

# Ve492: Introduction to Artificial Intelligence

## Hidden Markov Models I



Paul Weng

UM-SJTU Joint Institute

Slides adapted from <http://ai.berkeley.edu>, AIMA, UM, CMU

# Reasoning over Time or Space

Often, we want to reason about a sequence of observations

- ❖ Speech recognition
- ❖ Robot localization
- ❖ User attention
- ❖ Medical monitoring

Need to introduce time (or space) into our models

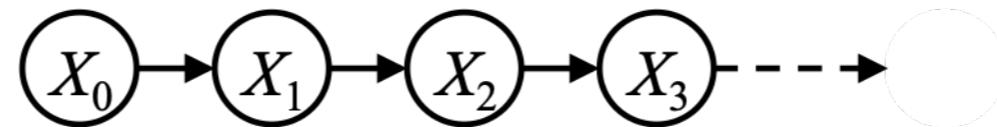
# Today

---

- ❖ Markov Models
  - ❖ Model
  - ❖ Stationary distribution
- ❖ Hidden Markov Models
  - ❖ Model
  - ❖ Forward algorithm for filtering

# Markov Models

- ❖ Value of  $X$  at a given time is called the **state**



$P(X_0)$

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

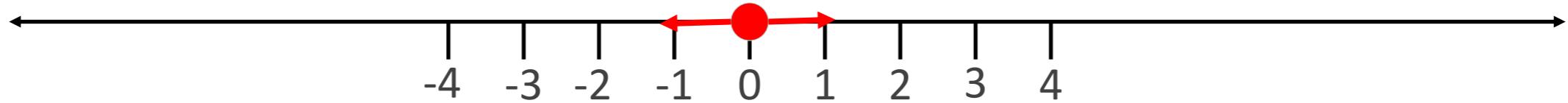
- ❖ Parameters:
  - ❖ Initial state probabilities
  - ❖ Transition probabilities (or dynamics) specify how state evolves over time
  - ❖ Stationarity assumption: transition probabilities the same at all times  $P(X'|X)$
  - ❖ Markov assumption: “future is independent of the past given the present”
    - ❖  $X_{t+1}$  is independent of  $X_0, \dots, X_{t-1}$  given  $X_t$
    - ❖ First-order Markov model ( $k$ -th-order = dependencies on  $k$  earlier steps)
  - ❖ Joint distribution  $P(X_0, \dots, X_T) = P(X_0) \prod_{t=1}^T P(X_t | X_{t-1})$

# Relation to Previous Models



- ❖ **Bayes' net**
  - ❖ Markov chain is a (growable) BN
  - ❖ We can always use generic BN reasoning on it if we truncate the chain at a fixed length
- ❖ **Markov decision process**
  - ❖ Markov chain is an MDP with one action per state (or MDP with fixed policy)

# Example: Random Walk in One Dimension



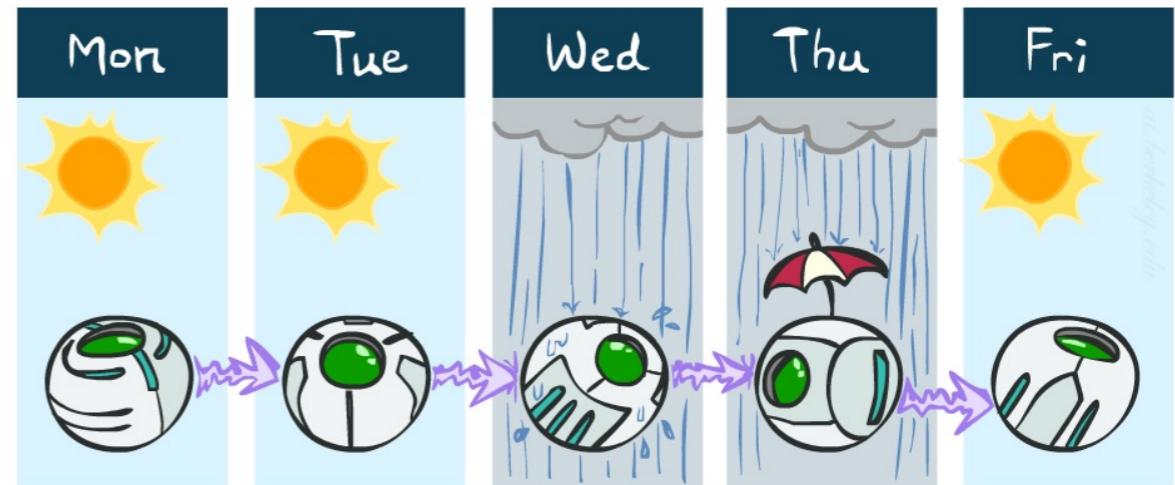
- ❖ State: location on the unbounded integer line
- ❖ Initial probability: starts at 0
- ❖ Transition model:  $P(X_t = k \pm 1 | X_{t-1} = k) = 0.5$
- ❖ Applications: particle motion in crystals, stock prices, gambling, genetics, etc.
- ❖ Questions:
  - ❖ How far does it get as a function of  $t$ ?
    - ❖ Expected distance is  $O(\sqrt{t})$
  - ❖ Does it get back to 0 or can it go off for ever and not come back?
    - ❖ In 1D and 2D, returns w.p. 1; in 3D, returns w.p. 0.34053733

# Example: n-gram Models

- ❖ State: word at position  $t$  in text (can also build letter n-grams)
- ❖ Transition model (probabilities come from empirical frequencies):
  - ❖ Unigram (zero-order):  $P(Word_t = i)$ 
    - ❖ “logical are as are confusion a may right tries agent goal the was . . .”
  - ❖ Bigram (first-order):  $P(Word_t = i | Word_{t-1} = j)$ 
    - ❖ “systems are very similar computational approach would be represented . . .”
  - ❖ Trigram (second-order):  $P(Word_t = i | Word_{t-1} = j, Word_{t-2} = k)$ 
    - ❖ “planning and scheduling are integrated the success of naive Bayes model is . . .”
- ❖ Applications: text classification, spam detection, author identification, language classification, speech recognition

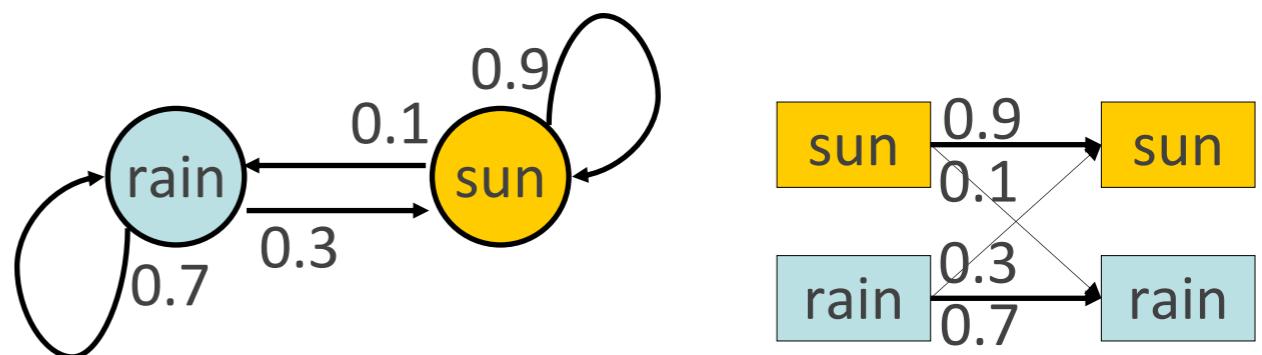
# Example: Weather Prediction

- ❖ **State:** sun or rain
- ❖ **Initial distribution:**  $<0.5, 0.5>$
- ❖ **Transition model:**



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

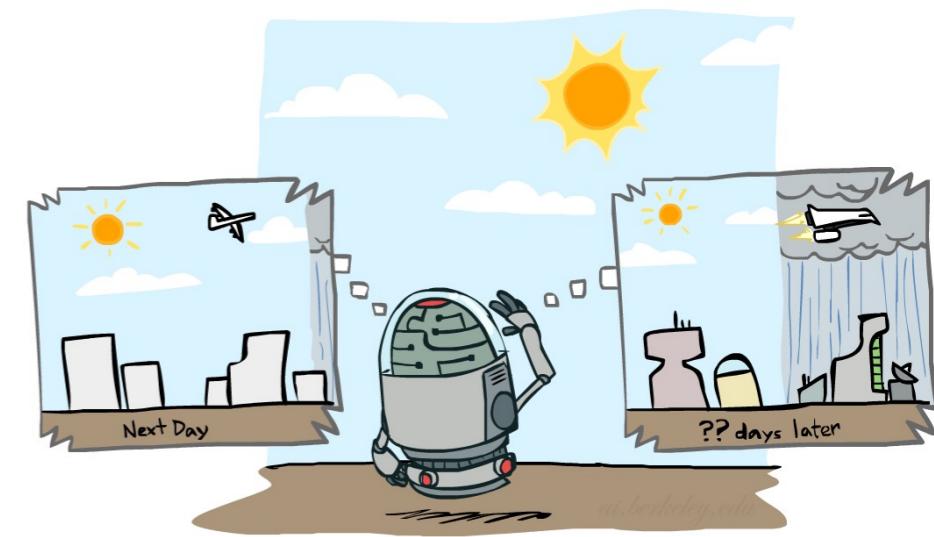
Two new ways of representing the same CPT



# Weather Prediction ctd.

- ❖ What is the weather like at time 1?

$$\begin{aligned}P(X_1) &= \sum_{x_0} P(X_1, X_0=x_0) \\&= \sum_{x_0} P(X_0=x_0) P(X_1 | X_0=x_0) \\&= 0.5<0.9, 0.1> + 0.5<0.3, 0.7> = <0.6, 0.4>\end{aligned}$$



- ❖ What is the weather like at time 2?

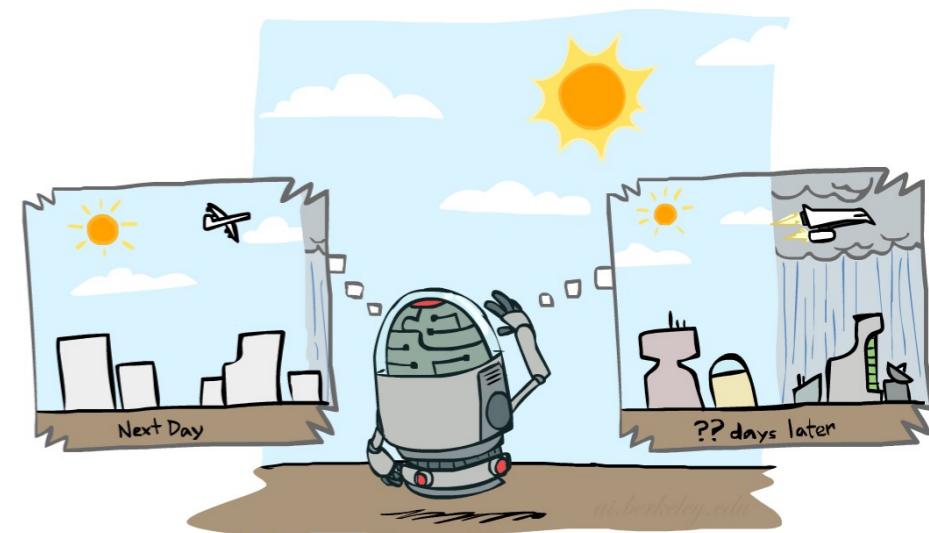
$$\begin{aligned}P(X_2) &= \sum_{x_1} P(X_2, X_1=x_1) \\&= \sum_{x_1} P(X_1=x_1) P(X_2 | X_1=x_1) \\&= 0.6<0.9, 0.1> + 0.4<0.3, 0.7> = <0.66, 0.34>\end{aligned}$$

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

# Quiz: Weather Prediction

- ❖ Time 2:  $<0.66, 0.34>$

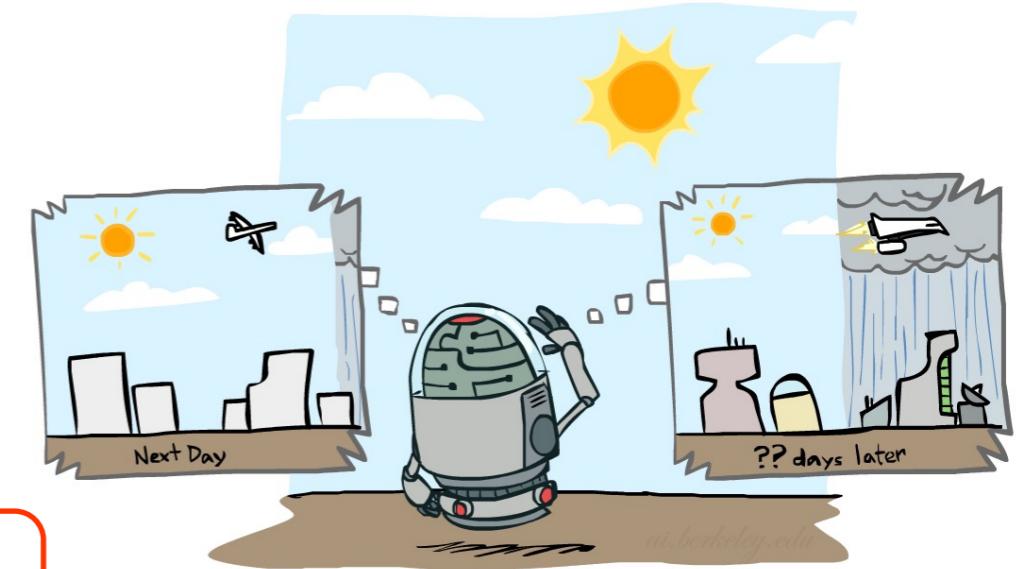
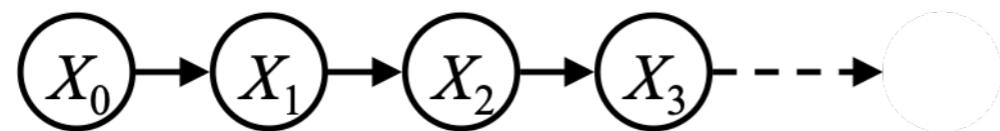
- ❖ What is the weather like at time 3?



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

# Simple Forward Algorithm

- ❖ Question: What's  $P(X)$  on some day  $t$ ?



$P(X_0)$  known

$$\begin{aligned} P(X_t) &= \sum_{x_{t-1}} P(X_t, X_{t-1}=x_{t-1}) \\ &= \sum_{x_{t-1}} P(X_{t-1}=x_{t-1}) P(X_t | X_{t-1}=x_{t-1}) \end{aligned}$$

Probability from previous iteration

Transition model

# Matrix Form

- ❖ What is the weather like at time 2?

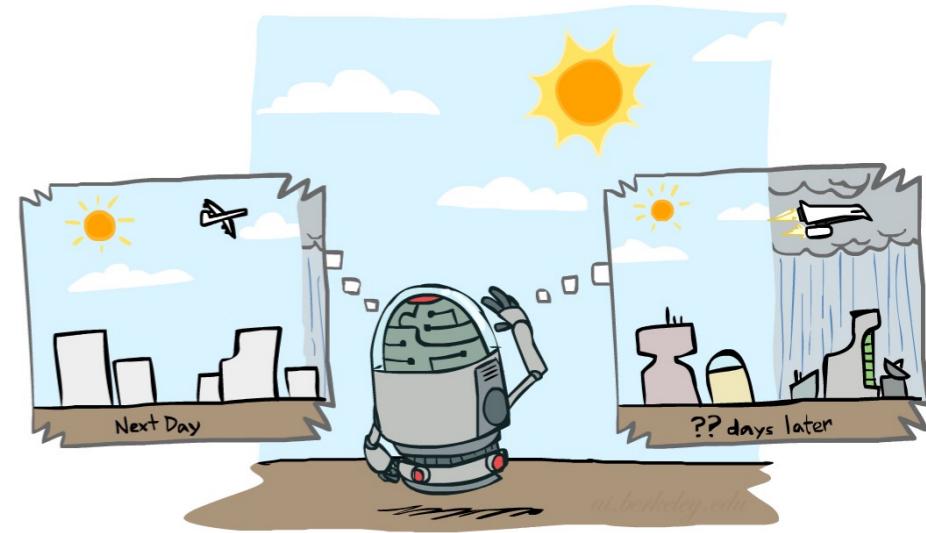
$$P(X_2) = 0.6\langle 0.9, 0.1 \rangle + 0.4\langle 0.3, 0.7 \rangle = \langle 0.66, 0.34 \rangle$$

- ❖ In matrix-vector form:

$$P(X_2) = \begin{pmatrix} 0.9 & 0.3 \\ 0.1 & 0.7 \end{pmatrix} \begin{pmatrix} 0.6 \\ 0.4 \end{pmatrix} = \begin{pmatrix} 0.66 \\ 0.34 \end{pmatrix}$$

- ❖ More generally,

$$P(X_{t+1}) = T^T P(X_t)$$



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

# Example Runs of Simple Forward Algorithm

- ❖ From initial observation of sun

$$\begin{array}{ccccc} \left\langle \begin{matrix} 1.0 \\ 0.0 \end{matrix} \right\rangle & \left\langle \begin{matrix} 0.9 \\ 0.1 \end{matrix} \right\rangle & \left\langle \begin{matrix} 0.84 \\ 0.16 \end{matrix} \right\rangle & \left\langle \begin{matrix} 0.804 \\ 0.196 \end{matrix} \right\rangle & \xrightarrow{\hspace{1cm}} \left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle \\ P(X_0) & P(X_1) & P(X_2) & P(X_3) & P(X_\infty) \end{array}$$

- ❖ From initial observation of rain

$$\begin{array}{ccccc} \left\langle \begin{matrix} 0.0 \\ 1.0 \end{matrix} \right\rangle & \left\langle \begin{matrix} 0.3 \\ 0.7 \end{matrix} \right\rangle & \left\langle \begin{matrix} 0.48 \\ 0.52 \end{matrix} \right\rangle & \left\langle \begin{matrix} 0.588 \\ 0.412 \end{matrix} \right\rangle & \xrightarrow{\hspace{1cm}} \left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle \\ P(X_0) & P(X_1) & P(X_2) & P(X_3) & P(X_\infty) \end{array}$$

- ❖ From yet another initial distribution  $P(X_1)$ :

$$\begin{array}{ccc} \left\langle \begin{matrix} p \\ 1-p \end{matrix} \right\rangle & \cdots & \xrightarrow{\hspace{1cm}} \left\langle \begin{matrix} 0.75 \\ 0.25 \end{matrix} \right\rangle \\ P(X_0) & & P(X_\infty) \end{array}$$

# Stationary Distributions

- ❖ The limiting distribution (if it exists) is called the ***stationary distribution***  $P_\infty$  of the chain
- ❖ It satisfies  $P_\infty = P_{\infty+1} = T^\top P_\infty$
- ❖ Solving for  $P_\infty$  in the example:

$$\begin{pmatrix} 0.9 & 0.3 \\ 0.1 & 0.7 \end{pmatrix} \begin{pmatrix} p \\ 1-p \end{pmatrix} = \begin{pmatrix} p \\ 1-p \end{pmatrix}$$

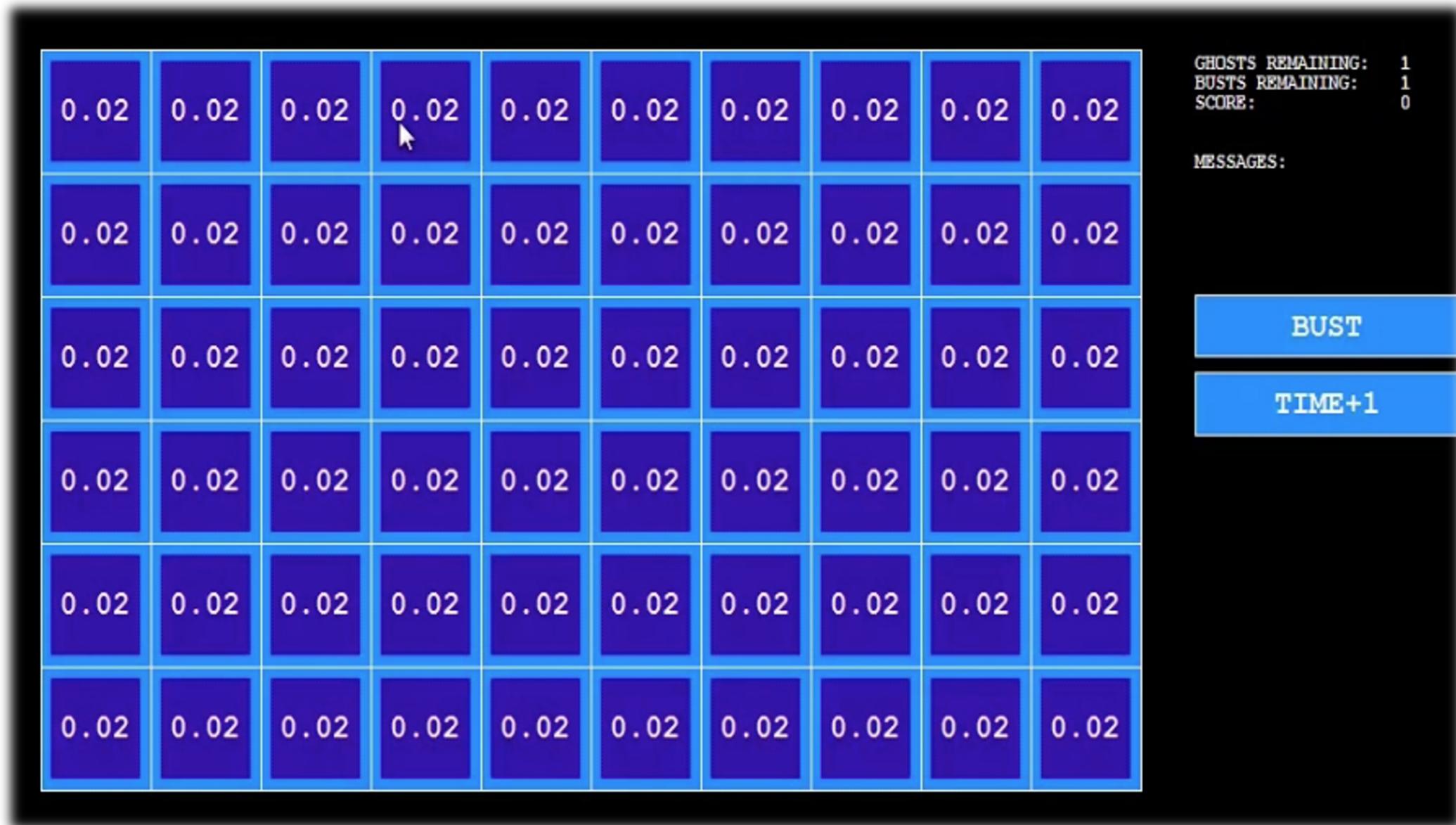
$$0.9p + 0.3(1-p) = p$$

$$p = 0.75$$

Stationary distribution is  $\langle 0.75, 0.25 \rangle$  ***regardless of starting distribution***



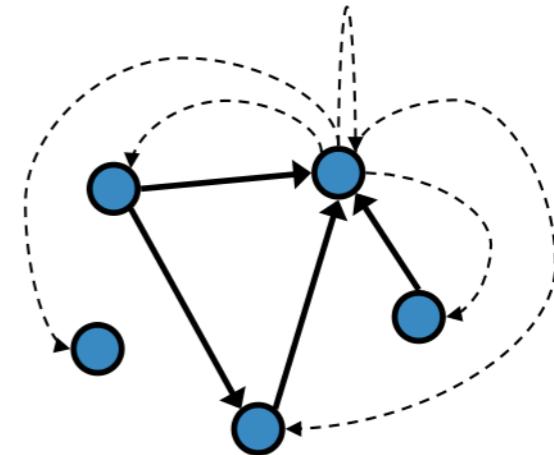
# Ghostbusters - Circular Dynamics



# Application of Stationary Distribution: Web Link Analysis

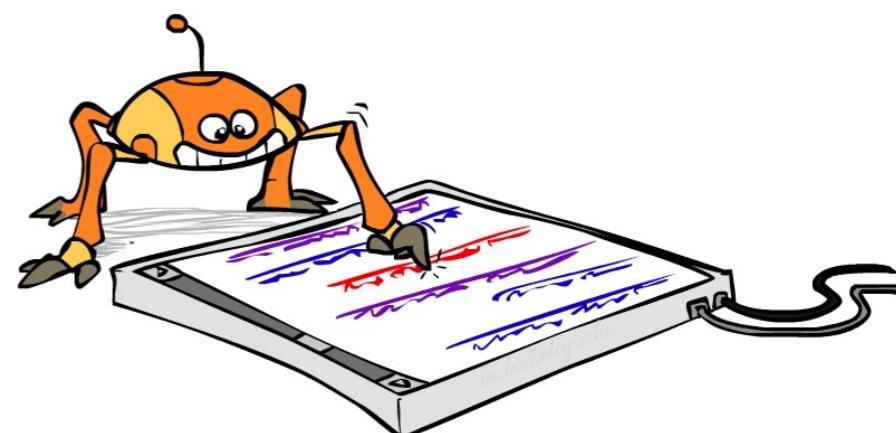
## Web browsing

- ❖ State = webpage
- ❖ 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
- ❖ Google 1.0 (Pagerank) returned the set of pages containing all your keywords in decreasing rank, now all search engines use link analysis along with many other factors (rank actually getting less important over time)



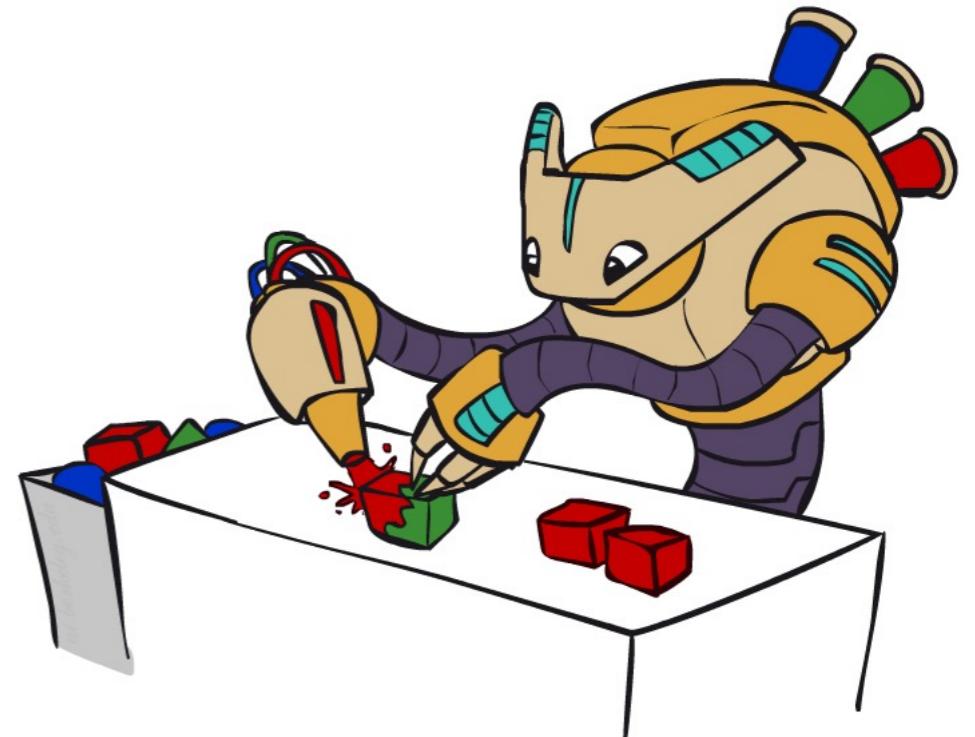
# Application of Stationary Distributions: Gibbs Sampling\*

## Gibbs Sampling

- ❖ State = joint instantiation over all hidden and query variables;  $\{X_1, \dots, X_n\} = H \cup Q$
- ❖ Some initial distribution
- ❖ Transitions:
  - ❖ With probability  $1/n$  resample variable  $X_j$  according to  $P(X_j | x_1, x_2, \dots, x_{j-1}, x_{j+1}, \dots, x_n, e_1, \dots, e_m)$

## Stationary distribution:

- ❖ Conditional distribution  $P(X_1, X_2, \dots, X_n | e_1, \dots, e_m)$
- ❖ Means that when running Gibbs sampling long enough we get a sample from the desired distribution
- ❖ Requires some proof to show this is true!



# Hidden Markov Models



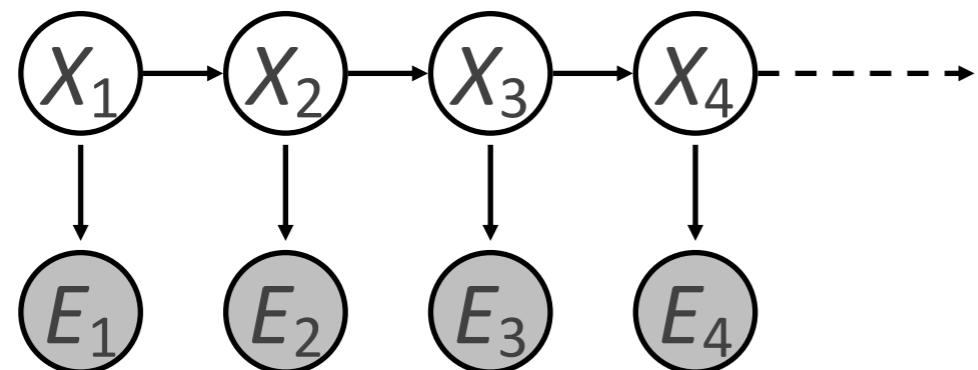
# Hidden Markov Models

Markov chains not so useful for most agents

- ❖ Need observations to update your beliefs

Hidden Markov models (HMMs)

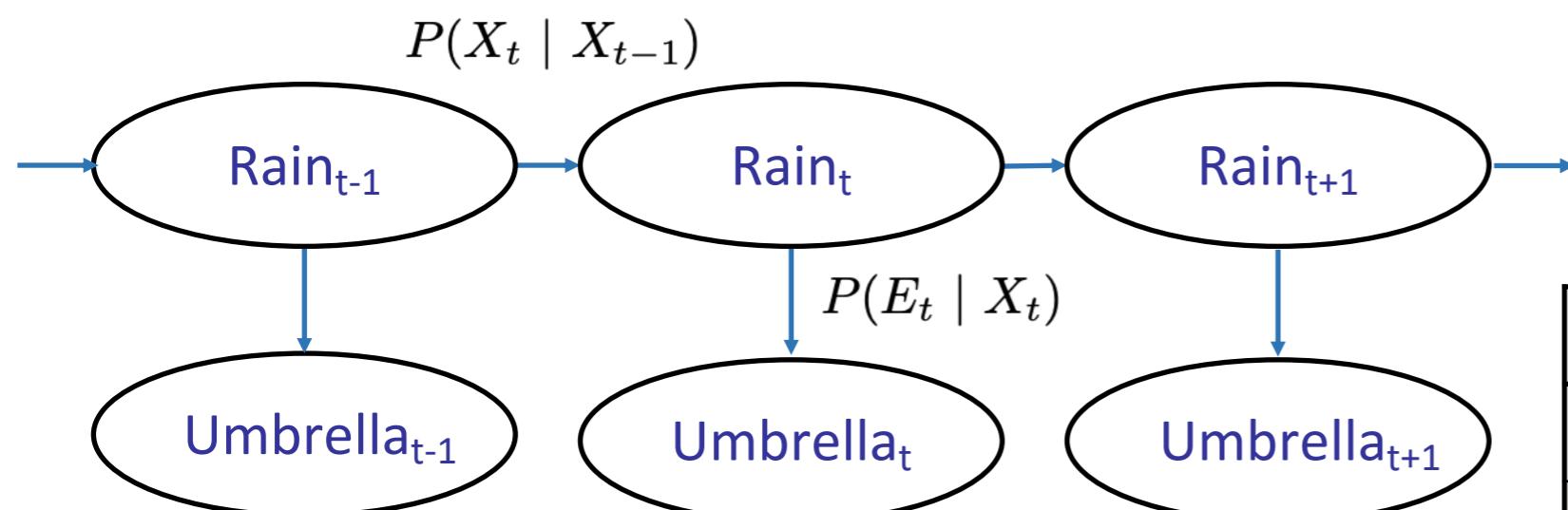
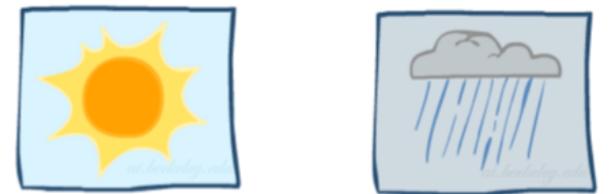
- ❖ Underlying Markov chain over states  $X$
- ❖ You observe outputs (effects) at each time step



# HMM Definition and Weather Example

An HMM is defined by:

- ❖ Initial distribution:  $P(X_0)$  or  $P(X_1)$
- ❖ Transitions:  $P(X_t | X_{t-1})$
- ❖ Emissions:  $P(E_t | X_t)$

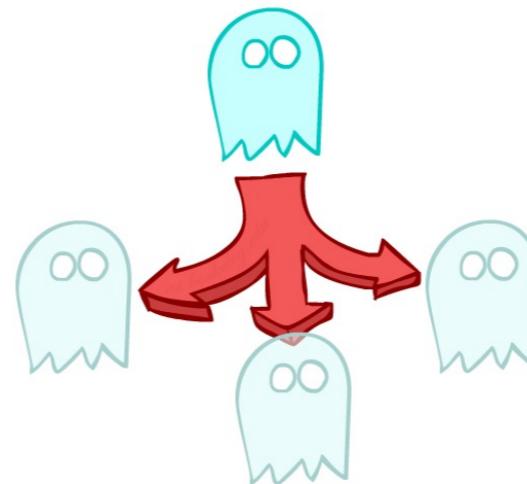


R <sub>t-1</sub>	R <sub>t</sub>	P(R <sub>t</sub>   R <sub>t-1</sub> )	R <sub>t</sub>	U <sub>t</sub>	P(U <sub>t</sub>   R <sub>t</sub> )
+r	+r	0.7	+r	+u	0.9
+r	-r	0.3	+r	-u	0.1
-r	+r	0.3	-r	+u	0.2
-r	-r	0.7	-r	-u	0.8

# Example: Ghostbusters HMM

$P(X_0) = \text{uniform}$

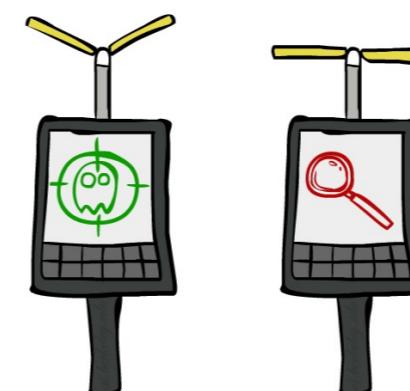
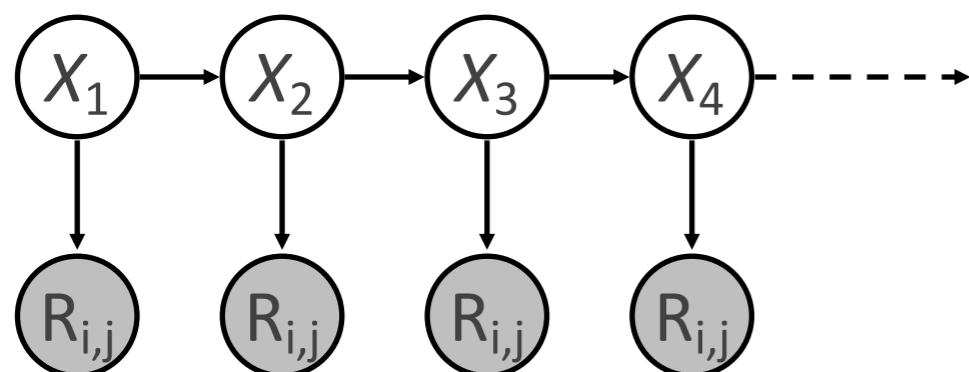
$P(X' | X)$  = usually move clockwise, but sometimes move in a random direction or stay in place



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

$P(X_0)$

$P(R_{ij} | X)$  = same sensor model as before:  
red means close, green means far away.



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

$P(X' | X = \langle 1, 2 \rangle)$

# HMM as Probability Model

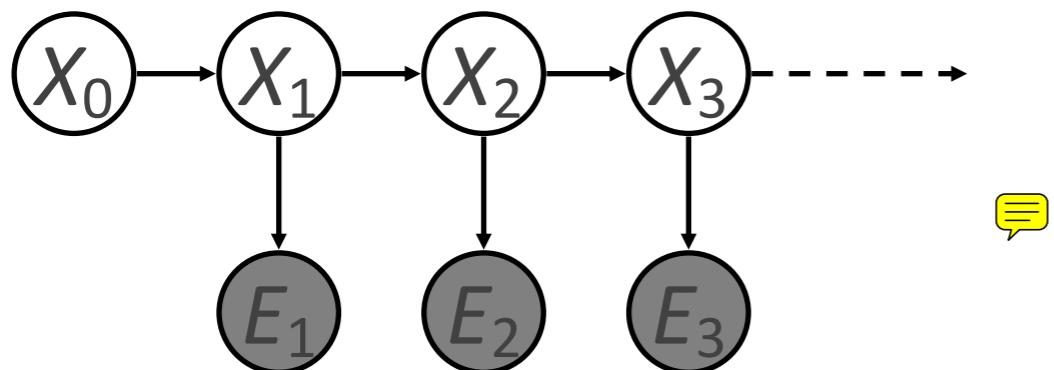
- ❖ Joint distribution for Markov model:

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

Joint distribution for hidden Markov model:

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

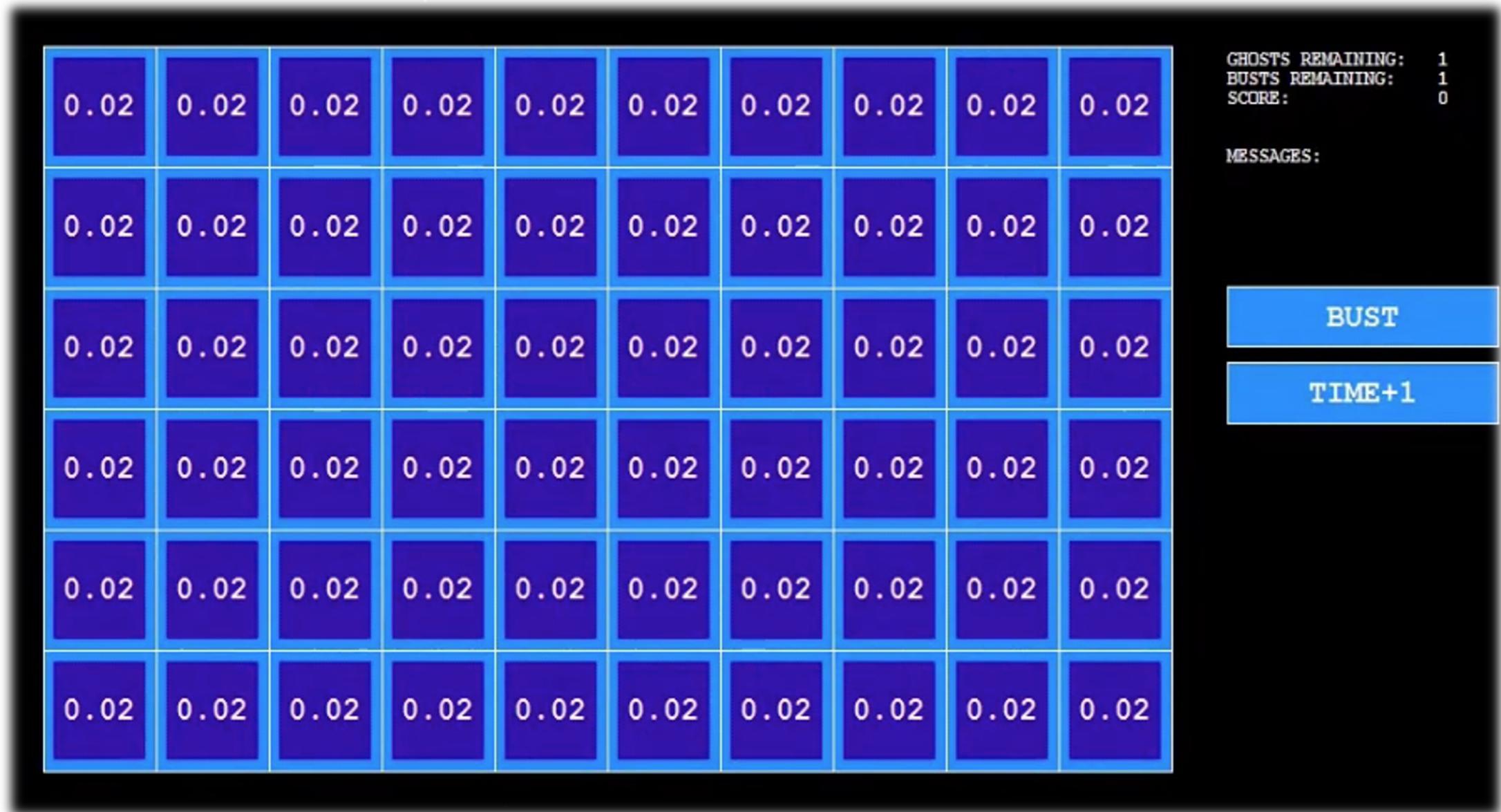
- ❖ Markov assumption
  - ❖ Future states are independent of the past given the present
  - ❖ Current evidence is independent of everything else given current state
- ❖ Are evidence variables independent of each other?



Useful notation:

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

# Ghostbusters – Circular Dynamics - HMM



# Real HMM Examples

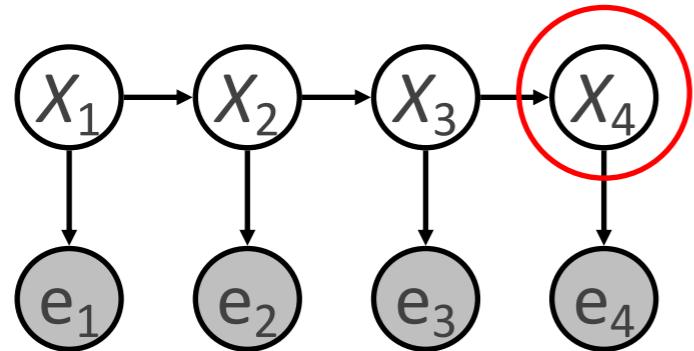
- ❖ **Speech recognition HMMs:**
  - ❖ Observations are acoustic signals (continuous valued)
  - ❖ States are specific positions in specific words (so, tens of thousands)
- ❖ **Machine translation HMMs:**
  - ❖ Observations are words (tens of thousands)
  - ❖ States are translation options
- ❖ **Robot 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.

# Inference tasks

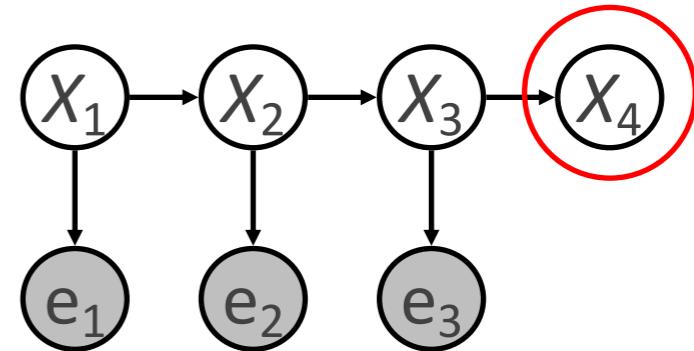
- ❖ **Filtering:**  $P(X_t | e_{1:t})$ 
  - ❖ *Belief state*—input to the decision process of a rational agent
- ❖ **Prediction:**  $P(X_{t+k} | e_{1:t})$  for  $k > 0$ 
  - ❖ Evaluation of possible sequences; like filtering without the evidence
- ❖ **Smoothing:**  $P(X_k | e_{1:t})$  for  $0 \leq k < t$ 
  - ❖ Better estimate of past states, essential for learning
- ❖ **Most likely explanation:**  $\arg \max_{x_{1:t}} P(x_{1:t} | e_{1:t})$ 
  - ❖ Speech recognition, decoding with a noisy channel

# HMM Queries

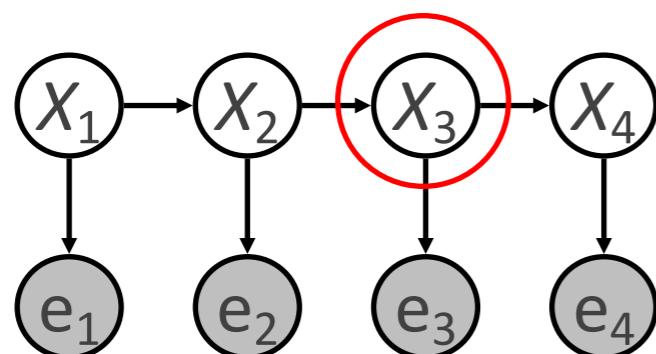
Filtering:  $P(X_t | e_{1:t})$



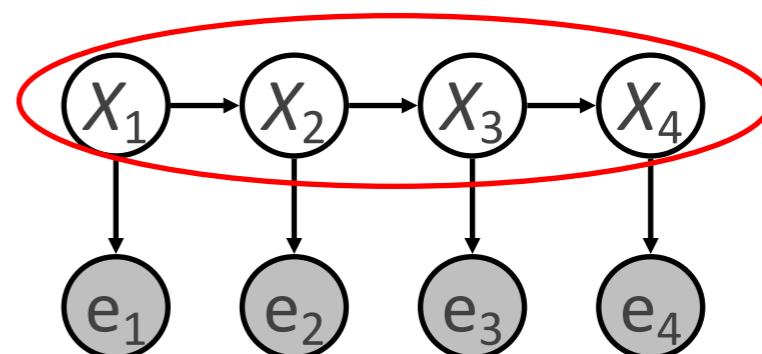
Prediction:  $P(X_{t+h} | e_{1:t-1})$



Smoothing:  $P(X_t | e_{1:N})$ ,  $t < N$



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

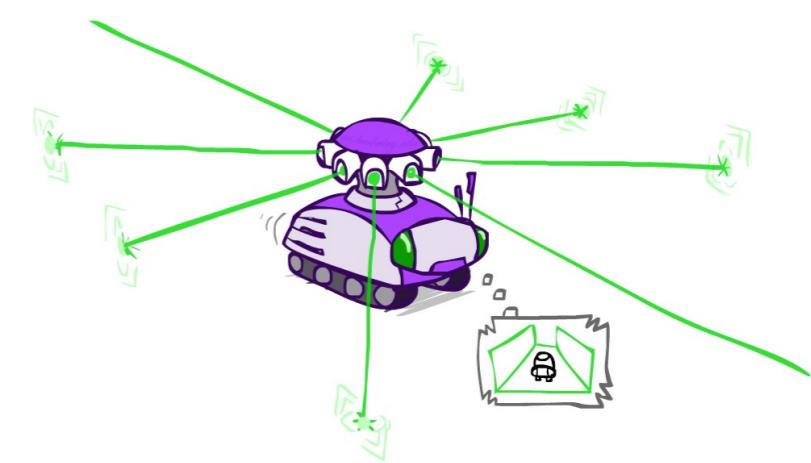
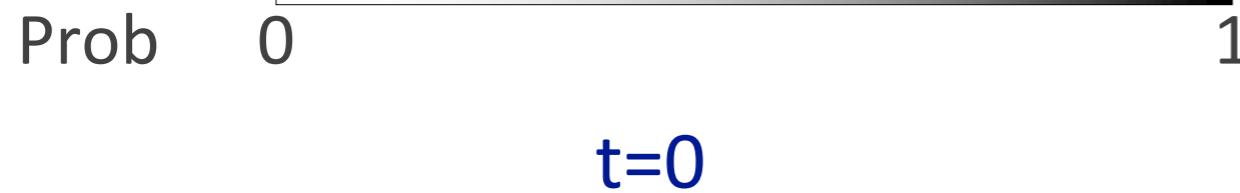
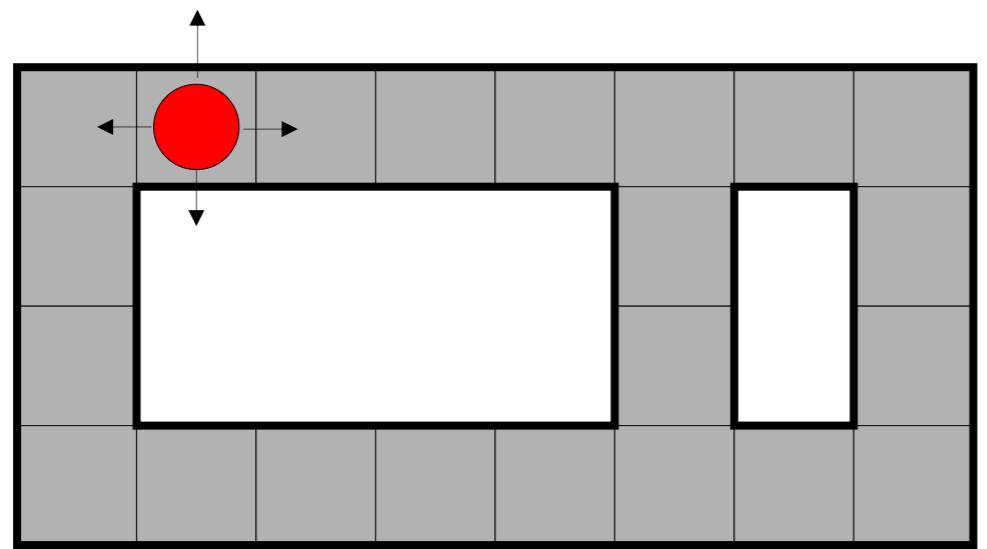


# Filtering

- ❖ Filtering (or monitoring or state estimation) is the task of maintaining the distribution  $f_{1:t} = P(X_t | e_{1:t})$  over time
- ❖ We start with  $f_0$  in an initial setting, usually uniform
- ❖ Filtering is a fundamental task in engineering and science
- ❖ The Kalman filter (continuous variables, linear dynamics, Gaussian noise) was invented in 1960 and used for trajectory estimation in the Apollo program; core ideas used by Gauss for planetary observations

# Example: Robot Localization

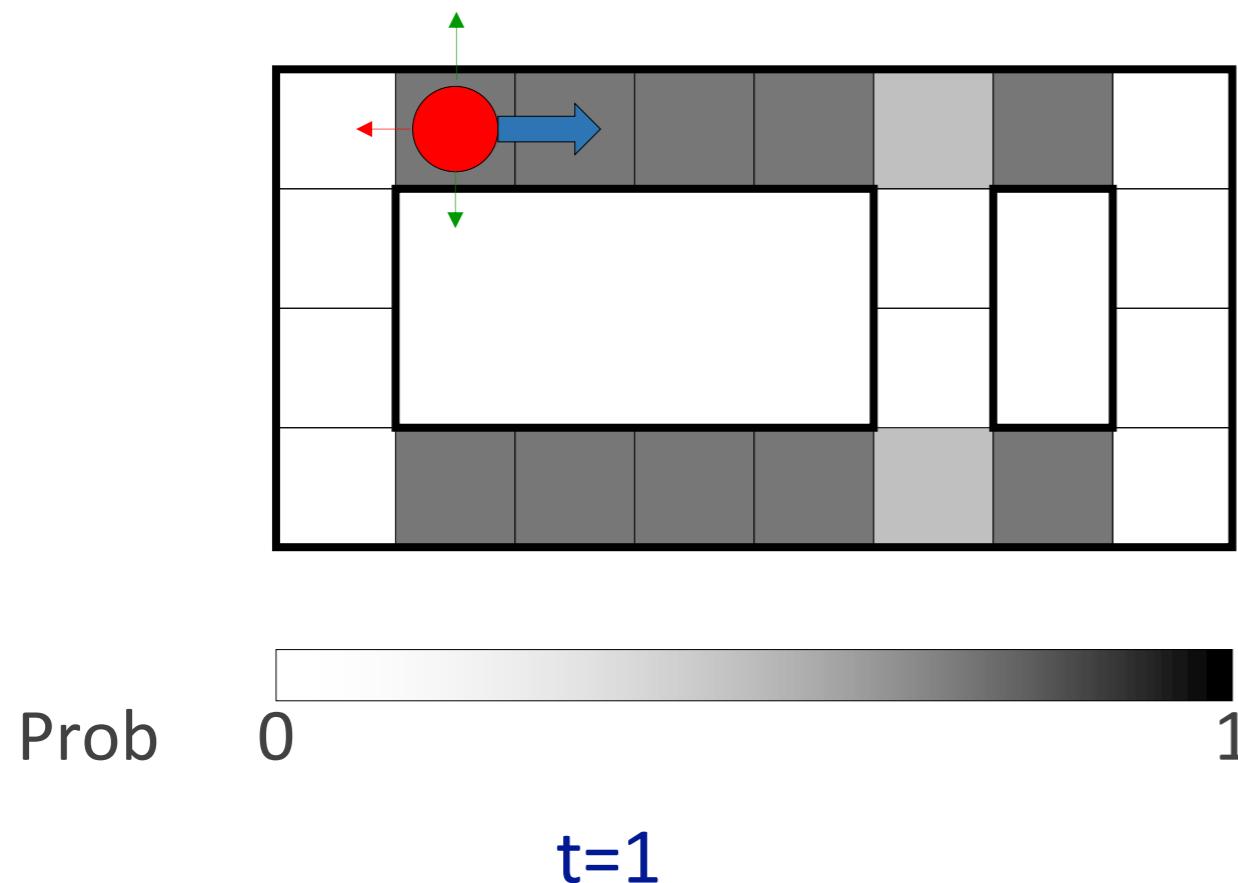
Example from  
Michael  
Pfeiffer



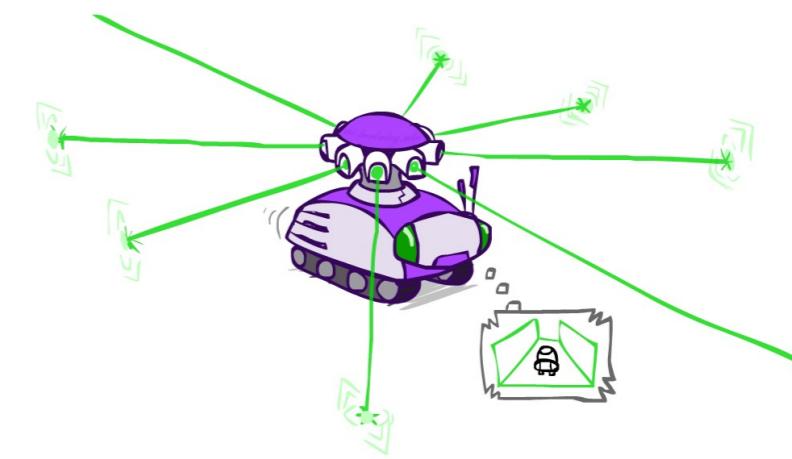
Sensor model: can read in which directions there is a wall, never more than 1 mistake

Motion model: may not execute action with small prob.

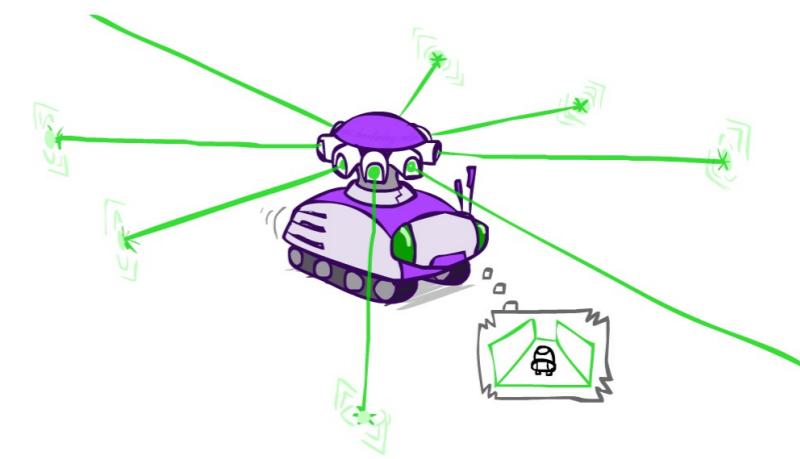
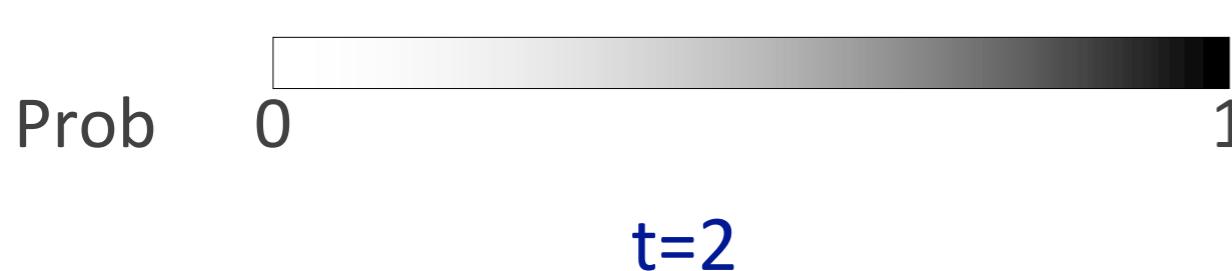
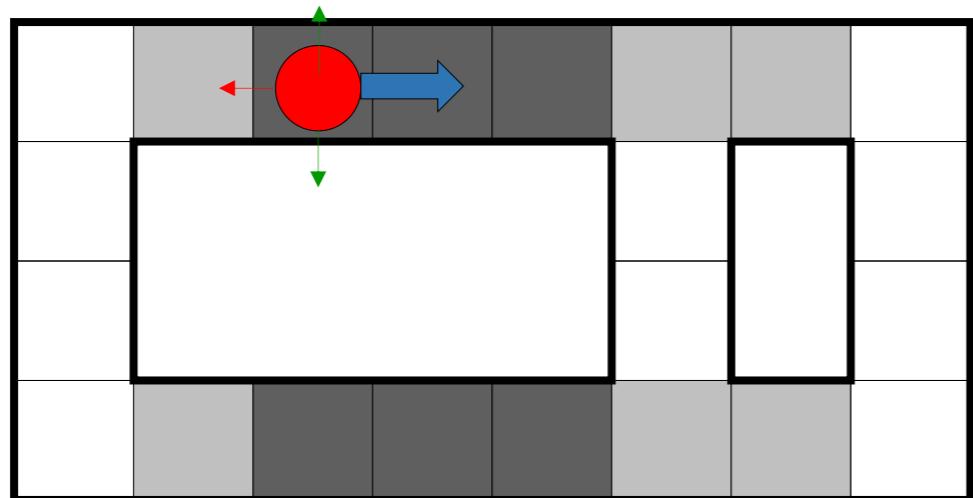
# Example: Robot Localization



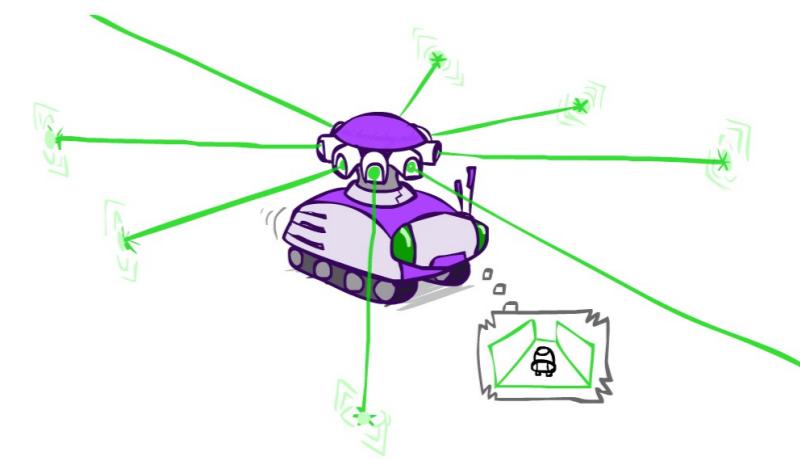
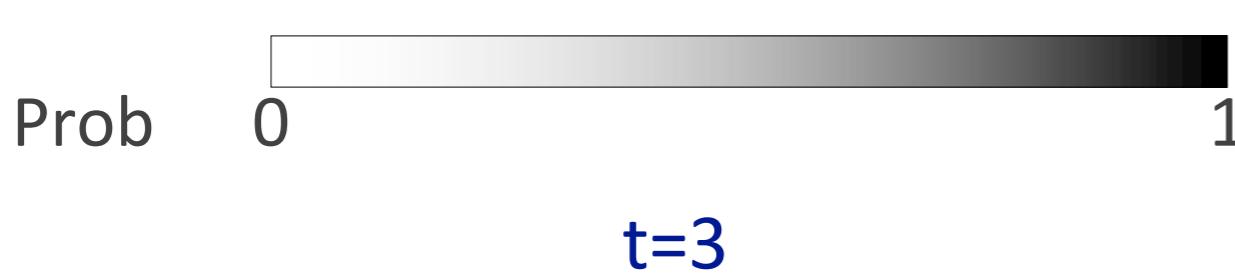
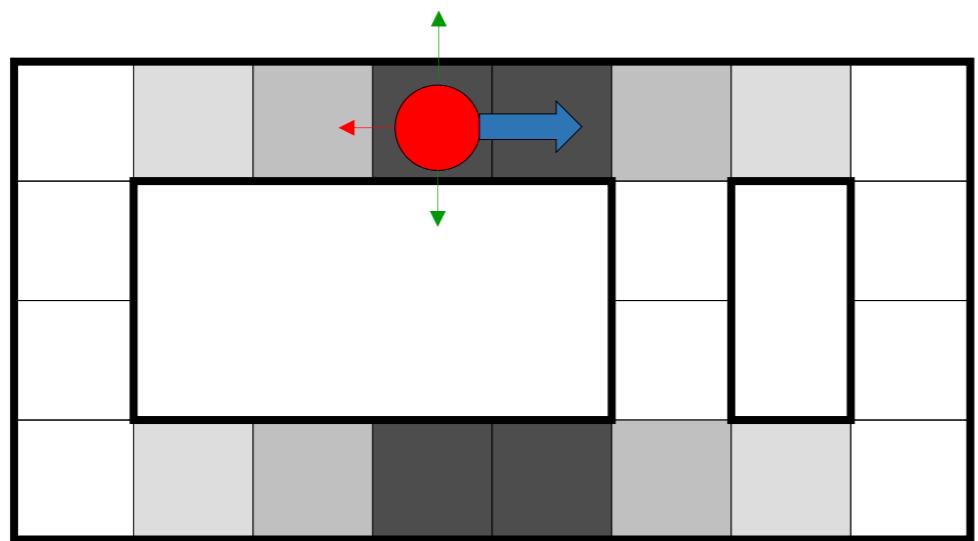
Lighter grey: was possible to get the reading, but less likely b/c required 1 mistake



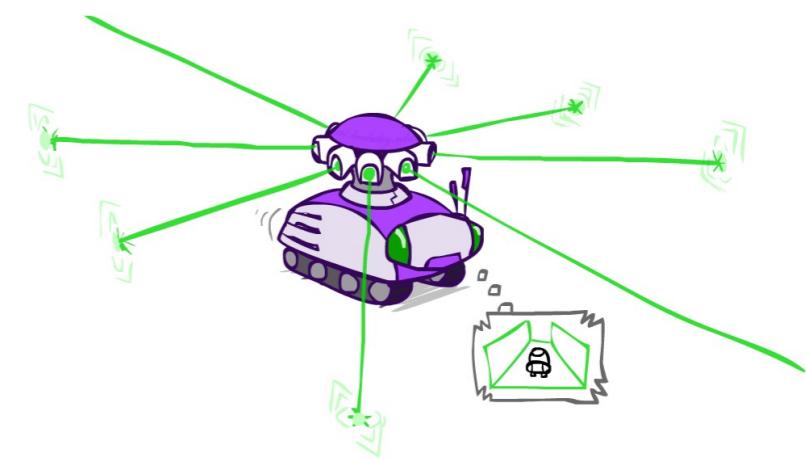
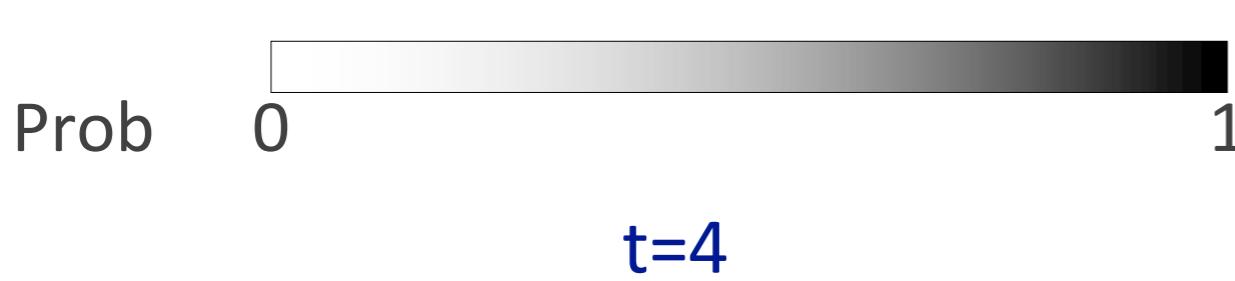
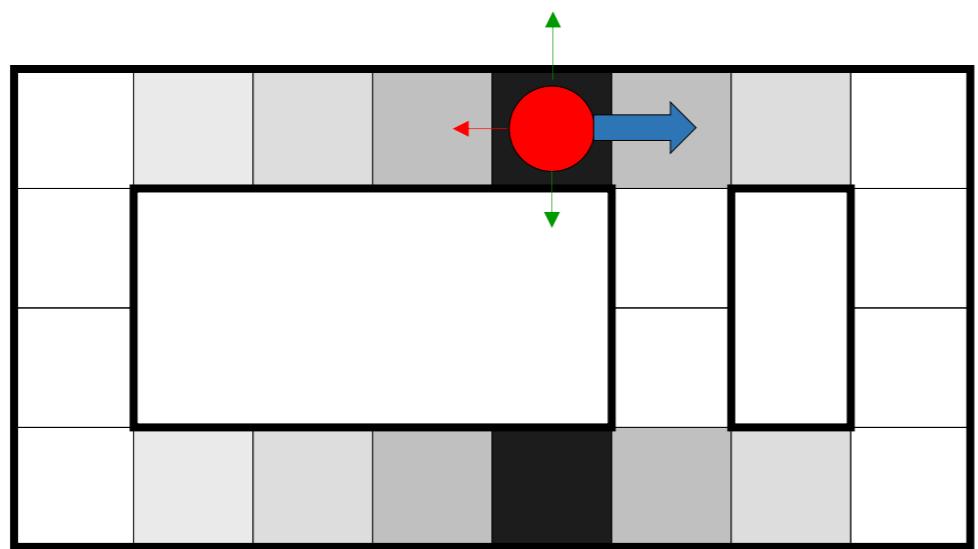
# Example: Robot Localization



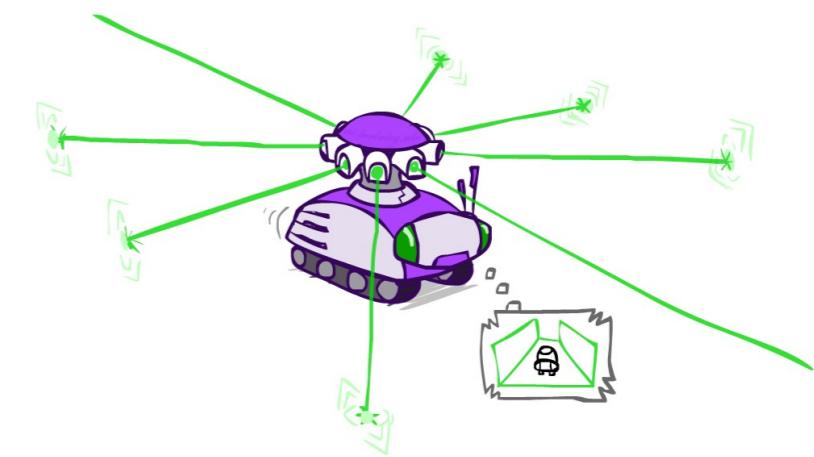
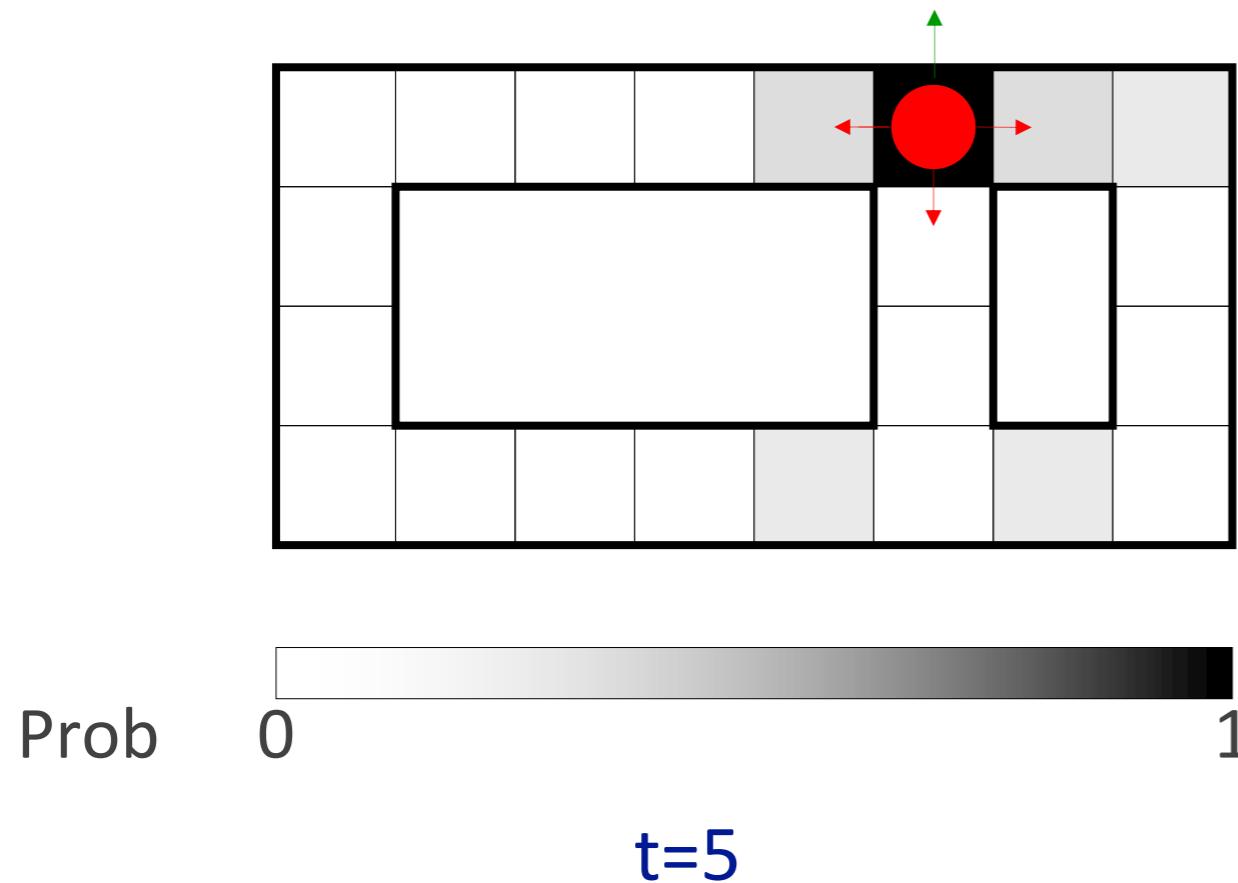
# Example: Robot Localization



# Example: Robot Localization



# Example: Robot Localization



$t=5$

# Filtering Algorithm

- ❖ Goal: design a **recursive filtering algorithm** of the form

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

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

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

Bayes rule  
 $\alpha$  normalization

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

Conditional ind.

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

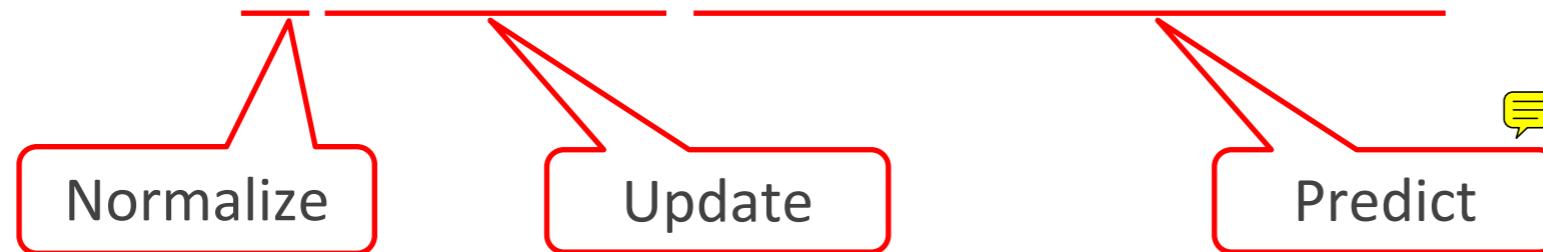
Law of total prob.

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

Conditional ind.

# Forward Algorithm

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



$$\diamond f_{1:t+1} = \text{FORWARD}(f_{1:t}, e_{t+1})$$

- ❖ 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

# Matrix Form

- ❖ Transition matrix  $T$ , observation matrix  $O_t$ 
  - ❖ Observation matrix has state likelihoods for  $E_t$  along diagonal
    - ❖ e.g., for  $U_1 = \text{true}$ ,  $O_1 = \begin{pmatrix} 0.2 & 0 \\ 0 & 0.9 \end{pmatrix}$
- ❖ Filtering algorithm becomes
  - ❖  $f_{1:t+1} = \alpha O_{t+1} T^\top f_{1:t}$

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

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

# 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

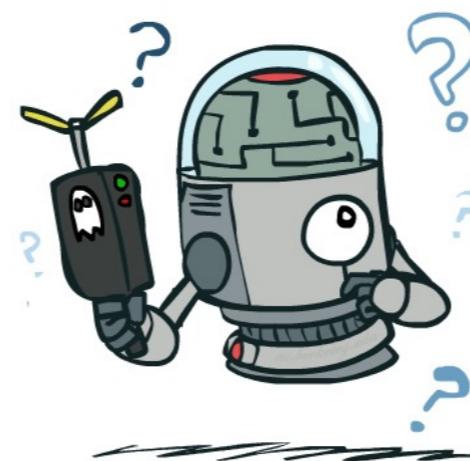
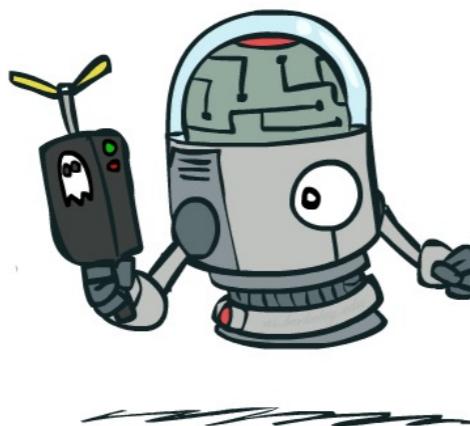
$T = 1$

<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”

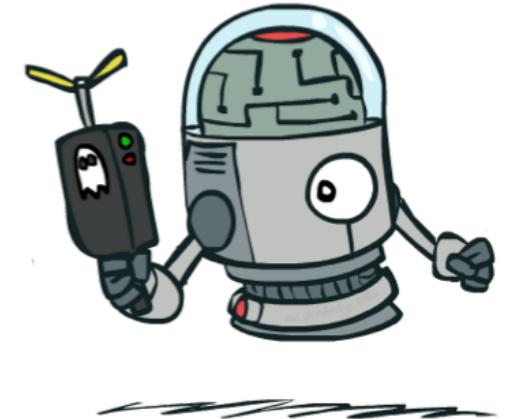
0.05	0.01	0.05	<0.01	<0.01	<0.01
0.02	0.14	0.11	0.35	<0.01	<0.01
0.07	0.03	0.05	<0.01	0.03	<0.01
0.03	0.03	<0.01	<0.01	<0.01	<0.01

Before observation



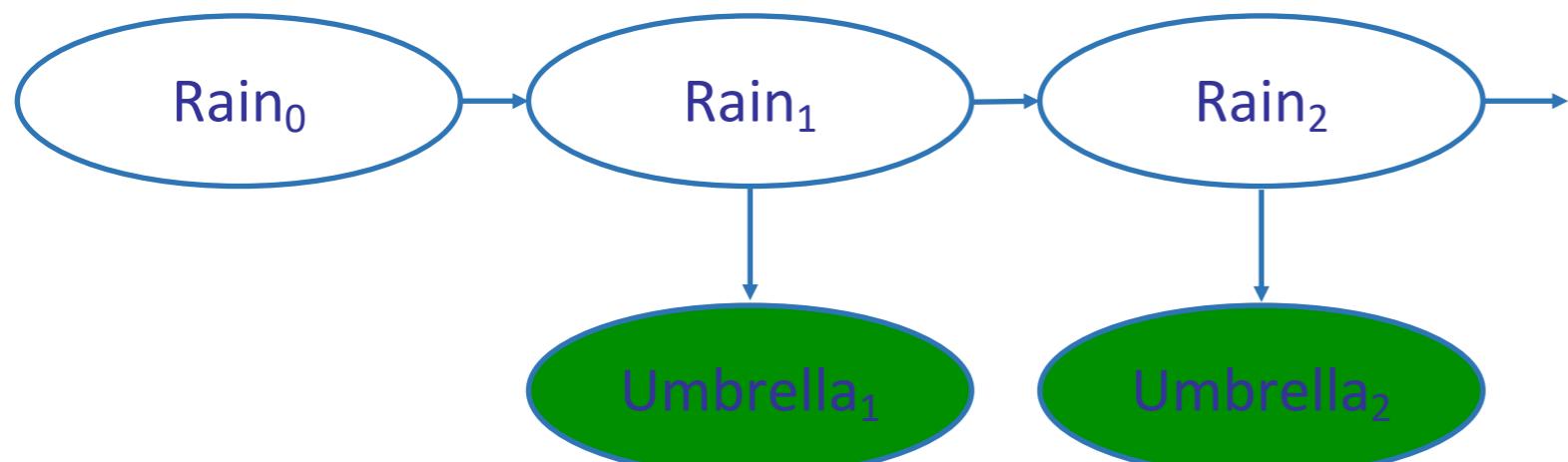
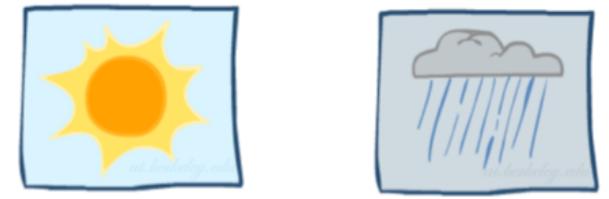
<0.01	<0.01	<0.01	<0.01	0.02	<0.01
<0.01	<0.01	<0.01	0.83	0.02	<0.01
<0.01	<0.01	0.11	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01

After observation



# Example: Weather HMM

$$\begin{array}{ll}
 \text{predict} & f'(+r) = 0.5 \\
 & f'(-r) = 0.5 \\
 \text{update} & f(+r) = 0.818 \\
 & f(-r) = 0.182
 \end{array}
 \quad
 \begin{array}{ll}
 \text{predict} & f'(+r) = 0.627 \\
 & f'(-r) = 0.373 \\
 \text{update} & f(+r) = 0.883 \\
 & f(-r) = 0.117
 \end{array}$$



$R_t$	$R_{t+1}$	$P(R_{t+1} R_t)$	$R_t$	$U_t$	$P(U_t R_t)$
+r	+r	0.7	+r	+u	0.9
+r	-r	0.3	+r	-u	0.1
-r	+r	0.3	-r	+u	0.2
-r	-r	0.7	-r	-u	0.8

# Pacman – Sonar

