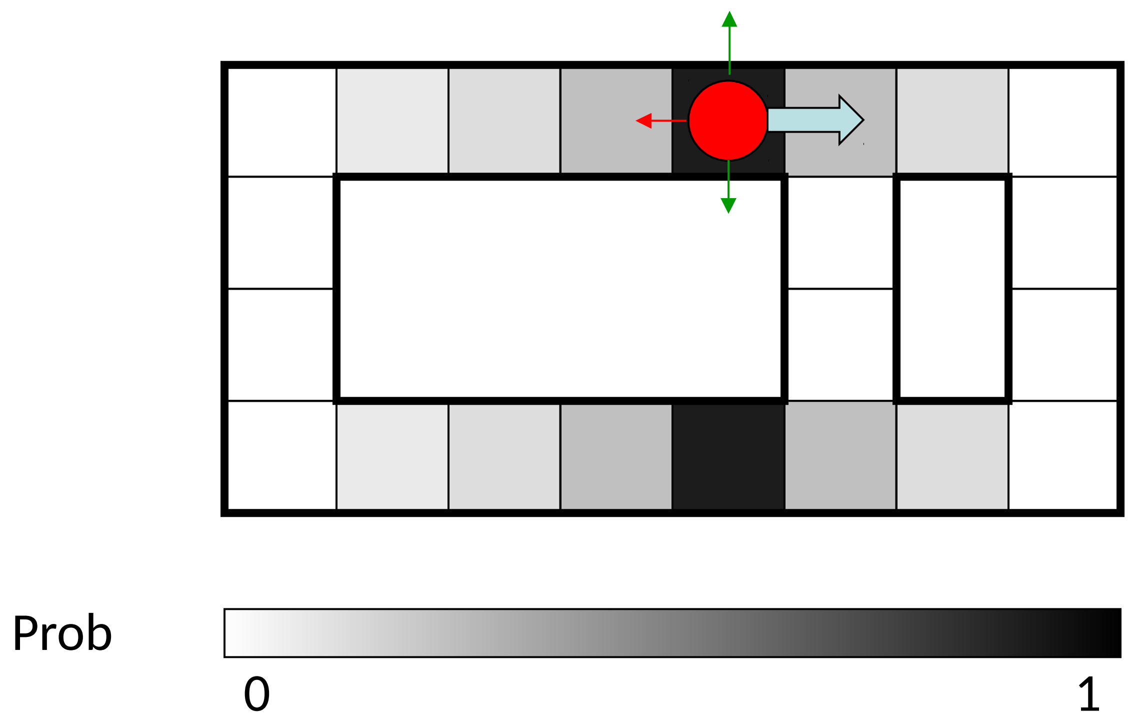


Image: Berkeley CS188 course notes (downloaded Summer 2015)

Motivation: robot localization

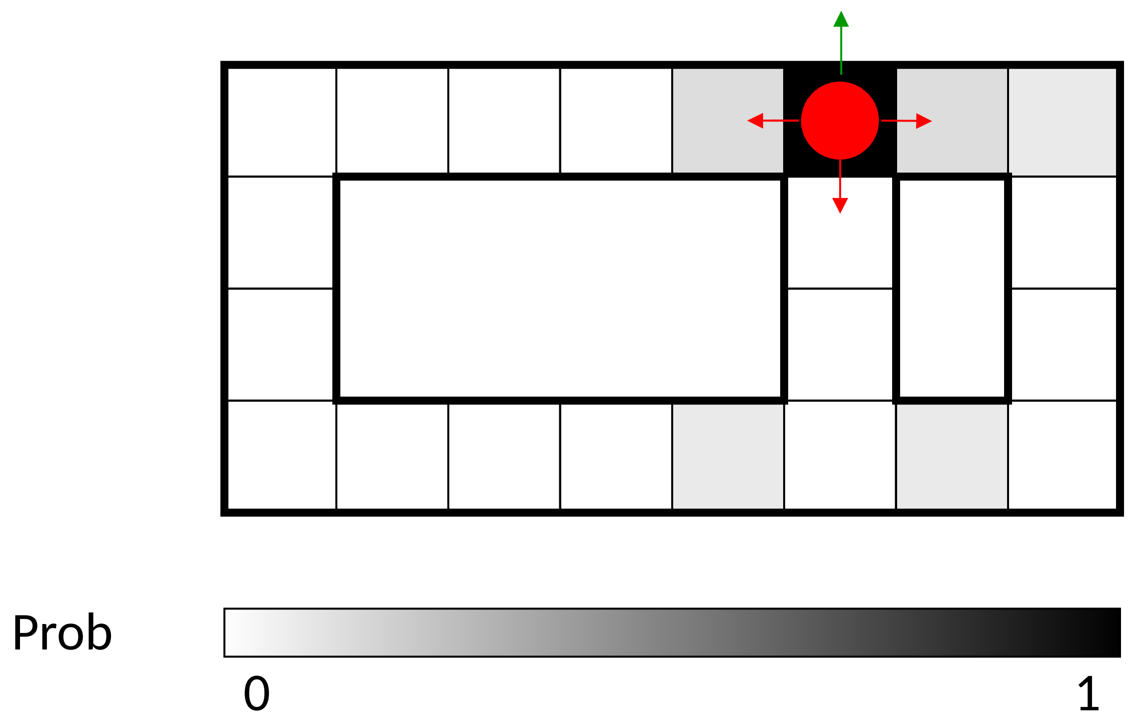
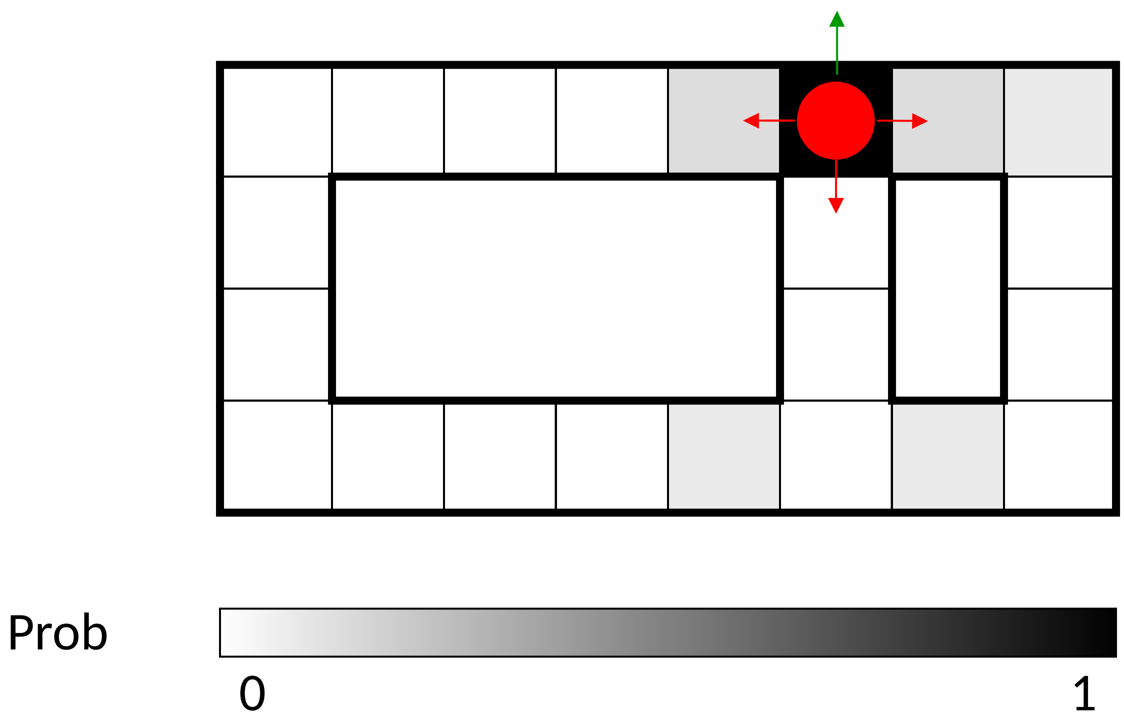


Image: Berkeley CS188 course notes (downloaded Summer 2015)

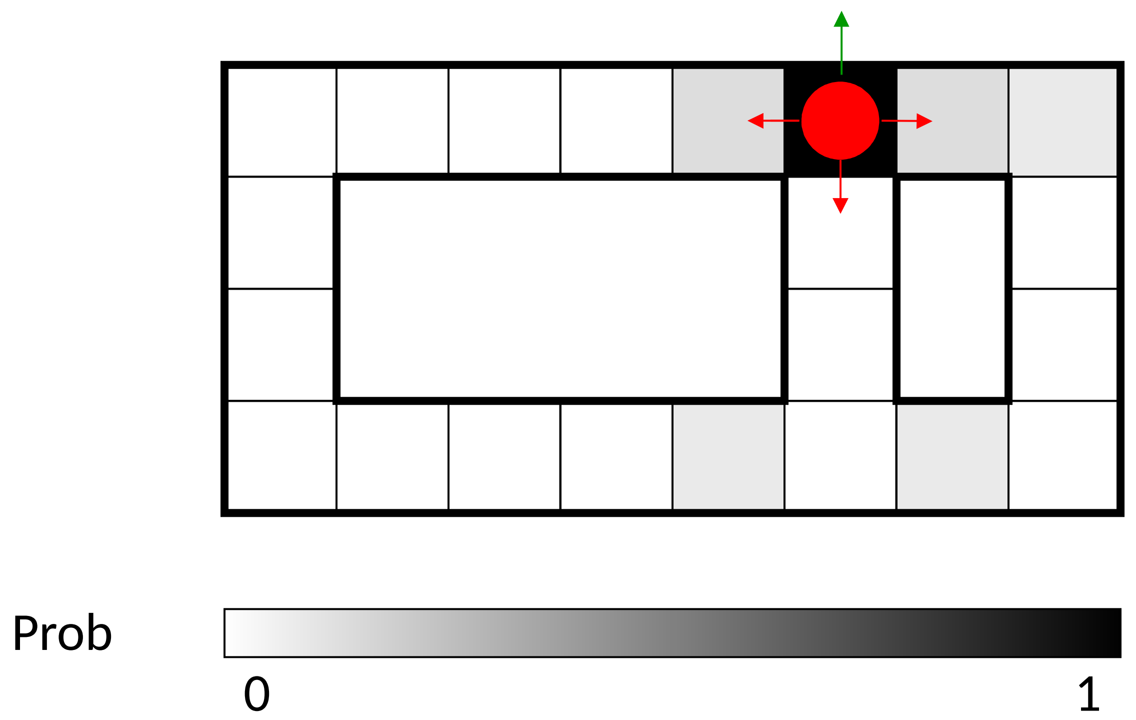
Motivation: robot localization



How does this work?

Image: Berkeley CS188 course notes (downloaded Summer 2015)

Motivation: robot localization



How do we update this probability distribution from time *t* to *t+1*?

Image: Berkeley CS188 course notes (downloaded Summer 2015)

Markov Models

We have already seen that an MDP provides a useful framework for modeling stochastic control problems.

Markov Models: model any kind of temporally dynamic system.

Probability recap

- Conditional probability $$P(x|y) = \frac{P(x,y)}{P(y)}$$
- Product rule $$P(x,y) = P(x|y)P(y)$$
- Chain rule $$\begin{aligned} P(X_1, X_2, \ldots X_n) &= P(X_1)P(X_2|X_1)P(X_3|X_1,X_2)\ldots \\ &= \prod_{i=1}^{n} P(X_i|X_1,\ldots,X_{i-1}) \end{aligned}$$
- X, Y independent if and only if: $$\forall x,y : P(x,y) = P(x)P(y)$$
- X and Y are conditionally independent given Z if and only if:

$$X \perp\!\!\!\perp Y | Z \qquad \forall x,y,z : P(x,y|z) = P(x|z)P(y|z)$$

Probability again: Independence

Two random variables, *x* and *y*, are independent when:

$$\forall(x, y), P(x, y) = P(x)P(y) \quad \Longleftrightarrow \quad x \perp\!\!\!\perp y$$

$$x \not\perp\!\!\!\perp y$$

The outcomes of two different coin flips are usually independent of each other

Probability again: Independence

If: $$P(x, y) = P(x)P(y)$$

Then: $$P(x) = P(x|y)$$

$$P(y) = P(y|x)$$

Why?

Probability again: Independence

Two random variables, *x* and *y*, are independent when:

$$\forall(x,y), P(x,y) = P(x)P(y) \iff x \perp\!\!\!\perp y$$

$$x \not\perp\!\!\!\perp y$$

The outcomes of two different coin flips are usually independent of each other

Example: Independence

	winter	!winter
snow	0.1	0.1
!snow	0.3	0.5

Example: Independence

	winter	!winter
snow	0.1	0.1
!snow	0.3	0.5

Are snow and winter independent variables?

Example: Independence

	winter	!winter
snow	0.1	0.1
!snow	0.3	0.5

Are snow and winter independent variables?

P(snow) = 0.2

P(winter) = 0.4

Example: Independence

	winter	!winter
snow	0.1	0.1
!snow	0.3	0.5

Are snow and winter independent variables?

P(snow) = 0.2

P(winter) = 0.4

What would the distribution look like if snow, winter were independent?

Conditional independence

Independence: $$\forall (x, y), P(x, y) = P(x)P(y)$$

$$x \perp\!\!\!\perp y$$

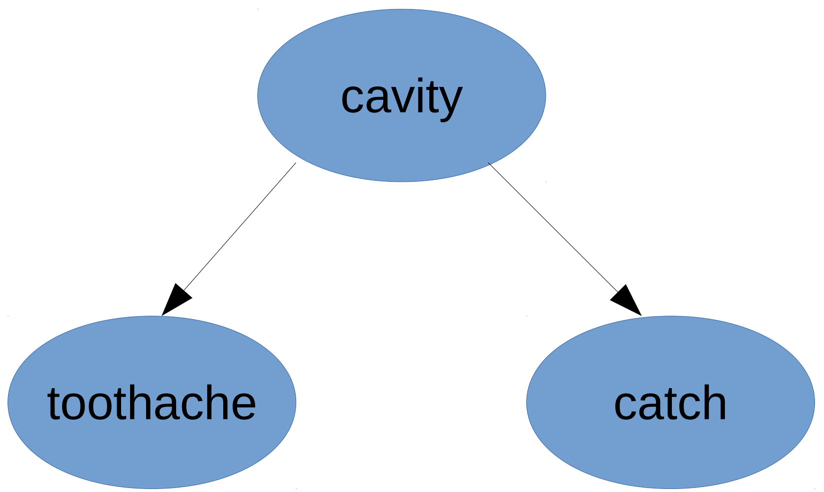
Conditional independence: $$\forall (x, y, z), P(x, y|z) = P(x|z)P(y|z)$$

$$x \perp\!\!\!\perp y|z$$

Equivalent statements of conditional independence: $$P(x|z) = P(x|z, y)$$

$$P(y|z) = P(y|z, x)$$

Conditional independence: example



P(toothache, catch | cavity) = P(toothache | cavity) P(catch | cavity)

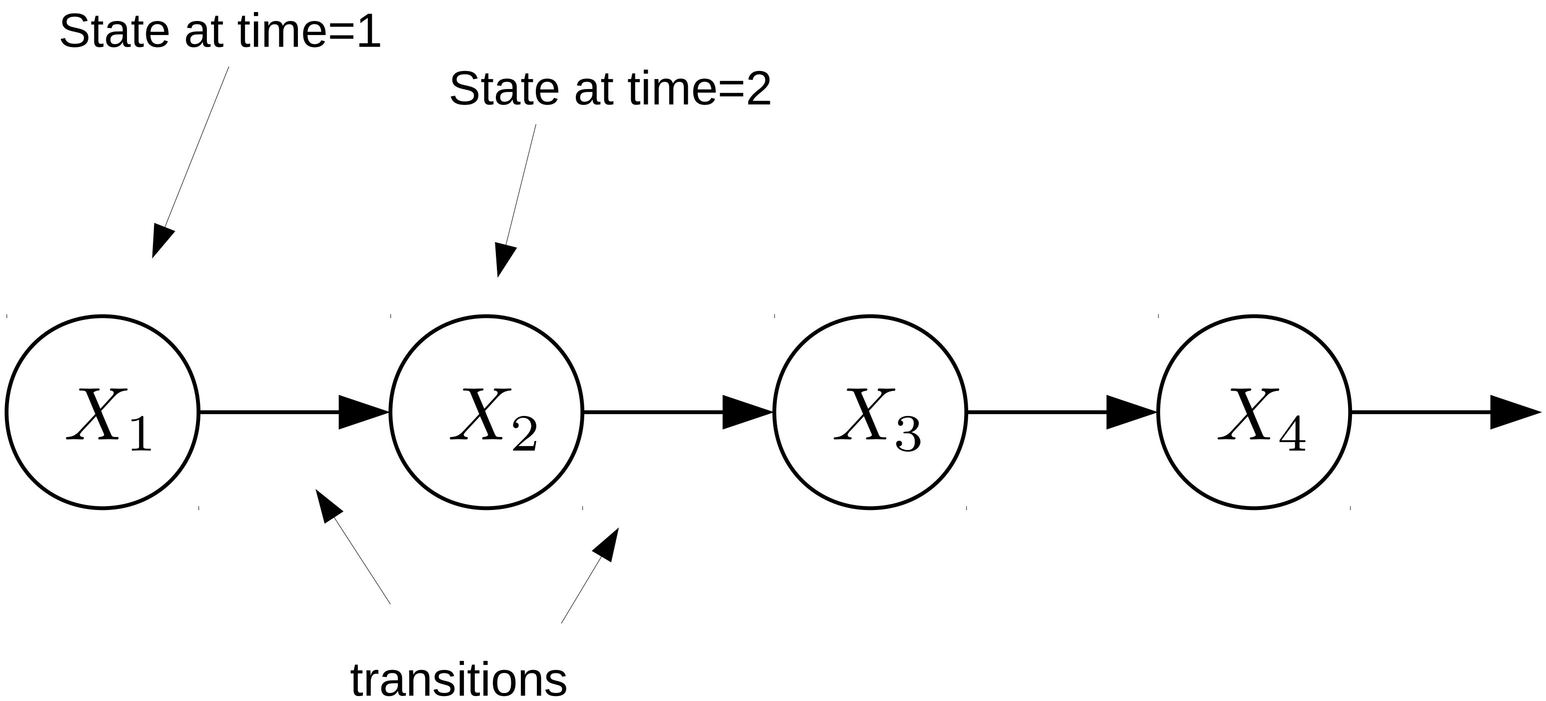
Toothache and catch are conditionally independent given cavity – this is the “common cause” scenario covered in Bayes Nets...

Examples of conditional independence

What are the conditional independence relationships in the following?

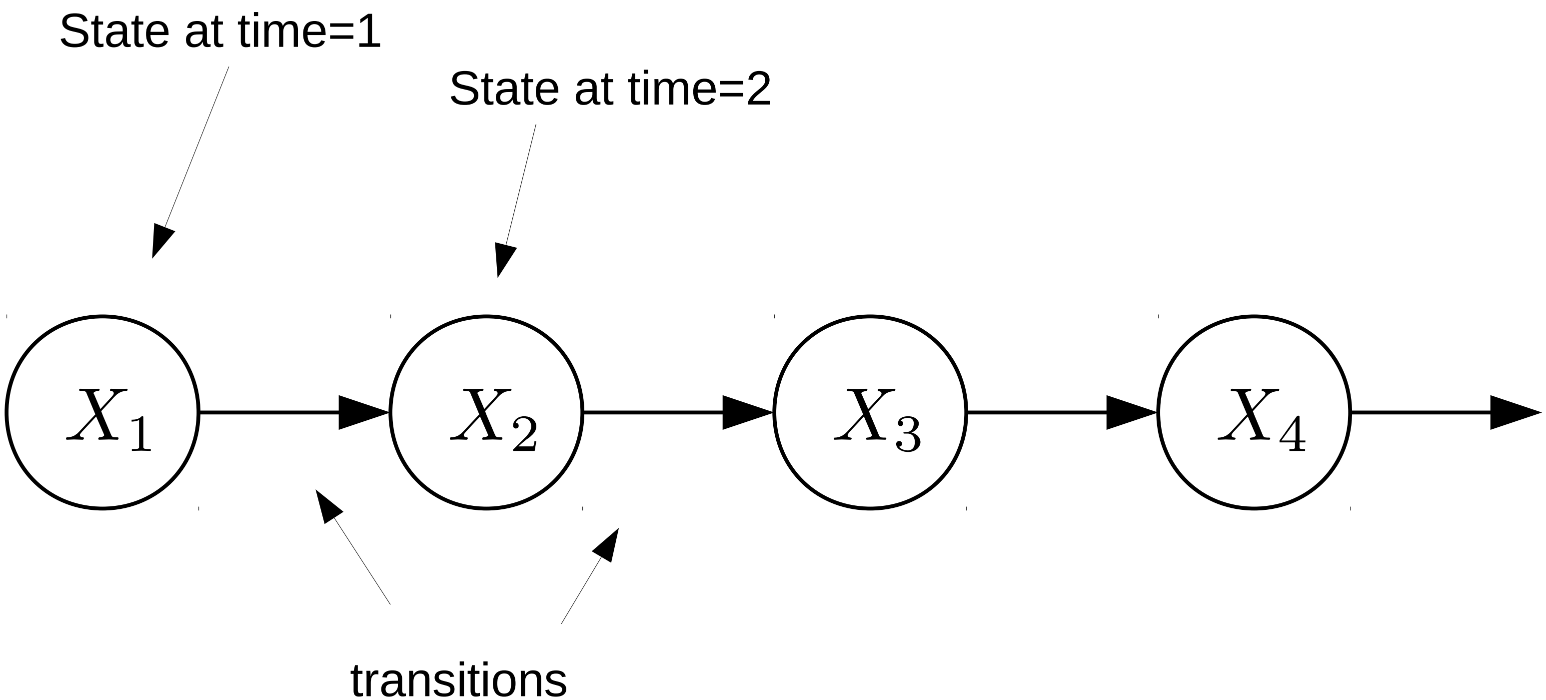
– traffic, raining, late for work
– snow, cloudy, crash
– fire, smoke, alarm

Markov Processes



Markov model can be used to model any sequential time process
– the weather
– traffic
– news cycle
– text to speech
...

Markov Processes

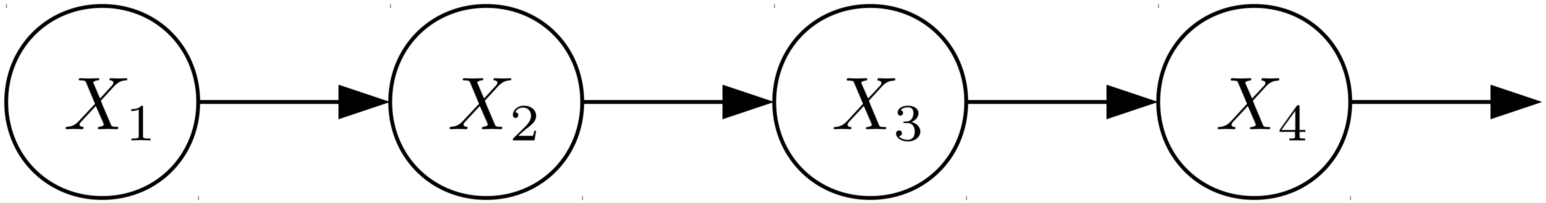


Since this is a Markov process, we assume transitions are Markov:

Process model: $$P(X_t|X_{t-1}) = P(X_t|X_{t-1}, \ldots, X_1)$$

Markov assumption: $$X_t \perp\!\!\!\perp X_{t-2}|X_{t-1}$$

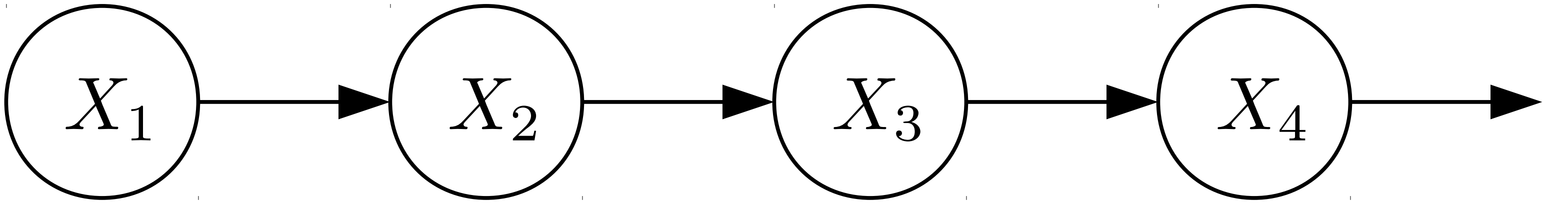
Markov Processes



How do we calculate: $P(X_1, X_2, X_3, X_4) =?$

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X_3|X_2, X_1)P(X_4|X_3, X_2, X_1)$$

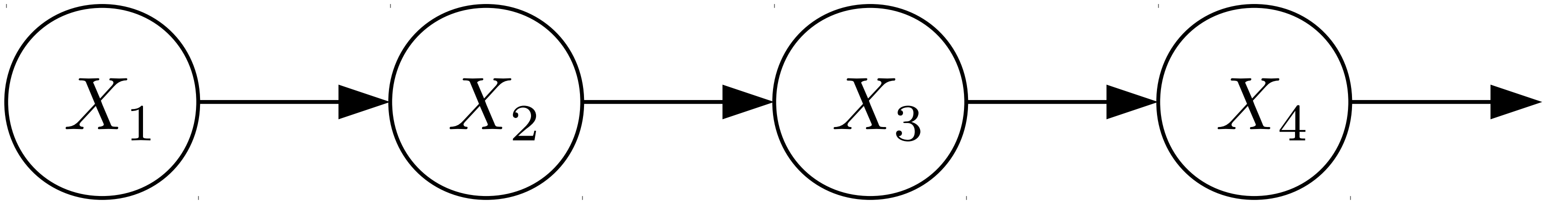
Markov Processes



How do we calculate: $P(X_1, X_2, X_3, X_4) = ?$

$$P(X_1, X_2, X_3, X_4) = \underbrace{P(X_1)P(X_2|X_1)}_{P(X_2, X_1)}P(X_3|X_2, X_1)P(X_4|X_3, X_2, X_1)$$

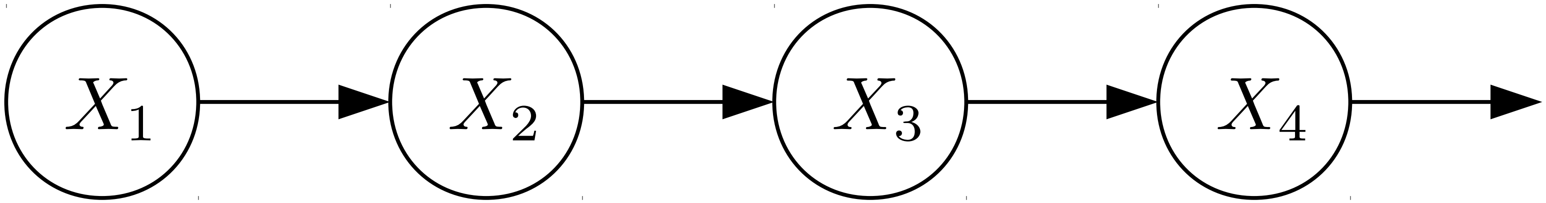
Markov Processes



How do we calculate: $P(X_1, X_2, X_3, X_4) = ?$

$$P(X_1, X_2, X_3, X_4) = \underbrace{P(X_1)P(X_2|X_1)P(X_3|X_2, X_1)}_{P(X_3, X_2, X_1)}P(X_4|X_3, X_2, X_1)$$

Markov Processes

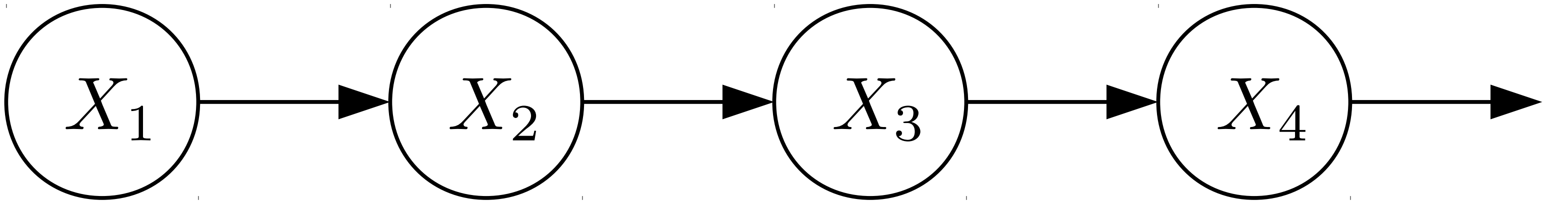


How do we calculate: $P(X_1, X_2, X_3, X_4) = ?$

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X_3|X_2, X_1)P(X_4|X_3, X_2, X_1)$$

Can we simplify this expression?

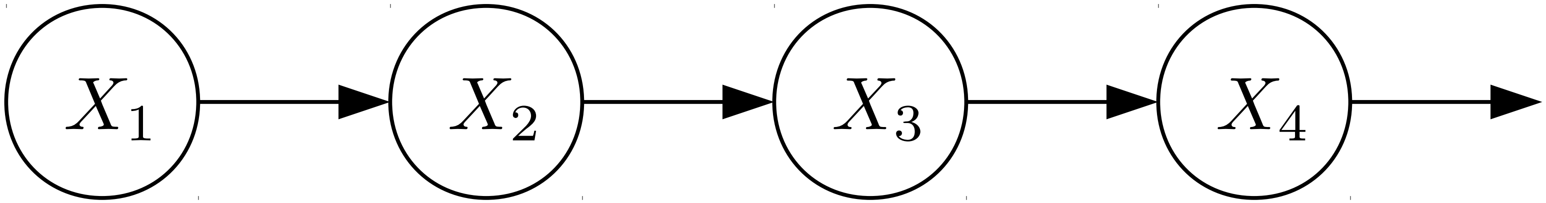
Markov Processes



How do we calculate: $P(X_1, X_2, X_3, X_4) = ?$

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)\underbrace{P(X_3|X_2, X_1)}_{P(X_3|X_2)}\underbrace{P(X_4|X_3, X_2, X_1)}_{P(X_4|X_3)}$$

Markov Processes

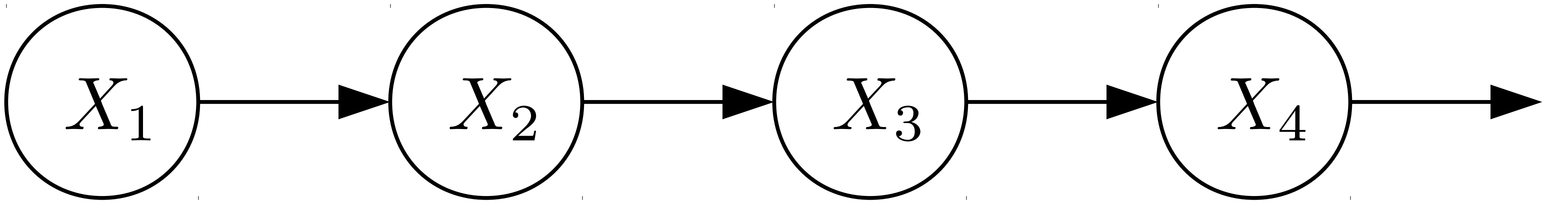


How do we calculate: $P(X_1, X_2, X_3, X_4) = ?$

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X_3|X_2, X_1)P(X_4|X_3, X_2, X_1)$$

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X_3|X_2)P(X_4|X_3)$$

Markov Processes



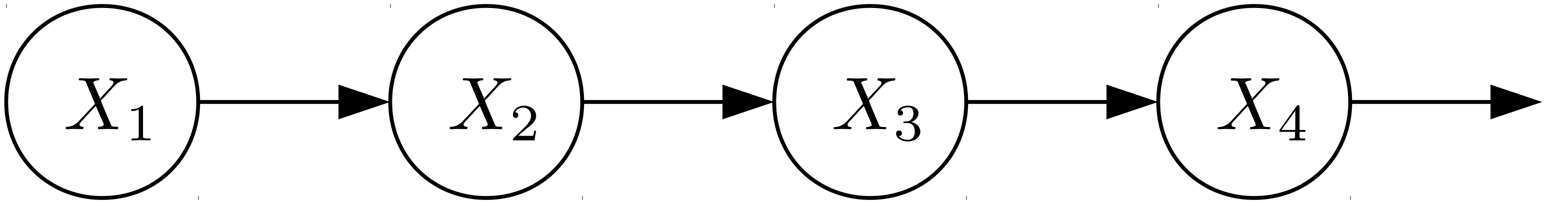
How do we calculate: $P(X_1, X_2, X_3, X_4) = ?$

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X_3|X_2, X_1)P(X_4|X_3, X_2, X_1)$$

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X_3|X_2)P(X_4|X_3)$$

In general: $P(X_1, X_2, \ldots, X_T) = P(X_1) \prod_{t=1}^{T-1} P(X_{t+1}|X_t)$

Markov Processes



How do we calculate: $P(X_1, X_2, X_3, X_4) = ?$

$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X$ … $X_2, X_1)$

Process model

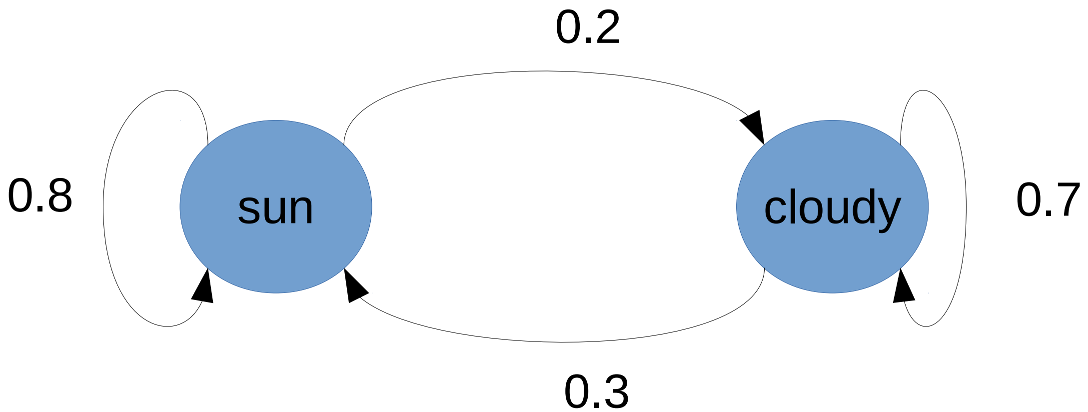
$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2|X_1)P(X_3|X_2)P(X_4|X_3)$$

In general: $$P(X_1, X_2, \ldots, X_T) = P(X_1)\prod_{t=1}^{T-1} \boxed{P(X_{t+1}|X_t)}$$

Markov Processes: example

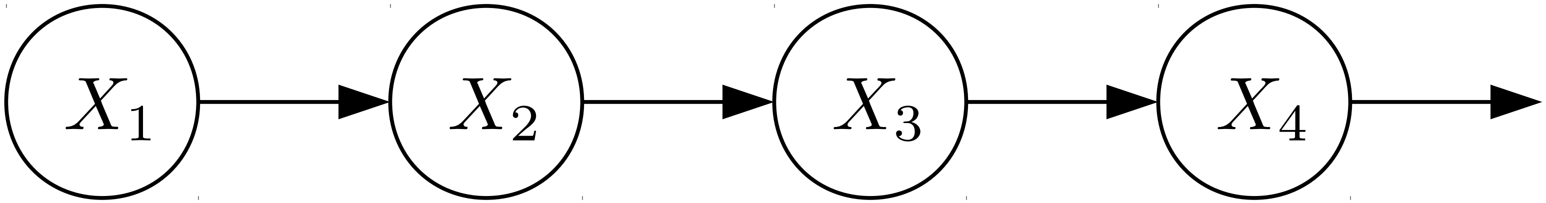
Two states: cloudy, sunny

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7



$$t \in \{\text{mon}, \text{tues}, \text{weds}, \text{thurs}, \text{fri}\}$$

Simulating dynamics forward



Joint distribution: $P(X_1, X_2, \ldots, X_T) = P(X_1) \prod_{t=1}^{T-1} P(X_{t+1}|X_t)$

But, suppose we want to predict the state at time T, given a prior distribution at time 1?

$$P(X_2) = \sum_{X_1} P(X_1)P(X_2|X_1)$$

$$P(X_3) = \sum_{X_2} P(X_2)P(X_3|X_2)$$

$$\vdots$$

$$P(X_T) = \sum_{X_{T-1}} P(X_{T-1})P(X_T|X_{T-1})$$

Simulating dynamics forward

Suppose is it sunny on mon... $P(x_1) = 1$

Simulating dynamics forward

Suppose is it sunny on mon... $P(x_1) = 1$

Prob sunny tues
$$\begin{aligned} P(x_2) &= P(x_2|x_1)P(x_1) \\ &= 0.8 \end{aligned}$$

Simulating dynamics forward

Suppose is it sunny on mon... $P(x_1) = 1$

Prob sunny tues
$$\begin{aligned} P(x_2) &= P(x_2|x_1)P(x_1) \\ &= 0.8 \end{aligned}$$

Prob sunny weds
$$\begin{aligned} P(x_3) &= P(x_3|x_2)P(x_2) + P(x_3|\bar{x}_2)P(\bar{x}_2) \\ &= 0.64 + 0.06 = 0.7 \end{aligned}$$

Prob sunny thurs
$$\begin{aligned} P(x_4) &= P(x_4|x_3)P(x_3) + P(x_4|\bar{x}_3)P(\bar{x}_3) \\ &= 0.56 + 0.09 = 0.65 \end{aligned}$$

Prob sunny fri
$$\begin{aligned} P(x_5) &= P(x_5|x_4)P(x_4) + P(x_5|\bar{x}_4)P(\bar{x}_4) \\ &= 0.52 + 0.105 = 0.625 \end{aligned}$$

$$P(x_\infty) = 0.6$$

Simulating dynamics forward

Suppose is it cloudy on mon... $P(x_1) = 0$

Prob sunny tues
$$\begin{aligned} P(x_2) &= P(x_2|x_1)P(x_1) + P(x_2|\bar{x}_1)P(\bar{x}_1) \\ &= 0 + 0.3 = 0.3 \end{aligned}$$

Prob sunny weds
$$\begin{aligned} P(x_3) &= P(x_3|x_2)P(x_2) + P(x_3|\bar{x}_2)P(\bar{x}_2) \\ &= 0.24 + 0.21 = 0.45 \end{aligned}$$

Prob sunny thurs
$$\begin{aligned} P(x_4) &= P(x_4|x_3)P(x_3) + P(x_4|\bar{x}_3)P(\bar{x}_3) \\ &= 0.36 + 0.165 = 0.53 \end{aligned}$$

Prob sunny fri
$$\begin{aligned} P(x_5) &= P(x_5|x_4)P(x_4) + P(x_5|\bar{x}_4)P(\bar{x}_4) \\ &= 0.424 + 0.141 = 0.565 \end{aligned}$$

$$P(x_\infty) = 0.6$$

Simulating dynamics forward

Suppose is it cloudy on mon... $P(x_1) = 0$

Prob sunny tues
$$\begin{aligned} P(x_2) &= P(x_2|x_1)P(x_1) + P(x_2|\bar{x}_1)P(\bar{x}_1) \\ &= 0 + 0.3 = 0.3 \end{aligned}$$

Prob sunny weds
$$\begin{aligned} P(x_3) &= P(x_3|x_2)P(x_2) + P(x_3|\bar{x}_2)P(\bar{x}_2) \\ &= 0.24 + 0.21 = 0.45 \end{aligned}$$

Prob sunny thurs
$$\begin{aligned} P(x_4) &= P(x_4|x_3)P(x_3) + P(x_4|\bar{x}_3)P(\bar{x}_3) \\ &= 0.36 + 0.165 = 0.53 \end{aligned}$$

Prob sunny $x_5|\bar{x}_4)P(\bar{x}_4)$

Converge to same distribution regardless of starting point
– called the “stationary distribution”

.5

$$\boxed{P(x_\infty) = 0.6}$$

An aside: the stationary distribution

How might you calculate the stationary distribution?

Let: $p_t = \begin{pmatrix} p(sun) \\ p(cloudy) \end{pmatrix}$ $\quad T = \begin{pmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{pmatrix}$

Then: $p_{t+1} = Tp_t$

Stationary distribution is the value for p such that: $p = Tp$

An aside: the stationary distribution

How might you calculate the stationary distribution?

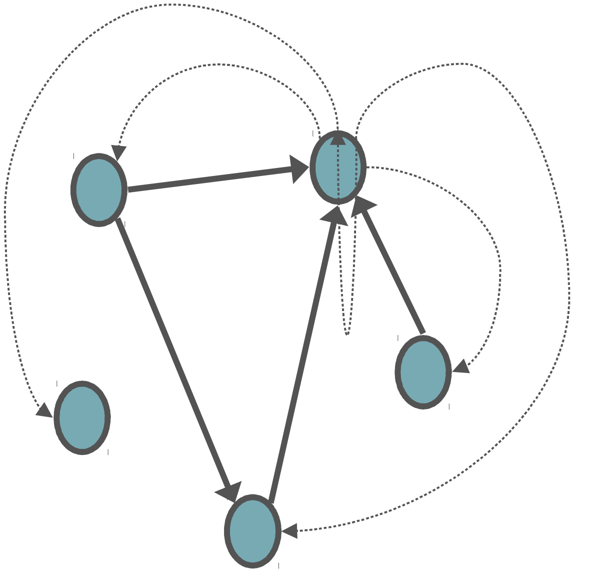
Let: $p_t = \begin{pmatrix} p(sun) \\ p(cloudy) \end{pmatrix}$ $T = \begin{pmatrix} 0.8 & 0.3 \\ 0.2 & 0.7 \end{pmatrix}$

Then: $p_{t+1} = T p_t$

Stationary distribution is the value for p such that: $p = Tp$

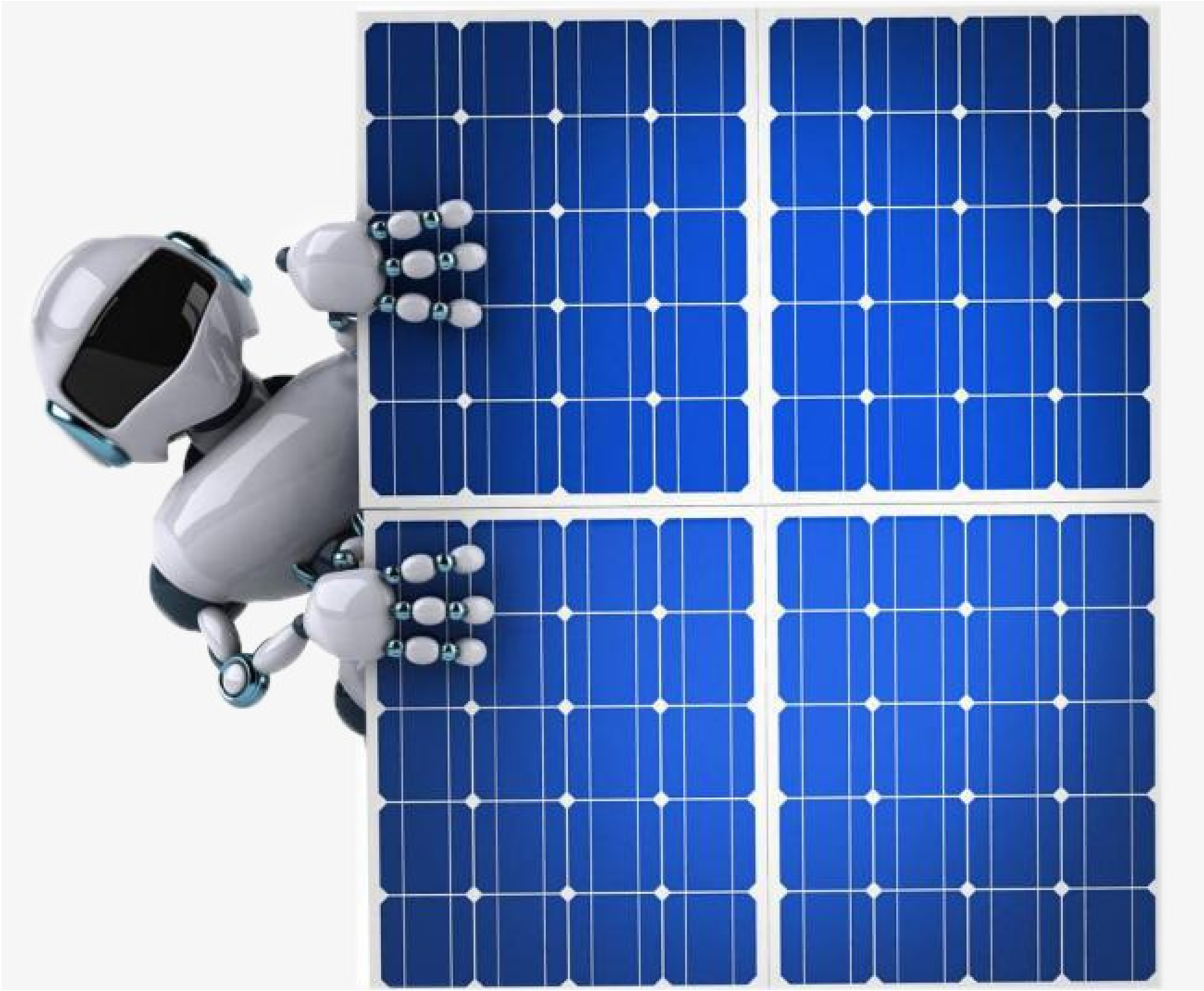
How calculate p that satisfies this eqn?

Case Study: Web link Analysis

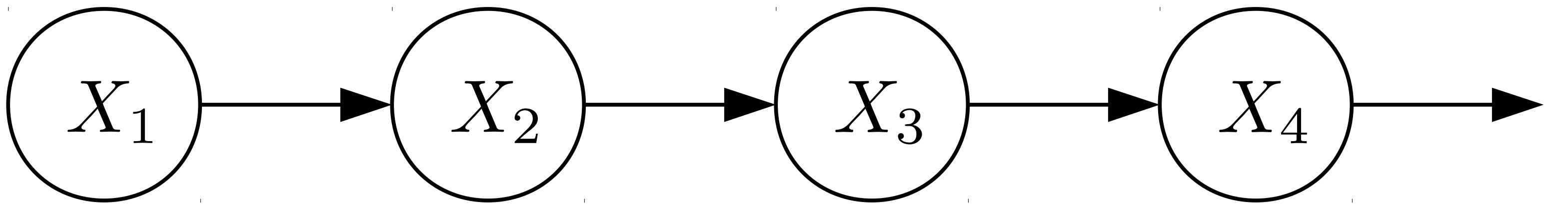


- PageRank over a web graph
  - Each web page is a state
  - Initial distribution: uniform over pages
  - Transitions:
    - With prob. c, uniform jump to a random page (dotted lines, not all shown)
    - With prob. 1-c, follow a random outlink (solid lines)
- Stationary distribution
  - Will spend more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

Hidden Markov Models (HMMs)

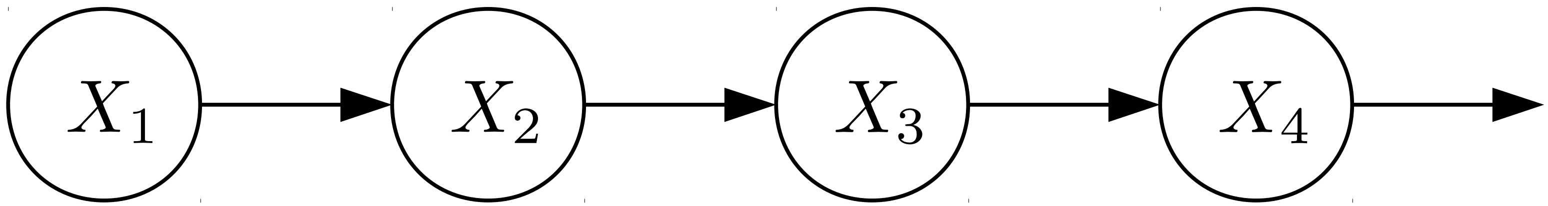


Markov Model



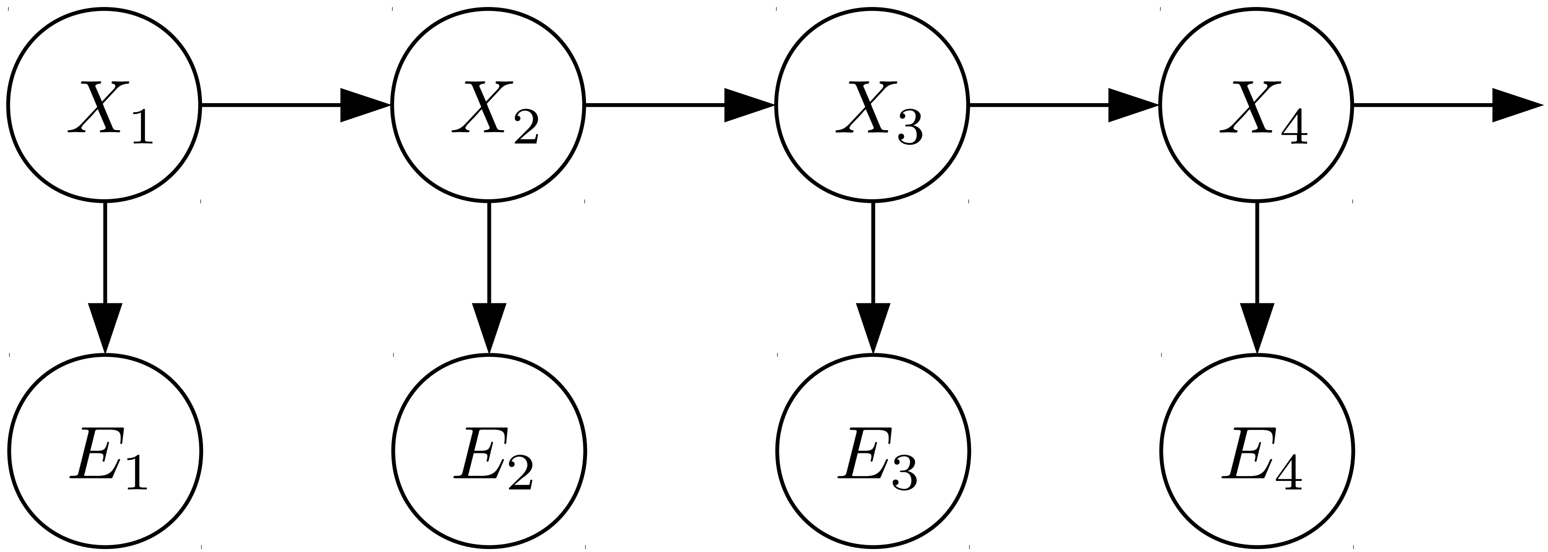
State, X_t , is observed

Hidden Markov Model



State, X_t , is assumed to be unobserved

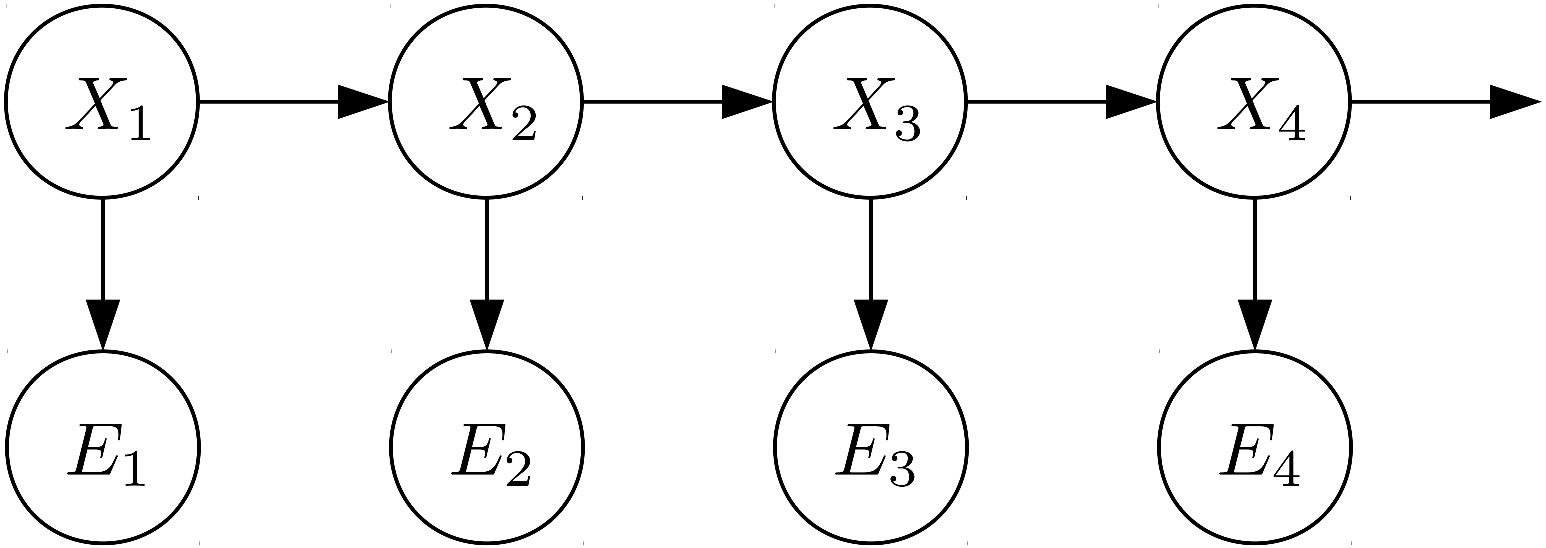
Hidden Markov Model



State, X_t , is assumed to be unobserved

However, you get to make one observation, E_t , on each timestep.

Hidden Markov Model

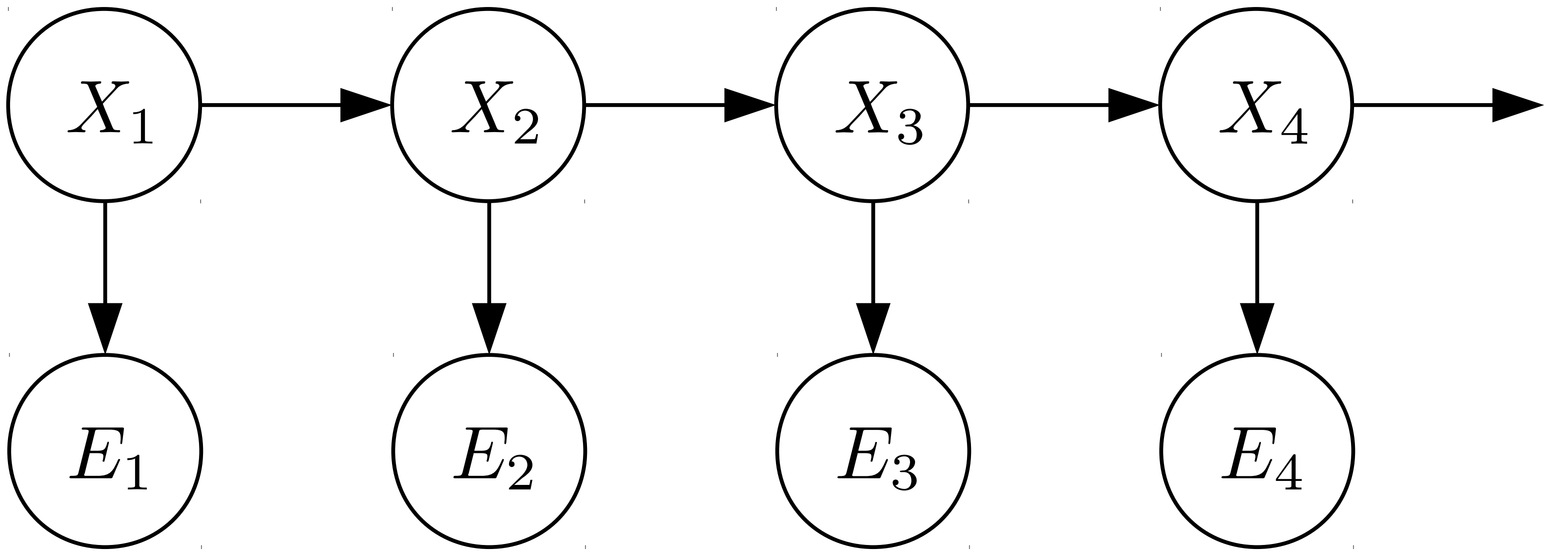


Called an “emission”

State, X_t , is assumed to be unobserved

However, you get to make one observation, E_t , on each timestep.

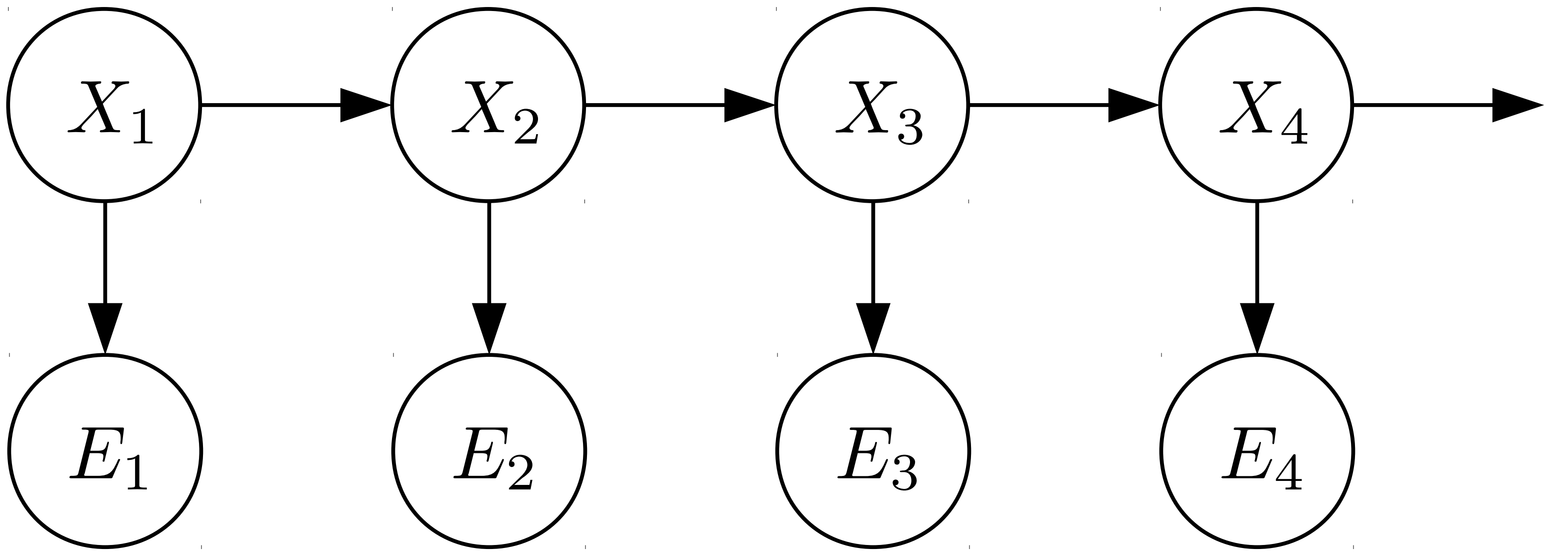
Hidden Markov Models (HMMs)



Process dynamics: $P(X_t|X_{t-1})$ How the system changes from one time step to the next

Observation dynamics: $P(E_t|X_t)$ What gets observed as a function of what state the system is in

Hidden Markov Models (HMMs)

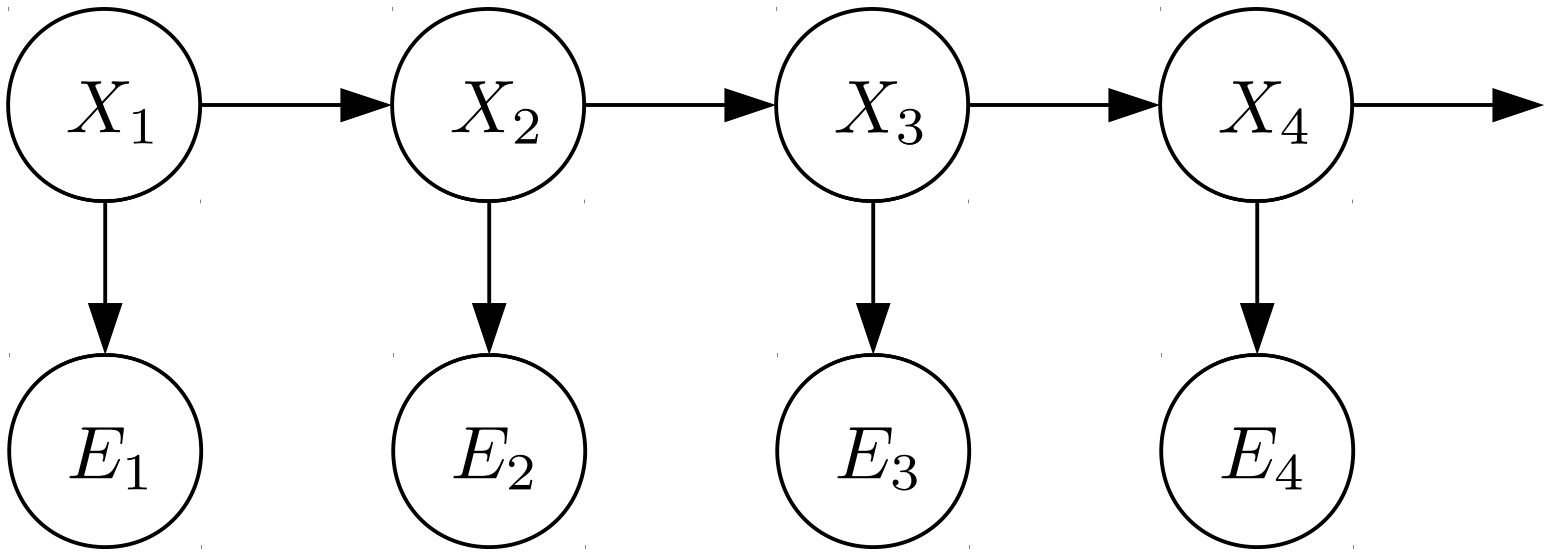


Process dynamics: $P(X_t|X_{t-1})$ — How the system changes from one time step to the next

Observation dynamics: $P(E_t|X_t)$ — What gets observed as a function of what state the system is in

Let's assume (for now) that these probability distributions are given to us.

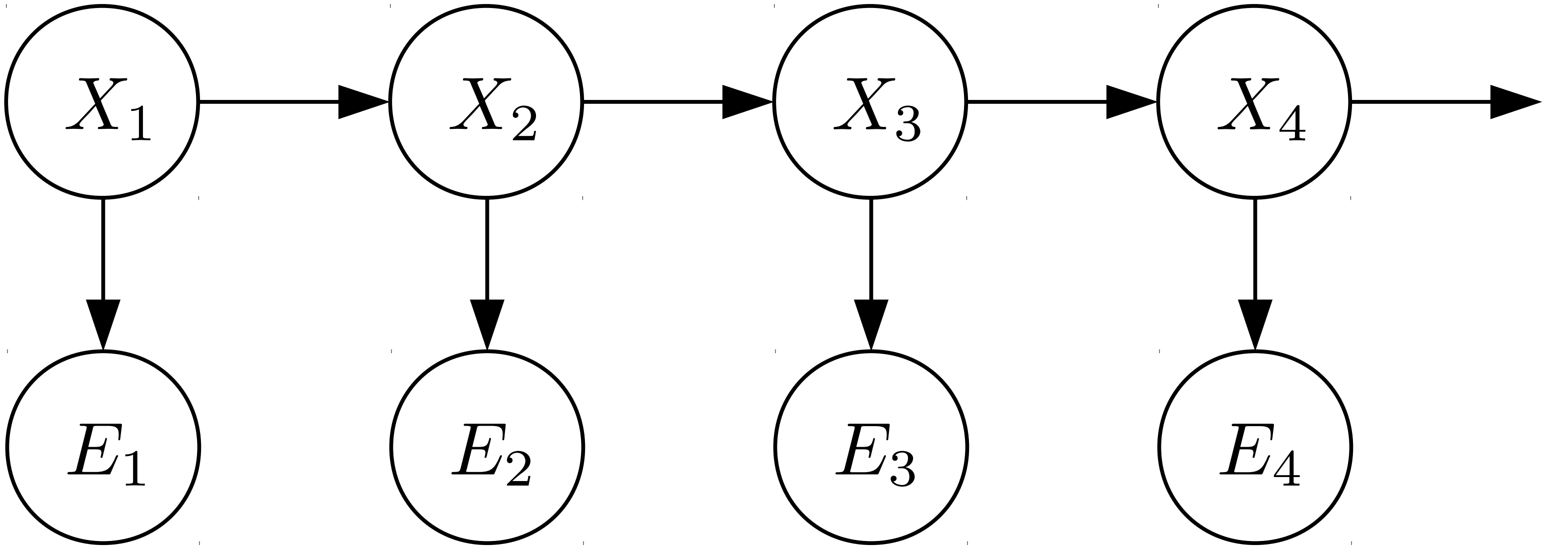
Hidden Markov Models (HMMs)



Process dynamics: $P(X_t|X_{t-1}) = P(X_t|X_{t-1}, \ldots, X_1)$

Observation dynamics: $P(E_t|X_t) = P(E_t|X_t, X_{t-1}, \ldots, X_1)$

Hidden Markov Models (HMMs)



Process dynamics: $P(X_t|X_{t-1}) = P(X_t|X_{t-1}, \ldots, X_1)$

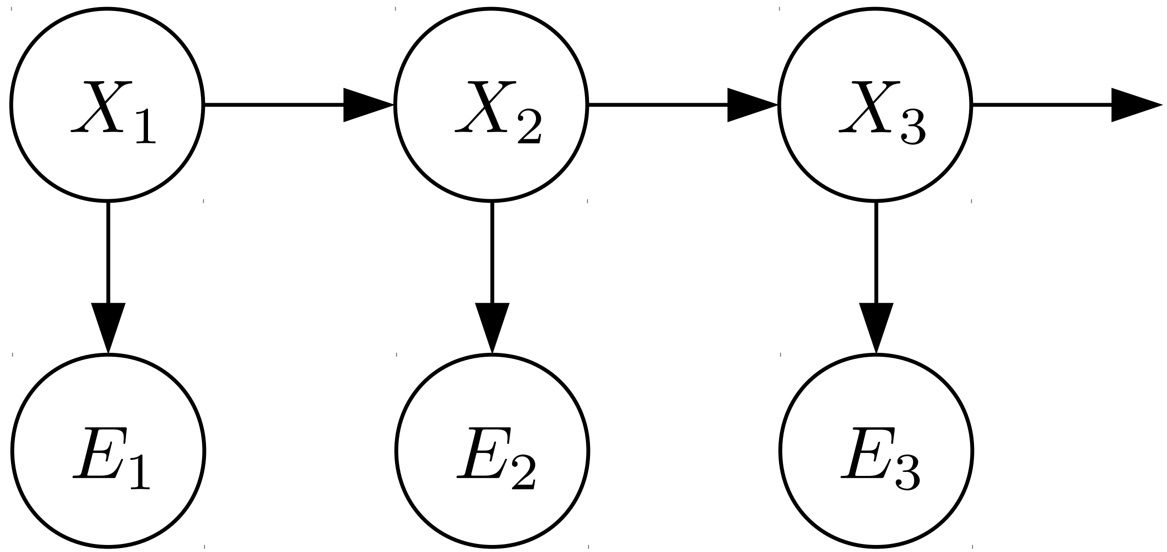
Observation dynamics: $P(E_t|X_t) = P(E_t|X_t, X_{t-1}, \ldots, X_1)$

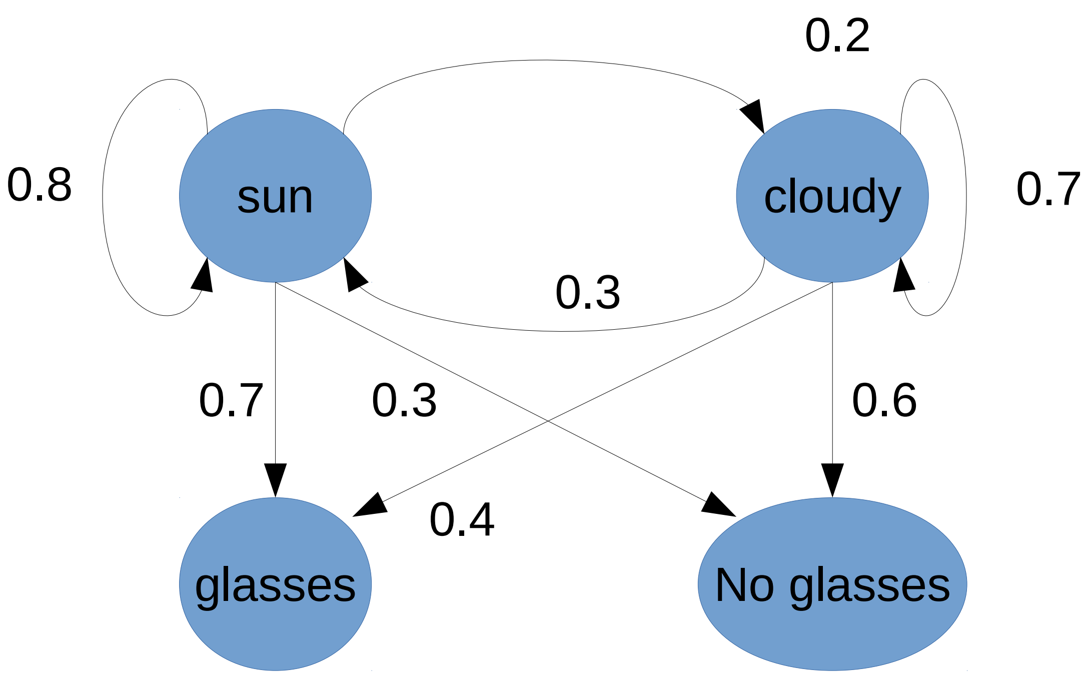
Markov assumptions

HMM example 1

$X \in \{\text{sun}, \text{cloudy}\}$
(state is unobserved)

$E \in \{\text{glasses}, \text{no glasses}\}$
(only observations are observed)





You live underground...

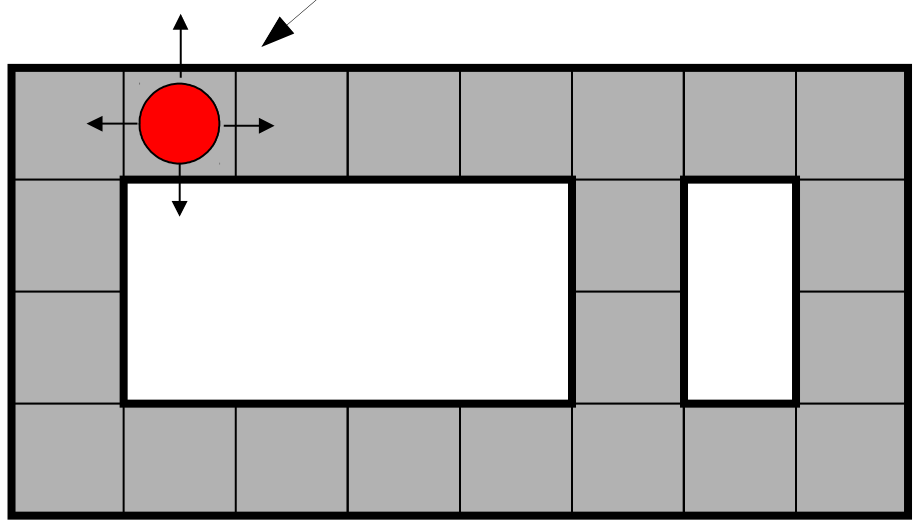
Every day, you're boss comes in either wearing sunglasses or not

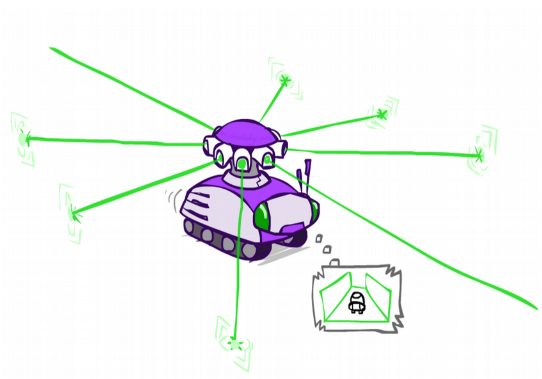
Can you infer whether it's sunny out based on whether you see the glasses over a sequence of days?

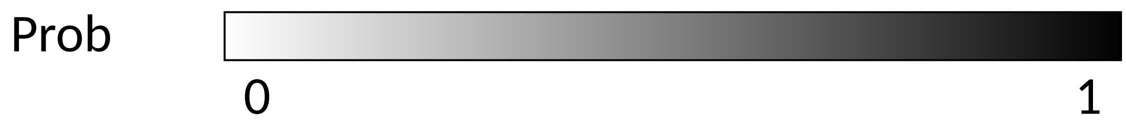
– e.g. what's the prob it's sunny out today if you've seen your boss wear glasses three days in a row?

HMM example 2

Robot is actually located here, but it doesn't know it.





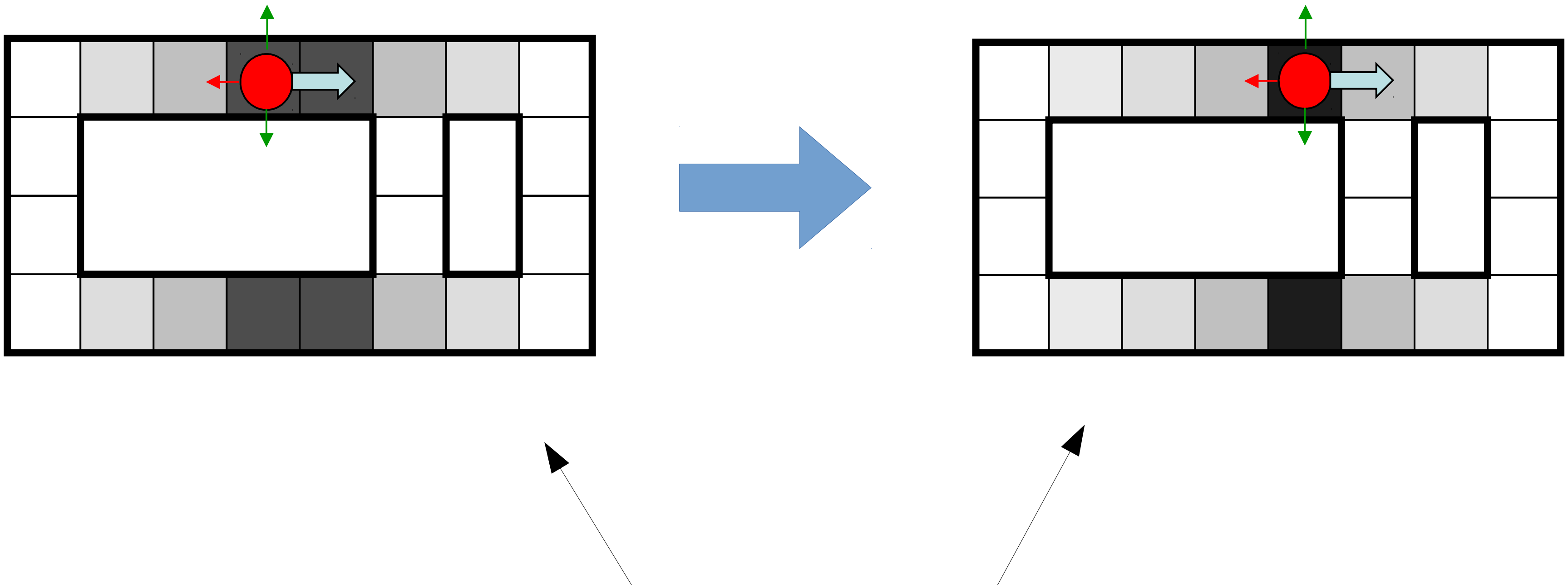


Gray level denotes estimated probability that robot is in that square

<u>Goal:</u> localize the robot based on sequential observations
– robot is given a map of the world; robot could be in any square
– initially, robot doesn't know which square it's in

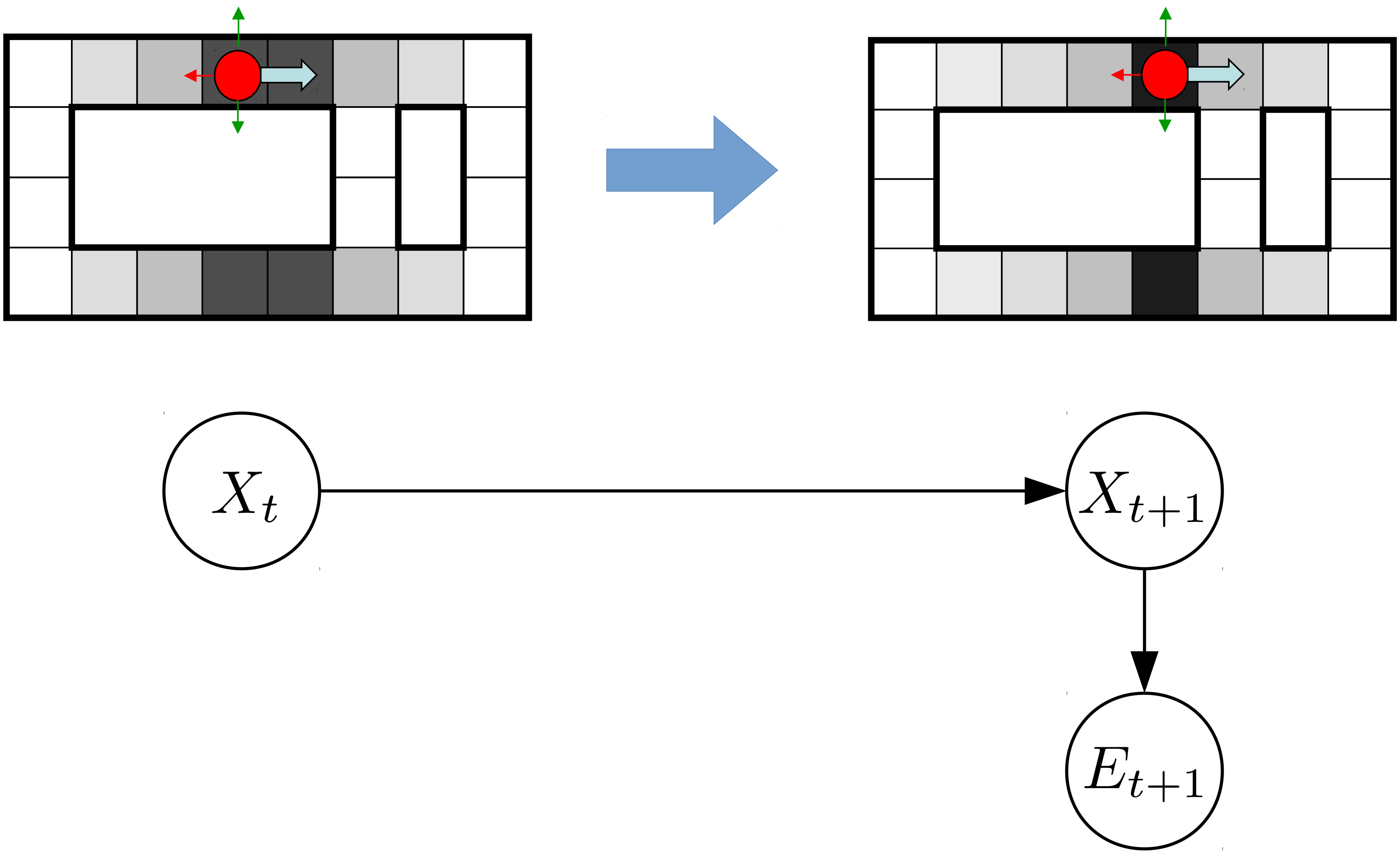
Image: Berkeley CS188 course notes (downloaded Summer 2015)

Bayes Filtering

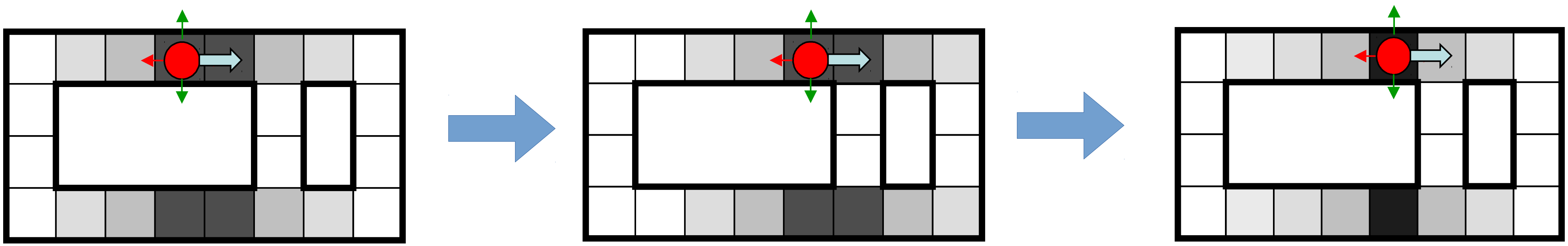


How do we go from <u>this</u> distribution to <u>this</u> distribution?

Bayes Filtering



Bayes Filtering



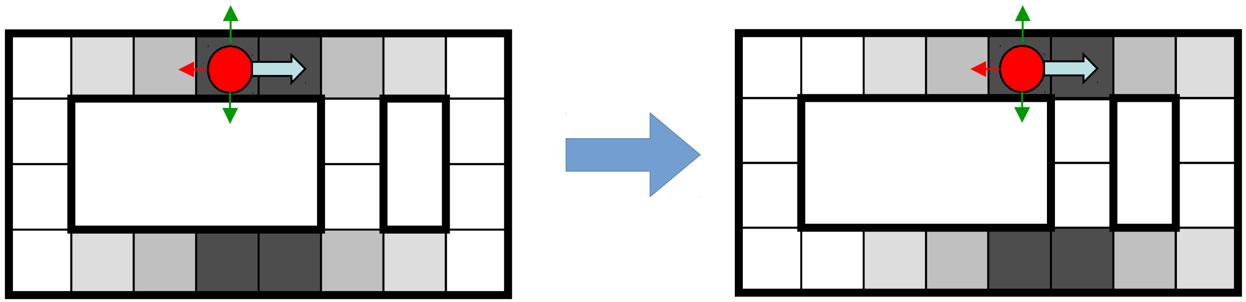
$$B(X_t) \qquad B'(X_{t+1}) \qquad B(X_{t+1})$$

$$\boxed{P(X_t|E_{1:t})} \longrightarrow \boxed{P(X_{t+1}|E_{1:t})} \longrightarrow \boxed{P(X_{t+1}|E_{1:t+1})}$$

Process update

Observation update

Process update

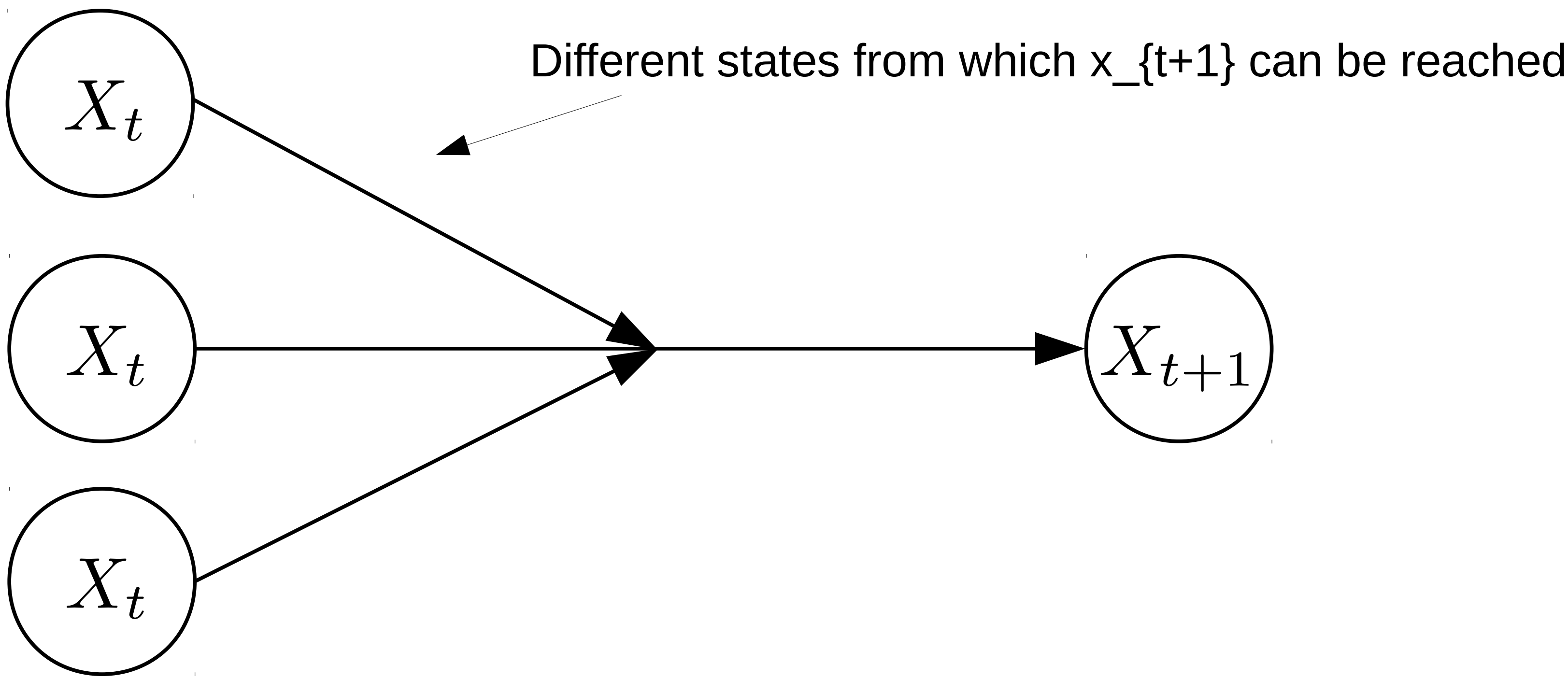


$$B(X_t) \qquad\qquad B'(X_{t+1})$$

$$\boxed{P(X_t|e_{1:t})} \longrightarrow \boxed{P(X_{t+1}|e_{1:t})}$$

Process update

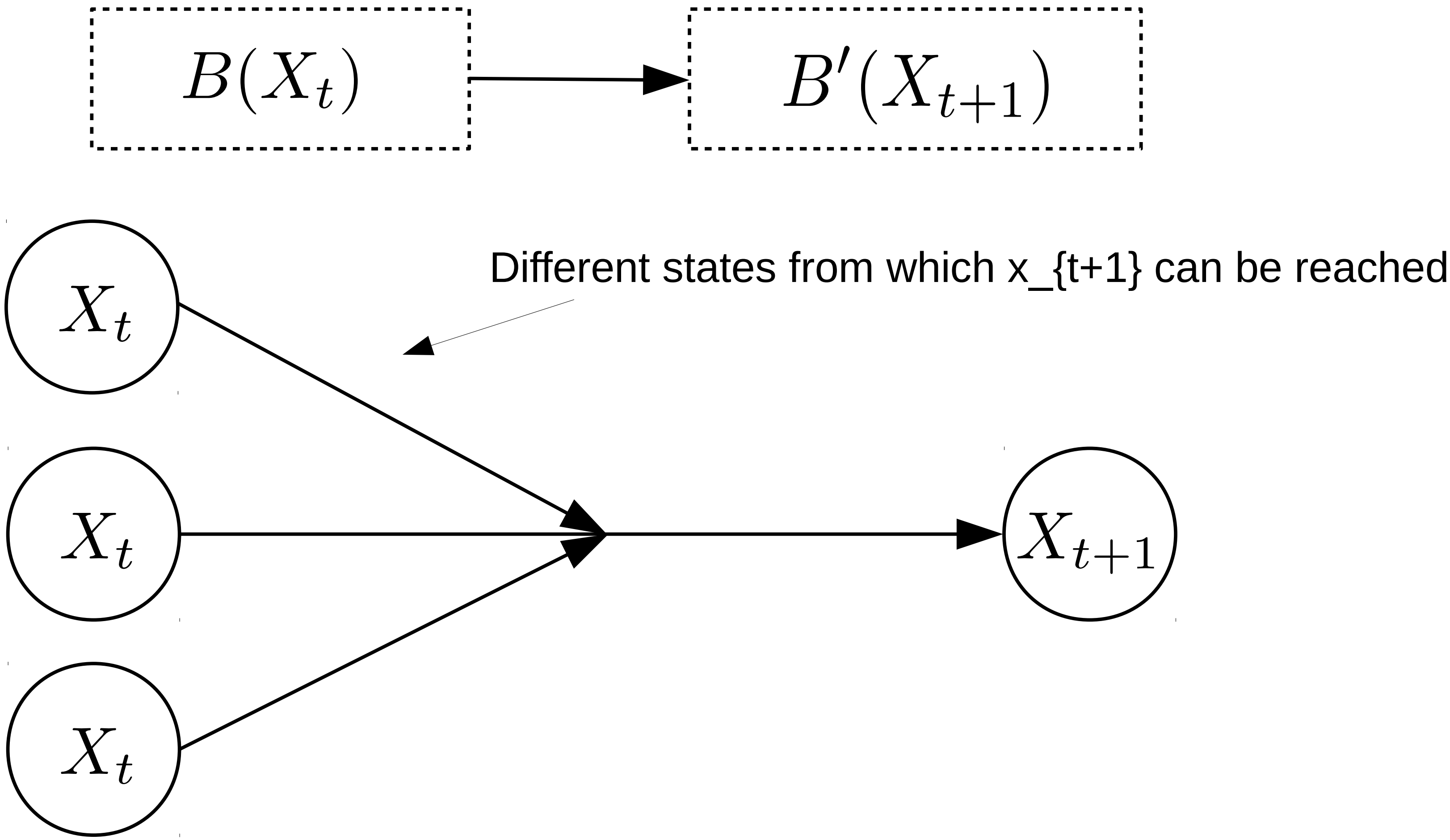
$$P(X_t|e_{1:t}) \longrightarrow P(X_{t+1}|e_{1:t})$$



$$P(X_{t+1}|e_{1:t}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t})P(X_t|e_{1:t})$$

Marginalize over
next states

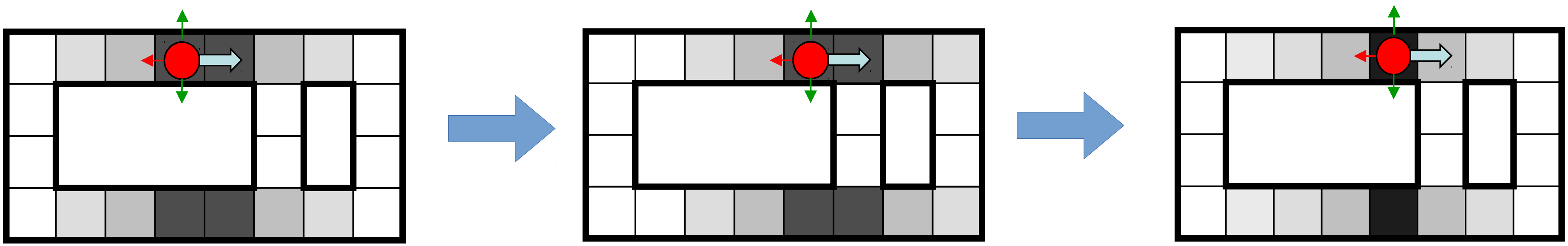
Process update



$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1} | X_t, e_{1:t}) B(X_t)$$

Marginalize over
next states

Bayes Filtering



$B(X_t)$ $\qquad$ $B'(X_{t+1})$ $\qquad$ $B(X_{t+1})$

$$\boxed{P(X_t|E_{1:t})} \longrightarrow \boxed{P(X_{t+1}|E_{1:t})} \longrightarrow \boxed{P(X_{t+1}|E_{1:t+1})}$$

Process update $\qquad$ Observation update

Observation update

$$\overset{B'(X_{t+1})}{\boxed{P(X_{t+1}|e_{1:t})}} \longrightarrow \overset{B(X_{t+1})}{\boxed{P(X_{t+1}|e_{1:t+1})}}$$

$$P(X_{t+1}|e_{1:t+1}) = \eta P(e_{t+1}|X_{t+1})P(X_{t+1}|e_{1:t})$$

Observation update

$$B'(X_{t+1}) \qquad\qquad B(X_{t+1})$$

$$\boxed{P(X_{t+1}|e_{1:t})} \longrightarrow \boxed{P(X_{t+1}|e_{1:t+1})}$$

$$P(X_{t+1}|e_{1:t+1}) = \eta \underbrace{P(e_{t+1}|X_{t+1})} P(X_{t+1}|e_{1:t})$$

Probability of seeing observation e_{t+1} from state X_{t+1}

Observation update

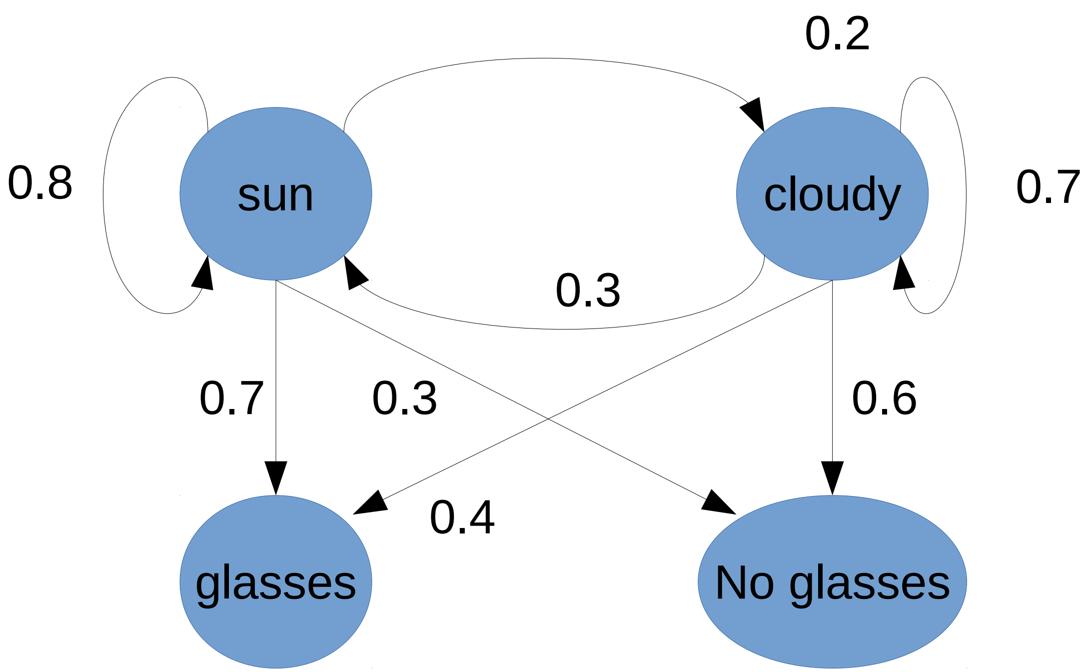
$$\underset{P(X_{t+1}|e_{1:t})}{B'(X_{t+1})} \longrightarrow \underset{P(X_{t+1}|e_{1:t+1})}{B(X_{t+1})}$$

$$P(X_{t+1}|e_{1:t+1}) = \eta P(e_{t+1}|X_{t+1})P(X_{t+1}|e_{1:t})$$

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

Where $\eta = \dfrac{1}{P(e_{t+1})}$ is a normalization factor

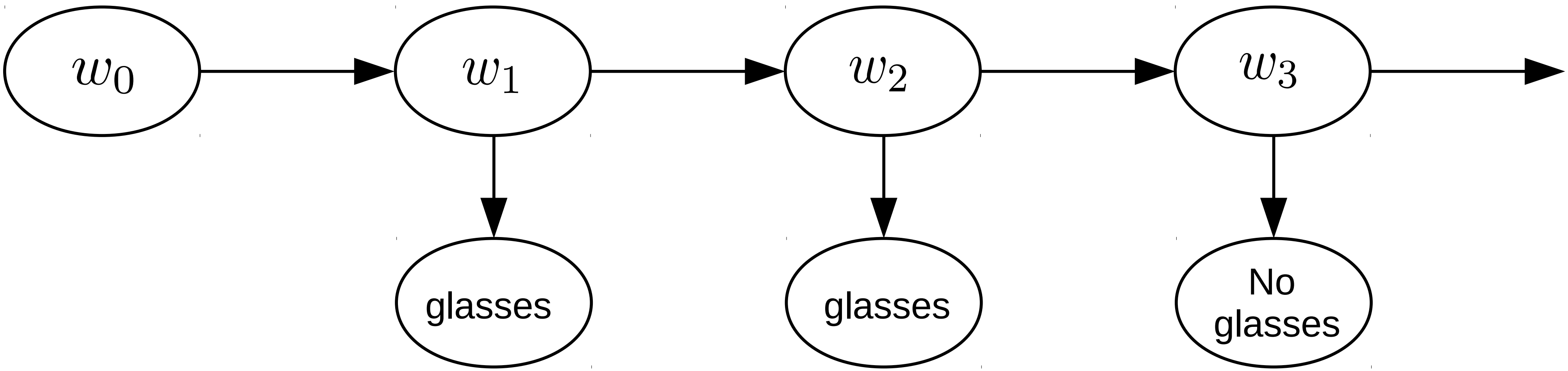
Weather HMM example



Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



w_t	P(w_t)
sun	0.5
cloudy	0.5

Process update:

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t})B(X_t)$$

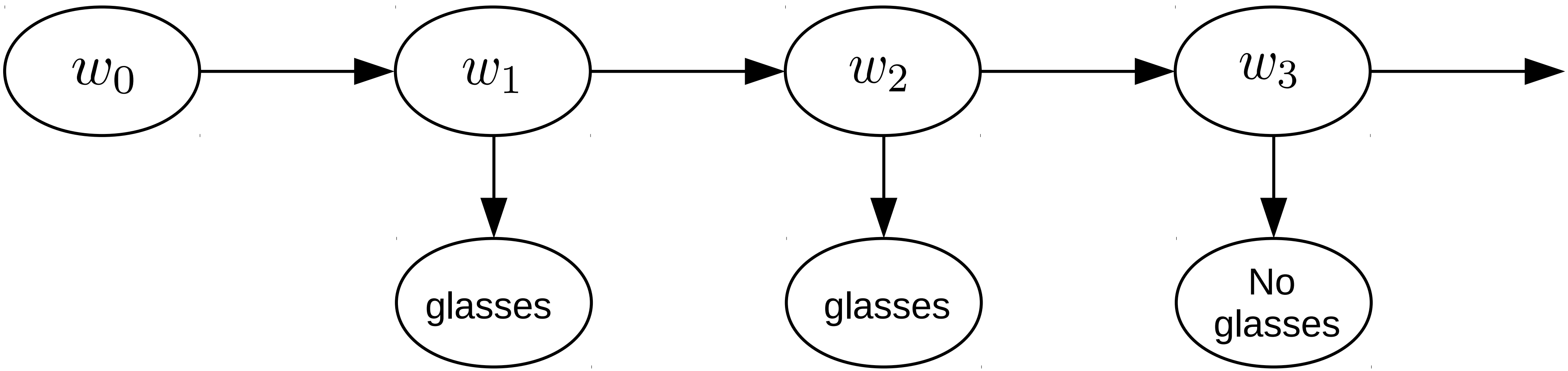
Observation update:

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



w_t	P(w_t)
sun	0.5
cloudy	0.5

w_t	P(w_t)
sun	?
cloudy	?

Process update:

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t})B(X_t)$$

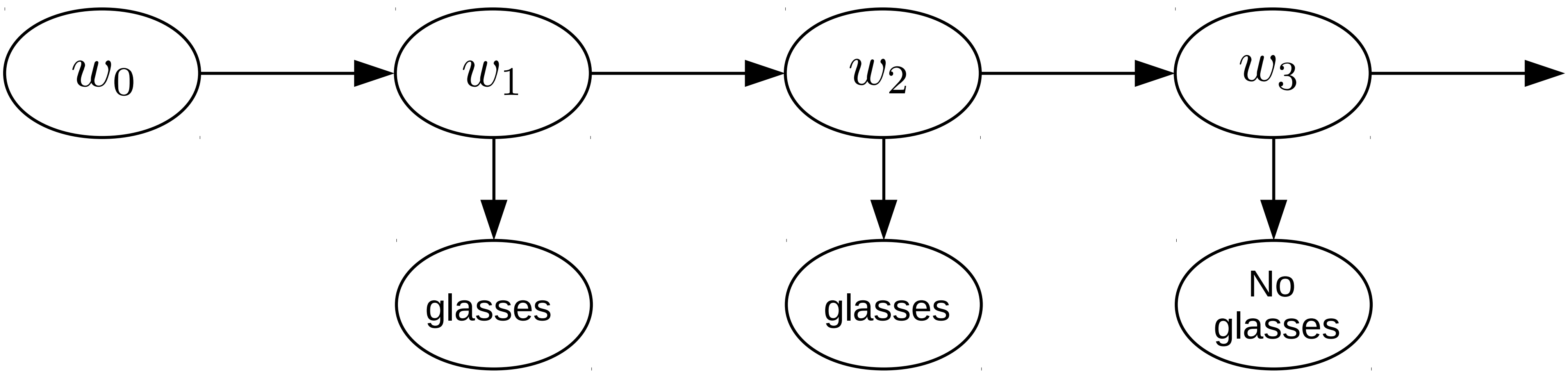
Observation update:

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



w_t	P(w_t)
sun	0.5
cloudy	0.5

w_t	P(w_t)
sun	0.55
cloudy	0.45

Process update:

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t}) B(X_t)$$

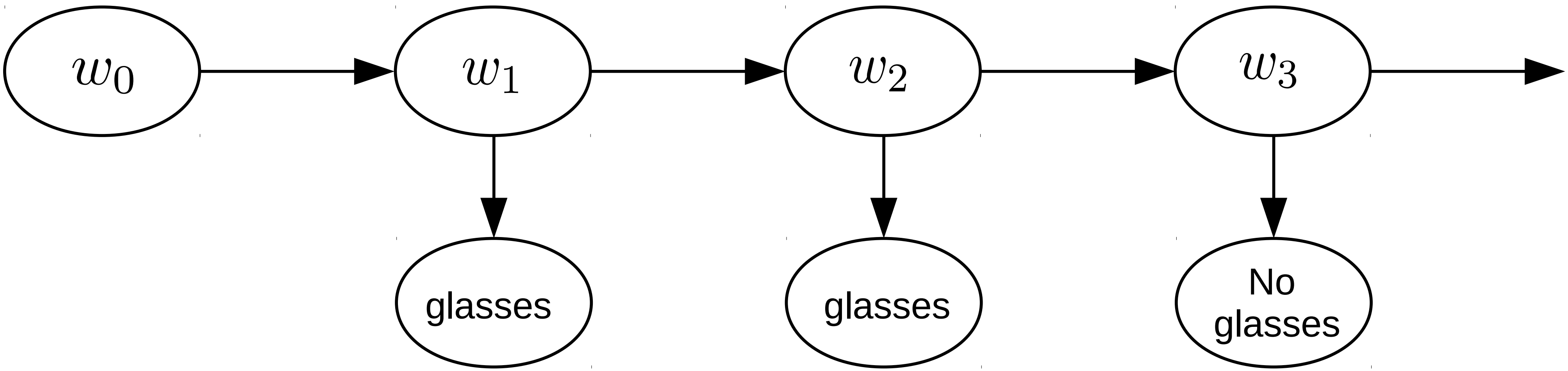
Observation update:

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1}) B'(X_{t+1})$$

Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



w_t	P(w_t)
sun	0.5
cloudy	0.5

w_t	P(w_t)
sun	0.55
cloudy	0.45

w_t	P(w_t)
sun	?
cloudy	?

Process update:

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t})B(X_t)$$

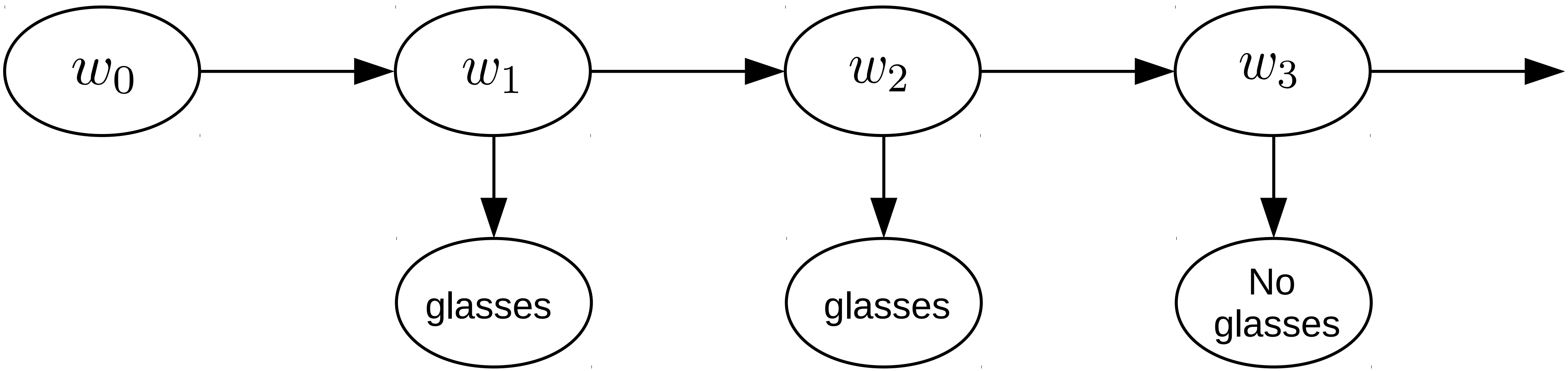
Observation update:

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



w_t	P(w_t)
sun	0.5
cloudy	0.5

w_t	P(w_t)
sun	0.55
cloudy	0.45

w_t	P(w_t)
sun	0.68
cloudy	0.31

<u>Process update:</u>

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t})B(X_t)$$

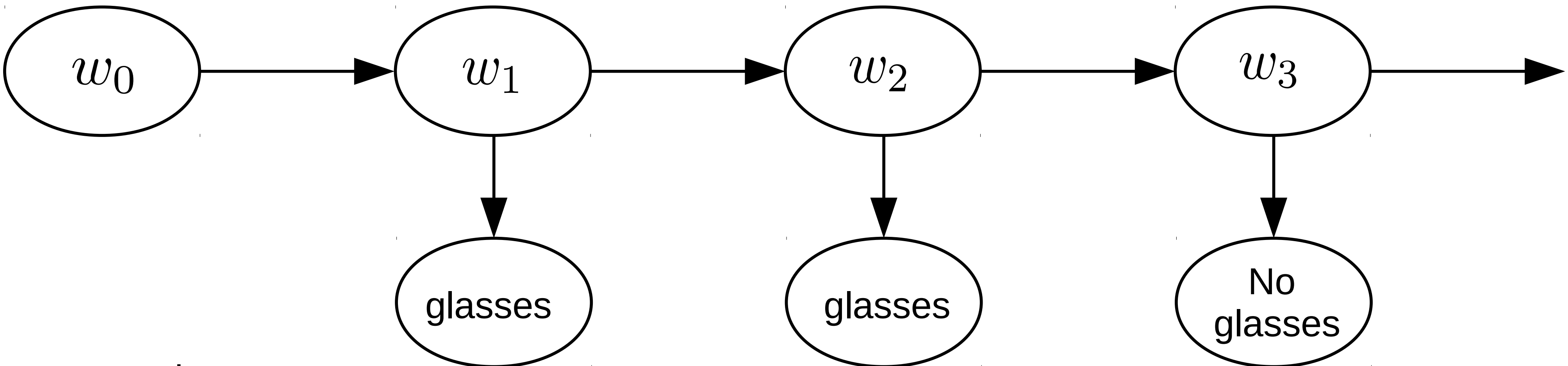
<u>Observation update:</u>

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



Process update:

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t})B(X_t)$$

Observation update:

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

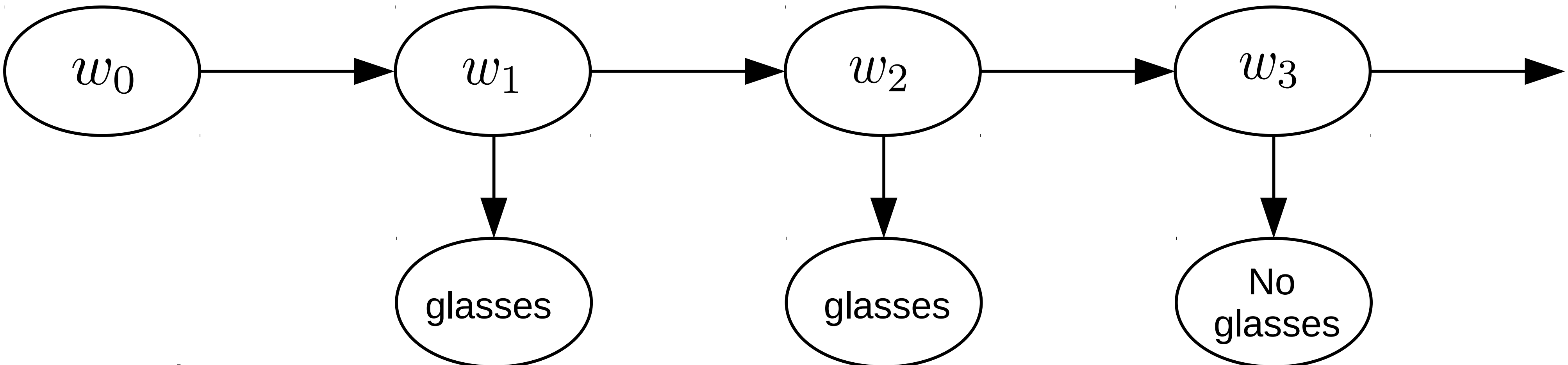
w_t	P(w_t)
sun	?
cloudy	?

w_t	P(w_t)
sun	0.68
cloudy	0.31

Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



<u>Process update:</u>

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t})B(X_t)$$

<u>Observation update:</u>

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

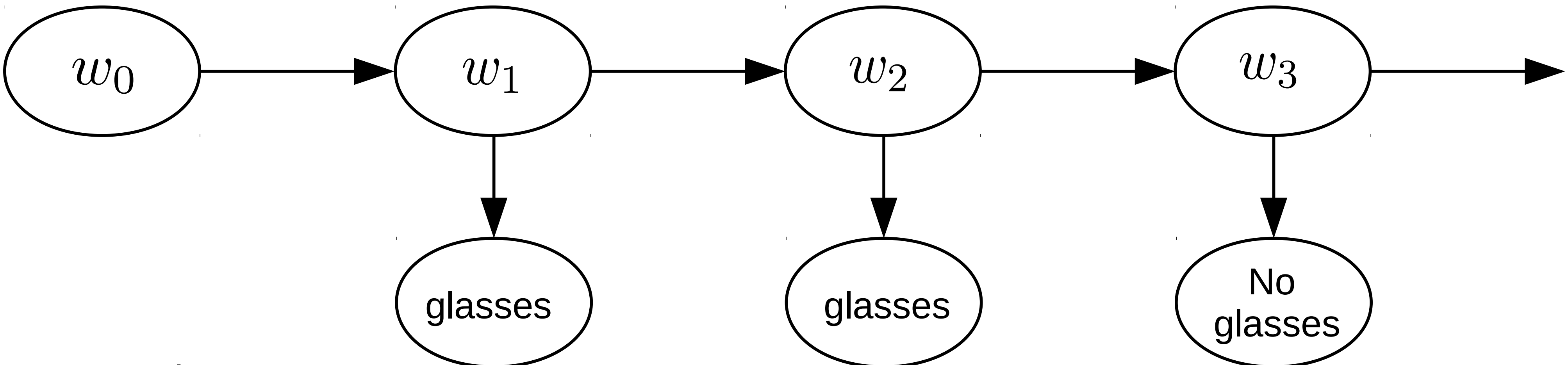
w_t	P(w_t)
sun	0.64
cloudy	0.36

w_t	P(w_t)
sun	0.68
cloudy	0.31

Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



Process update:

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t}) B(X_t)$$

Observation update:

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1}) B'(X_{t+1})$$

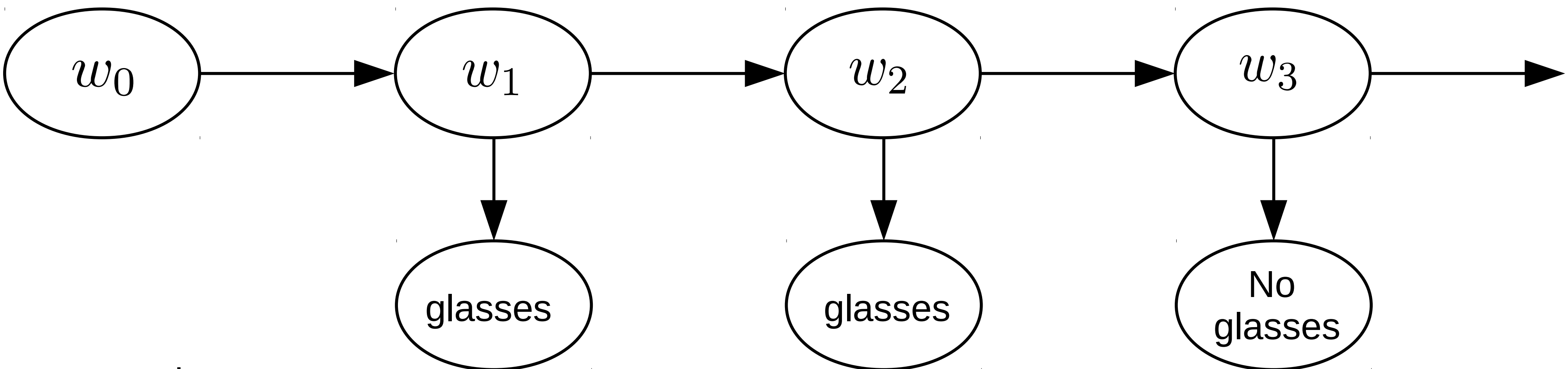
w_t	P(w_t)
sun	0.64
cloudy	0.36

w_t	P(w_t)
sun	?
cloudy	?

Weather HMM example

X_{t-1}	X_t	X_t
sun	sun	0.8
sun	cloudy	0.2
cloudy	sun	0.3
cloudy	cloudy	0.7

X_t	P(g_t|X_t)
sun	0.7
cloudy	0.4



Process update:

$$B'(X_{t+1}) = \sum_{X_t} P(X_{t+1}|X_t, e_{1:t})B(X_t)$$

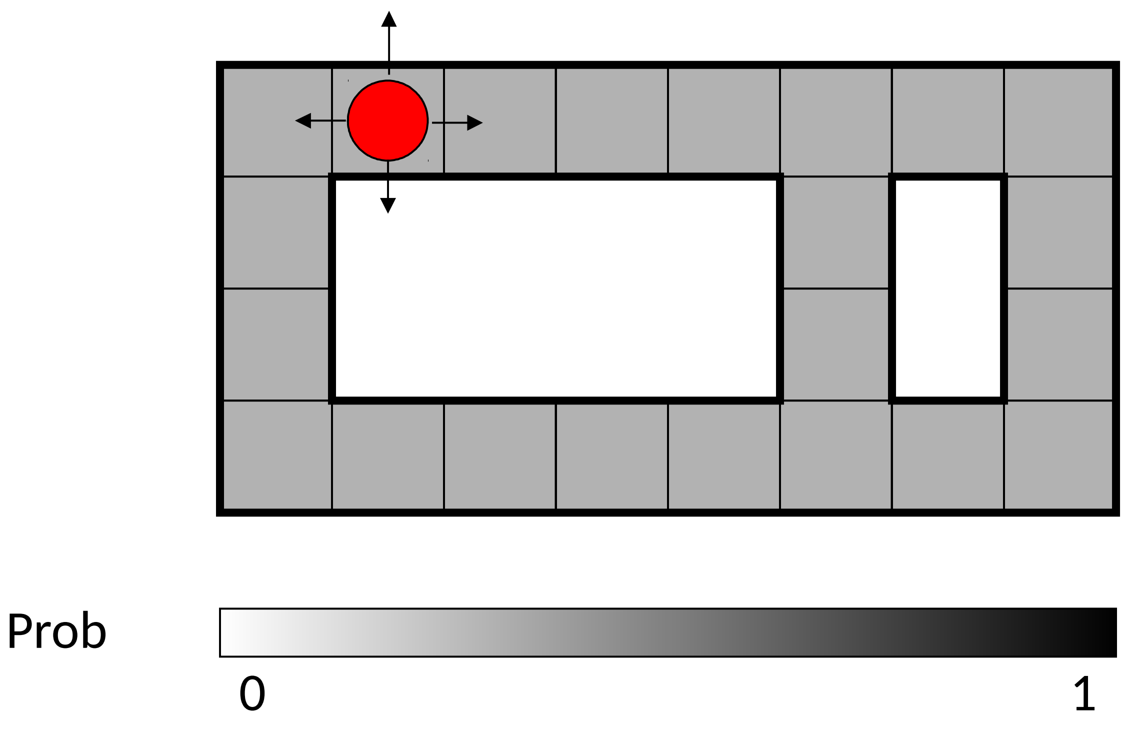
Observation update:

$$B(X_{t+1}) = \eta P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

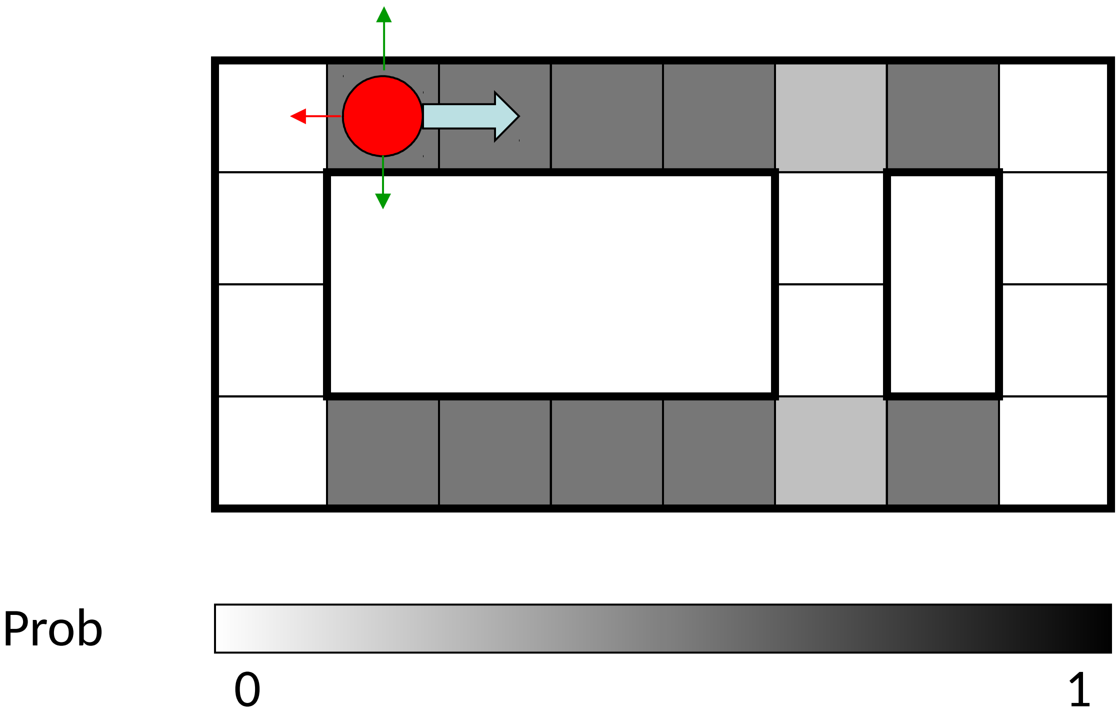
w_t	P(w_t)
sun	0.64
cloudy	0.36

w_t	P(w_t)
sun	0.76
cloudy	0.24

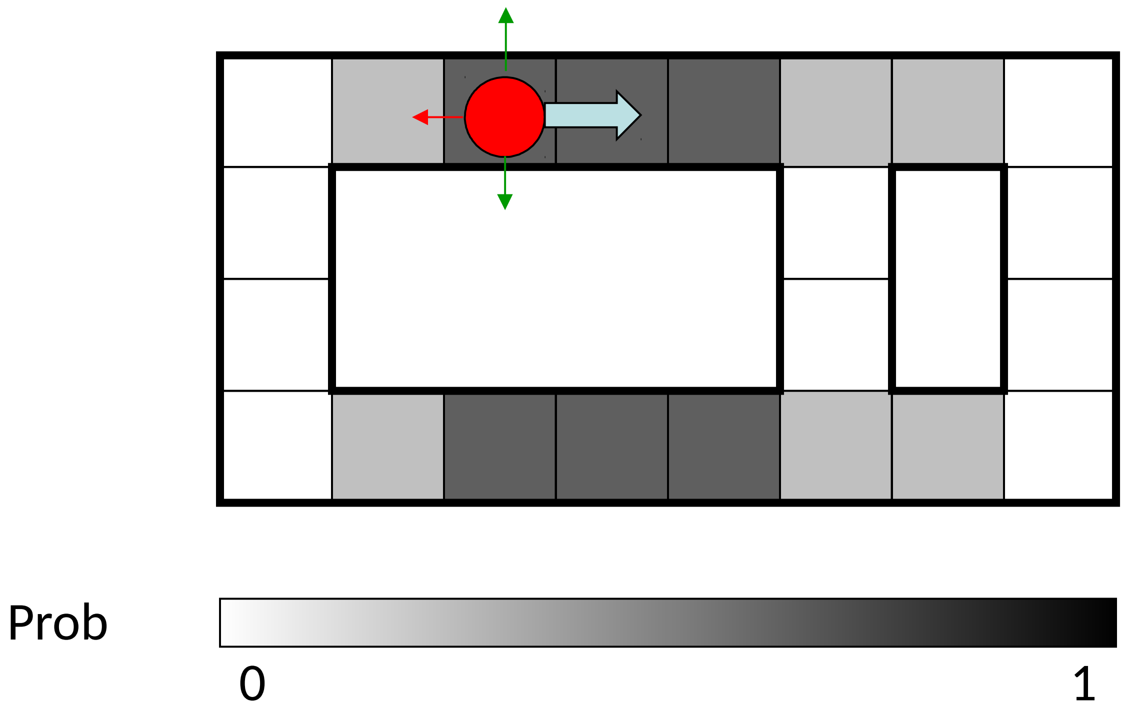
Robot localization example



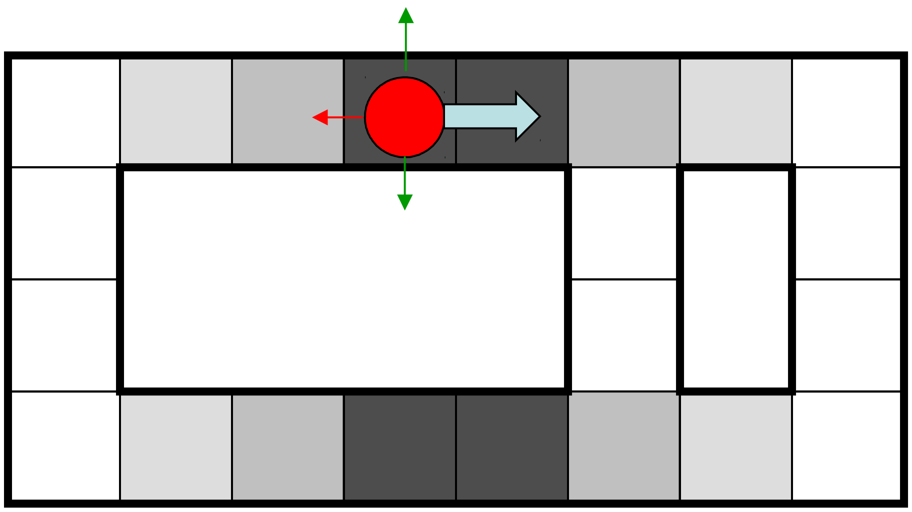
Robot localization example



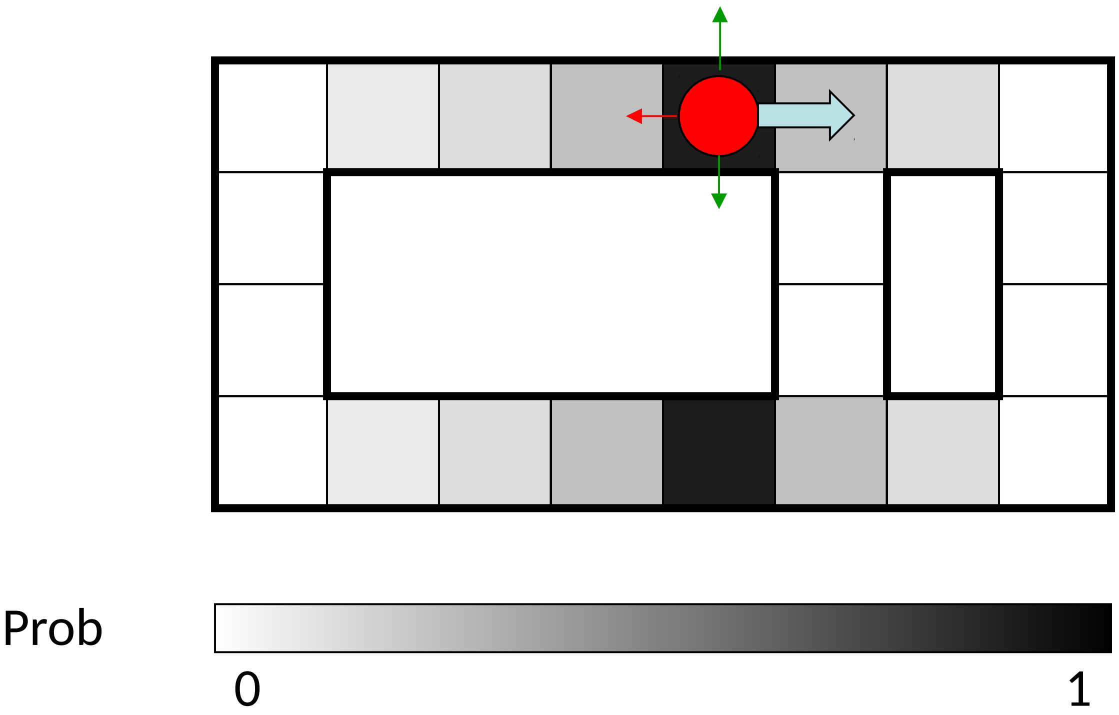
Robot localization example



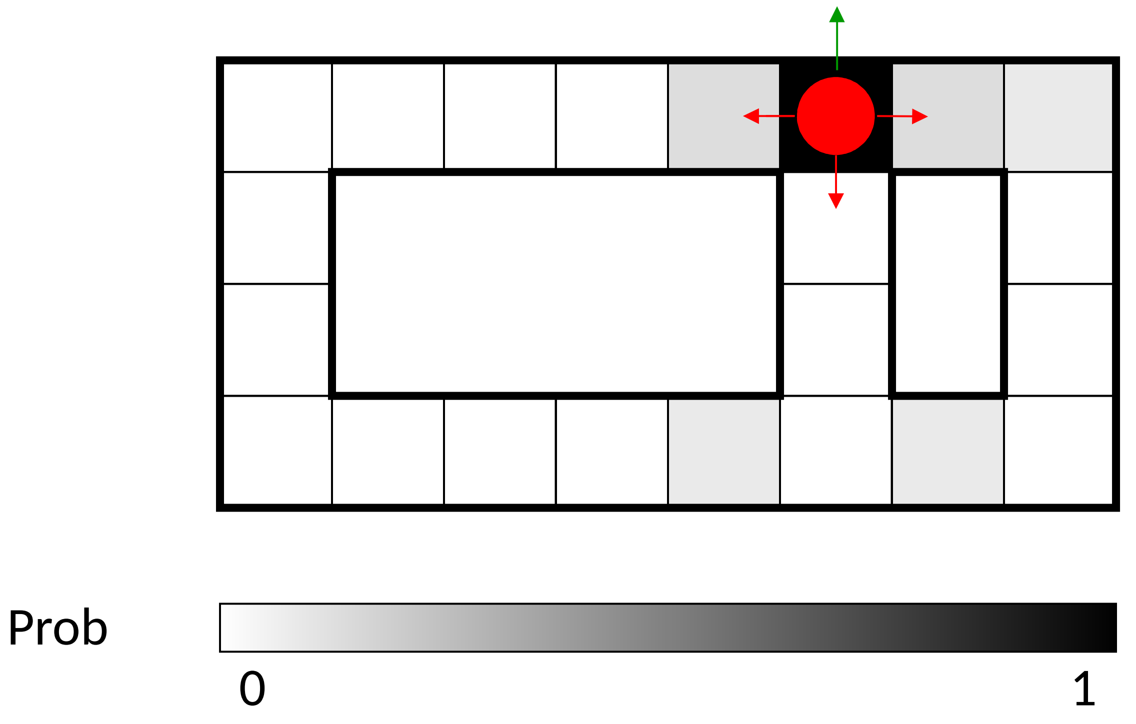
Robot localization example



Robot localization example



Robot localization example



Real world HMMs

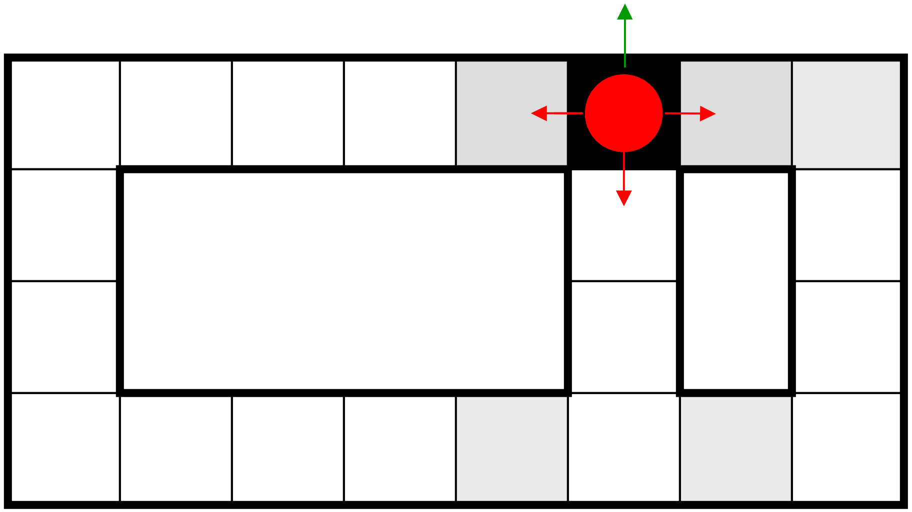
- Speech recognition HMMs:
- Observations are acoustic signals (continuous valued)
- States are specific positions in specific words (so, tens of thousands)

- 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)

Particle Filter

Why must I be confined to this grid?



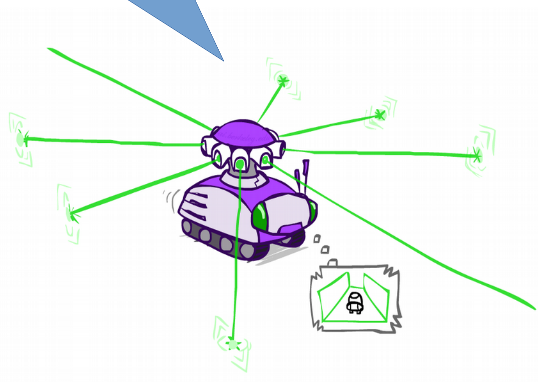
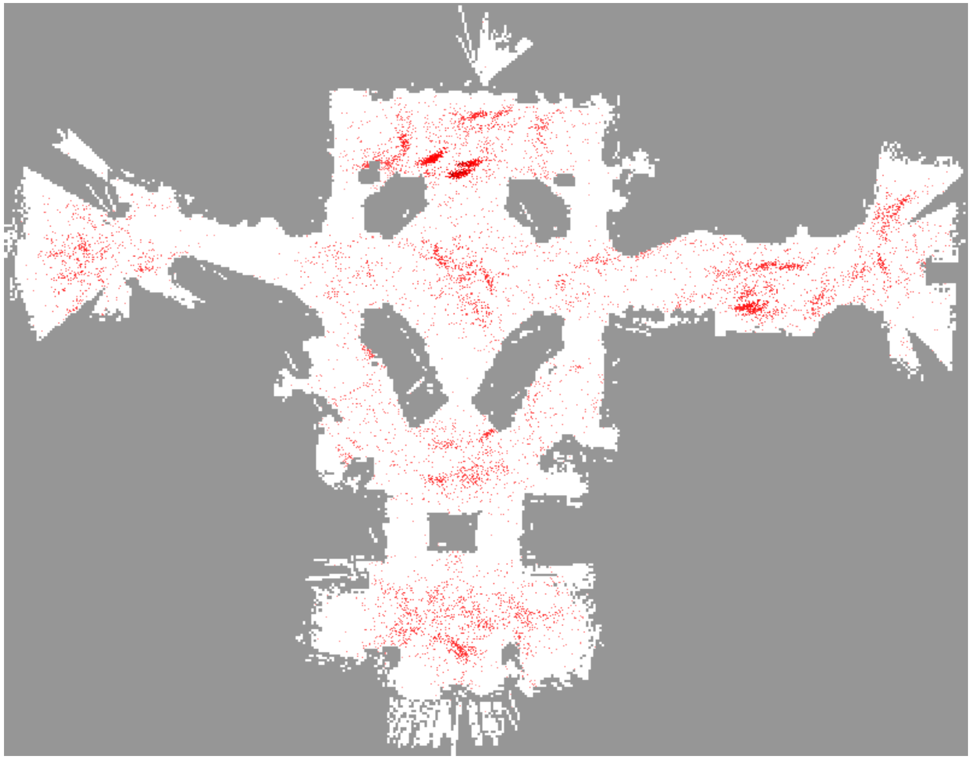


Image: Berkeley CS188 course notes (downloaded Summer 2015)

Particle Filter: a solution for continuous state spaces



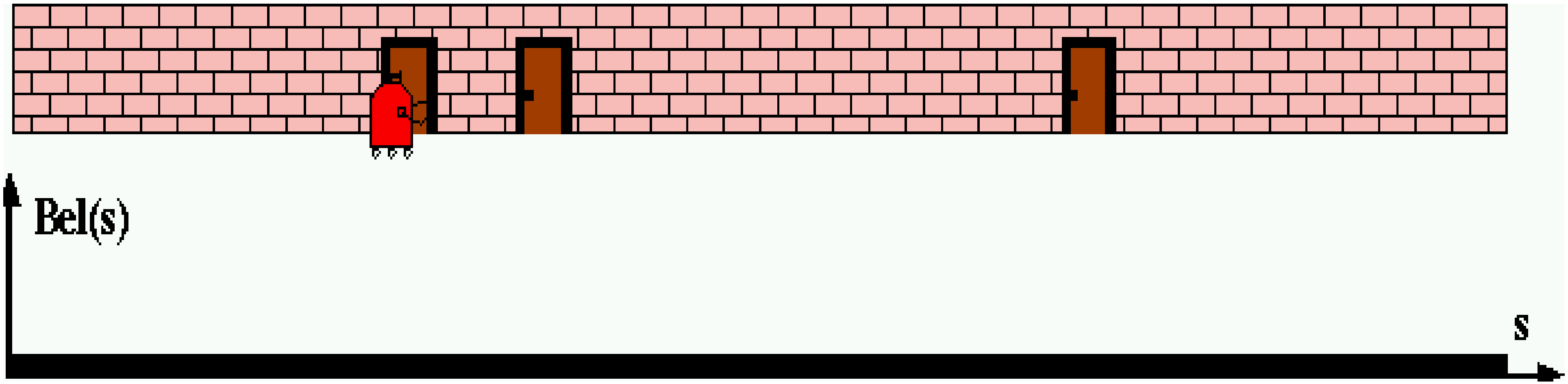
Sequential Bayes Filtering is great, but it's not great for continuous state spaces.
– you need to discretize the state space (e.g. a grid) in order to use Bayes filtering
  – but, doing filtering on a grid is not efficient...

Therefore:
- particle filters
- Kalman filters

Two different ways of filtering in continuous state spaces

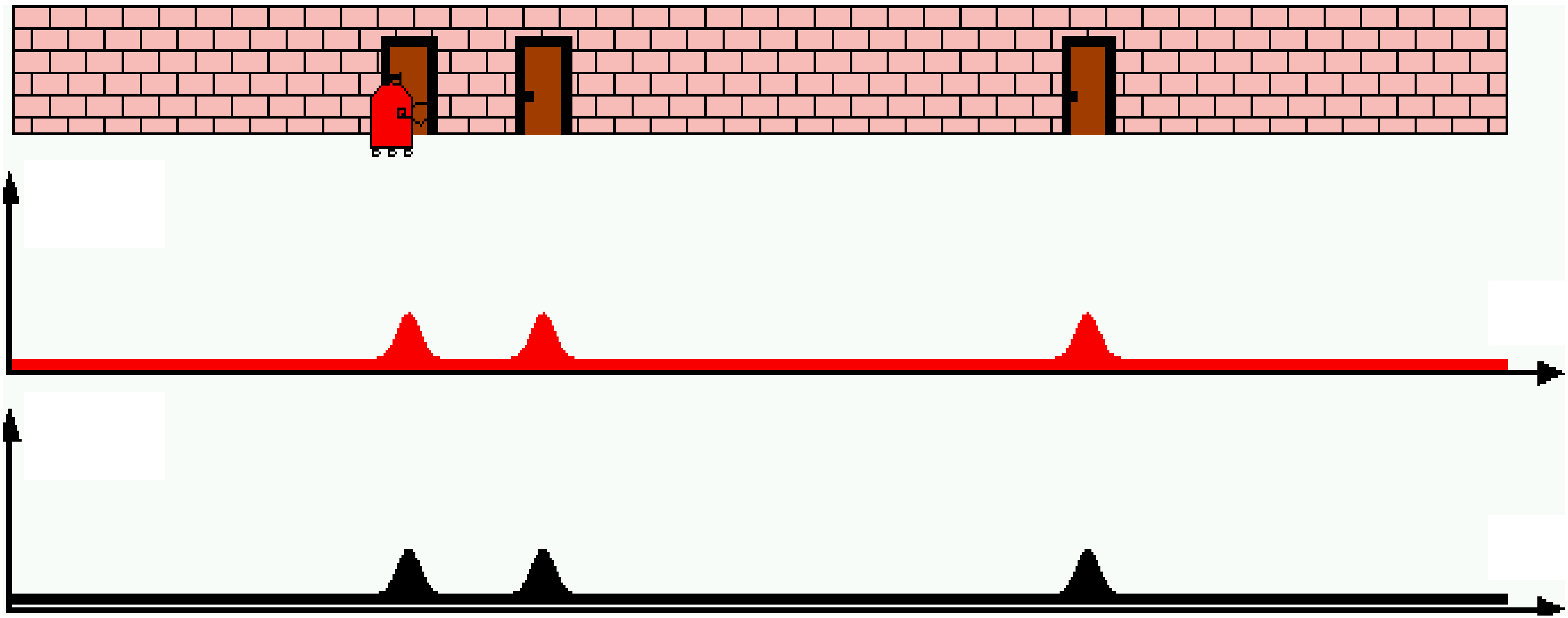
Image: Thrun CS223b Course Notes (downloaded Summer 2015)

Bayes Filtering in a Continuous Space: ideal case



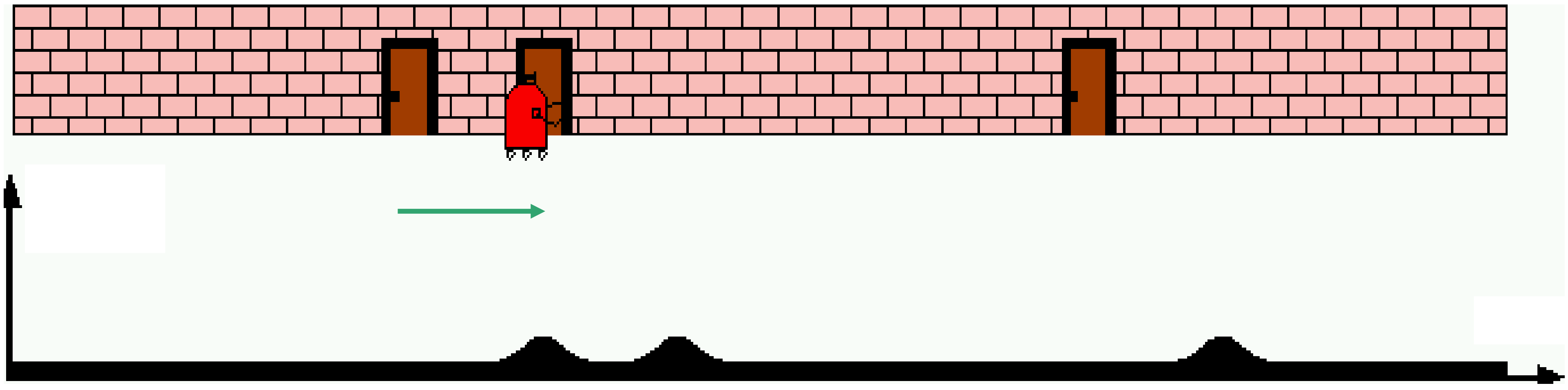
Prior belief distribution

Bayes Filtering in a Continuous Space: ideal case



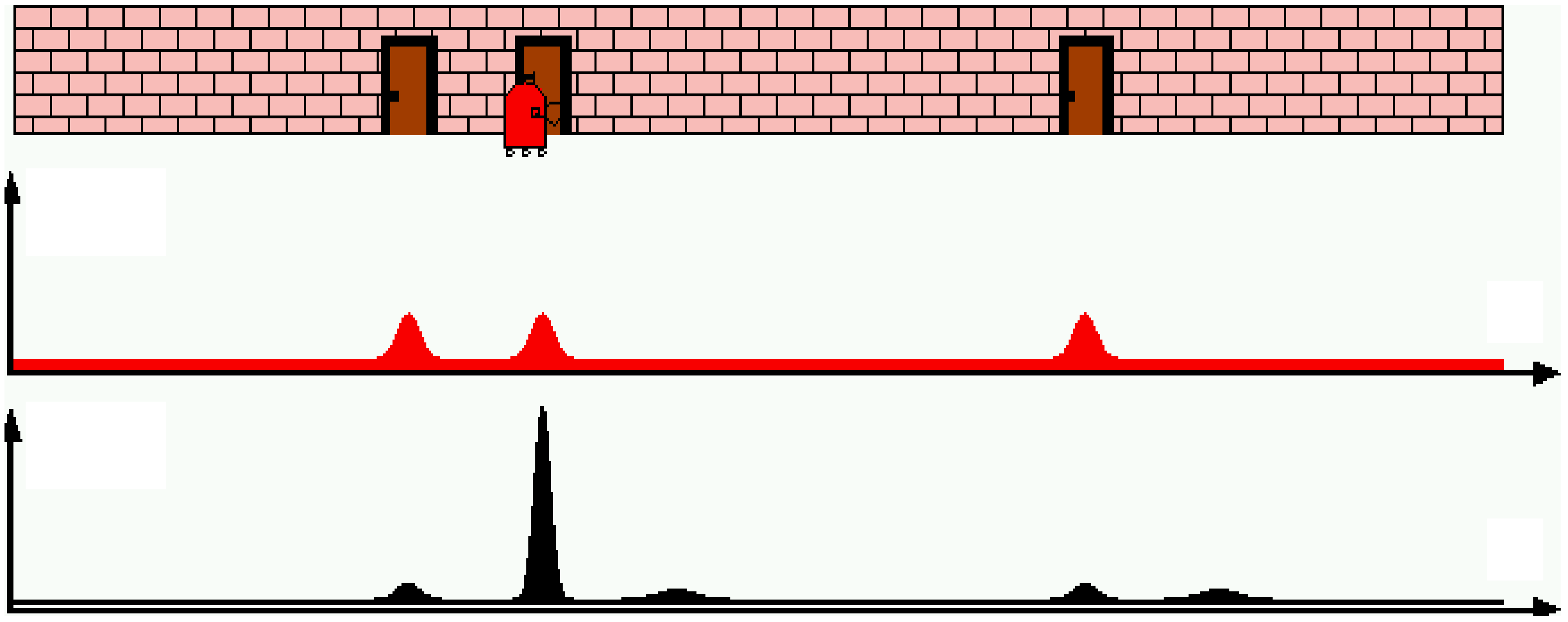
Observation update

Bayes Filtering in a Continuous Space: ideal case



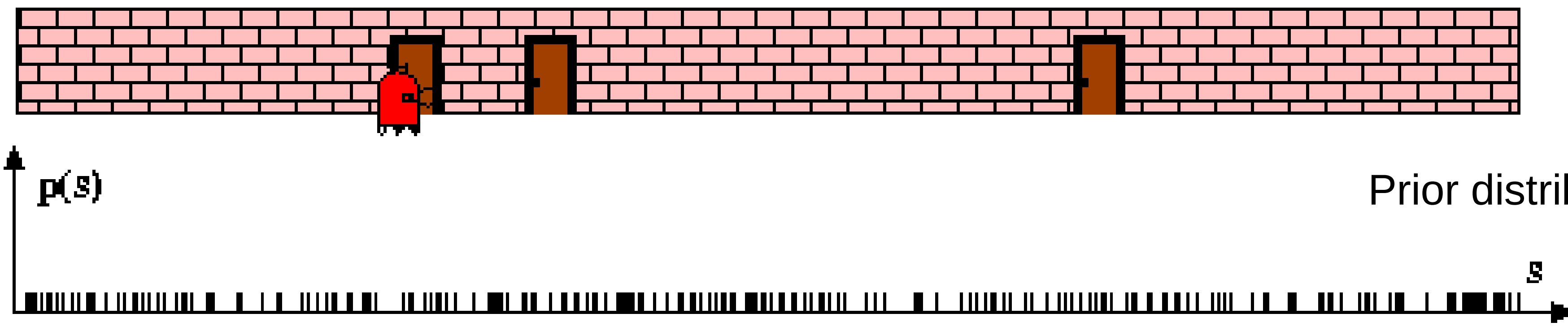
Process/Measurement update

Bayes Filtering in a Continuous Space: ideal case



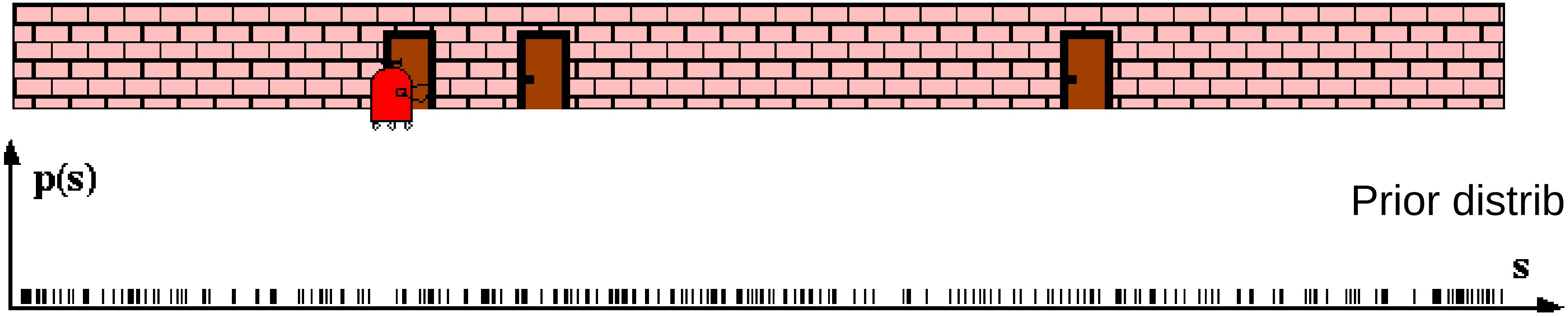
Posterior

Particle Filter

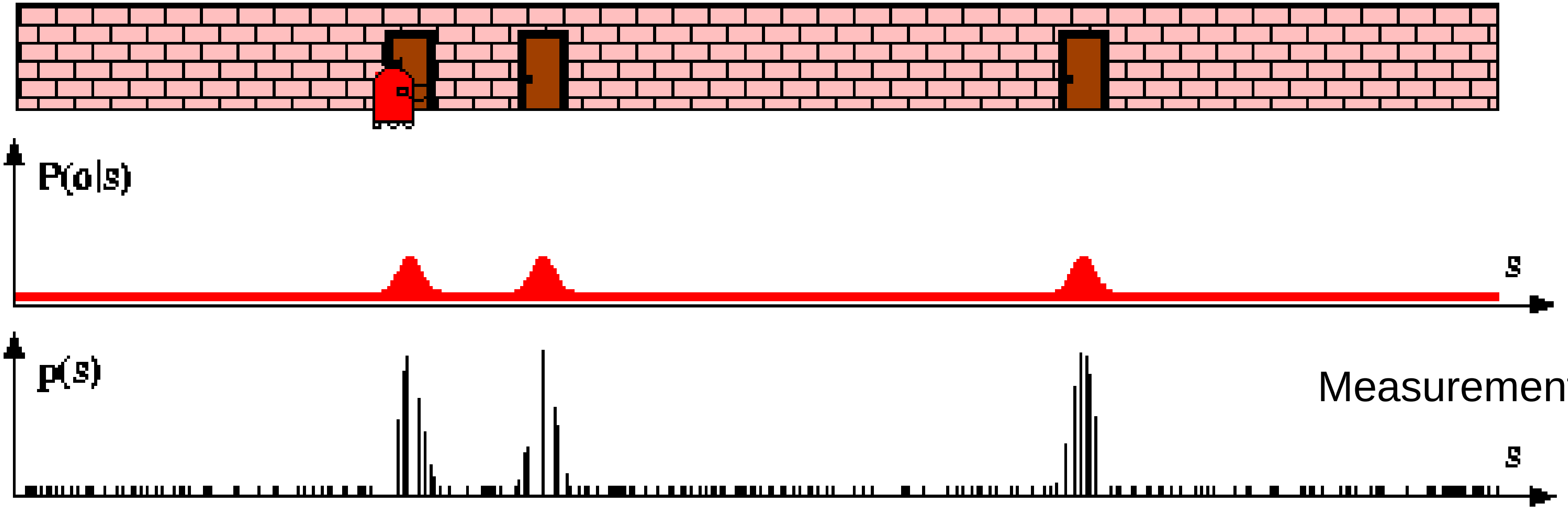


Prior distribution

Particle Filter

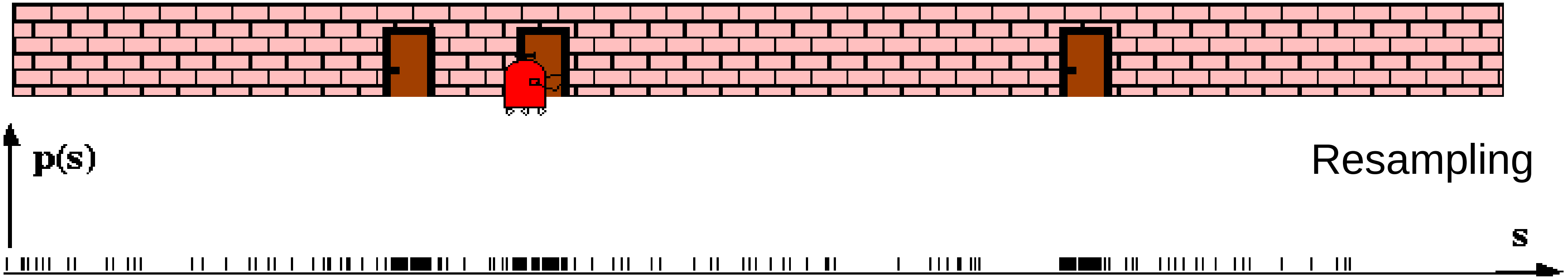


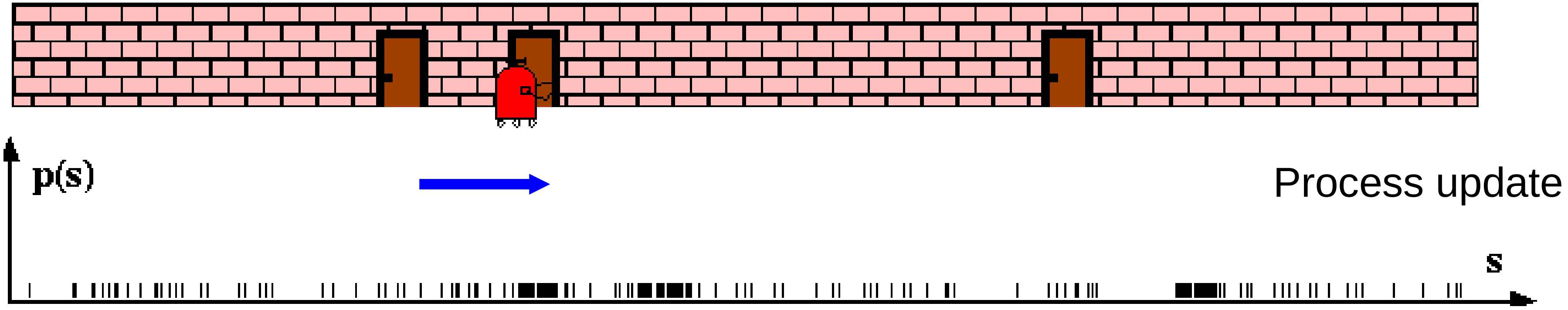
Prior distribution



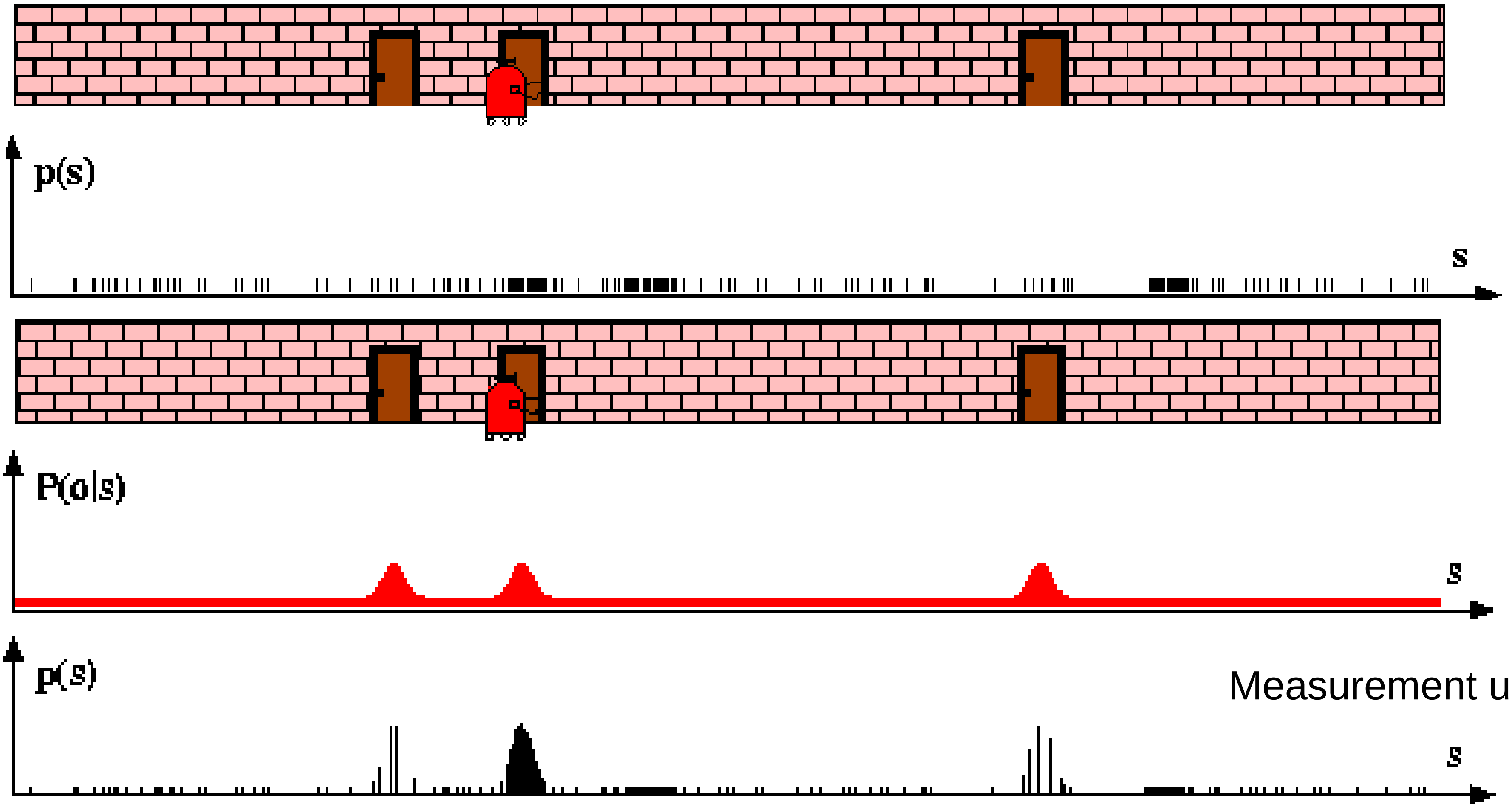
Measurement update

Particle Filter





Particle Filter



Measurement update

Time out for Monte Carlo Sampling!

Suppose we are given a probability distribution, $P(x)$

Suppose we are given a function, $f(x)$

How do we calculate the expected value of *f* over P?

$$E_{x \sim P(x)}(f(x)) = \int_x f(x)P(x)$$

Time out for Monte Carlo Sampling!

Suppose we are given a probability distribution, $P(x)$

Suppose we are given a function, $f(x)$

How do we calculate the expected value of *f* over P?

$$E_{x \sim P(x)}(f(x)) = \int_x f(x)P(x)$$

But, what if we don't have an analytical expression for P????

Time out for Monte Carlo Sampling!

Suppose we are given a probability distribution, $P(x)$

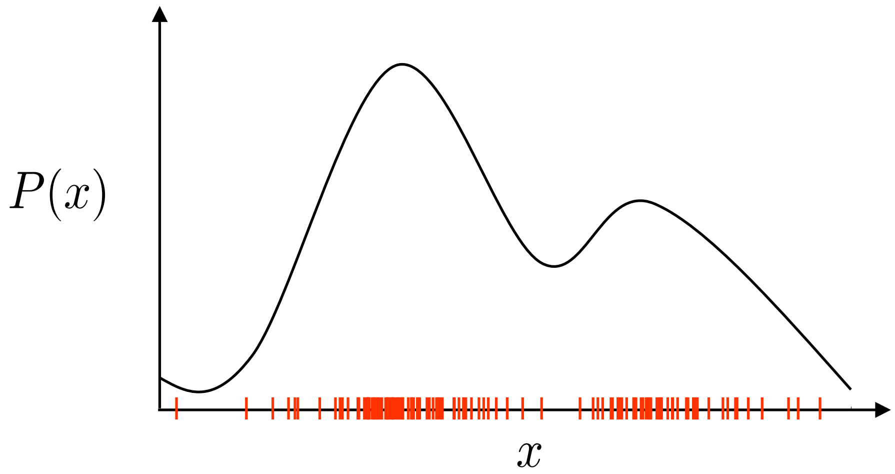
Suppose we are given a function, $f(x)$

How do we calculate the expected value of *f* over P?

$$E_{x \sim P(x)}(f(x)) = \int_x f(x) P(x)$$

But, what if we don’t have an analytical expression for P????

Sample!



Time out for Monte Carlo Sampling!

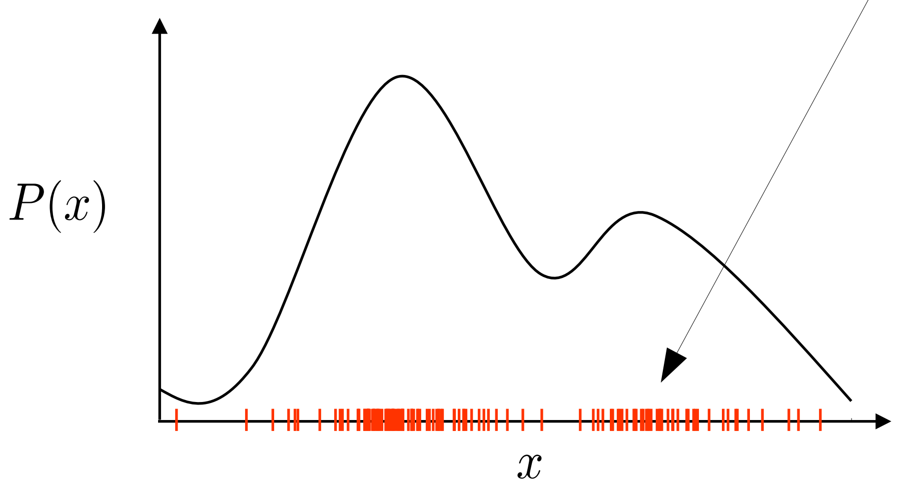
Suppose we are given a probability distribution, $P(x)$

Suppose we are given a function, $f(x)$

How do we calculate the expected value of *f* over P?

$$E_{x \sim P(x)}(f(x)) = \int_x f(x) P(x)$$

$$\approx \frac{1}{k} \sum_{i=1}^{k} f(x^i)$$ where x^i are samples from $P(x)$



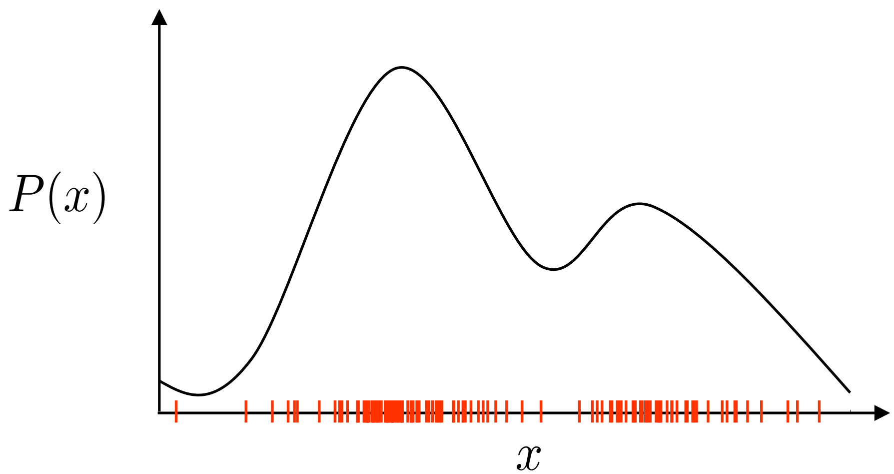
Time out for Monte Carlo Sampling!

Suppose we are given a probability distribution, $P(x)$

Suppose we are given a function, $f(x)$

How do we calculate the expected value of *f* over P?

But, what if we can't even sample from P?



Time out for Monte Carlo Sampling!

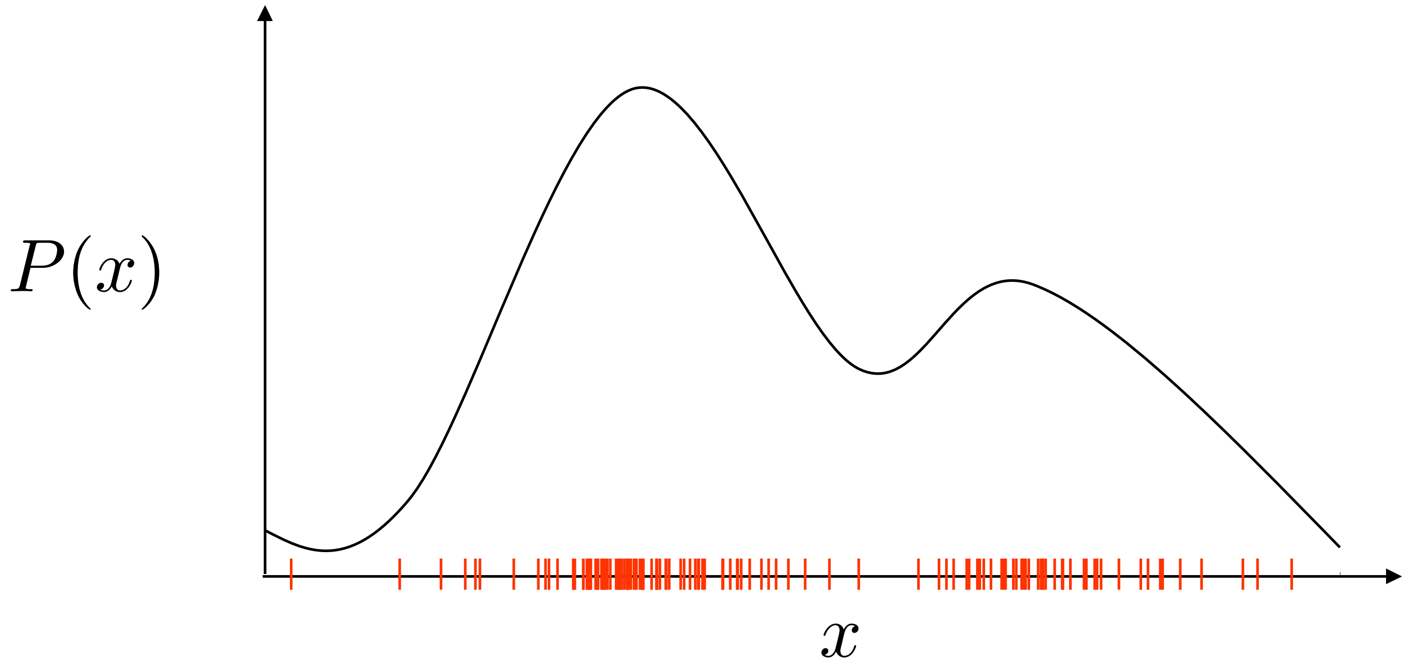
Suppose we are given a probability distribution, $P(x)$

Suppose we are given a function, $f(x)$

How do we calculate the expected value of *f* over P?

But, what if we can't even sample from P?

Particle Filter



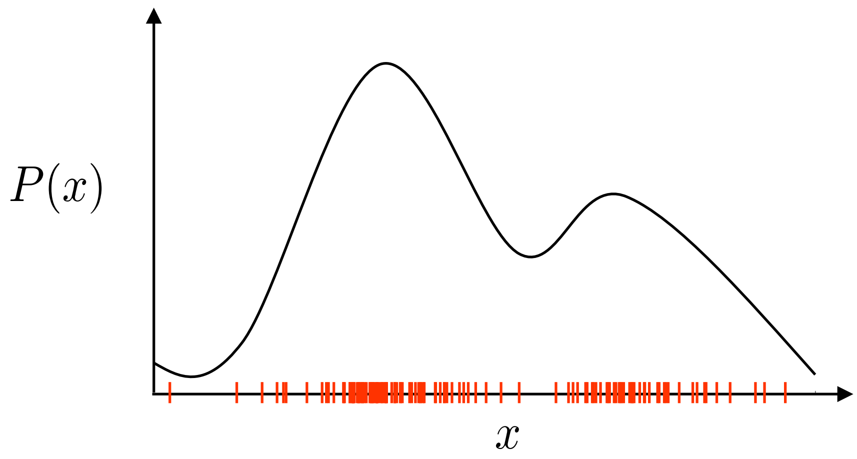
Key idea: represent a probability distribution as a finite set of points

– density of points encodes probability mass.

– particle filtering is an adaptation of Bayes filtering to this particle representation

Image: Thrun CS223b Course Notes (downloaded Summer 2015)

Monte Carlo Sampling



Suppose you are given an unknown probability distribution, $P(x)$

Suppose you can't evaluate the distribution analytically, but you can draw samples from it

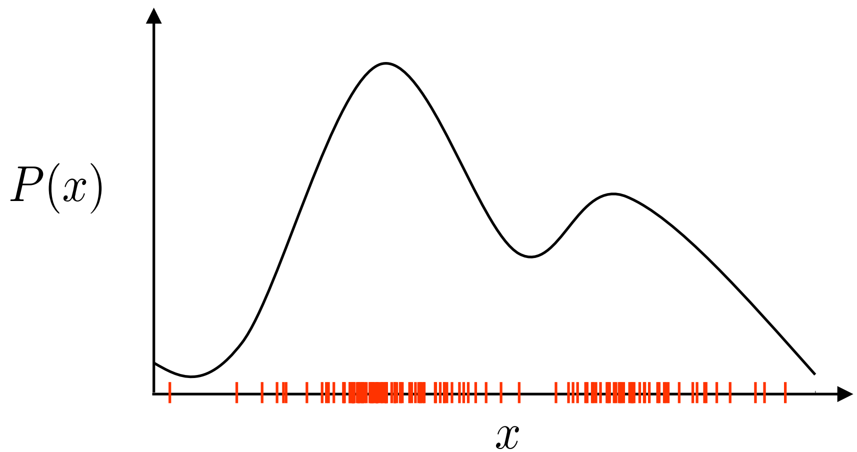
What can you do with this information?

$$E_{x \sim P(x)}(f(x)) = \int_x f(x) P(x)$$

$$\approx \frac{1}{k} \sum_{i=1}^{k} f(x^i)$$ where x^i are samples drawn from $P(x)$

Image: Thrun CS223b Course Notes (downloaded Summer 2015)

Monte Carlo Sampling



Suppose you are given an unknown probability distribution, $P(x)$

Suppose you can't evaluate the distribution analytically, <u>but you can draw samples from it</u>

What can you do with this information?

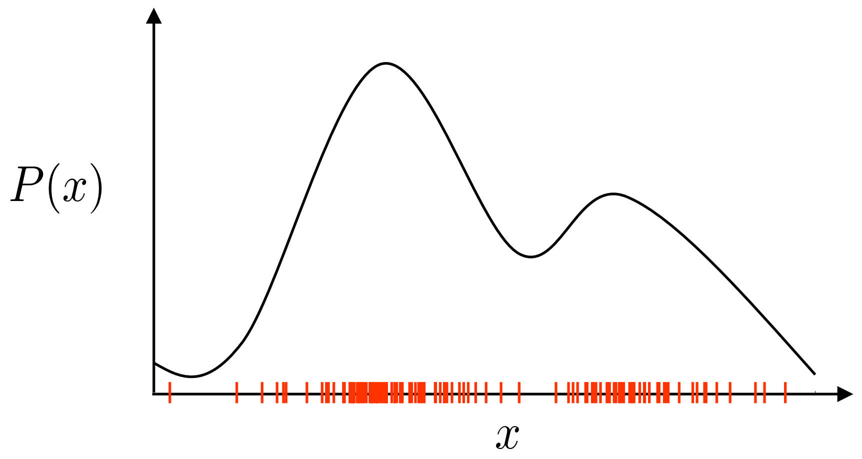
$$E_{x\sim P(x)}(f(x)) = \int_x f(x)P(x)$$

FYI:
You can use the same strategy to estimate other moments as well...

$$\approx \frac{1}{k}\sum_{i=1}^{k} f(x^i)$$ where x^i are samples drawn from $P(x)$

Image: Thrun CS223b Course Notes (downloaded Summer 2015)

Importance Sampling



Suppose you are given an unknown probability distribution, $P(x)$

Suppose you can't evaluate the distribution analytically, ~~but you can draw samples from it~~

What can you do with this information?

Suppose you can't even sample from it?

Suppose that all you can do is evaluate the function at a given point?

Image: Thrun CS223b Course Notes (downloaded Summer 2015)

Importance Sampling

Question: how estimate expected values if cannot draw samples from f(x)
– suppose all we can do is evaluate f(x) at a given point...

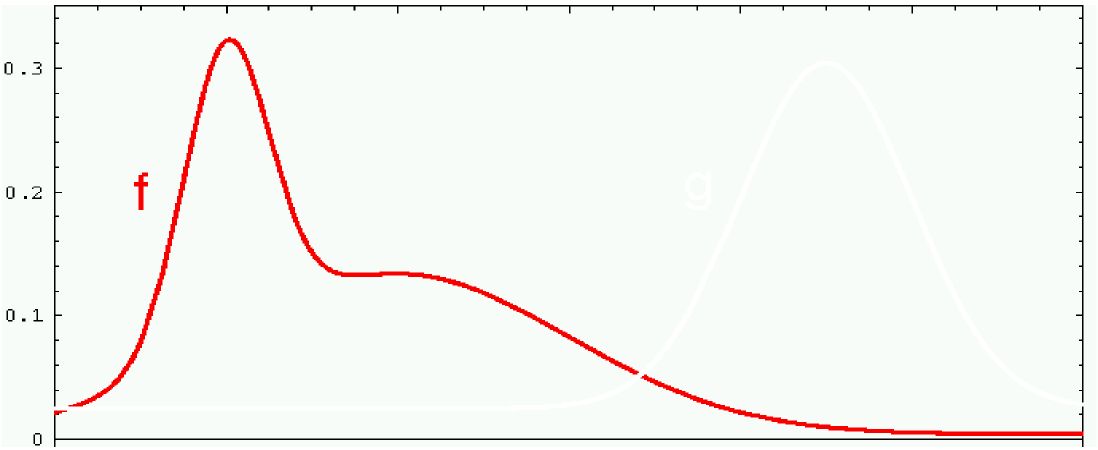
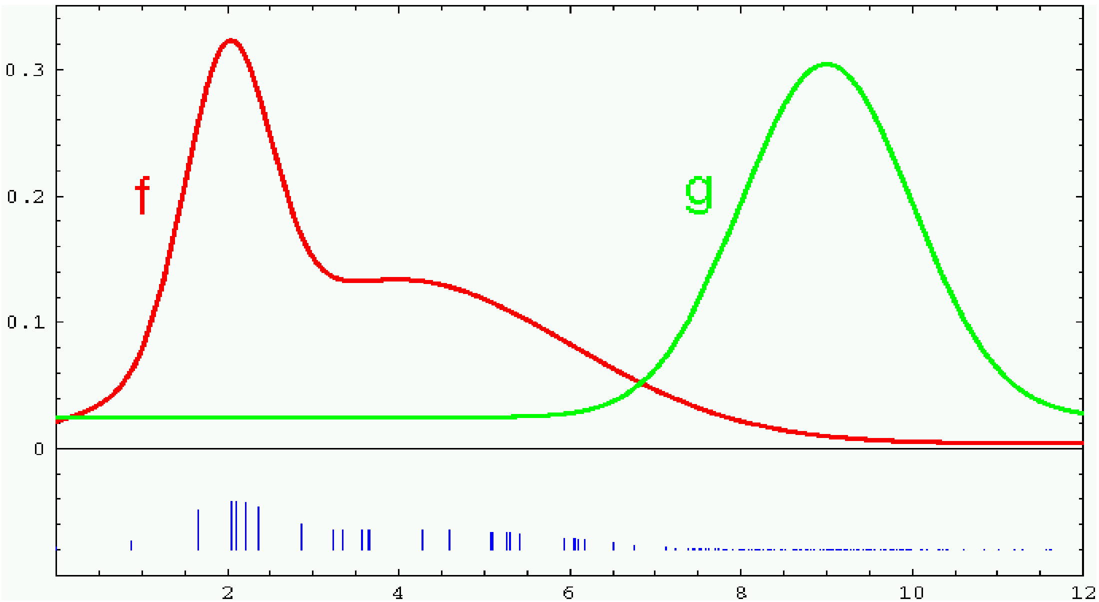


Image: Thrun CS223b Course Notes (downloaded Summer 2015)

Importance Sampling

Question: how estimate expected values if cannot draw samples from f(x)
– suppose all we can do is evaluate f(x) at a given point...

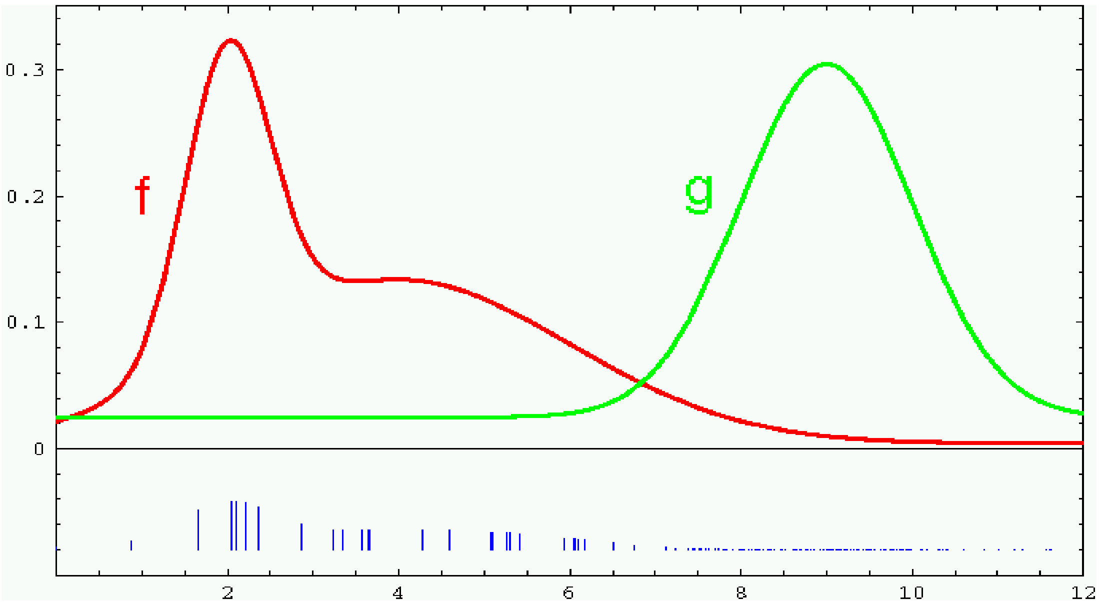


<u>Answer</u>: draw samples from a different distribution and weight them

Image: Thrun CS223b Course Notes (downloaded Summer 2015)

Importance Sampling

Question: how estimate expected values if cannot draw samples from f(x)
– suppose all we can do is evaluate f(x) at a given point...



Answer: draw samples from a different distribution and weight them

$$E_{x \sim f(x)}(h(x)) = \int_x h(x) \frac{f(x)}{g(x)} g(x)$$

$$\approx \frac{1}{k} \sum_{i=1}^{k} h(x^i) w_i$$

where x^i are samples drawn from $g(x)$

Proposal distribution

and $w_i = f(x^i)/g(x^i)$

Image: Thrun CS223b Course Notes (downloaded Summer 2015)

Particle Filter

Prior distribution

$$x_t^1, \ldots, x_t^n \qquad w_t^1, \ldots, w_t^n = 1$$

$$B(X_t) \qquad \boxed{P(X_t|E_{1:t})}$$

$$\downarrow$$

$$B'(X_t) \qquad \boxed{P(X_{t+1}|E_{1:t})}$$

$$\downarrow$$

$$B(X_{t+1}) \qquad \boxed{P(X_{t+1}|E_{1:t+1})}$$

Particle Filter

$B(X_t)$ $\boxed{P(X_t|E_{1:t})}$

$\downarrow$

$B'(X_t)$ $\boxed{P(X_{t+1}|E_{1:t})}$

$\downarrow$

$B(X_{t+1})$ $\boxed{P(X_{t+1}|E_{1:t+1})}$

<u>Prior distribution</u>

$$x_t^1, \ldots, x_t^n \qquad w_t^1, \ldots, w_t^n = 1$$

<u>Process update</u>

$$\bar{x}_{t+1}^i \sim P(X_{t+1}|x_t^i, e_{1:t})$$

Particle Filter

$B(X_t)$ $\boxed{P(X_t|E_{1:t})}$

$\downarrow$

$B'(X_t)$ $\boxed{P(X_{t+1}|E_{1:t})}$

$\downarrow$

$B(X_{t+1})$ $\boxed{P(X_{t+1}|E_{1:t+1})}$

<u>Prior distribution</u>

$$x_t^1, \ldots, x_t^n \qquad w_t^1, \ldots, w_t^n = 1$$

<u>Process update</u>

$$\bar{x}_{t+1}^i \sim P(X_{t+1}|x_t^i, e_{1:t})$$

<u>Observation update</u>

$$w_{t+1}^i = P(e_{t+1}|\bar{x}_{t+1}^i)w_t^i$$

Particle Filter

$B(X_t)$ $\boxed{P(X_t|E_{1:t})}$

$\downarrow$

$B'(X_t)$ $\boxed{P(X_{t+1}|E_{1:t})}$

$\downarrow$

$B(X_{t+1})$ $\boxed{P(X_{t+1}|E_{1:t+1})}$

Do this *n* times $\rightarrow$

Prior distribution

$$x_t^1, \dots, x_t^n \qquad w_t^1, \dots, w_t^n = 1$$

Process update

$$\bar{x}_{t+1}^i \sim P(X_{t+1}|x_t^i, e_{1:t})$$

Observation update

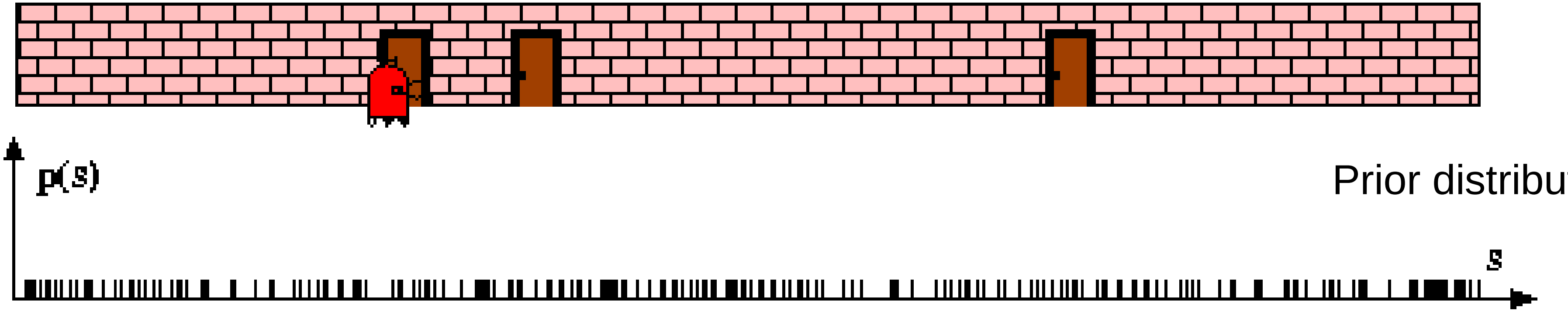
$$w_{t+1}^i = P(e_{t+1}|\bar{x}_{t+1}^i)w_t^i$$

Resample

$$X_{t+1} = \{\}$$

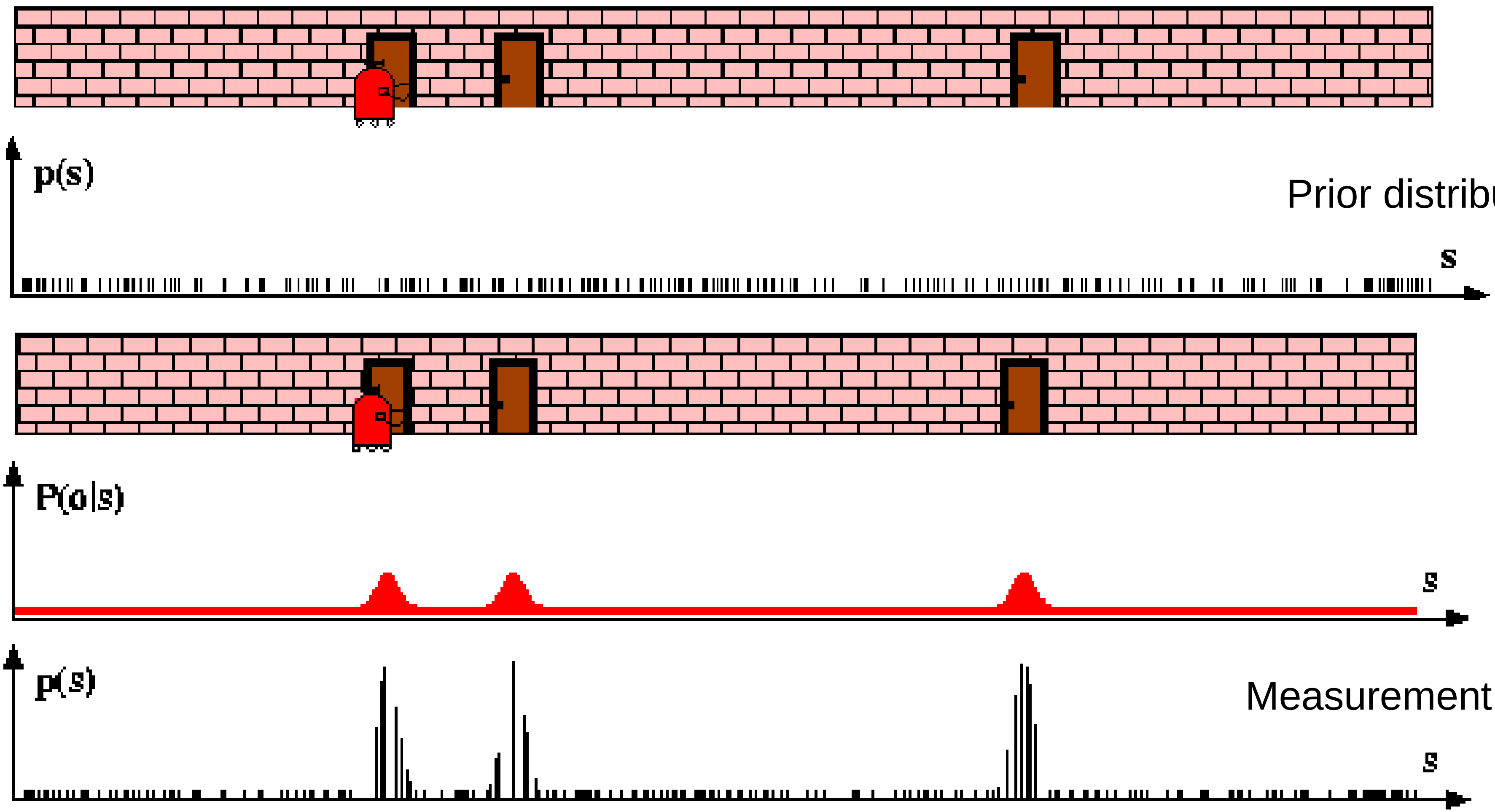
$X_{t+1} = X_{t+1} \cup \bar{x}_{t+1}^i$ w/ prob w_{t+1}^i

Particle Filter



Prior distribution

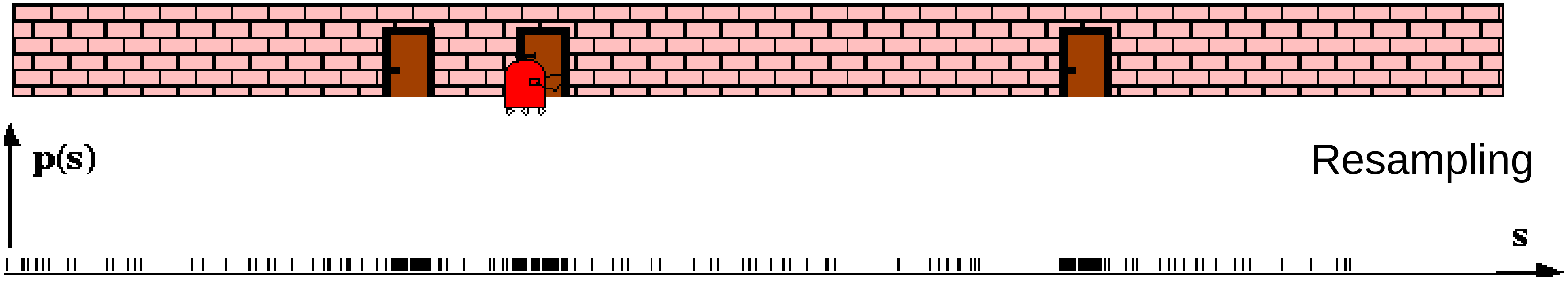
Particle Filter

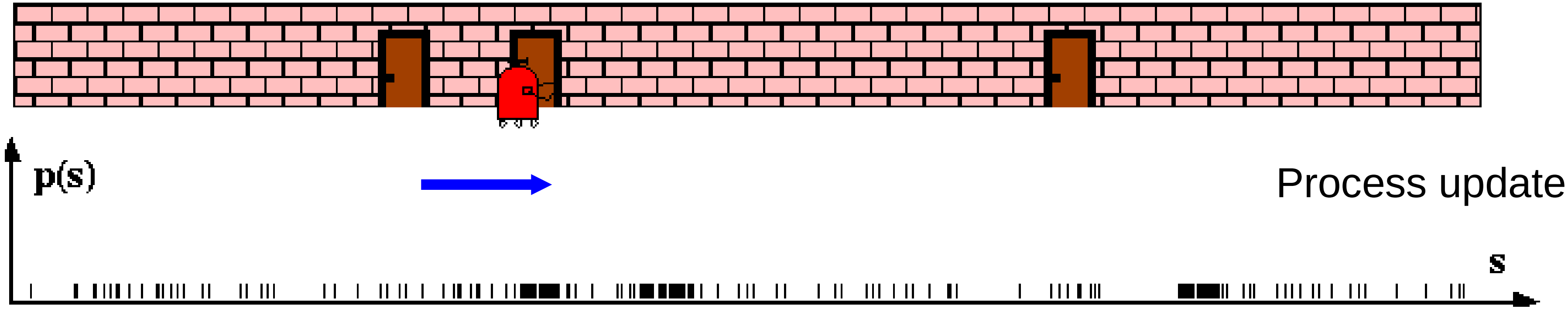


Prior distribution

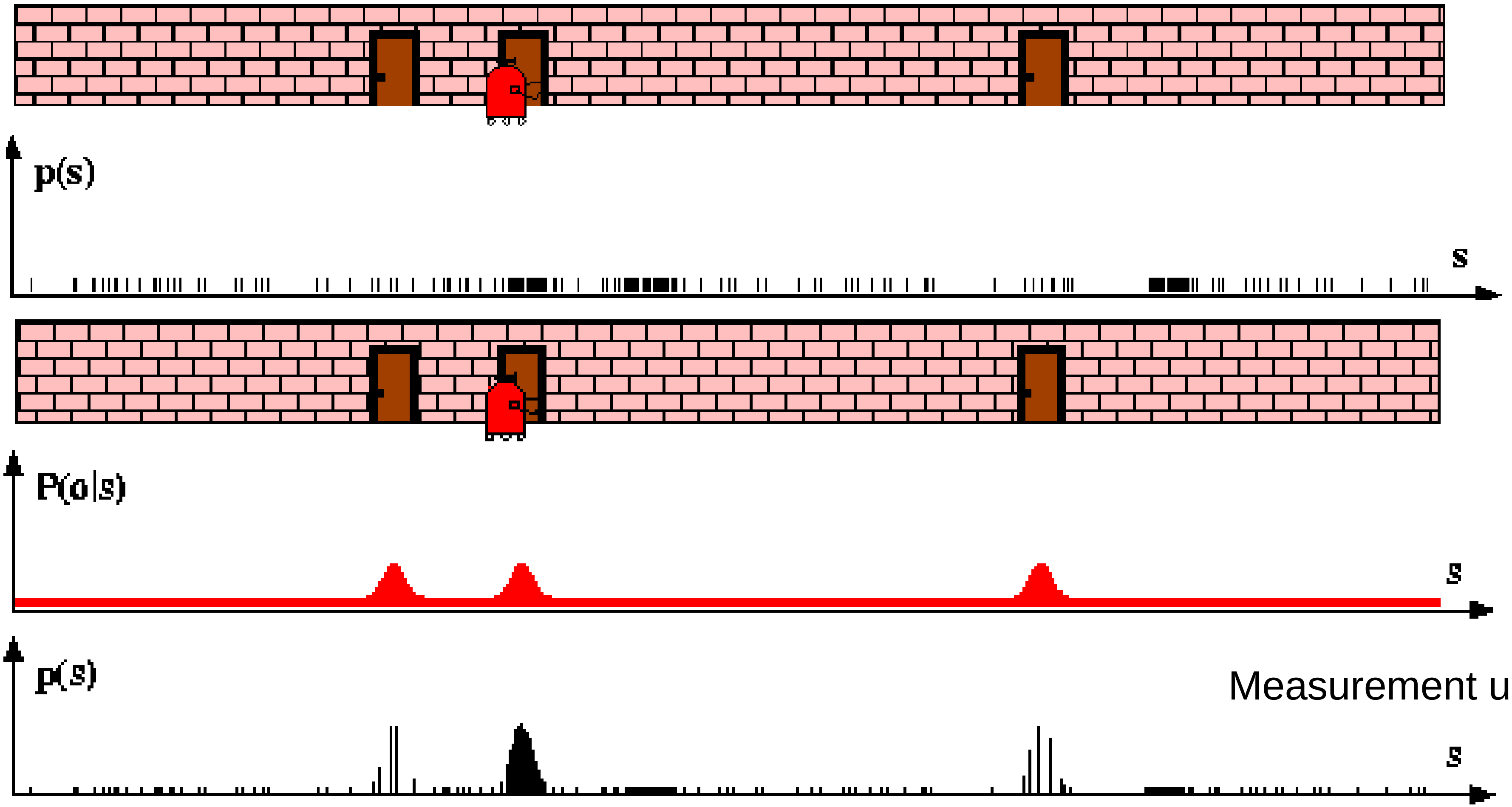
Measurement update

Particle Filter



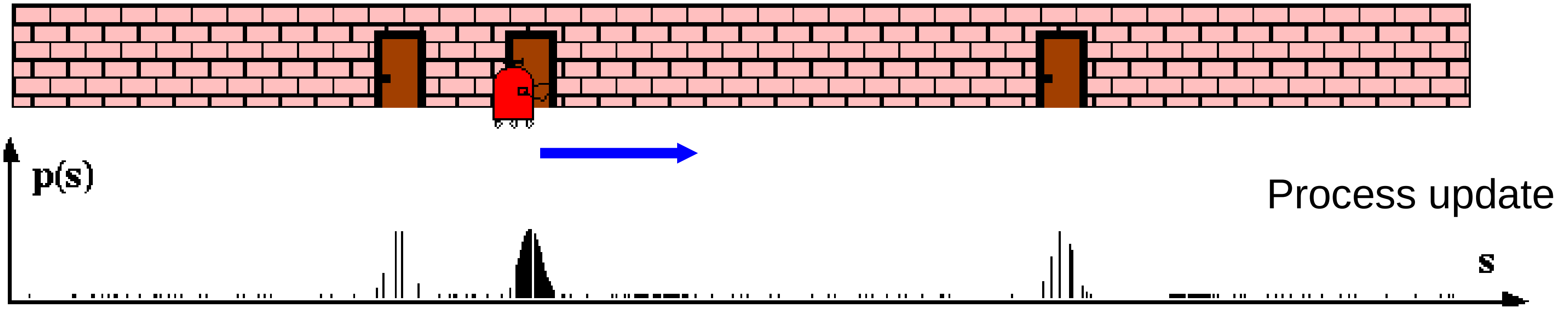


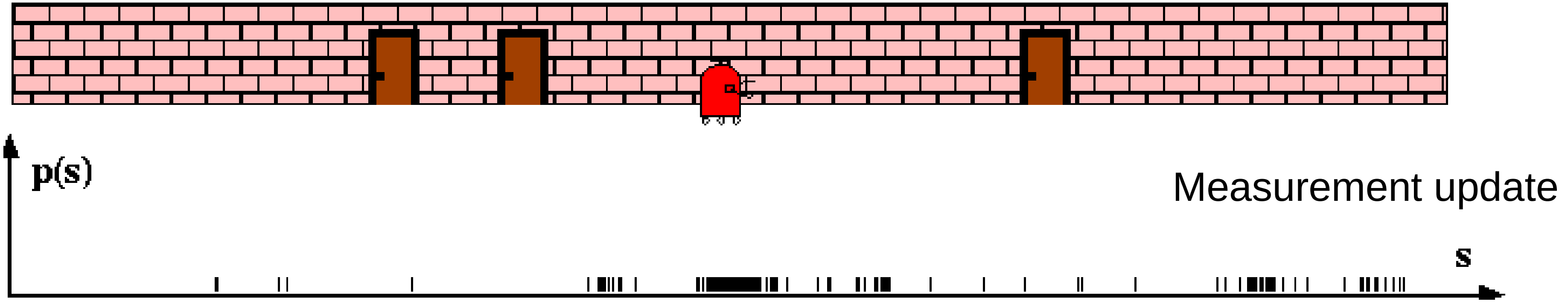
Particle Filter



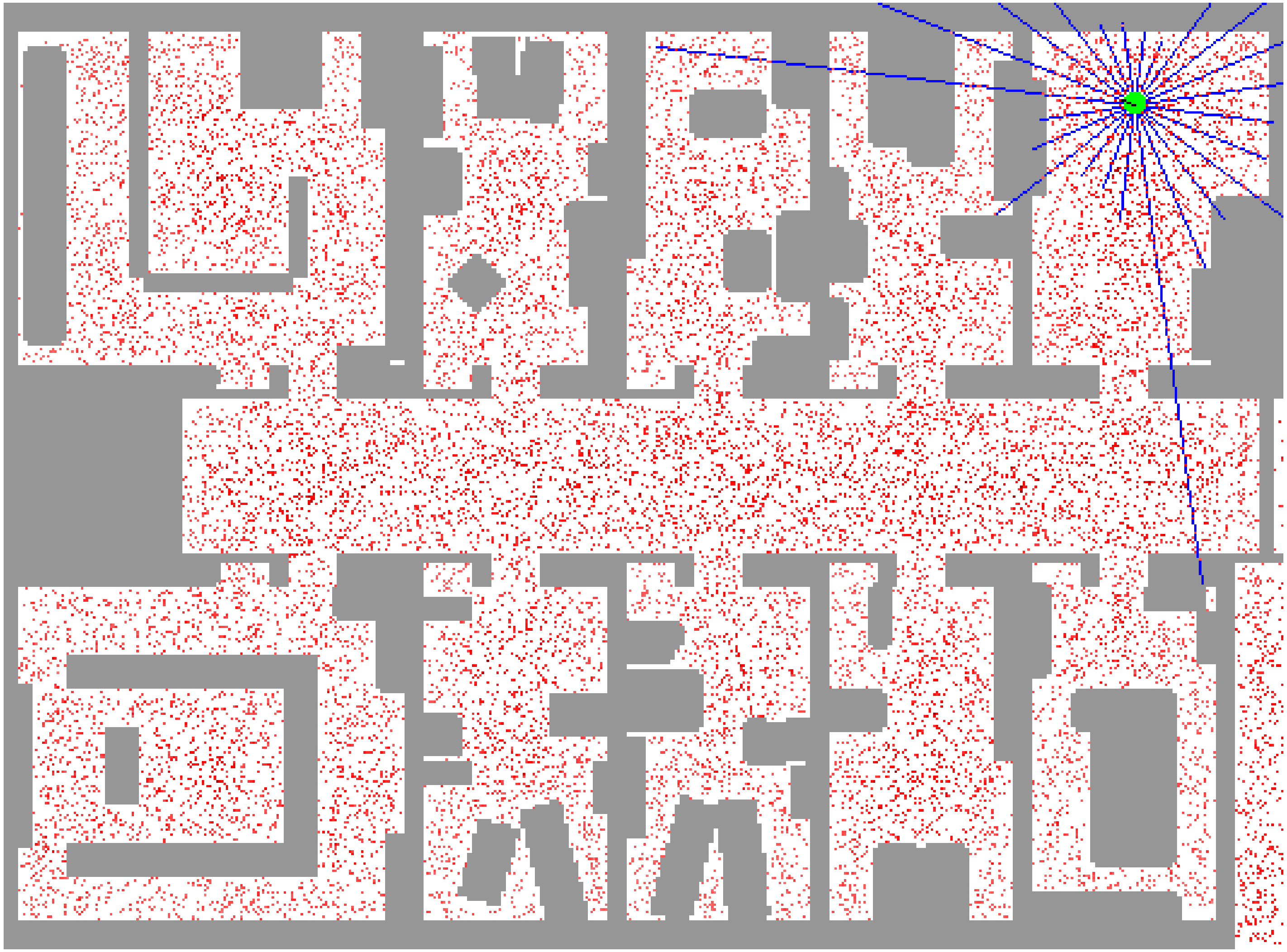
Measurement update

Particle Filter

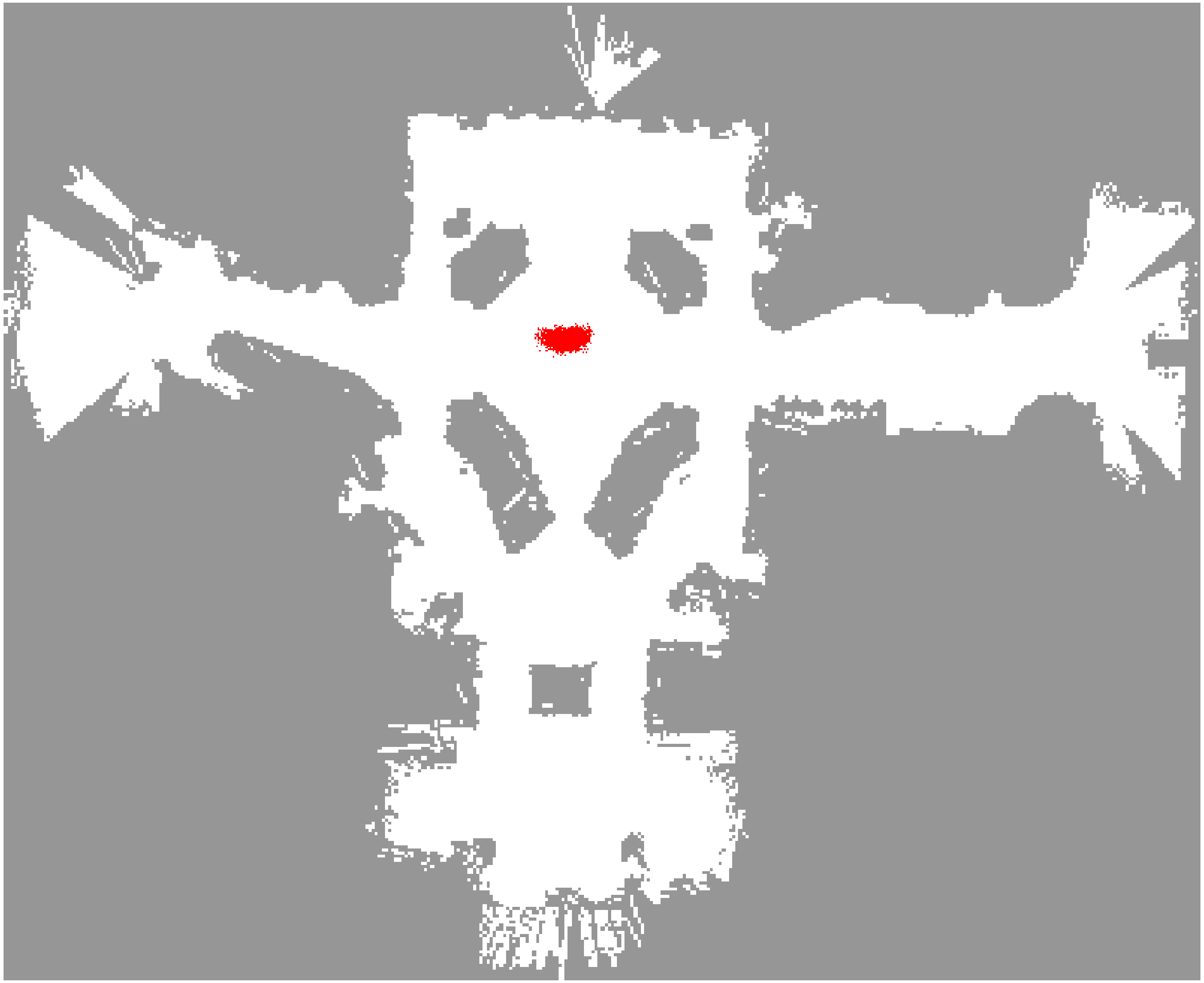




Particle Filter Example



Particle Filter Example



Particle Filtering

Pros:

– works in continuous spaces
– can represent multi-modal distributions

Cons:

– parameters to tune
– sample impoverishment

Sample Impoverishment

Pros:

– works in continuous spaces
– can represent multi-modal distributions

Cons:

– parameters to tune
– sample impoverishment

No particles nearby the true system state

Sample Impoverishment

If there aren't enough samples, then we might ``resample away'' the true state...

<u>Prior distribution</u>

$$x_t^n \qquad w_t^1, \ldots, w_t^n = 1$$

<u>Process update</u>

$$P(X_{t+1}|x_t^i, e_{1:t})$$

<u>Observation update</u>

$$w_{t+1}^i = P(e_{t+1}|\bar{x}_{t+1}^i)w_t^i$$

$$B(X_{t+1}) \quad \boxed{P(X_{t+1}|E_{1:t+1})}$$

Do this *n* times

<u>Resample</u>

$$X_{t+1} = \{\}$$

$$X_{t+1} = X_{t+1} \cup \bar{x}_{t+1}^i \text{ w/ prob } w_{t+1}^i$$

Sample Impoverishment

<u>Prior distribution</u>

$x_t^n \qquad w_t^1, \ldots, w_t^n = 1$

If there aren't enough samples, then we might ``resample away'' the true state...

<u>One solution</u>: add an additional k samples drawn completely at random

<u>Process update</u>

$$P(X_{t+1} | x_t^i, e_{1:t})$$

<u>Observation update</u>

$$w_{t+1} = P(e_{t+1} | \bar{x}_{t+1}^i) w_t^i$$

$$B(X_{t+1}) \quad \boxed{P(X_{t+1} | E_{1:t+1})}$$

Do this *n* times

<u>Resample</u>

$$X_{t+1} = \{\}$$

$X_{t+1} = X_{t+1} \cup \bar{x}_{t+1}^i$ w/ prob w_{t+1}^i

Sample Impoverishment

If there aren't enough samples, then we might ``resample away'' the true state...

One solution: add an additional k samples drawn completely at random

BUT: there's always a chance that the true state won't be represented well by the particles...

Prior distribution

$x_t^n \qquad w_t^1, \ldots, w_t^n = 1$

Process update

$P(X_{t+1} | x_t^i, e_{1:t})$

Observation update

$w_{t+1} = P(e_{t+1} | \bar{x}_{t+1}^i) w_t^i$

$B(X_{t+1}) \quad \boxed{P(X_{t+1} | E_{1:t+1})}$

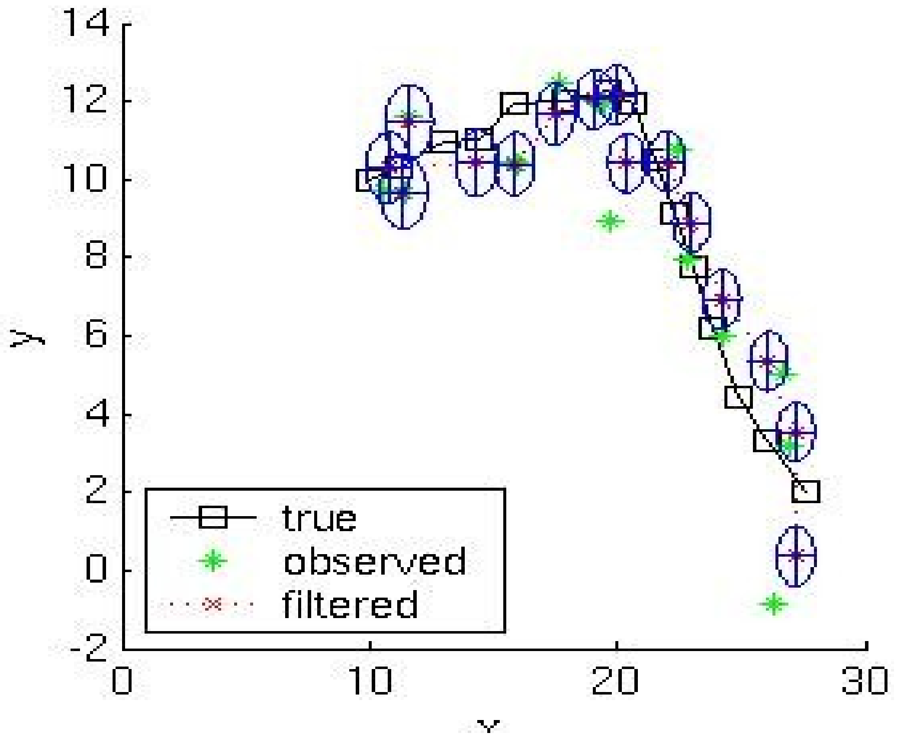
Do this *n* times

Resample

$X_{t+1} = \{\}$

$X_{t+1} = X_{t+1} \cup \bar{x}_{t+1}^i$ w/ prob w_{t+1}^i

Kalman Filtering



Another way to adapt Sequential Bayes Filtering to continuous state spaces

– relies on representing the probability distribution as a Gaussian

– first developed in the early 1960s (before general Bayes filtering); used in Apollo program

Image: UBC, Kevin Murphy Matlab toolbox

Kalman Idea

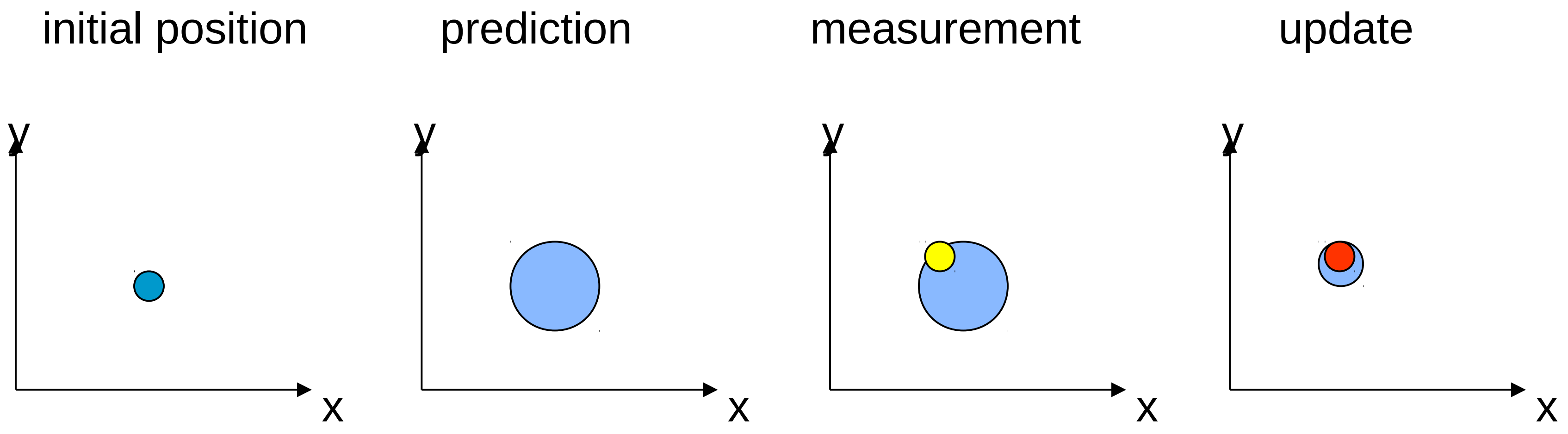


Image: Thrun *et al.*, *CS233B course notes*

Kalman Idea

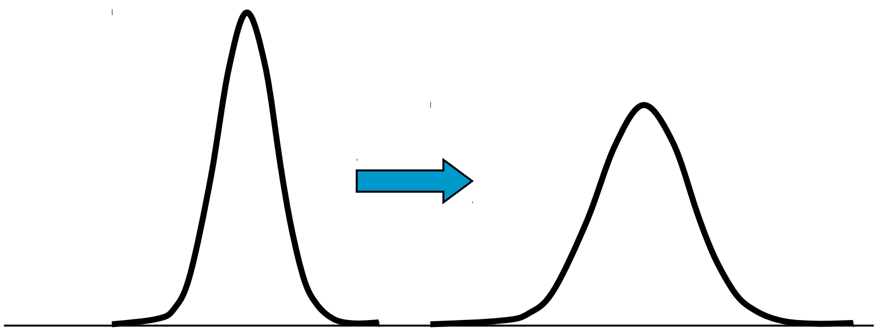


Image: Thrun *et al.*, *CS233B course notes*

$$P(x_{t+1}|z_{0:t}) = \sum_{x_t} P(x_{t+1}|x_t)P(x_t|z_{0:t})$$

$$P(x_{t+1}|z_{0:t+1}) = \eta P(z_{t+1}|x_{t+1})P(x_{t+1}|z_{0:t})$$

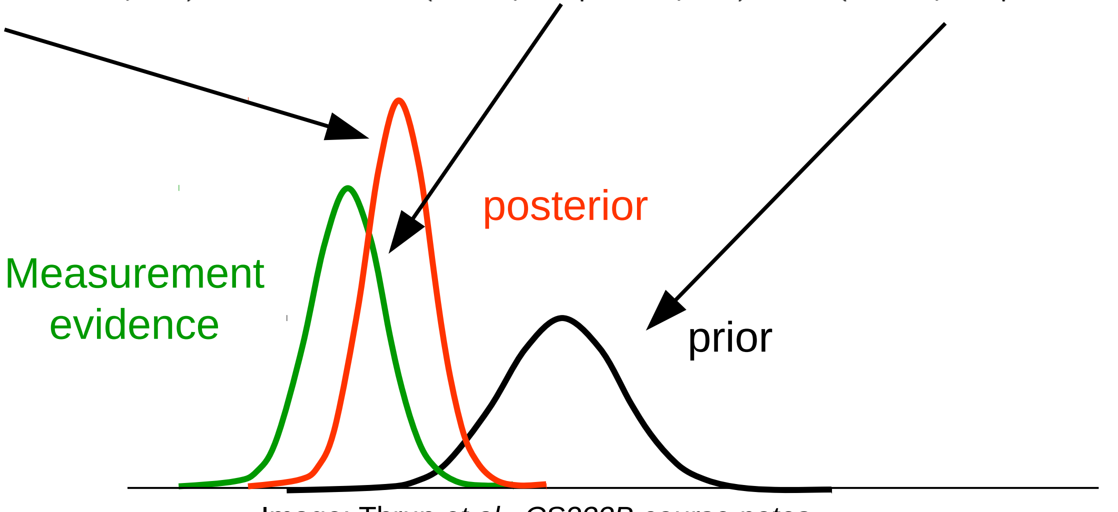


Image: Thrun *et al.*, *CS233B course notes*

Gaussians

- Univariate Gaussian:

$$P(x) = \eta e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}}$$

- Multivariate Gaussian:

$$P(x) = \eta e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$$

$$P(x) = N(x; \mu, \Sigma)$$

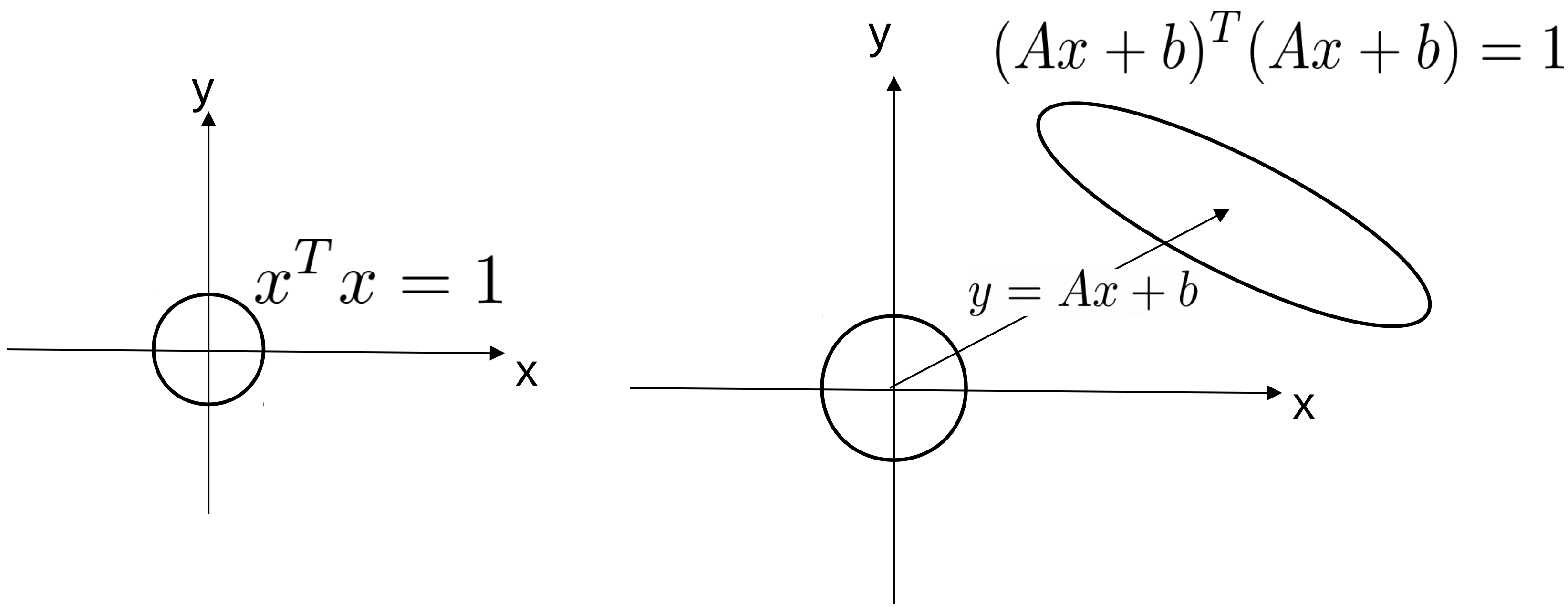
Playing w/ Gaussians

- Suppose: $P(x) = N(x; \mu, \Sigma)$

  $y = Ax + b$

- Calculate: $P(y) = ?$

$$P(y) = N(y; A\mu + b, A\Sigma A^T)$$



In fact

- Suppose: $P(x) = N(x; \mu, \Sigma)$
  $y = Ax + b$

- Then:

$$P\begin{pmatrix} x \\ y \end{pmatrix} = N\left[\begin{array}{c} x \\ y \end{array} : \begin{array}{c} \mu \\ A\mu + b \end{array}, \begin{pmatrix} \Sigma & \Sigma A^T \\ A\Sigma & A\Sigma A^T \end{pmatrix} \right]$$

Illustration

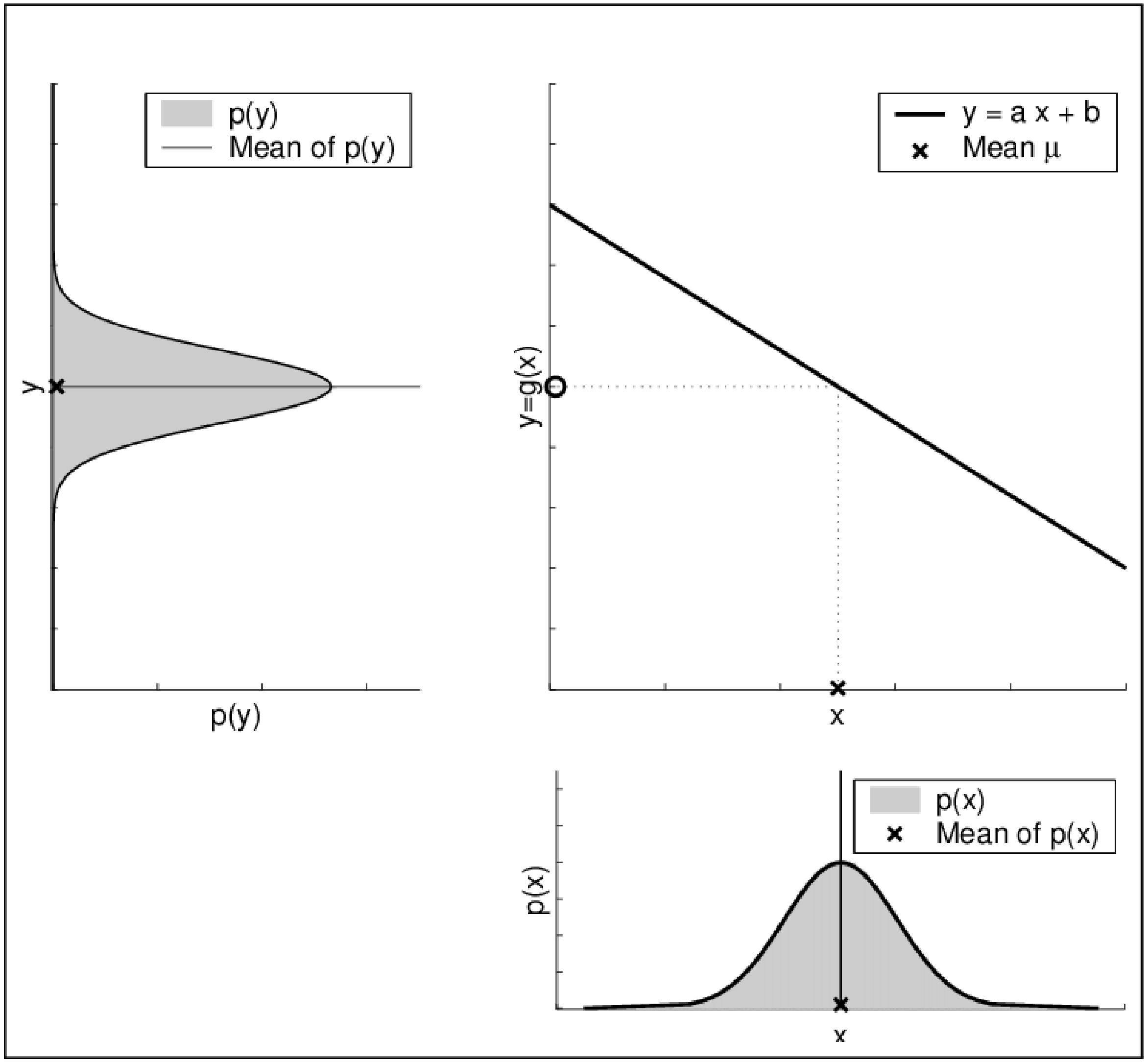


Image: Thrun *et al.*, *CS233B course notes*

And

Suppose: $P(x) = N(x; \mu, \Sigma)$

$$P(y|x) = N(y; Ax + b, R)$$

Then:

$$P\begin{pmatrix} x \\ y \end{pmatrix} = N\left[\begin{array}{c} x \\ y \end{array} : \begin{array}{c} \mu \\ A\mu + b \end{array}, \begin{pmatrix} \Sigma & \Sigma A^T \\ A\Sigma & A\Sigma A^T + R \end{pmatrix}\right]$$

$$P(y) = N(y; A\mu + b, A\Sigma A^T + R)$$

Marginal distribution

Does this remind us of anything?

Does this remind us of anything?

Process update (discrete):

$$P(x_{t+1}|z_{0:t}) = \sum_{x_t} P(x_{t+1}|x_t) P(x_t|z_{0:t})$$

Process update (continuous):

$$P(x_{t+1}|z_{0:t}) = \int_{x_t} P(x_{t+1}|x_t) P(x_t|z_{0:t})$$

Does this remind us of anything?

Process update (discrete):

$$P(x_{t+1}|z_{0:t}) = \sum_{x_t} P(x_{t+1}|x_t)P(x_t|z_{0:t})$$

Process update (continuous):

$$P(x_{t+1}|z_{0:t}) = \int_{x_t} P(x_{t+1}|x_t)P(x_t|z_{0:t})$$

$$N(x_{t+1}|Ax_t, Q)$$

transition dynamics

$$N(x_t|\mu_t, \Sigma_t)$$

prior

Does this remind us of anything?

Process update (discrete): $$P(x_{t+1}|z_{0:t}) = \sum_{x_t} P(x_{t+1}|x_t)P(x_t|z_{0:t})$$

Process update (continuous): $$P(x_{t+1}|z_{0:t}) = \int_{x_t} P(x_{t+1}|x_t)P(x_t|z_{0:t})$$

$$N(x_{t+1}|Ax_t, Q)$$

transition dynamics

$$N(x_t|\mu_t, \Sigma_t)$$

prior

$$P(x_{t+1}|z_{0:t}) = \int_{x_t} N(x_{t+1}|Ax_t, Q)N(x_t;\mu_t, \Sigma_t)$$

$$P(x_{t+1}|z_{0:t}) = N(x_{t+1}|A\mu_t, A\Sigma_t A^T + Q)$$

Observation update

Observation update:

$$P(x_{t+1}|z_{0:t+1}) = \eta P(z_{t+1}|x_{t+1}) P(x_{t+1}|z_{0:t})$$

$$N(z_{t+1}|Cx_{t+1}, R) \qquad N(x_t|\mu'_t, \Sigma'_t)$$

Where: $\mu'_t = A\mu_t$

$$\Sigma'_t = A\Sigma_t A^T + Q$$

Observation update

Observation update:

$$P(x_{t+1}|z_{0:t+1}) = \eta P(z_{t+1}|x_{t+1})P(x_{t+1}|z_{0:t})$$

$$N(z_{t+1}|Cx_{t+1}, R) \qquad N(x_t|\mu'_t, \Sigma'_t)$$

Where: $\mu'_t = A\mu_t$

$$\Sigma'_t = A\Sigma_t A^T + Q$$

$$P(z_{t+1}, x_{t+1}|z_{0:t}) = \eta N(z_{t+1}|Cx_t, R)N(x_t;\mu'_t, \Sigma'_t)$$

Observation update

Observation update:

$$P(x_{t+1}|z_{0:t+1}) = \eta P(z_{t+1}|x_{t+1}) P(x_{t+1}|z_{0:t})$$

$$N(x_t|\mu'_t, \Sigma'_t)$$

Where: $\mu'_t = A\mu_t$

$$P(z_{t+1}, x_{t+1}|z_{0:t}) = \eta N(z_{t+1}|Cx_t, R) N(x_t; \mu'_t, \Sigma'_t)$$

$$P(z_{t+1}, x_{t+1}|z_{0:t}) = N\left[\begin{array}{ccc} x_{t+1} & : & \mu'_t \\ z_{t+1} & & C\mu'_t \end{array}, \left(\begin{array}{cc} \Sigma'_t & \Sigma'_t C^T \\ C\Sigma'_t & C\Sigma'_t A^T + R \end{array}\right)\right]$$

Observation update

$$P(z_{t+1}, x_{t+1} | z_{0:t}) = N \left[\begin{array}{c} x_{t+1} \\ z_{t+1} \end{array} : \begin{array}{c} \mu'_t \\ C\mu'_t \end{array}, \begin{pmatrix} \Sigma'_t & \Sigma'_t C^T \\ C\Sigma'_t & C\Sigma'_t A^T + R \end{pmatrix} \right]$$

But we need: $P(x_{t+1} | z_{0:t+t}) = ?$

Another Gaussian identity...

Suppose: $N\left[\begin{array}{ccc} x & : & a \\ y & & b \end{array}, \begin{pmatrix} A & C \\ C^T & B \end{pmatrix}\right]$

Calculate: $P(y|x) = ?$

$$P(y|x) = N(y|b + C^T A^{-1}(x-a), B - C^T A^{-1} C)$$

Observation update

$$P(z_{t+1}, x_{t+1} | z_{0:t}) = N\left[\begin{array}{ccc} x_{t+1} & : & \mu'_t \\ z_{t+1} & & C\mu'_t \end{array}, \left(\begin{array}{cc} \Sigma & \Sigma C^T \\ C\Sigma & C\Sigma A^T + R \end{array} \right) \right]$$

But we need: $P(x_{t+1} | z_{0:t+1}) = ?$

$$P(x_{t+1} | z_{0:t+1}) = N(x_{t+1}; \mu_{t+1}, \Sigma_{t+1})$$

$$\mu_{t+1} = \mu'_t + \Sigma'_t C^T (R + C\Sigma'_t C^T)^{-1} (z_{t+1} - C\mu'_t)$$

$$\Sigma_{t+1} = \Sigma'_t - \Sigma'_t C^T (R + C\Sigma'_t C^T)^{-1} C\Sigma'_t$$

To summarize the Kalman filter

System: $$P(x_{t+1}|x_t) = N(x_{t+1}|Ax_t, Q)$$
$$P(z_{t+1}|x_{t+1}) = N(z_{t+1}|Cx_{t+1}, R)$$

Prior: μ_t

Σ_t

Process update: $\mu_t' = A\mu_t$

$$\Sigma_t' = A\Sigma_t A^T + Q$$

Measurement update:
$$\mu_{t+1} = \mu_t' + \Sigma_t' C^T (R + C\Sigma_t' C^T)^{-1}(z_{t+1} - C\mu_t')$$
$$\Sigma_{t+1} = \Sigma_t' - \Sigma_t' C^T (R + C\Sigma_t' C^T)^{-1} C\Sigma_t'$$

Suppose there is an action term...

System: $$P(x_{t+1}|x_t) = N(x_{t+1}|Ax_t + u_t, Q)$$
$$P(z_{t+1}|x_{t+1}) = N(z_{t+1}|Cx_{t+1}, R)$$

Prior: μ_t

Σ_t

Process update: $\mu'_t = A\mu_t + u_t$

Measurement update:
$$\mu_{t+1} = \mu'_t + \Sigma'_t C^T (R + C\Sigma'_t C^T)^{-1}(z_{t+1} - C\mu'_t)$$
$$\Sigma_{t+1} = \Sigma'_t - \Sigma'_t C^T (R + C\Sigma'_t C^T)^{-1} C\Sigma'_t$$

To summarize the Kalman filter

Prior: μ_t

Σ_t

Process update: $\mu_t' = A\mu_t$

$$\Sigma_t' = A\Sigma_t A^T + Q$$

Measurement update:

$$\mu_{t+1} = \mu_t' + \boxed{\Sigma_t' C^T (R + C\Sigma_t' C^T)^{-1}} (z_{t+1} - C\mu_t')$$

This factor is often called the "Kalman gain"

$$\Sigma_{t+1} = \Sigma_t' - \boxed{\Sigma_t' C^T (R + C\Sigma_t' C^T)^{-1}} C\Sigma_t'$$

Things to note about the Kalman filter

Process update: $\mu'_t = A\mu_t \qquad \Sigma'_t = A\Sigma_t A^T + Q$

Measurement update:

$$\mu_{t+1} = \mu'_t + \Sigma'_t C^T (R + C\Sigma'_t C^T)^{-1}(z_{t+1} - C\mu'_t)$$
$$\Sigma_{t+1} = \Sigma'_t - \Sigma'_t C^T (R + C\Sigma'_t C^T)^{-1} C\Sigma'_t$$

– covariance update is independent of observation

– Kalman is only optimal for linear-Gaussian systems

– the distribution “stays” Gaussian through this update

– the error term can be thought of as the different between the observation and the prediction

Kalman in 1D

System:

$$P(x_{t+1}|x_t) = N(x_{t+1} : x_t + u_t, q)$$
$$P(z_{t+1}|x_{t+1}) = N(z_{t+1}|2x_{t+1}, r)$$

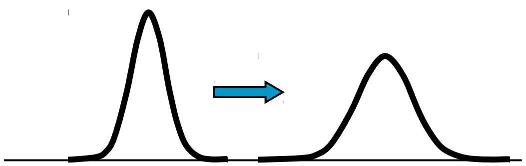


Image: Thrun *et al.*, *CS233B course notes*

Process update:

$$\bar{\mu}_t = \mu_t + u_t$$
$$\bar{\sigma}_t^2 = \sigma_t^2 + q$$

Measurement update:

$$\mu_{t+1} = \bar{\mu}_t + \frac{2\bar{\sigma}_t^2}{r + 4\bar{\sigma}_t^2}(z_{t+1} - \bar{\mu}_t)$$
$$\sigma_{t+1} = \bar{\sigma}_t^2 - \frac{4(\bar{\sigma}_t^2)^2}{r + 4\bar{\sigma}_t^2}$$

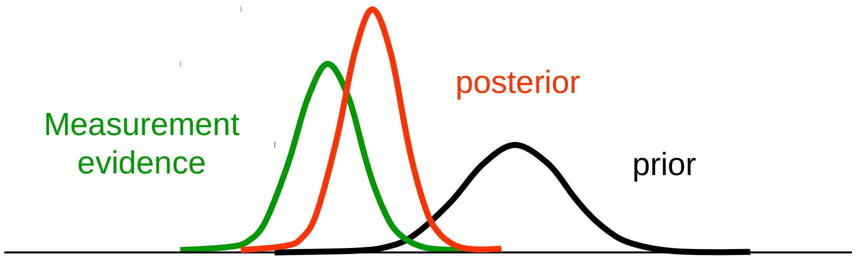


Image: Thrun *et al.*, *CS233B course notes*

Kalman Idea

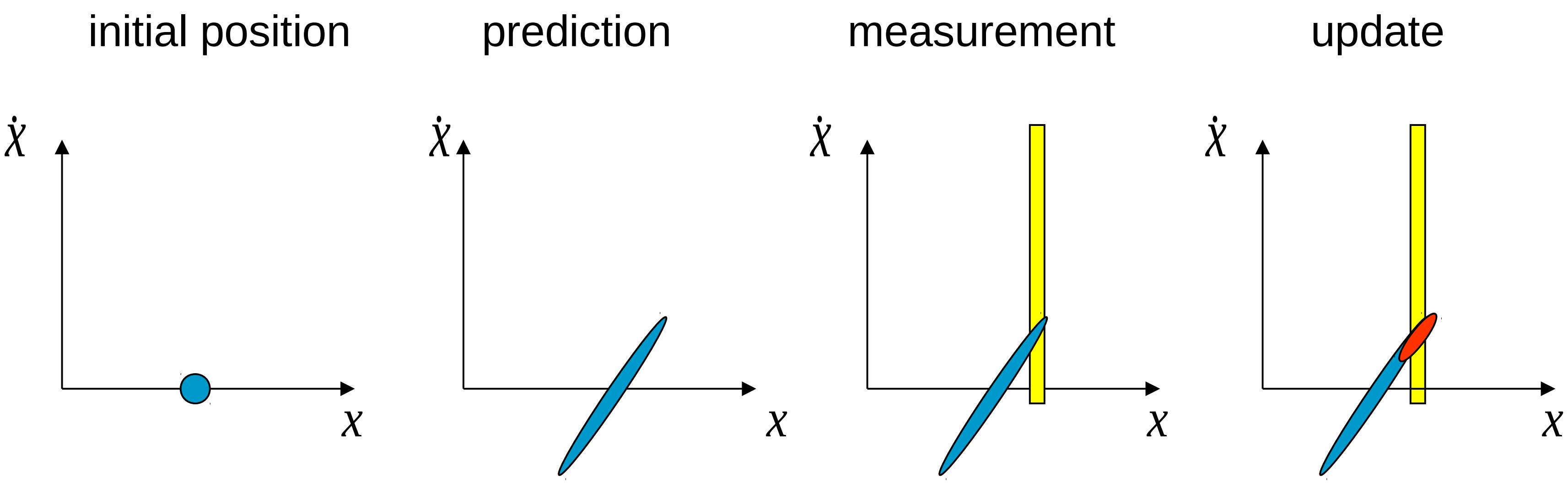


Image: Thrun *et al.*, *CS233B course notes*

Example: estimate velocity

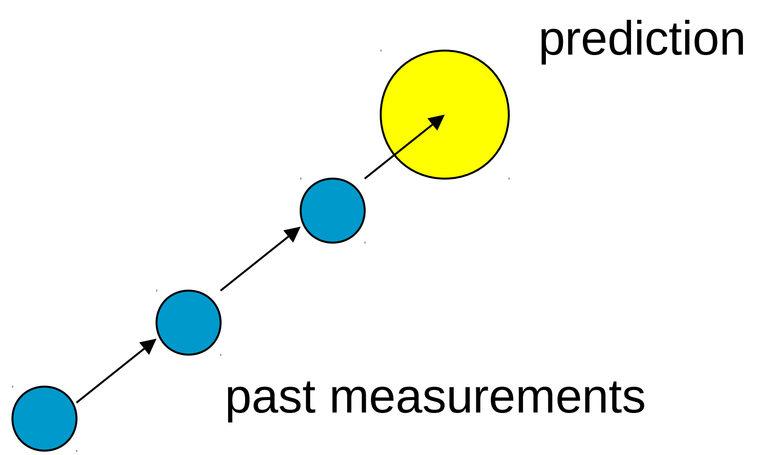


Image: Thrun *et al.*, *CS233B course notes*

Example: filling a tank

$$x = \begin{pmatrix} l \\ f \end{pmatrix}$$

Level of tank

Fill rate

$$l_{t+1} = l_t + f dt$$

Process: $$x_{t+1} = \begin{pmatrix} 1 & dt \\ 0 & 1 \end{pmatrix} x_t + q$$

Observation: $$z_{t+1} = \begin{pmatrix} 1 & 0 \end{pmatrix} x_{t+1} + r$$

Example: estimate velocity

$$x_{t+1} = Ax_t + w_t$$

$$\begin{pmatrix} x_{t+1} \\ y_{t+1} \\ \dot{x}_{t+1} \\ \dot{y}_{t+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_t \\ y_t \\ \dot{x}_t \\ \dot{y}_t \end{pmatrix} + \mathrm{w}_t$$

$$z_{t+1} = Cx_{t+1} + r_{t+1}$$

$$\begin{pmatrix} x_{t+1} \\ y_{t+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{t+1} \\ y_{t+1} \\ \dot{x}_{t+1} \\ \dot{y}_{t+1} \end{pmatrix} + \mathrm{r}_{t+1}$$

But, my system is NON-LINEAR!

$$
\begin{aligned}
x_{t+1} &= f(x_t, u_t) \\
&\neq Ax_t + Bu_t
\end{aligned}
$$

What should I do?

But, my system is NON-LINEAR!

$$
\begin{aligned}
x_{t+1} &= f(x_t, u_t) \\
&\neq Ax_t + Bu_t
\end{aligned}
$$

- What should I do?

Well, there are some options...

-

But, my system is NON-LINEAR!

$$\begin{aligned} x_{t+1} &= f(x_t, u_t) \\ &\neq Ax_t + Bu_t \end{aligned}$$

- What should I do?

Well, there are some options...

- But none of them are great.
-

But, my system is NON-LINEAR!

$$
\begin{aligned}
x_{t+1} &= f(x_t, u_t) \\
&\neq Ax_t + Bu_t
\end{aligned}
$$

- What should I do?

Well, there are some options...

But none of them are great.

Here's one: the Extended Kalman Filter

Extended Kalman filter

Take a Taylor expansion:

$$
\begin{aligned}
x_{t+1} &= f(x_t, u_t) \\
&\approx f(\mu_t, u_t) + A_t(x_t - \mu_t)
\end{aligned}
$$

Where: $A_t = \frac{\partial f}{\partial x}(\mu_t, u_t)$

$$
\begin{aligned}
z_{t+1} &= h(x_t) \\
&\approx h(\mu_t) + C_t(x_t - \mu_t)
\end{aligned}
$$

Where: $C_t = \frac{\partial h}{\partial x}(\mu_t)$

Extended Kalman filter

Take a Taylor expansion:

$$x_{t+1} = f(x_t, u_t)$$

$$\approx f(\mu_t, u_t) + A_t(x_t - \mu_t)$$

Where: $A_t = \frac{\partial f}{\partial x}(\mu_t, u_t)$

$$z_{t+1} = h(x_t)$$

$$\approx h(\mu_t) + C_t(x_t - \mu_t)$$

Where: $C_t = \frac{\partial h}{\partial x}(\mu_t)$

Then use the same equations...

To summarize the EKF

Prior: μ_t

Σ_t

Process update:

$$\mu'_t = f(\mu_t, u_t)$$

$$\Sigma'_t = A_t \Sigma_t A_t^T + Q$$

Measurement update:

$$\mu_{t+1} = \mu'_t + \Sigma'_t C^T (R + C \Sigma'_t C^T)^{-1} (z_{t+1} - h(\mu'_t))$$

$$\Sigma_{t+1} = \Sigma'_t - \Sigma'_t C^T (R + C \Sigma'_t C^T)^{-1} C \Sigma'_t$$

Extended Kalman filter

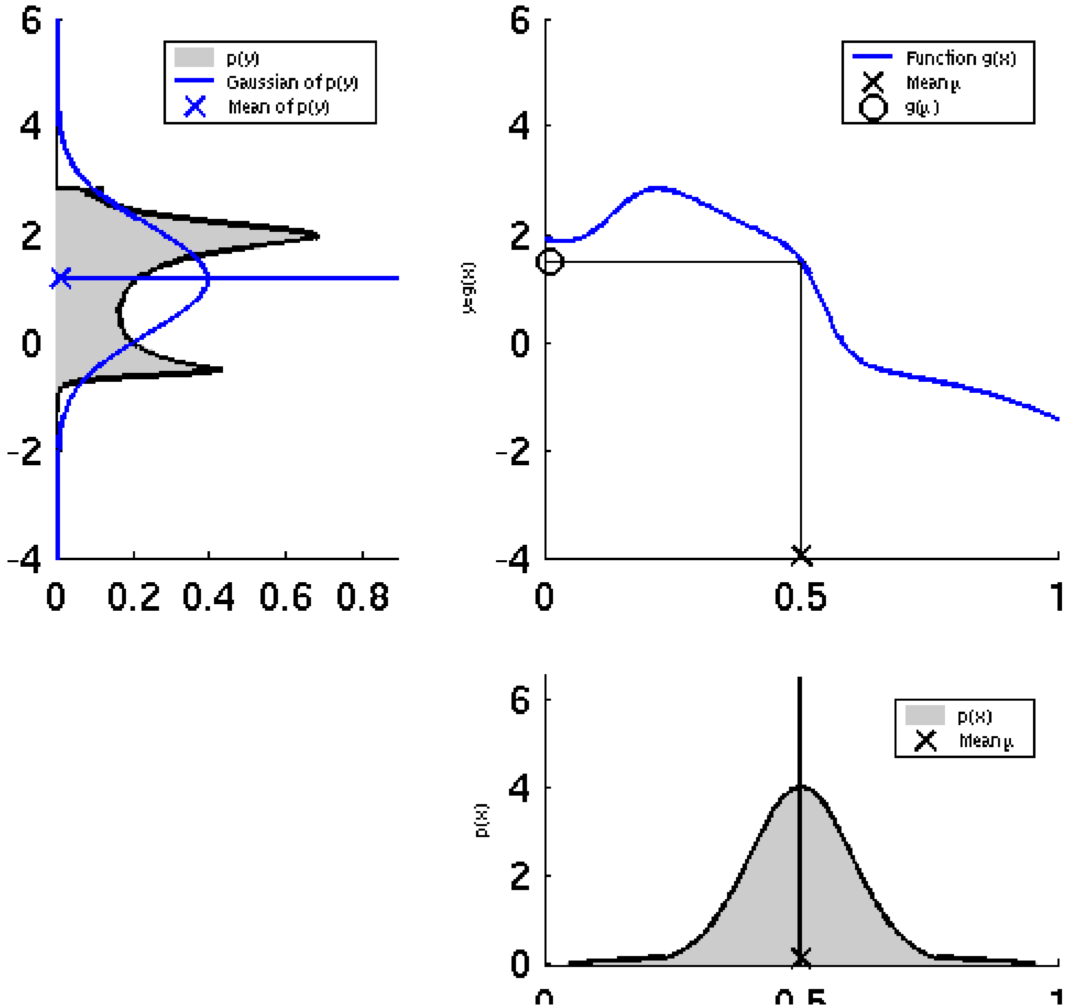


Image: Thrun *et al.*, *CS233B course notes*

Extended Kalman filter

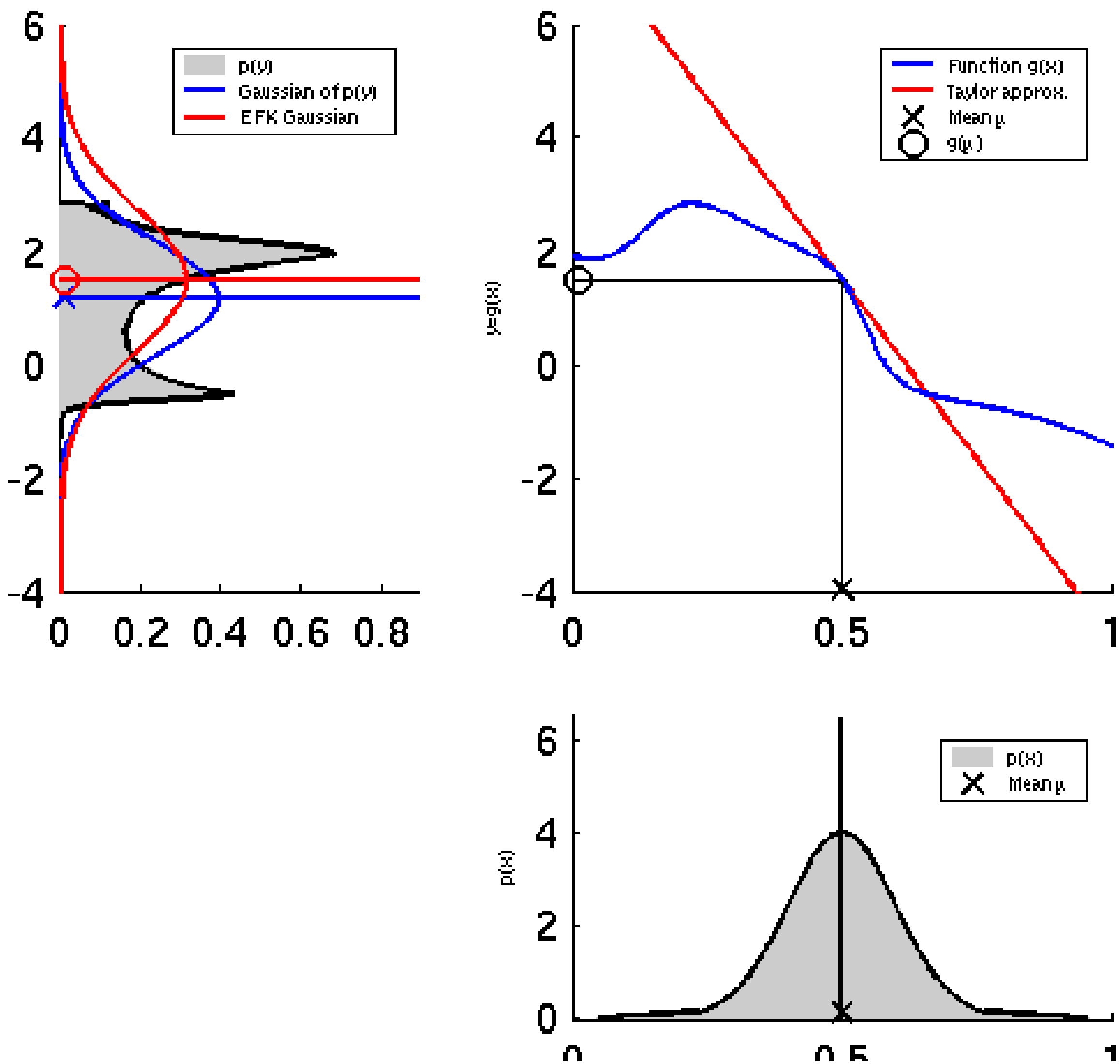
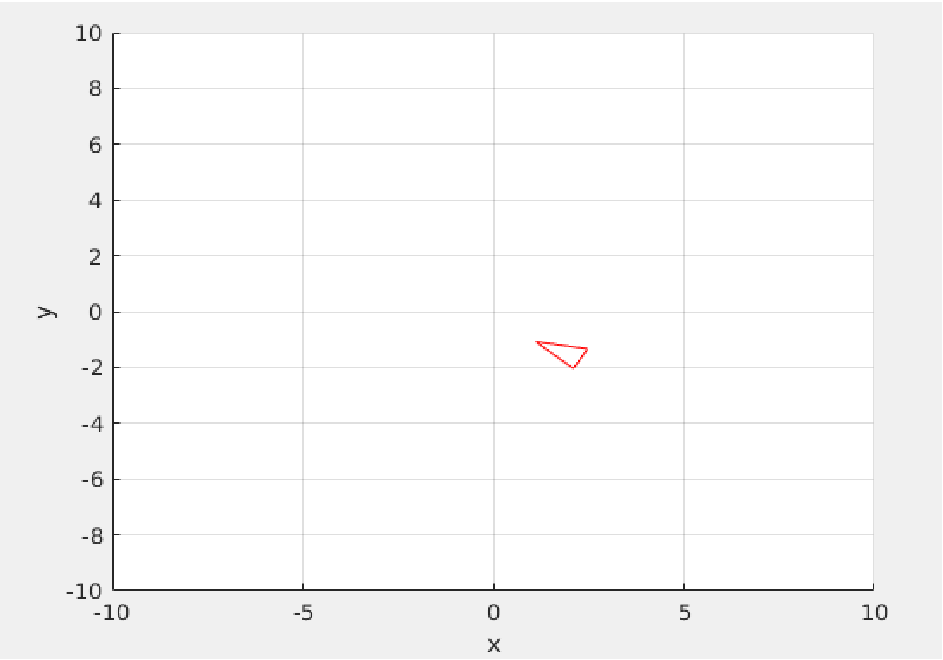


Image: Thrun *et al.*, *CS233B course notes*

EKF Mobile Robot Localization



Suppose we have a mobile robot wandering around in a 2-d world ...

noise

Process dynamics: $\boldsymbol{x}\langle k+1\rangle = \boldsymbol{f}\left(\boldsymbol{x}\langle k\rangle,\, \delta\langle k\rangle,\, \boldsymbol{v}\langle k\rangle\right)$

state

Odometry measurement

Process noise is assumed to be Gaussian: $\boldsymbol{v} = (\nu_d,\, \nu_\theta) \sim N(0,\, V)$

EKF Mobile Robot Localization

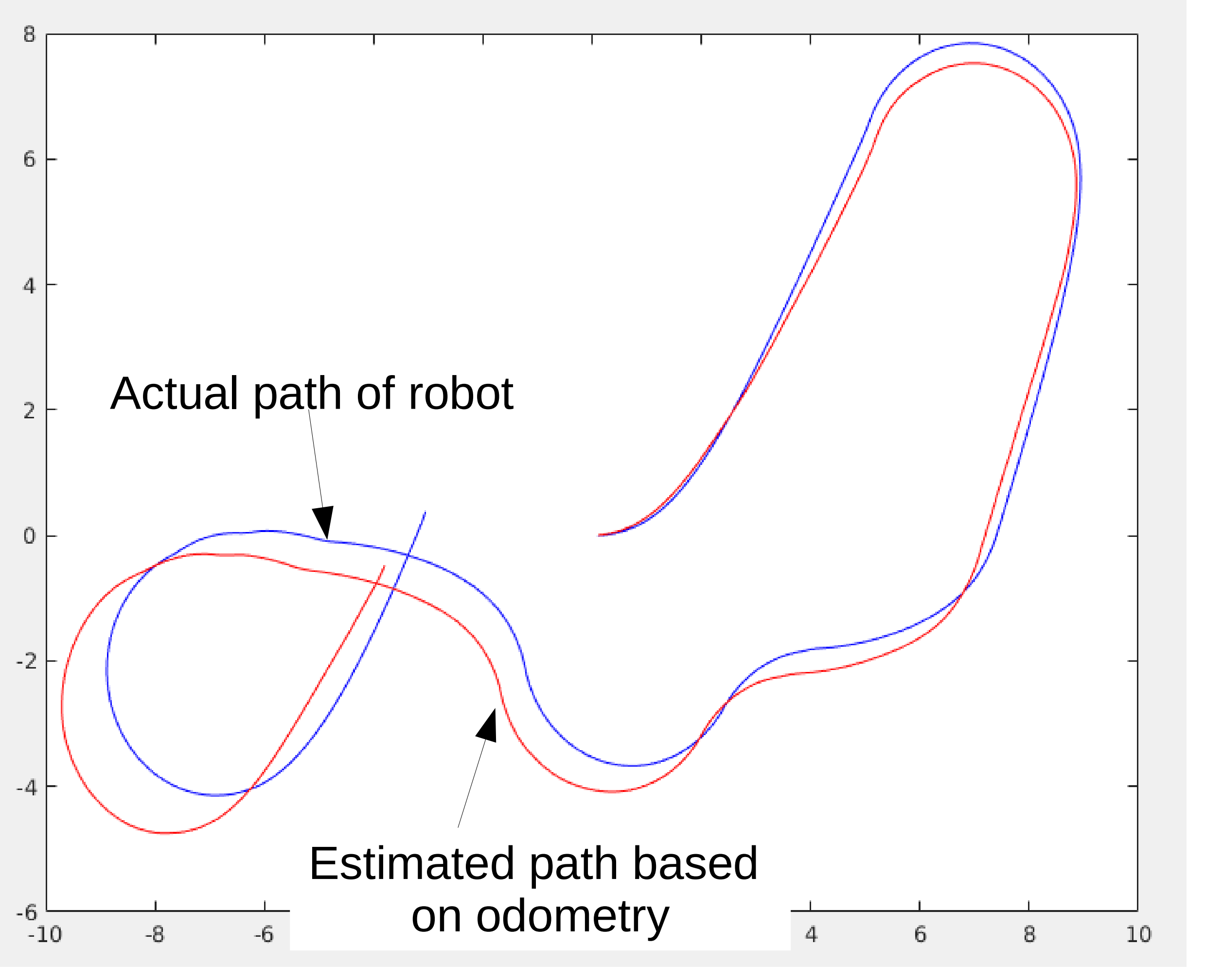
Process dynamics: $\boldsymbol{x}\langle k+1\rangle = \boldsymbol{f}\big(\boldsymbol{x}\langle k\rangle,\ \delta\langle k\rangle,\ \boldsymbol{v}\langle k\rangle\big)$

noise

$$\xi\langle k+1\rangle = \begin{pmatrix} x\langle k\rangle + \big(\delta_d\langle k\rangle + v_d\big)\cos\big(\theta\langle k\rangle + \delta_\theta + v_\theta\big) \\ y\langle k\rangle + \big(\delta_d\langle k\rangle + v_d\big)\sin\big(\theta\langle k\rangle + \delta_\theta + v_\theta\big) \\ \theta\langle k\rangle + \delta_\theta + v_\theta \end{pmatrix}$$

Odometry measurement

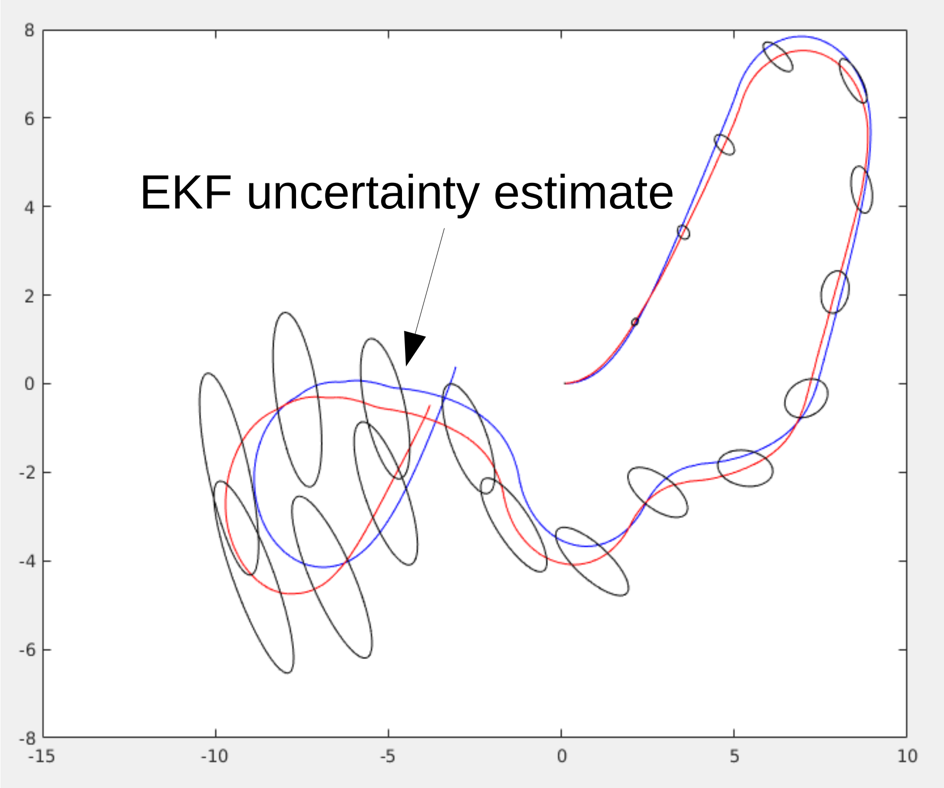
EKF Mobile Robot Localization



But, wheels slip – odometry is not always correct...

How do we localize? Extended Kalman Filter!

EKF Process Update



Dynamics: $\boldsymbol{x}\langle k+1\rangle = \boldsymbol{f}\big(\boldsymbol{x}\langle k\rangle,\ \delta\langle k\rangle,\ \boldsymbol{v}\langle k\rangle\big)$

Linearized dynamics:

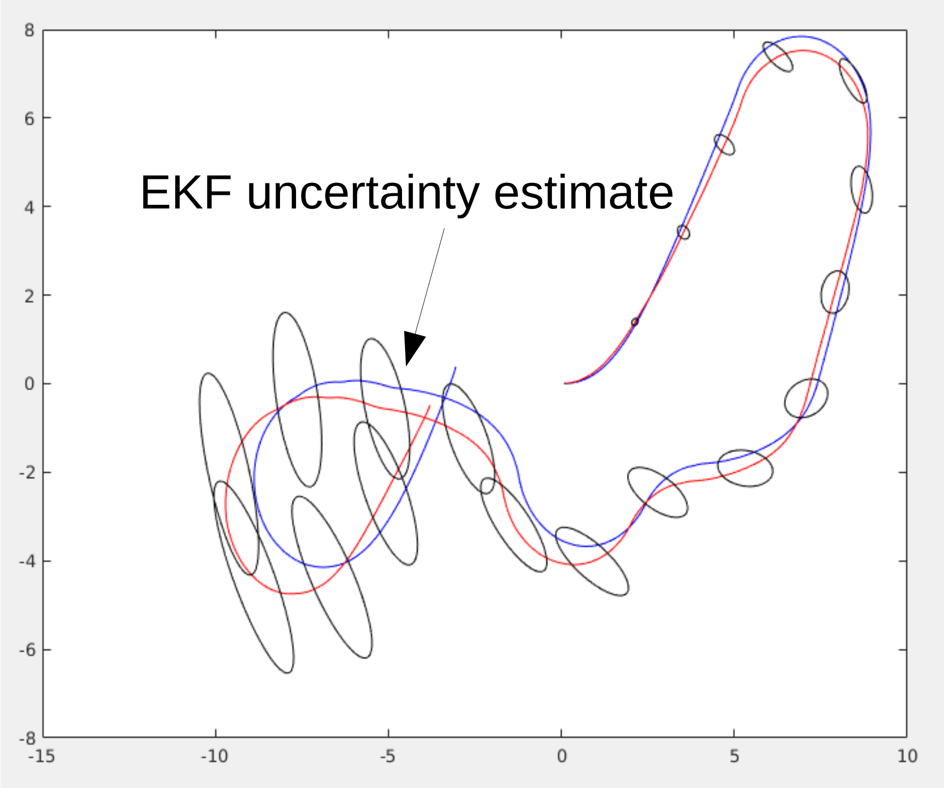
$$\hat{\boldsymbol{x}}\langle k+1\rangle = \hat{\boldsymbol{x}}\langle k\rangle + \boldsymbol{F}_x\big(\boldsymbol{x}\langle k\rangle - \hat{\boldsymbol{x}}\langle k\rangle\big) + \boldsymbol{F}_v \boldsymbol{v}\langle k\rangle$$

Where:

$$\boldsymbol{F}_x = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{v}=0} = \begin{pmatrix} 1 & 0 & -\delta_d\langle k\rangle - \sin\big(\theta\langle k\rangle + \delta_\theta\big) \\ 0 & 1 & \delta_d\langle k\rangle \cos\big(\theta\langle k\rangle + \delta_\theta\big) \\ 0 & 0 & 1 \end{pmatrix}$$

$$\boldsymbol{F}_v = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}}\right|_{\boldsymbol{v}=0} = \begin{pmatrix} \cos\big(\theta\langle k\rangle + \delta_\theta\big) & -\delta_d\langle k\rangle \sin\big(\theta\langle k\rangle + \delta_\theta\big) \\ \sin\big(\theta\langle k\rangle + \delta_\theta\big) & \delta_d\langle k\rangle \cos\big(\theta\langle k\rangle + \delta_\theta\big) \\ 0 & 1 \end{pmatrix}$$

EKF Process Update



Dynamics: $\boldsymbol{x}\langle k+1\rangle = \boldsymbol{f}\left(\boldsymbol{x}\langle k\rangle,\, \delta\langle k\rangle,\, \boldsymbol{v}\langle k\rangle\right)$

Linearized dynamics:

$$\hat{\boldsymbol{x}}\langle k+1\rangle = \hat{\boldsymbol{x}}\langle k\rangle + \boldsymbol{F}_x\left(\boldsymbol{x}\langle k\rangle - \hat{\boldsymbol{x}}\langle k\rangle\right) + \boldsymbol{F}_v \boldsymbol{v}\langle k\rangle$$

Where:

$$\boldsymbol{F}_x = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{v=0} = \begin{pmatrix} 1 & 0 & -\delta_d\langle k\rangle - \sin\left(\theta\langle k\rangle + \delta_\theta\right) \\ 0 & 1 & \delta_d\langle k\rangle \cos\left(\theta\langle k\rangle + \delta_\theta\right) \\ 0 & 0 & 1 \end{pmatrix}$$

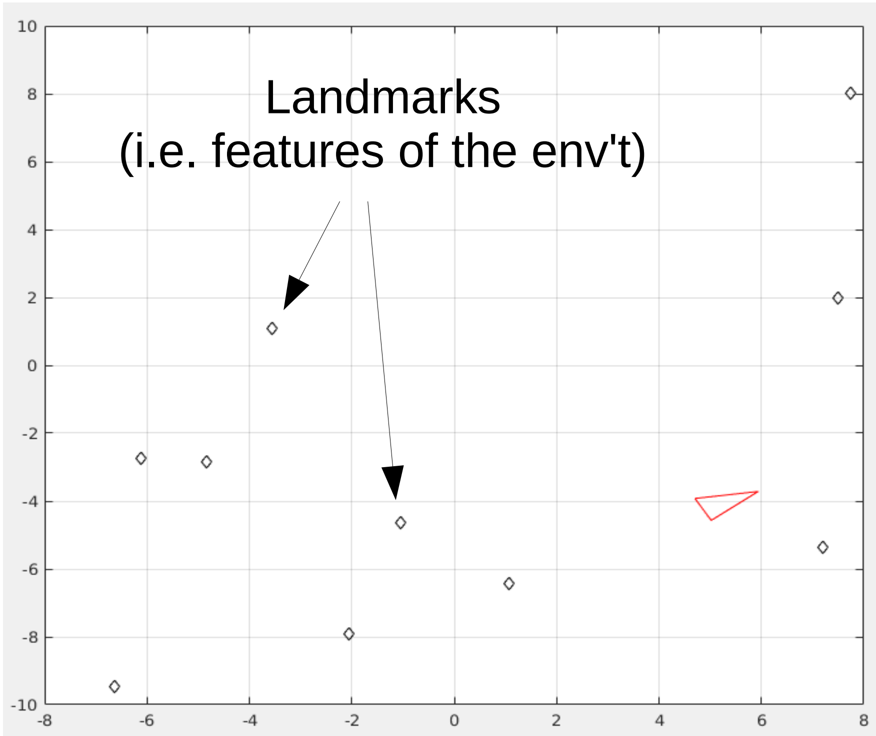
$$\boldsymbol{F}_v = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}}\right|_{v=0} = \begin{pmatrix} \cos\left(\theta\langle k\rangle + \delta_\theta\right) & -\delta_d\langle k\rangle \sin\left(\theta\langle k\rangle + \delta_\theta\right) \\ \sin\left(\theta\langle k\rangle + \delta_\theta\right) & \delta_d\langle k\rangle \cos\left(\theta\langle k\rangle + \delta_\theta\right) \\ 0 & 1 \end{pmatrix}$$

Process update:

$$\boxed{\begin{aligned} \hat{\boldsymbol{x}}\langle k+1|k\rangle &= \boldsymbol{f}\left(\hat{\boldsymbol{x}}\langle k\rangle,\, \delta\langle k\rangle,\, 0\right) \\ \hat{\boldsymbol{P}}\langle k+1|k\rangle &= \boldsymbol{F}_x\langle k\rangle \hat{\boldsymbol{P}}\langle k|k\rangle \boldsymbol{F}_x\langle k\rangle^T + \boldsymbol{F}_v\langle k\rangle \hat{\boldsymbol{V}} \boldsymbol{F}_v\langle k\rangle^T \end{aligned}}$$

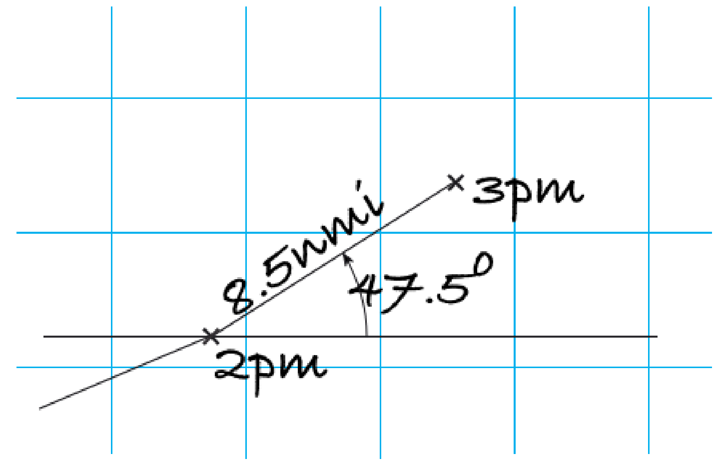
With no observations, uncertainty grows over time...

Observations



<u>Observations:</u>

– range and bearing of a landmark



Observations: $\boldsymbol{z} = \boldsymbol{h}(\boldsymbol{x}_v, \boldsymbol{x}_f, \boldsymbol{w})$

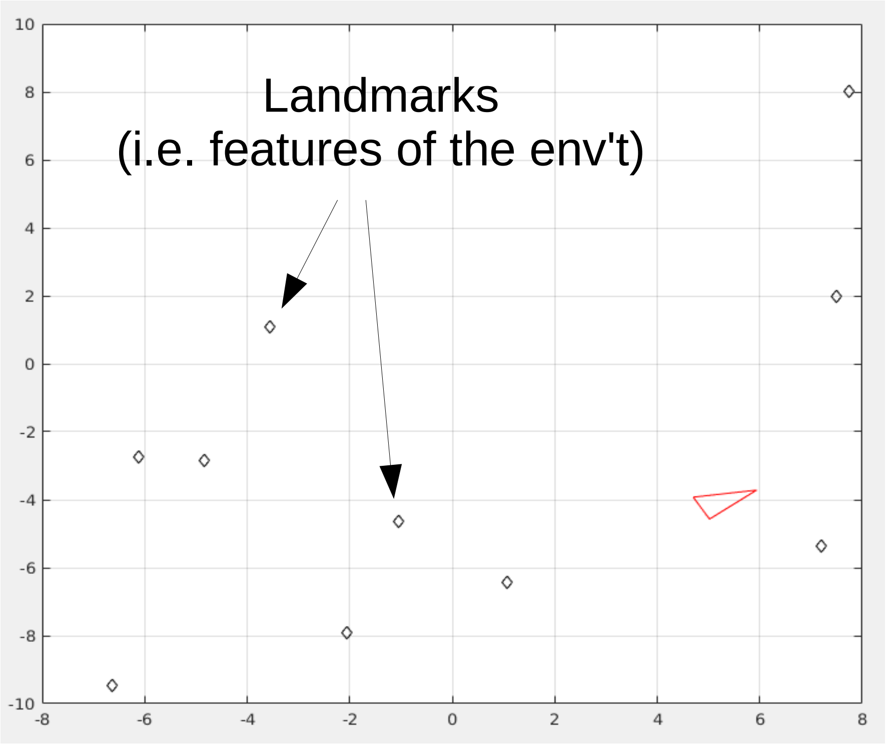
$$\boldsymbol{z} = \begin{pmatrix} \sqrt{(y_i - y_v)^2 + (x_i - x_v)^2} \\ \tan^{-1}(y_i - y_v)/(x_i - x_v) - \theta_v \end{pmatrix} + \begin{pmatrix} w_r \\ w_\beta \end{pmatrix}$$

range

bearing

$$\begin{pmatrix} w_r \\ w_\beta \end{pmatrix} \sim N(0, W), \quad W = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\beta^2 \end{pmatrix}$$

Observations



Observations:

$$\boldsymbol{z} = \boldsymbol{h}(\boldsymbol{x}_v, \boldsymbol{x}_f, \boldsymbol{w})$$

$$\boldsymbol{z} = \begin{pmatrix} \sqrt{(y_i - y_v)^2 + (x_i - x_v)^2} \\ \tan^{-1}(y_i - y_v)/(x_i - x_v) - \theta_v \end{pmatrix} + \begin{pmatrix} w_r \\ w_\beta \end{pmatrix}$$

$$\boldsymbol{z}\langle k\rangle = \hat{\boldsymbol{h}} + \boldsymbol{H}_x\left(\boldsymbol{x}\langle k\rangle - \hat{\boldsymbol{x}}\langle k\rangle\right) + \boldsymbol{H}_w \boldsymbol{w}\langle k\rangle$$

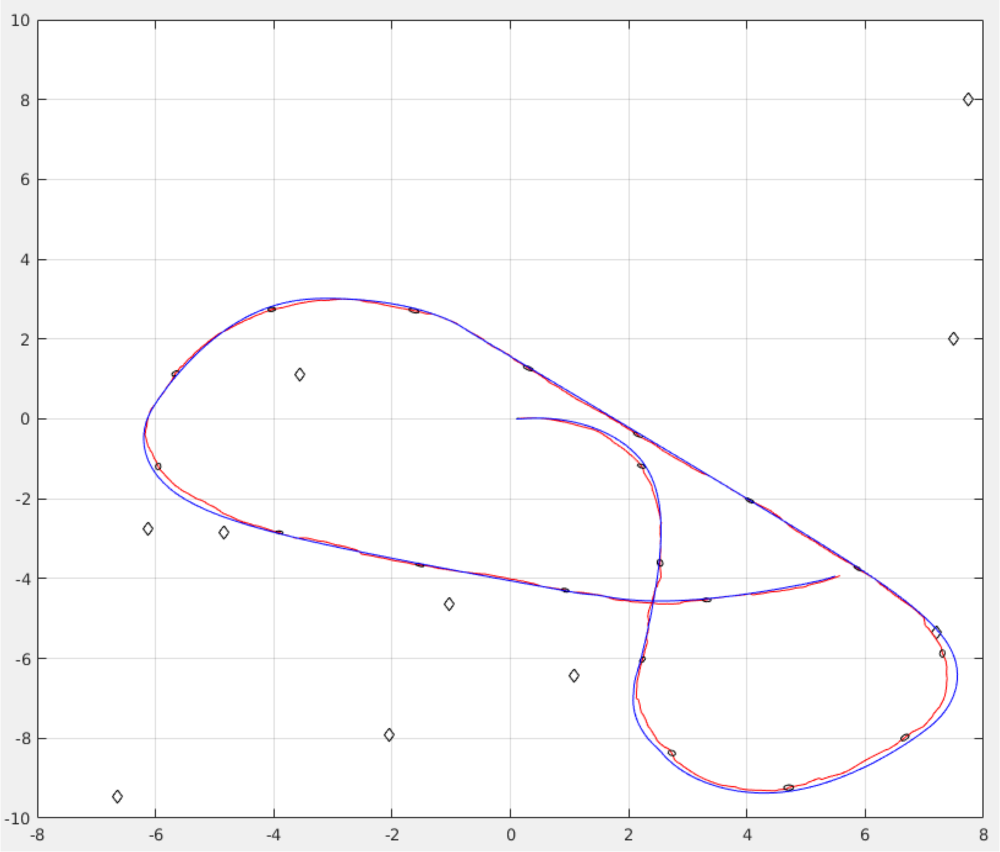
where:

$$\boldsymbol{H}_{x_i} = \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}_v}\right|_{w=0} = \begin{pmatrix} -\dfrac{x_i - x_v\langle k\rangle}{r} & -\dfrac{y_i - y_v\langle k\rangle}{r} & 0 \\ \dfrac{x_i - x_v\langle k\rangle}{r^2} & -\dfrac{y_i - y_v\langle k\rangle}{r^2} & -1 \end{pmatrix}$$

$$\boldsymbol{H}_w = \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{w}}\right|_{w=0} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} w_r \\ w_\beta \end{pmatrix} \sim N(0, \boldsymbol{W}) \qquad \boldsymbol{W} = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\beta^2 \end{pmatrix}$$

EKF Mobile Robot Localization



<u>Process Update:</u>

$$\hat{\boldsymbol{x}}\langle k+1|k\rangle = \boldsymbol{f}\left(\hat{\boldsymbol{x}}\langle k\rangle, \, \delta\langle k\rangle, \, 0\right)$$

$$\hat{\boldsymbol{P}}\langle k+1|k\rangle = \boldsymbol{F}_x\langle k\rangle \hat{\boldsymbol{P}}\langle k|k\rangle \boldsymbol{F}_x\langle k\rangle^T + \boldsymbol{F}_v\langle k\rangle \hat{\boldsymbol{V}} \boldsymbol{F}_v\langle k\rangle^T$$

<u>Observation Update:</u>

$$\hat{\boldsymbol{x}}\langle k+1|k+1\rangle = \hat{\boldsymbol{x}}\langle k+1|k\rangle + \boldsymbol{K}\langle k+1\rangle \nu\langle k+1\rangle$$

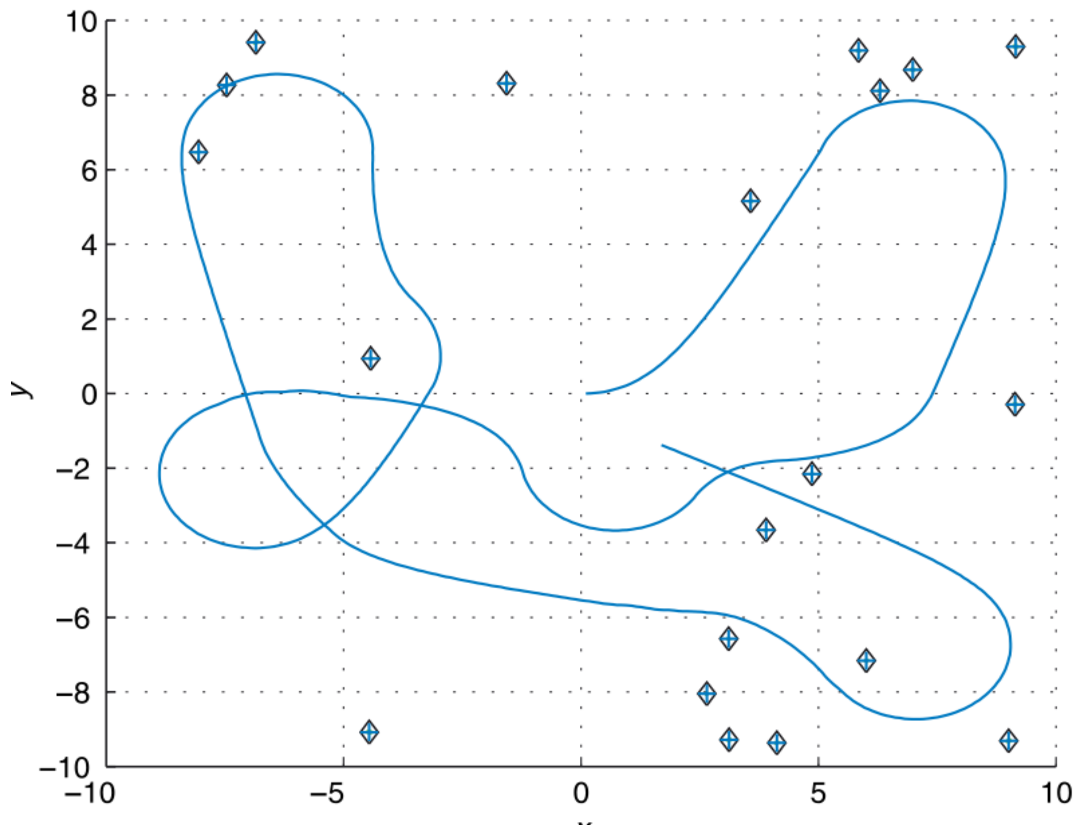
$$\hat{\boldsymbol{P}}\langle k+1|k+1\rangle = \hat{\boldsymbol{P}}\langle k+1|k\rangle \boldsymbol{F}_x\langle k\rangle^T - \boldsymbol{K}\langle k+1\rangle \boldsymbol{H}_x\langle k+1\rangle \hat{\boldsymbol{P}}\langle k+1|k\rangle$$

$$\boldsymbol{\nu}\langle k+1\rangle = \boldsymbol{z}\langle k+1\rangle - \boldsymbol{h}\left(\hat{x}(k+1|k), x_f, 0\right)$$

$$\boldsymbol{S}\langle k+1\rangle = \boldsymbol{H}_x\langle k+1\rangle \hat{\boldsymbol{P}}\langle k+1|k\rangle \boldsymbol{H}_x\langle k+1\rangle^T + \boldsymbol{H}_w\langle k+1\rangle \hat{\boldsymbol{W}}\langle k+1\rangle \boldsymbol{H}_w\langle k+1\rangle^T$$

$$\boldsymbol{K}\langle k+1\rangle = \hat{\boldsymbol{P}}\langle k+1|k\rangle \boldsymbol{H}_x\langle k+1\rangle^T \boldsymbol{S}\langle k+1\rangle^{-1}$$

Mapping using the EKF



How do we use the EKF to estimate landmark positions?

State: $\hat{\boldsymbol{x}} = (x_1, y_1, x_2, y_2, \cdots x_M, y_M)^T$

Positions of each of the M landmarks (base frame)