CS325 Artificial Intelligence
Robotics II – Navigation (Ch. 25)

Dr. Cengiz Günay, Emory Univ.

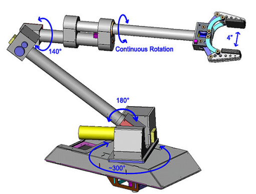
Robots with Different Degrees of Freedom

Different robots has different movements and degrees of freedom:

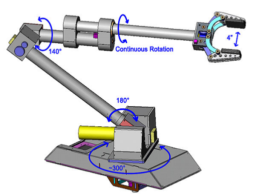
Robots with Different Degrees of Freedom

Different robots has different movements and degrees of freedom:

- robotic arm: only joints



Robots with Different Degrees of Freedom



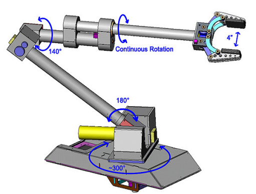
Different robots has different movements and degrees of freedom:

- robotic arm: only joints
- quadcopter/predator: all directions

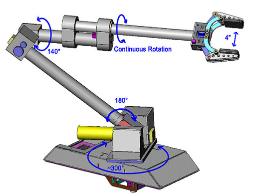
Robots with Different Degrees of Freedom

Different robots has different movements and degrees of freedom:

- robotic arm: only joints
- quadcopter/predator: all directions
- roomba: location + heading



Robots with Different Degrees of Freedom



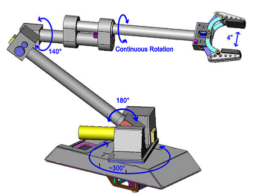
Different robots has different movements and degrees of freedom:

- robotic arm: only joints
- quadcopter/predator: all directions
- roomba: location + heading

Navigate these with:

particle filters: for **state estimation** and **future prediction**

Robots with Different Degrees of Freedom



Different robots has different movements and degrees of freedom:

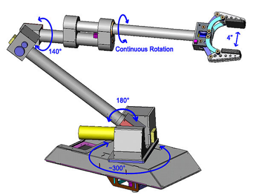
- robotic arm: only joints
- quadcopter/predator: all directions
- roomba: location + heading

Navigate these with:

particle filters: for **state estimation** and **future prediction**

planning: reach target from current state

Robots with Different Degrees of Freedom



Different robots has different movements and degrees of freedom:

- robotic arm: only joints
- quadcopter/predator: all directions
- roomba: location + heading

Navigate these with:

particle filters: for **state estimation** and **future prediction**

planning: reach target from current state

We'll make our own self driving car :)

Entry/Exit Surveys

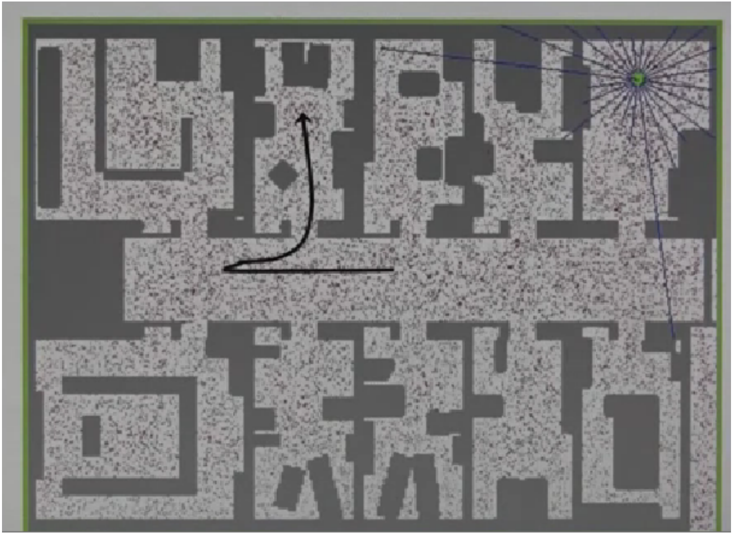
Exit survey: Robotics I – Autonomous Robots

- Which parameters do you have in the *dynamic* state of the roomba?
- How can we use the dynamic state parameters to estimate the current robot state?

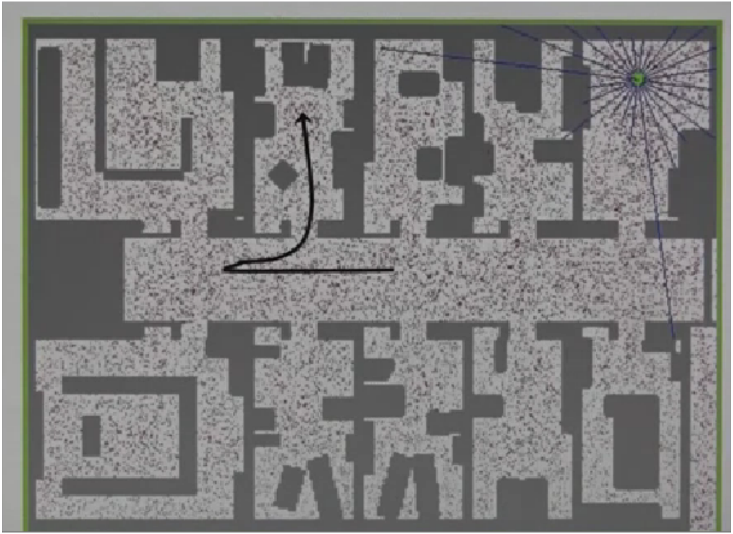
Entry survey: Robotics II – Navigation (0.25 pts)

- What were the steps in the particle filter algorithm?
- In what task would a robot need to combine a particle filter with planning? Briefly explain their roles in at least one example.

Remember Particle Filters?



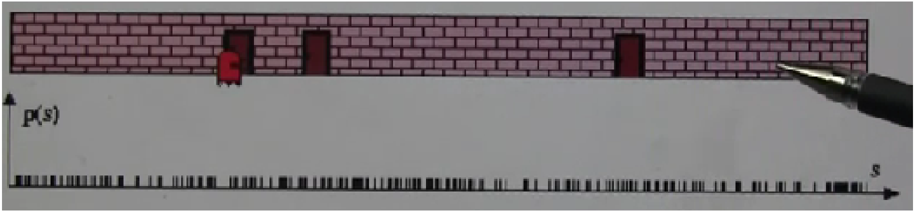
Remember Particle Filters?



Remember why we needed location and heading in particles?

Localization with Particle Filters

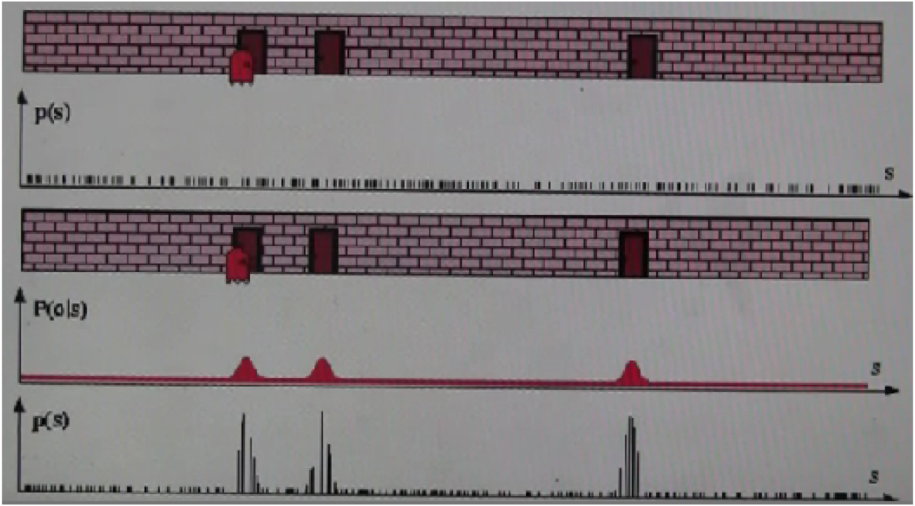
Particle filtering: weights show likelihood; pick particles, shift, and repeat.



Step 1: Initialize particles from homogeneous distribution.

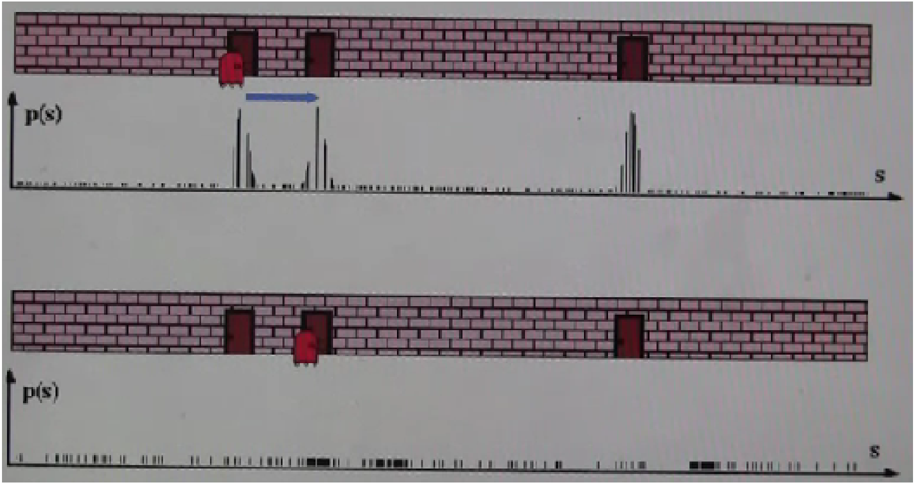
Localization with Particle Filters

Particle filtering: weights show likelihood; pick particles, shift, and repeat.



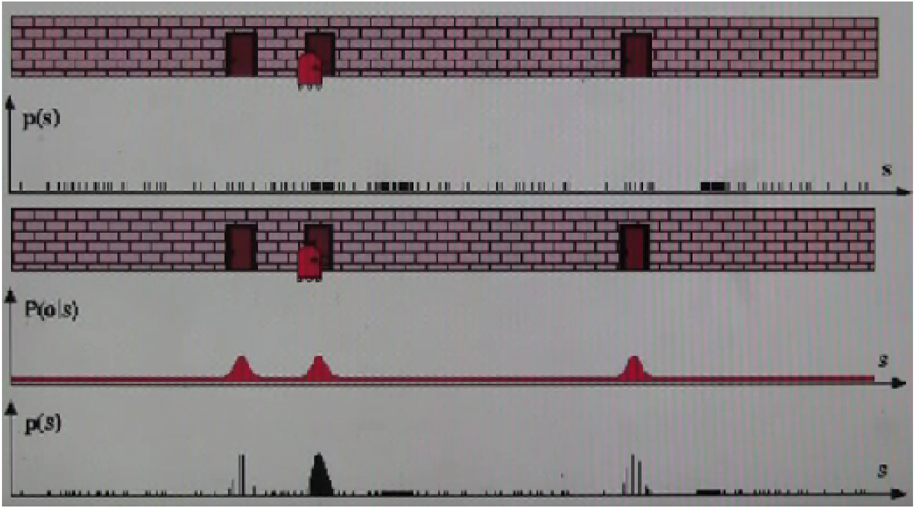
Step 2: Use sensors to **estimate likely locations**.

Particle filtering: weights show likelihood; pick particles, shift, and repeat.



Step 3: Resample likely particles and **predict next state**.

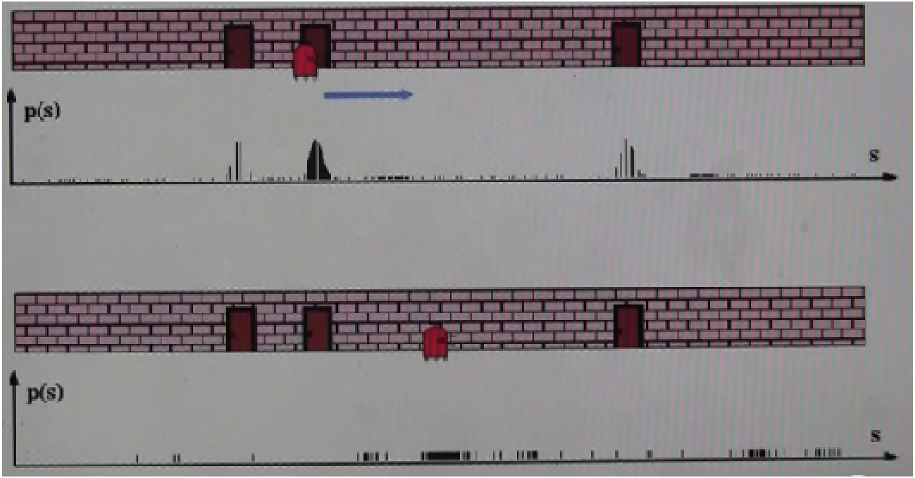
Particle filtering: weights show likelihood; pick particles, shift, and repeat.



Step 4: (again) Estimate location from sensors.

Localization with Particle Filters

Particle filtering: weights show likelihood; pick particles, shift, and repeat.



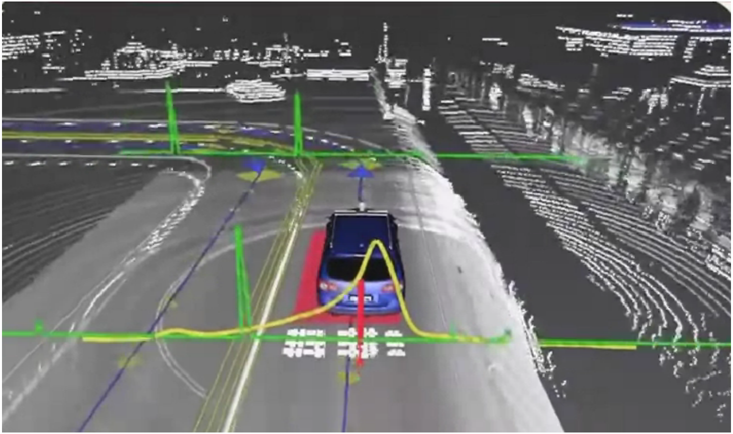
Step 5: (again) Resample and predict state from movement.

Particle Filter Algorithm

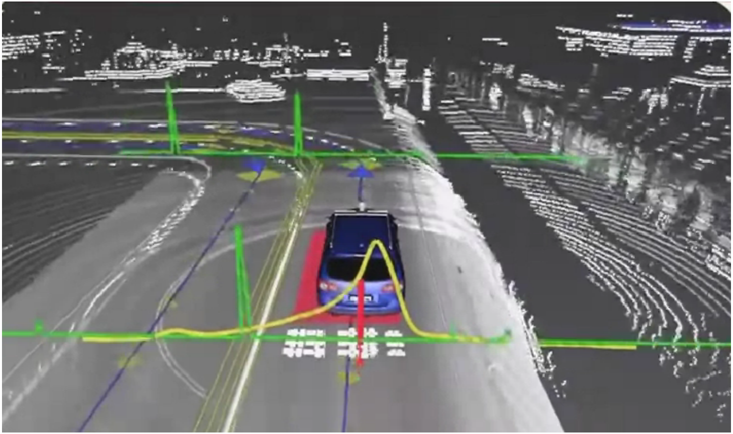
- S: Particle set $\{< x, w >, \ldots\}$,
- U: Control vector (e.g., map),
- Z: Measure vector

$S' = \emptyset,\ \eta = 0$
For i=1…n
  sample $j \sim \{w\}$ w/ replacement
  $x' \sim P(x'|U, S_j)$
  $w' = P(Z|x')$
  $\eta = \eta + w'$
  $S' = S' \cup \{< x', w' >\}$
End
For i=1…n // Normalization step
  $w_i = \frac{1}{\eta} w_i$
End

Particle Filter for Finding Road Boundaries



Particle Filter for Finding Road Boundaries



Particles following the white lane lines so the car knows where it is.

Particles are Like Small Cars

Particle's dynamic state to estimate next state:

$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$$

Particles are Like Small Cars

Particle's dynamic state to estimate next state:

$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \quad \& \quad \begin{pmatrix} v \\ \omega \end{pmatrix}$$

Particles are Like Small Cars

Particle's dynamic state to estimate next state:

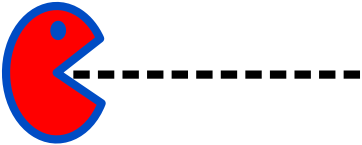
$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \quad \& \quad \begin{pmatrix} v \\ \omega \end{pmatrix} \rightarrow \begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix}$$

Particles are Like Small Cars

Particle's dynamic state to estimate next state:

$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \quad \& \quad \begin{pmatrix} v \\ \omega \end{pmatrix} \rightarrow \begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix}$$

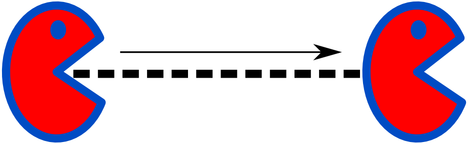
Also add some noise to account for uncertainty:



Particle's dynamic state to estimate next state:

$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \quad \& \quad \begin{pmatrix} v \\ \omega \end{pmatrix} \rightarrow \begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix}$$

Also add some noise to account for uncertainty:

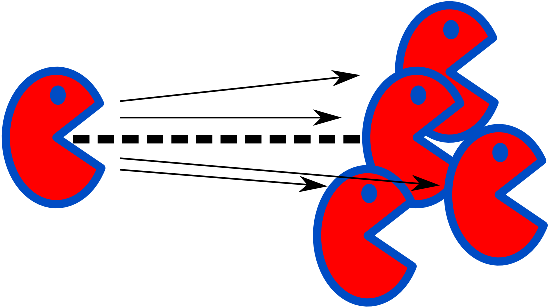


Particles are Like Small Cars

Particle's dynamic state to estimate next state:

$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \quad \& \quad \begin{pmatrix} v \\ \omega \end{pmatrix} \rightarrow \begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix}$$

Also add some noise to account for uncertainty:



Factor In Measurements

Pacman particles try to stay on the road lines:

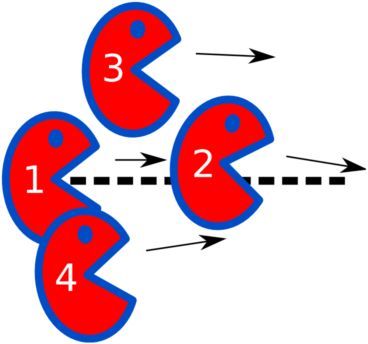
- Measures pattern on ground, z.

Factor In Measurements

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

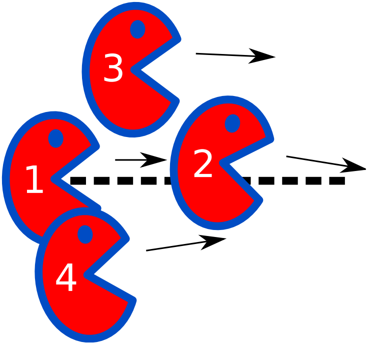
- $P(\text{dashes}|\text{on the line}) = 0.7$

Factor In Measurements

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



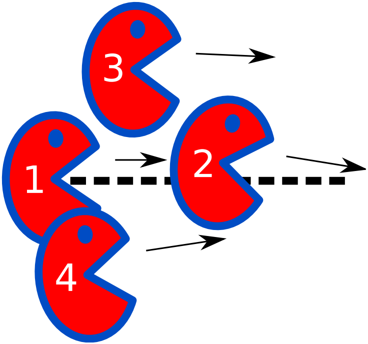
Likelihood based on measurement:

- $P(\text{dashes}|\text{on the line}) = 0.7$
- $P(\text{no dashes}|\text{off the line}) = 0.8$

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

- $P(\text{dashes}|\text{on the line}) = 0.7$
- $P(\text{no dashes}|\text{off the line}) = 0.8$

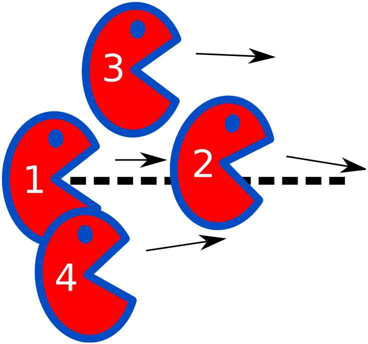
Particle weights, w, for 1 & 4?

Factor In Measurements

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

- $P(\text{dashes}|\text{on the line}) = 0.7$
- $P(\text{no dashes}|\text{off the line}) = 0.8$

Particle weights, w, for 1 & 4?

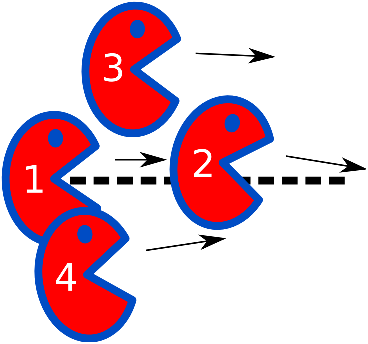
- $w_1 = 0.7$

Factor In Measurements

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

- $P(\text{dashes}|\text{on the line}) = 0.7$
- $P(\text{no dashes}|\text{off the line}) = 0.8$

Particle weights, w, for 1 & 4?

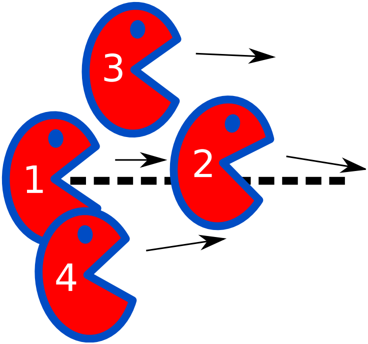
- $w_1 = 0.7$ & $w_4 = 0.2$

Factor In Measurements

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

- $P(\mathrm{dashes} | \mathrm{on\ the\ line}) = 0.7$
- $P(\mathrm{no\ dashes} | \mathrm{off\ the\ line}) = 0.8$

Particle weights, w, for 1 & 4?

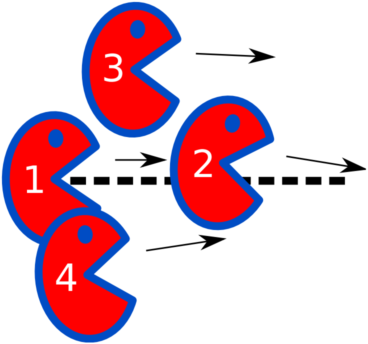
- $w_1 = 0.7$ & $w_4 = 0.2$ **(before normalization)**

Factor In Measurements

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

- $P(\text{dashes}|\text{on the line}) = 0.7$
- $P(\text{no dashes}|\text{off the line}) = 0.8$

Particle weights, w, for 1 & 4?

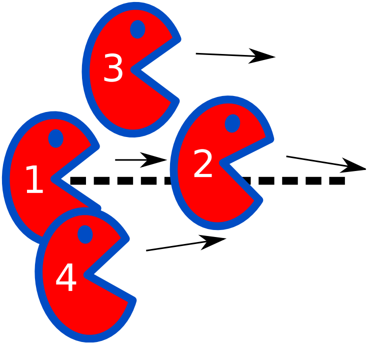
- $w_1 = 0.7$ & $w_4 = 0.2$
  (before normalization)
- $\text{Total} = 0.7 + 0.7 + 0.2 + 0.2 = 1.8$

Factor In Measurements

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

- $P(\text{dashes}|\text{on the line}) = 0.7$
- $P(\text{no dashes}|\text{off the line}) = 0.8$

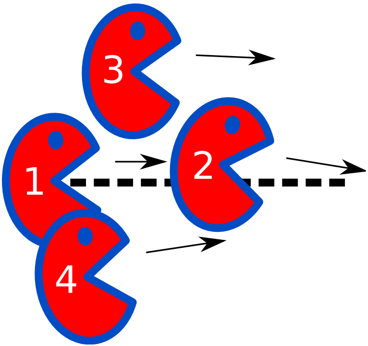
Particle weights, w, for 1 & 4?

- $w_1 = 0.7$ & $w_4 = 0.2$
  (before normalization)
- $\text{Total} = 0.7 + 0.7 + 0.2 + 0.2 = 1.8$
  $w_1 = 7/18$ & $w_4 = 2/18$
  (after normalization)

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

- $P(\text{dashes}|\text{on the line}) = 0.7$
- $P(\text{no dashes}|\text{off the line}) = 0.8$

Particle weights, w, for 1 & 4?

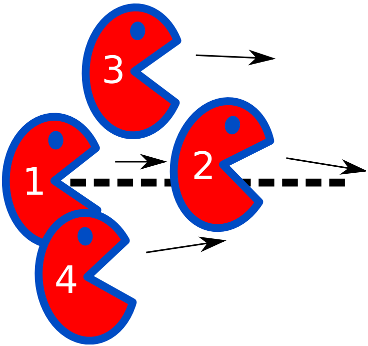
- $w_1 = 0.7$ & $w_4 = 0.2$
  (before normalization)
- $\text{Total} = 0.7 + 0.7 + 0.2 + 0.2 = 1.8$
  $w_1 = 7/18$ & $w_4 = 2/18$
  (after normalization)

Why?

Pacman particles try to stay on the road lines:

- Measures pattern on ground, z.

Question:



Likelihood based on measurement:

- $P(\text{dashes}|\text{on the line}) = 0.7$
- $P(\text{no dashes}|\text{off the line}) = 0.8$

Particle weights, w, for 1 & 4?

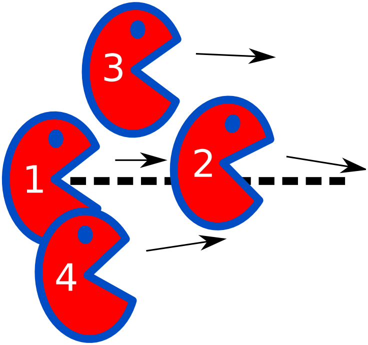
- $w_1 = 0.7$ & $w_4 = 0.2$
  (before normalization)
- $\text{Total} = 0.7 + 0.7 + 0.2 + 0.2 = 1.8$
  $w_1 = 7/18$ & $w_4 = 2/18$
  (after normalization)

Why? So that weights are **probabilities** that add up to 1.

Then, Resample New Particles

Pick new 4 particles based on probs:

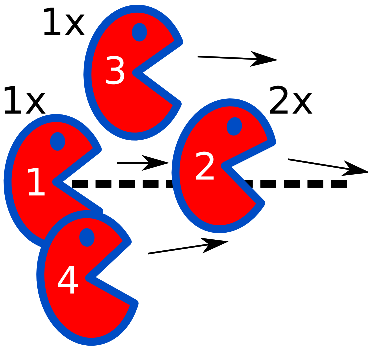
- $w_1 = w_2 = 7/18$ and $w_3 = w_4 = 2/18$



Then, Resample New Particles

Pick new 4 particles based on probs:

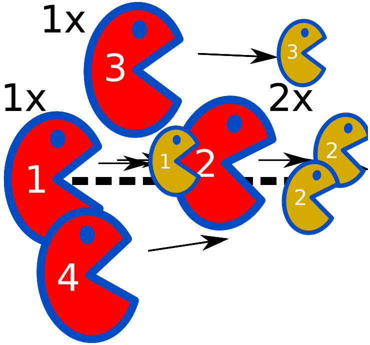
- $w_1 = w_2 = 7/18$ and $w_3 = w_4 = 2/18$



Then, Resample New Particles

Pick new 4 particles based on probs:

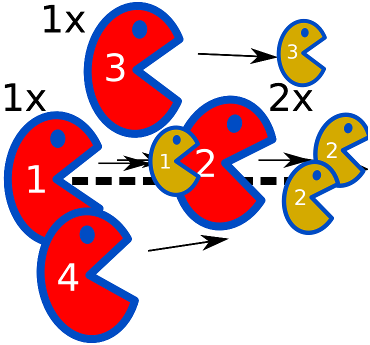
- $w_1 = w_2 = 7/18$ and $w_3 = w_4 = 2/18$



Then, Resample New Particles

Pick new 4 particles based on probs:

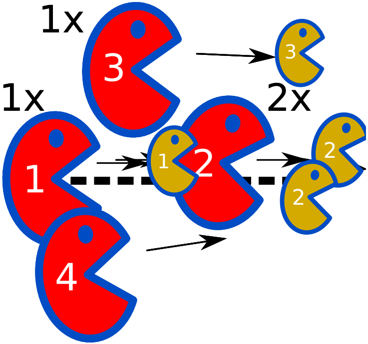
- $w_1 = w_2 = 7/18$ and $w_3 = w_4 = 2/18$



Then, Resample New Particles

Pick new 4 particles based on probs:

- $w_1 = w_2 = 7/18$ and $w_3 = w_4 = 2/18$

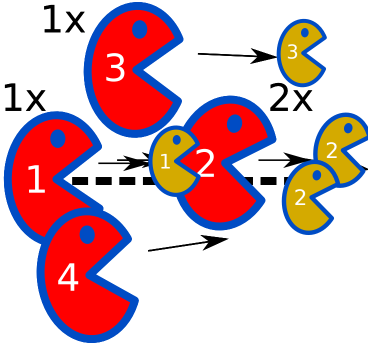


Wash, rinse and repeat!

Then, Resample New Particles

Pick new 4 particles based on probs:

- $w_1 = w_2 = 7/18$ and $w_3 = w_4 = 2/18$



Wash, rinse and repeat!

Repeat the process:

1. Estimate
2. Measure
3. Resample

Particle Filters Solution to Every Problem?

So Particle Filters allow us to

Particle Filters Solution to Every Problem?

So Particle Filters allow us to stay within the road.
Do they also help the robot reach the **goal**?

Particle Filters Solution to Every Problem?

So Particle Filters allow us to stay within the road.
Do they also help the robot reach the **goal**? **No.**

Particle Filters Solution to Every Problem?

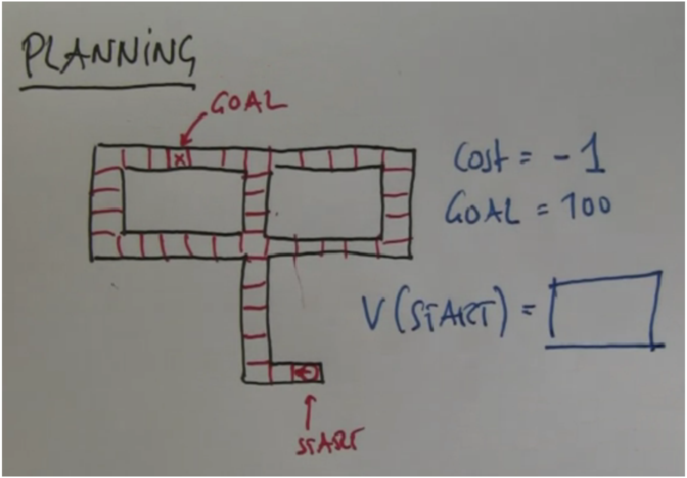
So Particle Filters allow us to stay within the road.
Do they also help the robot reach the **goal**? **No.**

- Still need **planning**. Let's start with **MDP**s.

Particle Filters Solution to Every Problem?

So Particle Filters allow us to stay within the road.
Do they also help the robot reach the **goal**? **No.**

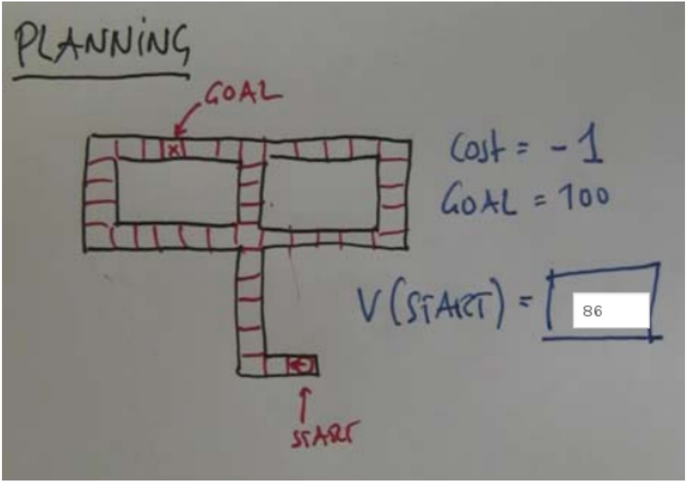
- Still need **planning**. Let's start with **MDP**s.



Particle Filters Solution to Every Problem?

So Particle Filters allow us to stay within the road.
Do they also help the robot reach the **goal**? **No.**

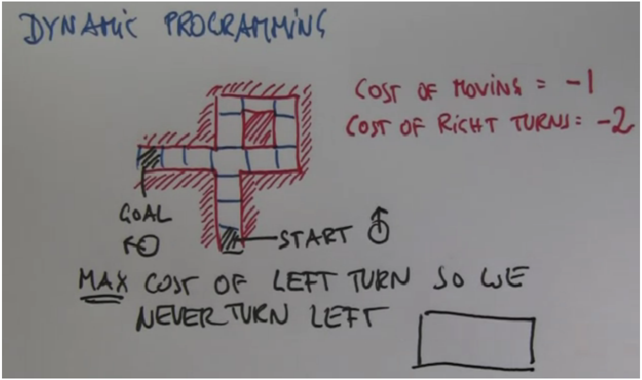
- Still need **planning**. Let's start with **MDP**s.



Planning: MDP in City Navigation

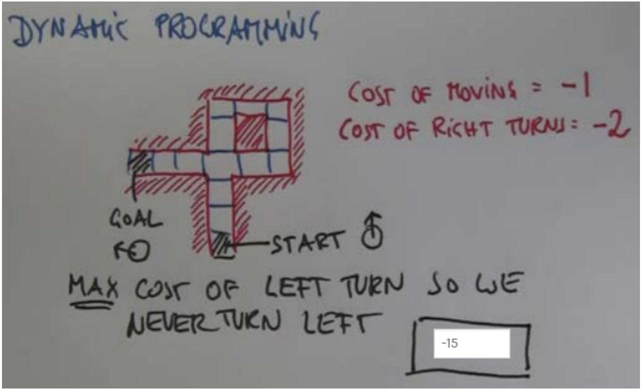
- Includes heading direction, one-way circle

MDP with Heading and Turns



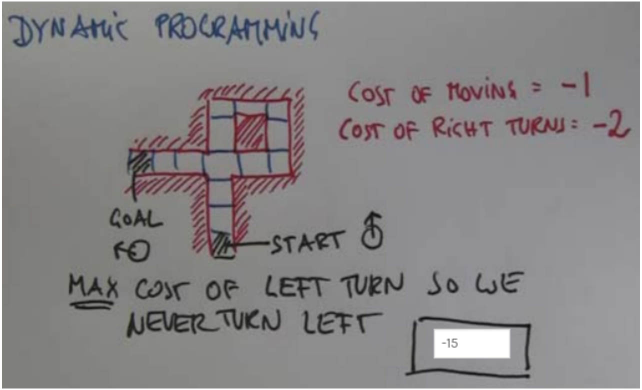
(Count turns only at the intersection.)

MDP with Heading and Turns



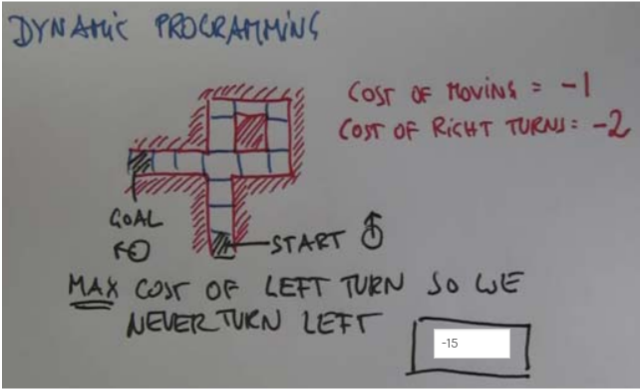
(Count turns only at the intersection.)

MDP with Heading and Turns



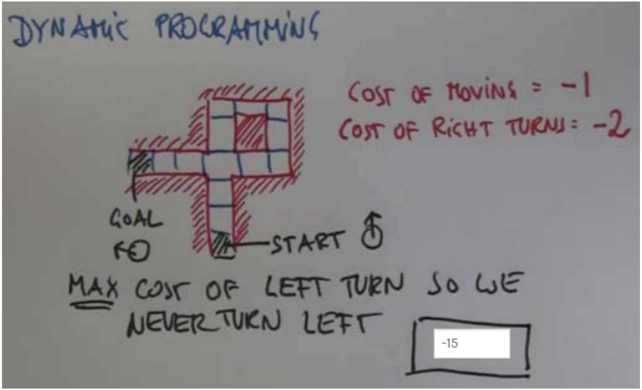
(Count turns only at the intersection.)

MDP with Heading and Turns



(Count turns only at the intersection.)

- Direct movement distance to goal: 6
- With the right loop:
  $6 + 8$ (additional steps) $+6$ (3 right turns) $= 6 + 14$

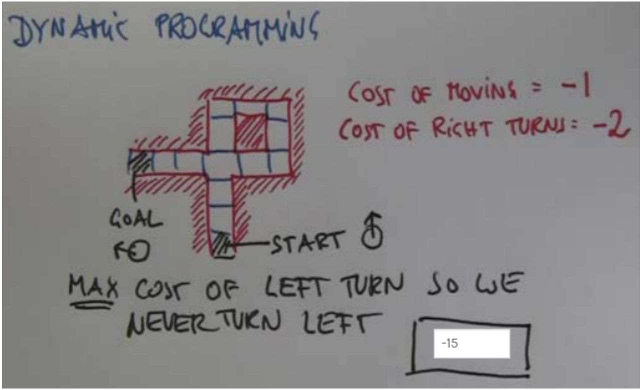


(Count turns only at the intersection.)

- Direct movement distance to goal: 6
- With the right loop:
  $6 + 8$ (additional steps) $+6$ (3 right turns) $= 6 + 14$

How many values per grid?

MDP with Heading and Turns

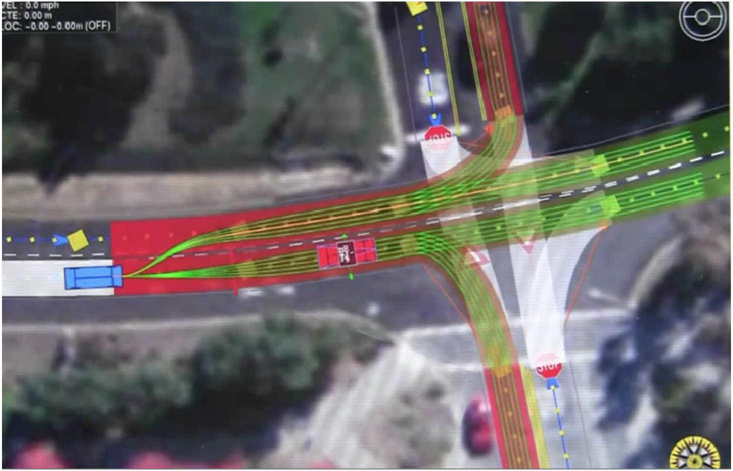


(Count turns only at the intersection.)

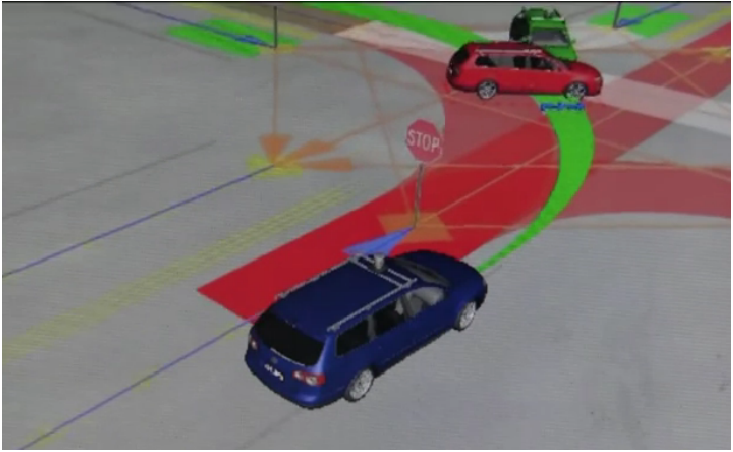
- Direct movement distance to goal: 6
- With the right loop:
  $6 + 8$ (additional steps) $+6$ (3 right turns) $= 6 + 14$

How many values per grid? More than one.

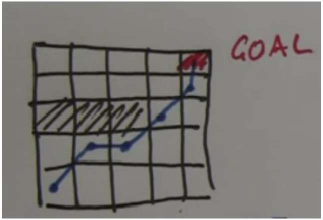
MDP in Action for Car Driving: Lane Change



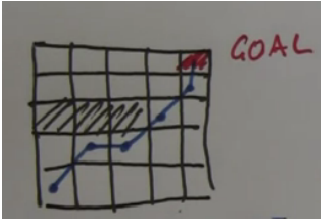
MDP in Action for Car Driving: Roadblock



A* for Robot Planning

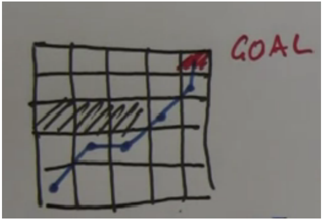


A* for Robot Planning



Any problems with this?

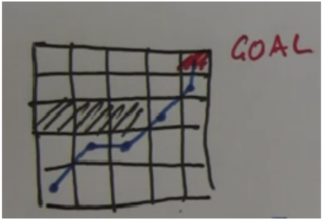
A* for Robot Planning



Any problems with this?

- A* is dicrete: sharp turns for car

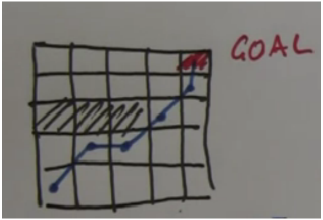
A* for Robot Planning



Any problems with this?

- A* is dicrete: sharp turns for car
- Robot car: needs continuous

A* for Robot Planning

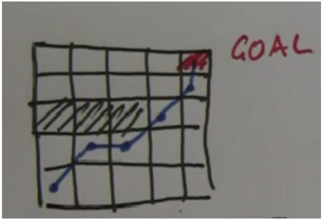


Any problems with this?

- A* is dicrete: sharp turns for car
- Robot car: needs continuous

Hybrid of discrete & continuous:

A* for Robot Planning

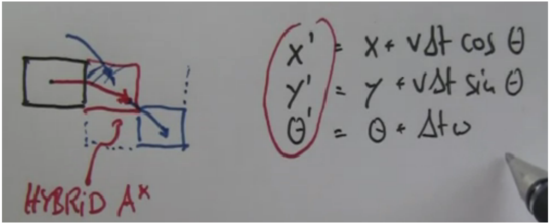


Any problems with this?

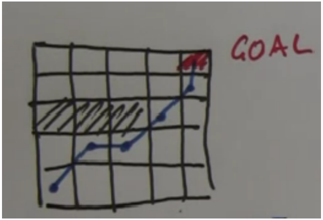
- A* is dicrete: sharp turns for car
- Robot car: needs continuous

Hybrid of discrete & continuous:

- Save location & heading within grid block.



A* for Robot Planning

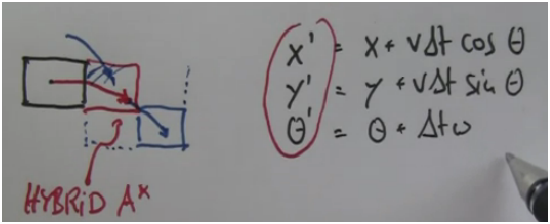


Any problems with this?

- A* is dicrete: sharp turns for car
- Robot car: needs continuous

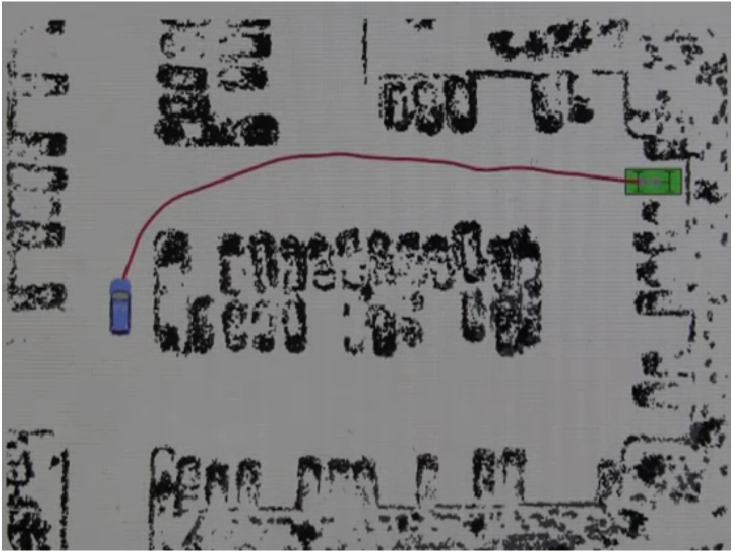
Hybrid of discrete & continuous:

- Save location & heading within grid block.



Also: prune loops, which makes it incomplete.

Hybrid A*: Driving in Parking Lot



Summary: Robotic Navigation

What we did so far:

Summary: Robotic Navigation

What we did so far:

Perception: Particle filter

Planning: MDP and A*

Summary: Robotic Navigation

What we did so far:

Perception: Particle filter

Planning: MDP and A*

Hope we gave you a flavor, more algorithms are also used: **Reinforcement learning**.