# Probabilistic Models

Sanjiban Choudhury

TAs: Matthew Rockett, Gilwoo Lee, Matt Schmittle

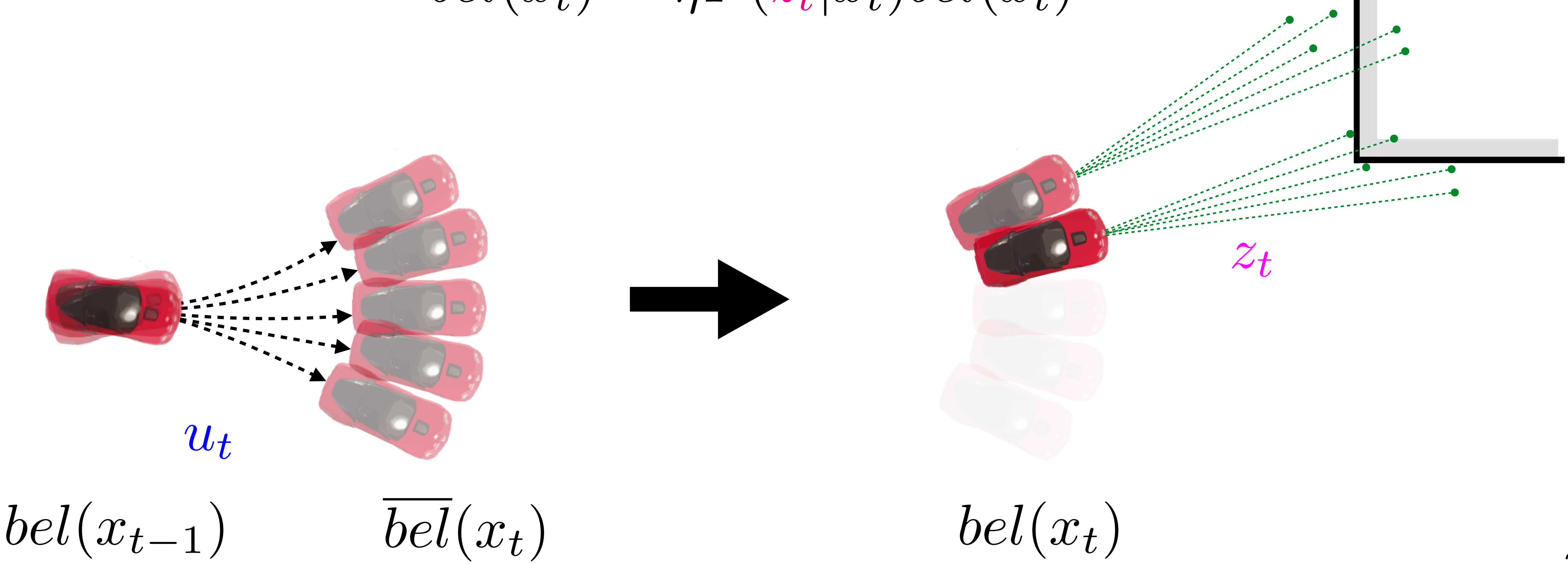
# Bayes filter in a nutshell

Step 1: Prediction - push belief through dynamics given action

$$\overline{bel}(x_t) = \int P(x_t|u_t, x_{t-1})bel(x_{t-1})dx_{t-1}$$

Step 2: Correction - apply Bayes rule given measurement

$$bel(x_t) = \eta P(z_t|x_t)\overline{bel}(x_t)$$



# Today's objective

1. Briefly discuss different paradigms of Bayes filtering

2. Probabilistic motion model

3. (If time remaining) Having fun with 1-D Kalman filter!

# Bayes filter is a powerful tool

Localization Mapping SLAM POMDP

# Casting different tasks as Bayes filtering

| Tasks | State | Action | Measurement |
| --- | --- | --- | --- |
| Localization | | | |
| Mapping | | | |
| SLAM | | | |
| Pursuit-Evasion | | | |

# Casting different tasks as Bayes filtering

| Tasks | State | Action | Measurement |
| --- | --- | --- | --- |
| Localization | Pose of the robot | Motor commands | GPS / Laser scans / RGB(D) images |
| Mapping | | | |
| SLAM | | | |
| Pursuit-Evasion | | | |

# Casting different tasks as Bayes filtering

| Tasks | State | Action | Measurement |
|---|---|---|---|
| Localization | Pose of the robot | Motor commands | GPS / Laser scans / RGB(D) images |
| Mapping | Objects in the world | NOP | Exact pose + Laser scans / RGB(D) images |
| SLAM | | | |
| Pursuit-Evasion | | | |

# Casting different tasks as Bayes filtering

| Tasks | State | Action | Measurement |
|---|---|---|---|
| Localization | Pose of the robot | Motor commands | GPS / Laser scans / RGB(D) images |
| Mapping | Objects in the world | NOP | Exact pose + Laser scans / RGB(D) images |
| SLAM | Pose of robot / Objects in the world | Motor commands (for pose), NOP (for objects) | GPS / Laser scans / RGB(D) images |
| Pursuit-Evasion | | | |

# Casting different tasks as Bayes filtering

| Tasks | State | Action | Measurement |
|---|---|---|---|
| Localization | Pose of the robot | Motor commands | GPS / Laser scans / RGB(D) images |
| Mapping | Objects in the world | NOP | Exact pose + Laser scans / RGB(D) images |
| SLAM | Pose of robot / Objects in the world | Motor commands (for pose), NOP (for objects) | GPS / Laser scans / RGB(D) images |
| Pursuit-Evasion | Pose of target | Guess where target can move | Camera image |

# Assembling Bayes filter

# Assembling Bayes filter

Tasks

Localization

P(pose | data)

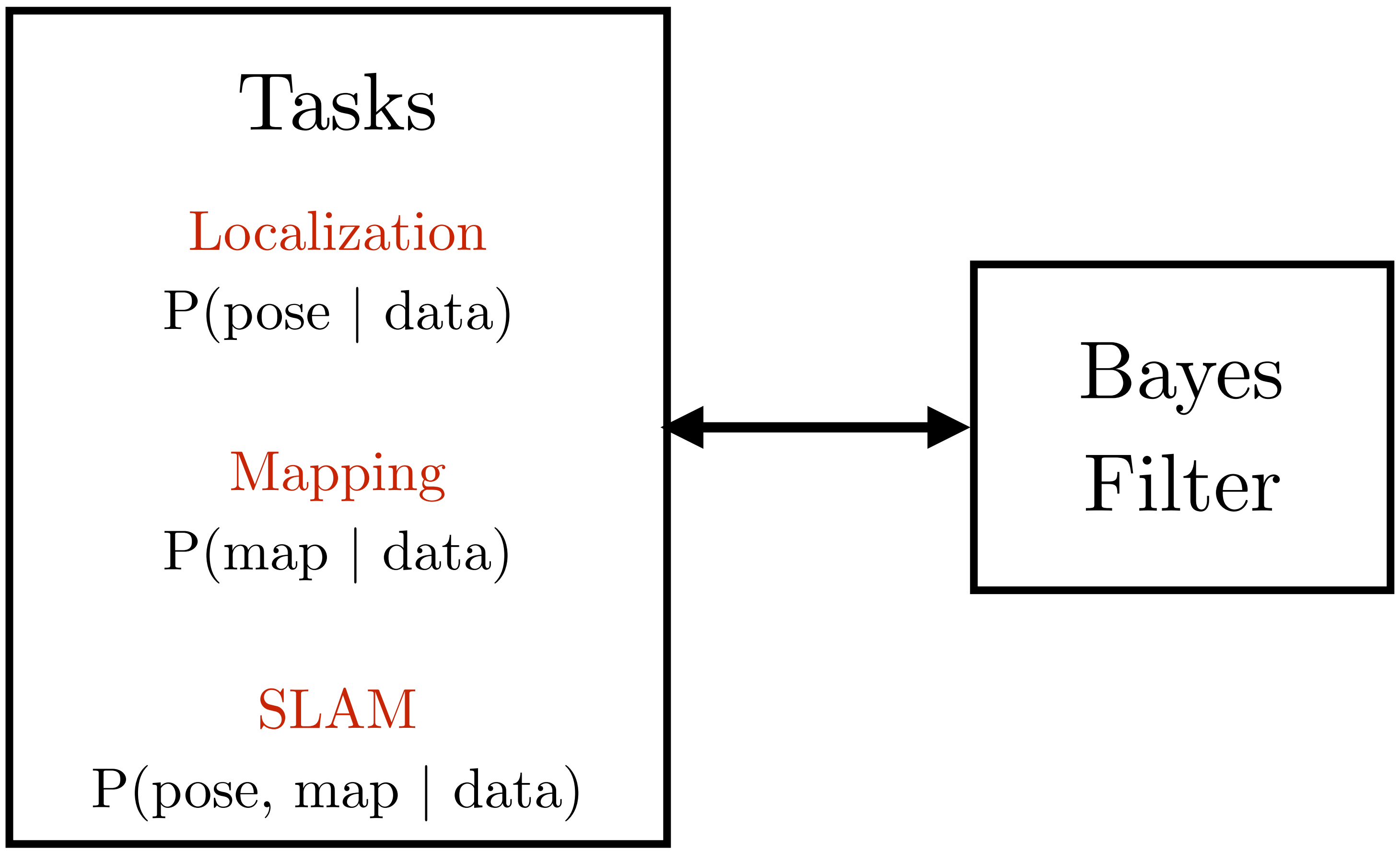
Mapping

P(map | data)

SLAM

P(pose, map | data)

# Assembling Bayes filter



# Assembling Bayes filter

Tasks

Localization

P(pose | data)

Mapping

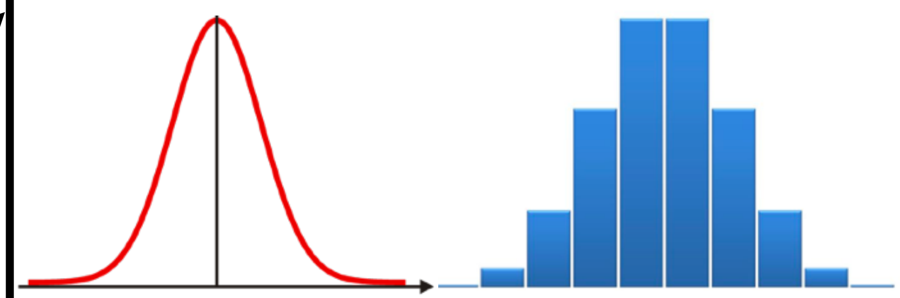
P(map | data)

SLAM

P(pose, map | data)

Bayes
Filter

Belief Representations



# Assembling Bayes filter

**Tasks**

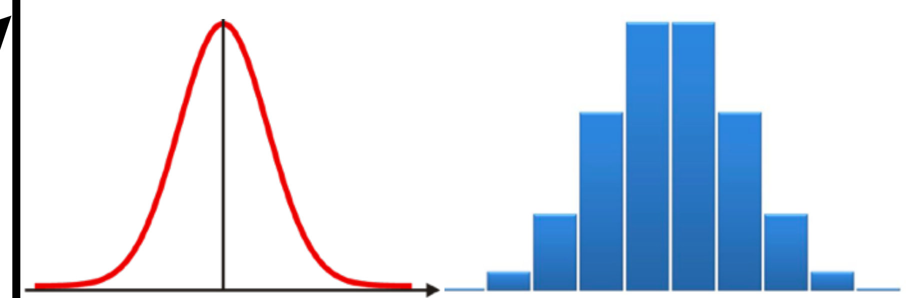
Localization
P(pose | data)

Mapping
P(map | data)

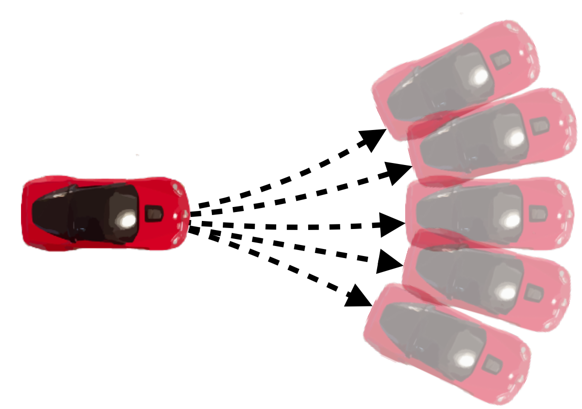
SLAM
P(pose, map | data)

**Bayes Filter**

**Belief Representations**



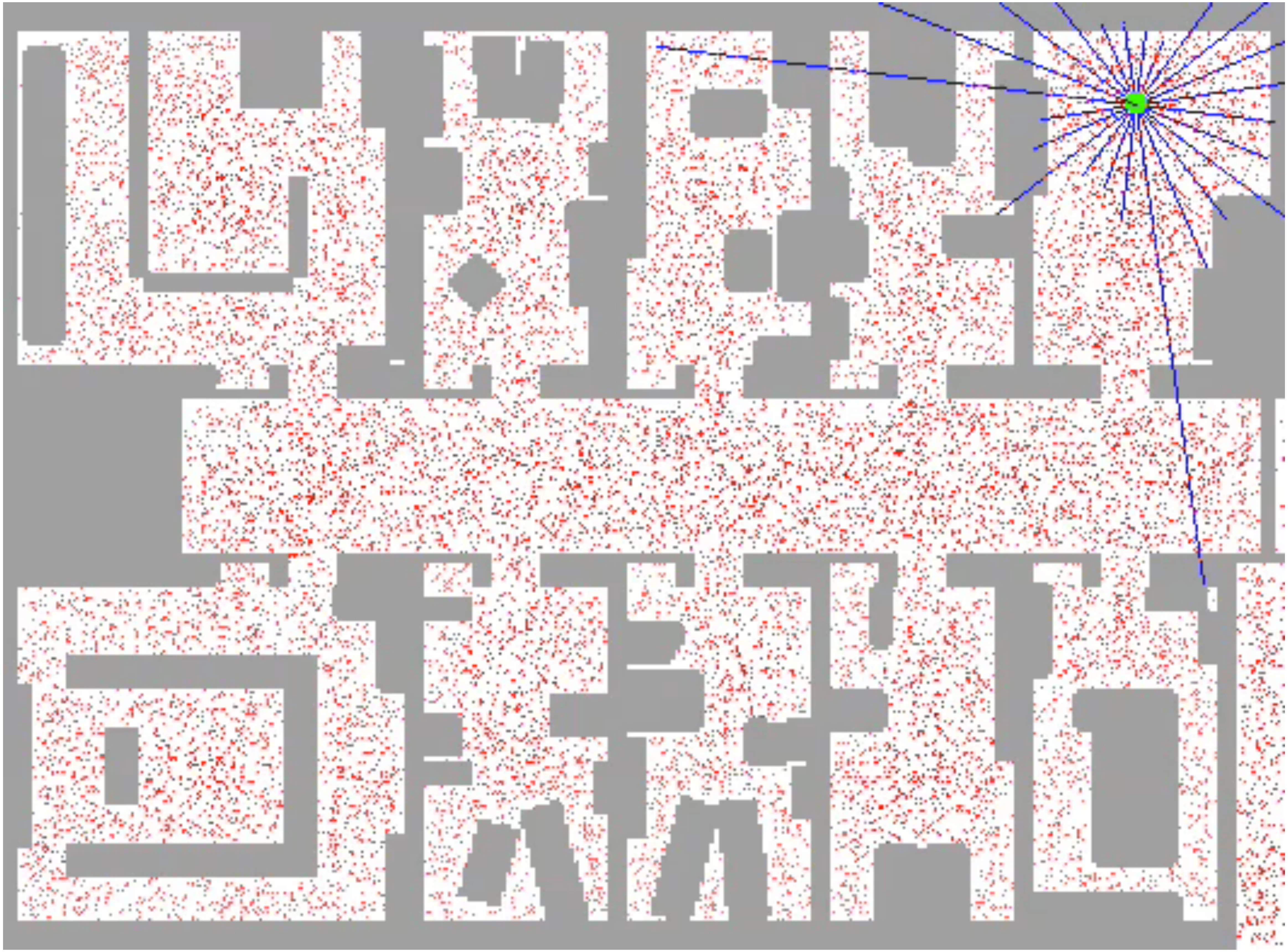
**Probabilistic Models**



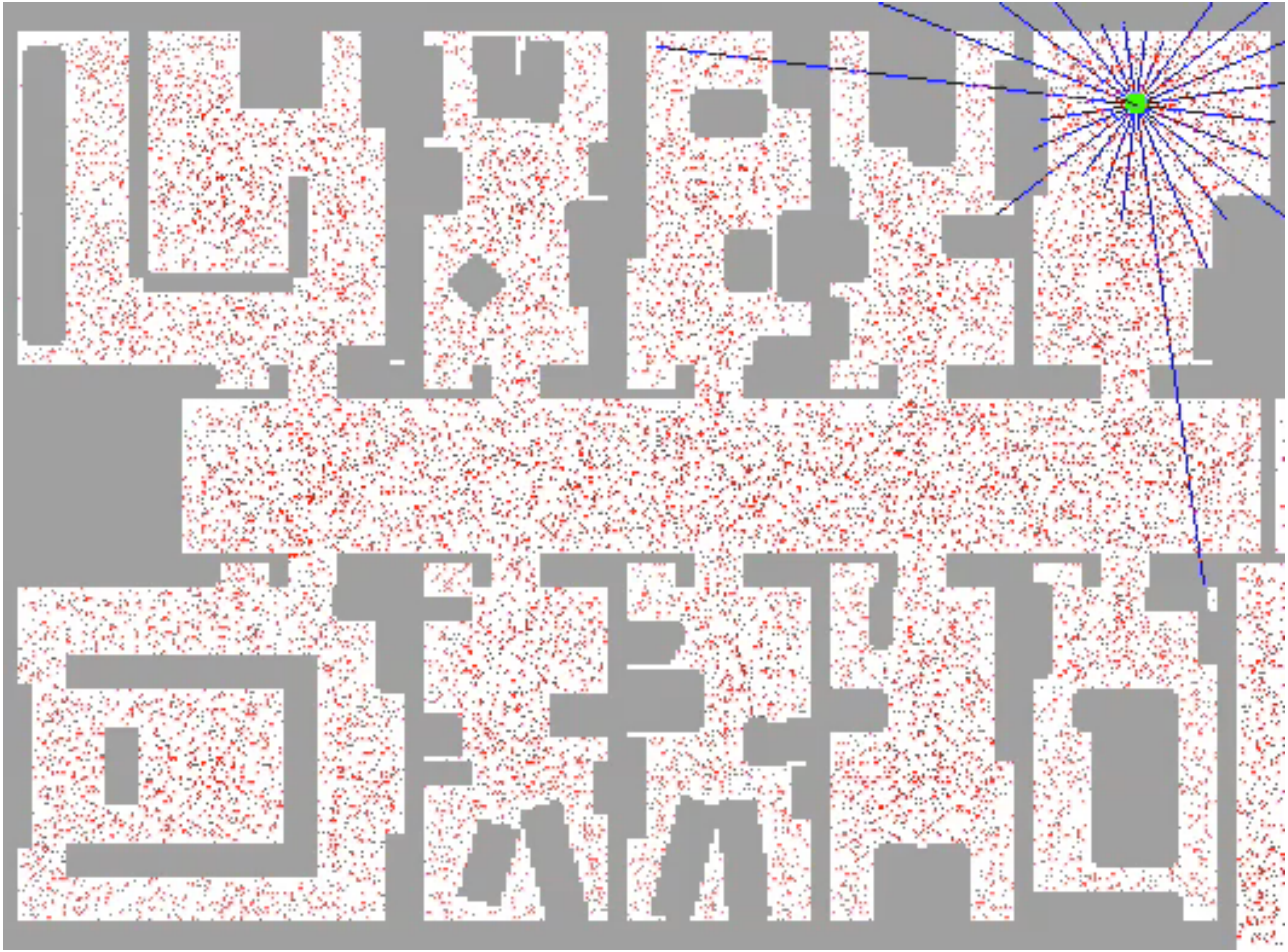
# Tasks that we will cover

| Tasks | Belief Representation | Probabilistic Models |
|---|---|---|
| Localization<br>P(pose \| data)<br>(Week 3) | Gaussian / Particles | Motion model<br>Measurement model |
| Mapping<br>P(map \| data)<br>(Week 4) | Discrete (binary) | Inverse measurement model |
| SLAM<br>P(pose, map \| data)<br>(Week 4) | Particles+Gaussian<br>(pose, landmarks) | Motion model,<br>measurement model,<br>correspondence model |

# What is localization?



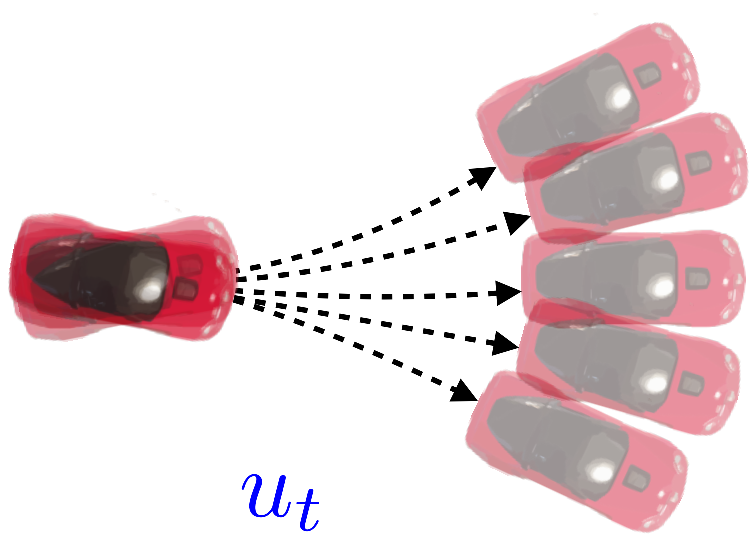
# What is localization?



# Probabilistic models in localization

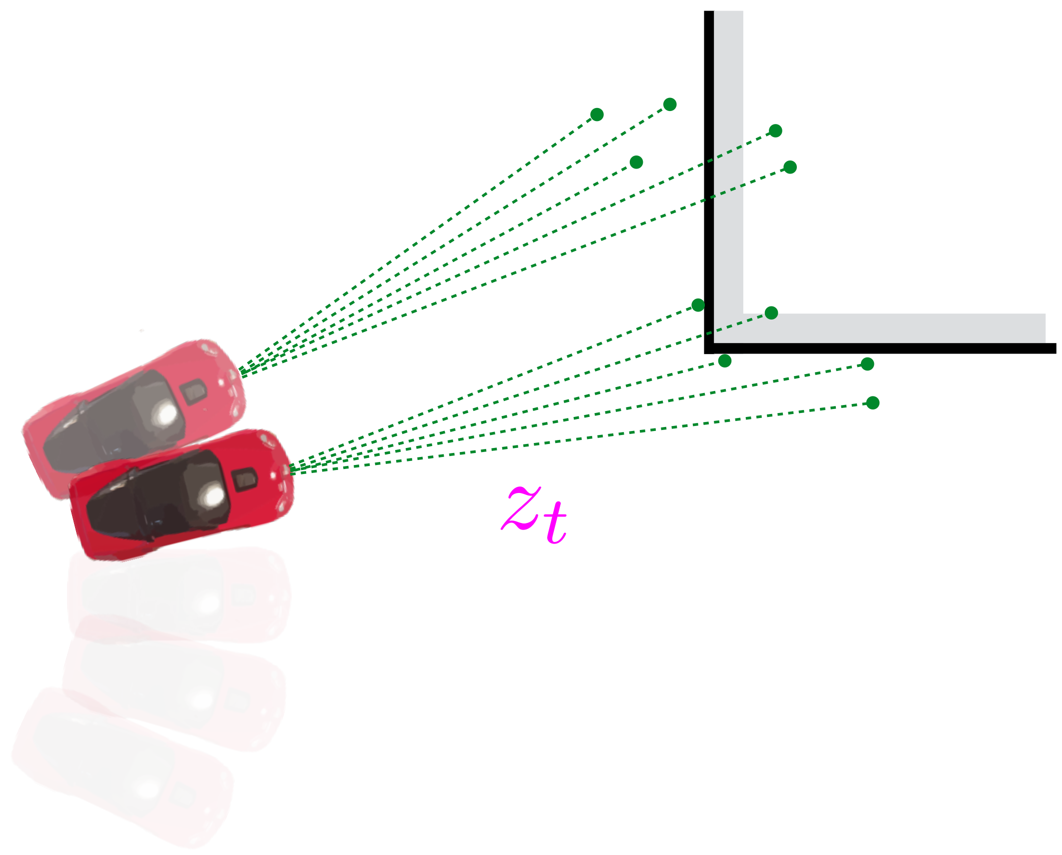
Motion model

$$P(x_t | u_t, x_{t-1})$$




Measurement model

$$P(z_t | x_t)$$



# How do we think about models?

# Three questions you should ask

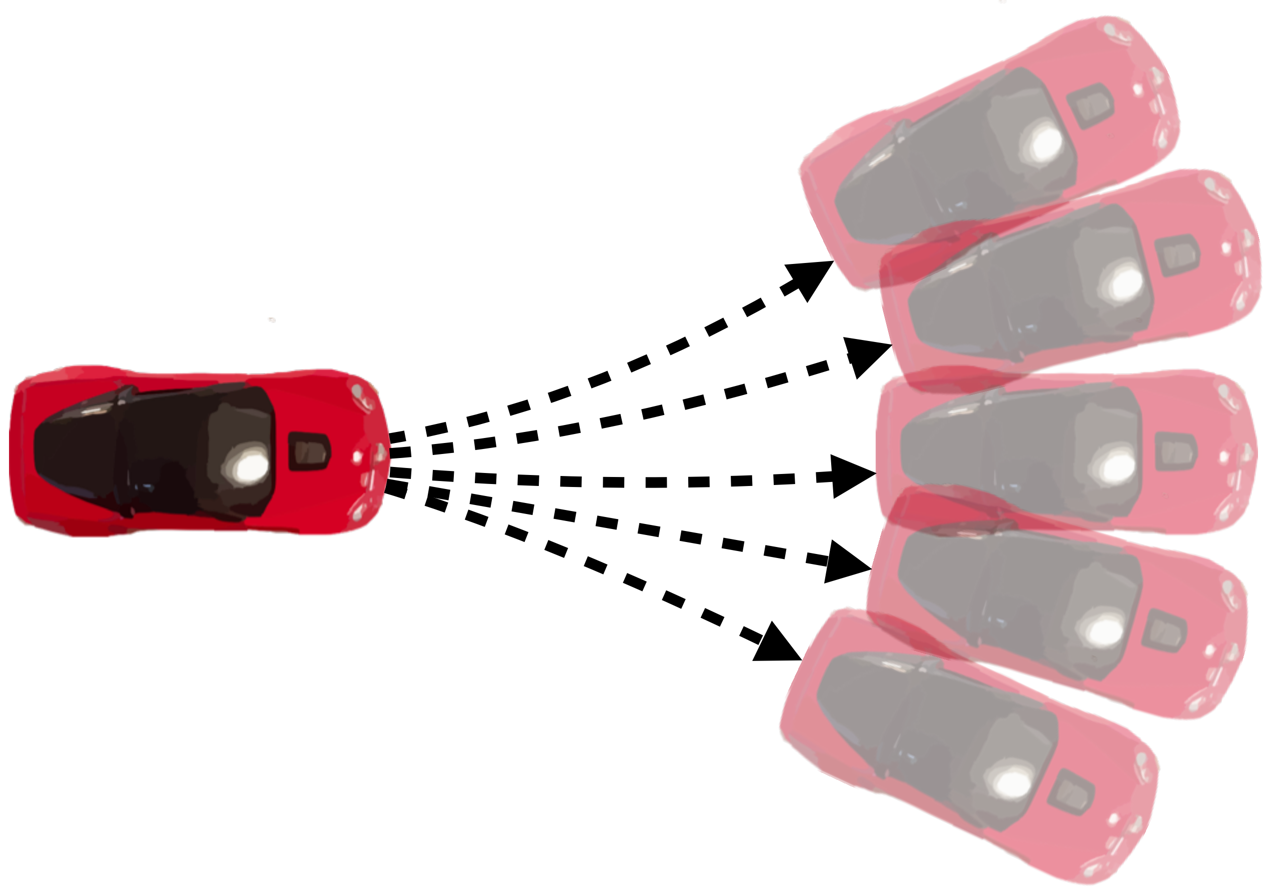
1. Why is the model probabilistic?

2. What defines a good model?

3. What model should I use for my robot?

# Motion Model

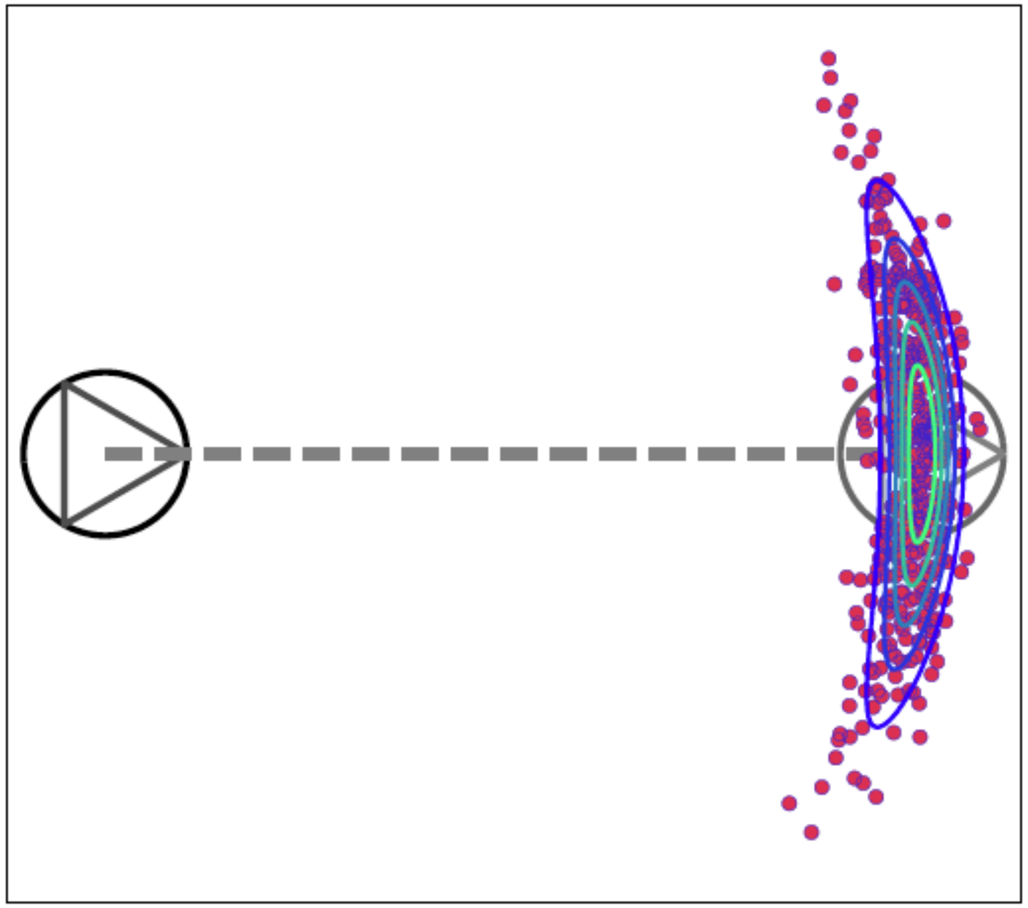
$$P(x_t | u_t, x_{t-1})$$



# Spectrum of motion models

(Redbull simulator)


vs



Highest fidelity models of everything

Simple model with lots of noise

# Three questions you should ask

1. Why is the model probabilistic?

2. What defines a good model?

3. What model should I use for my robot?

# Why is the motion model probabilistic?

If we know how to write out equations of motion,
we should be able to exactly predict where a body ends up, right?

# Why is the motion model probabilistic?

If we know how to write out equations of motion,
we should be able to exactly predict where a body ends up, right?

What are the sources of ~~noise~~ stochasticity?

# Why is the motion model probabilistic?

If we know how to write out equations of motion,
we should be able to exactly predict where a body ends up, right?

What are the sources of ~~noise~~ stochasticity?

| **Category** | **Example** |
|---|---|

# Why is the motion model probabilistic?

If we know how to write out equations of motion,
we should be able to exactly predict where a body ends up, right?

What are the sources of ~~noise~~ stochasticity?

| **Category** | **Example** |
| --- | --- |
| Control signal error | Voltage discretization, communication lag |

# Why is the motion model probabilistic?

If we know how to write out equations of motion,
we should be able to exactly predict where a body ends up, right?

What are the sources of ~~noise~~ stochasticity?

| **Category** | **Example** |
|---|---|
| Control signal error | Voltage discretization, communication lag |
| Unknown physics parameters | Friction of carpet, tire pressure |

# Why is the motion model probabilistic?

If we know how to write out equations of motion,
we should be able to exactly predict where a body ends up, right?

What are the sources of ~~noise~~ stochasticity?

| **Category** | **Example** |
|---|---|
| Control signal error | Voltage discretization, communication lag |
| Unknown physics parameters | Friction of carpet, tire pressure |
| Incorrect physics | Ignoring tire deformation, ignoring wheel slippage |

# Three questions you should ask

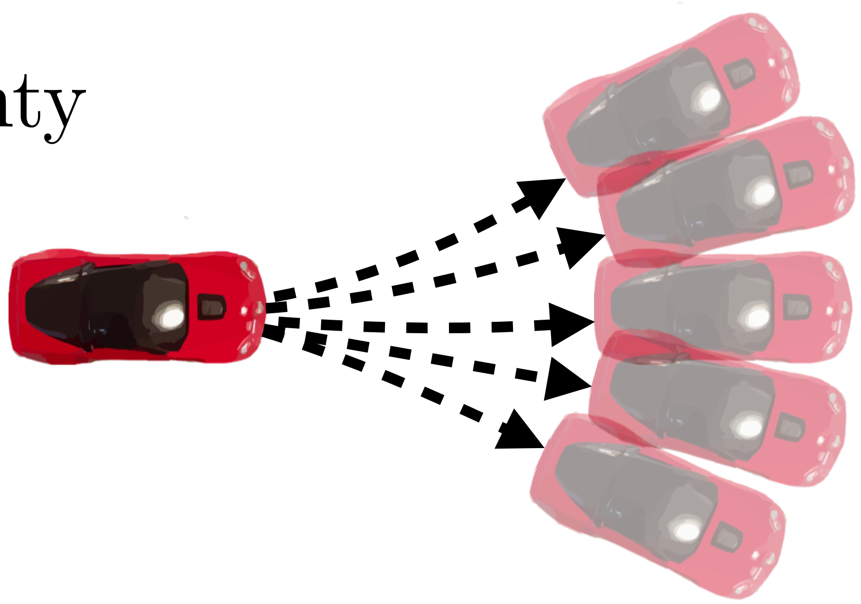
1. Why is the model probabilistic?

2. What defines a good model?

3. What model should I use for my robot?

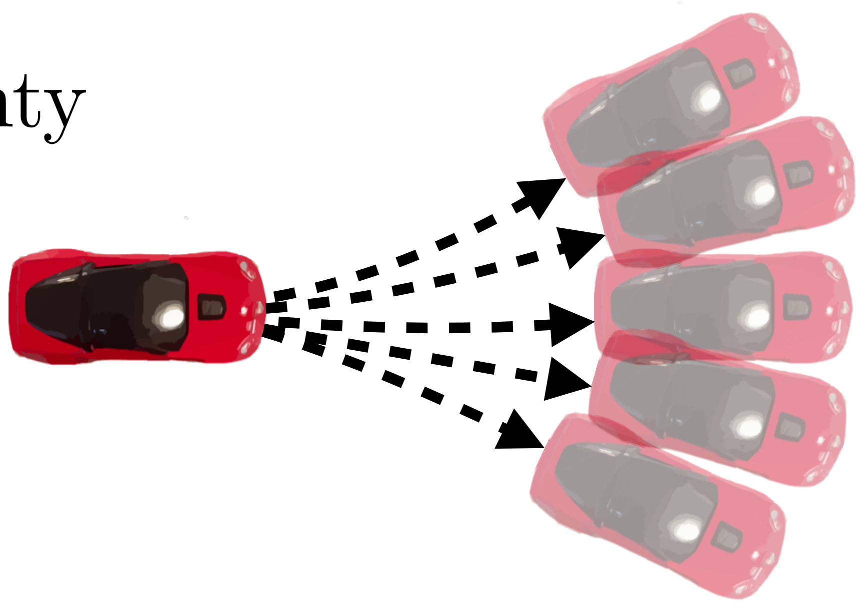
# What is the practical goal of modeling?

In theory - try to accurately model uncertainty



# What is the practical goal of modeling?

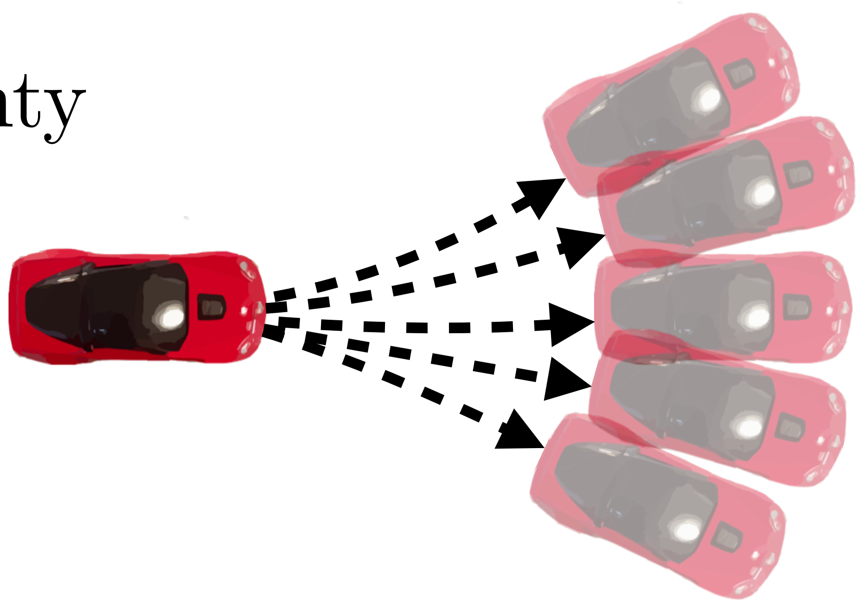
In theory - try to accurately model uncertainty



In practice - do we really need this?

# What is the practical goal of modeling?

In theory - try to accurately model uncertainty



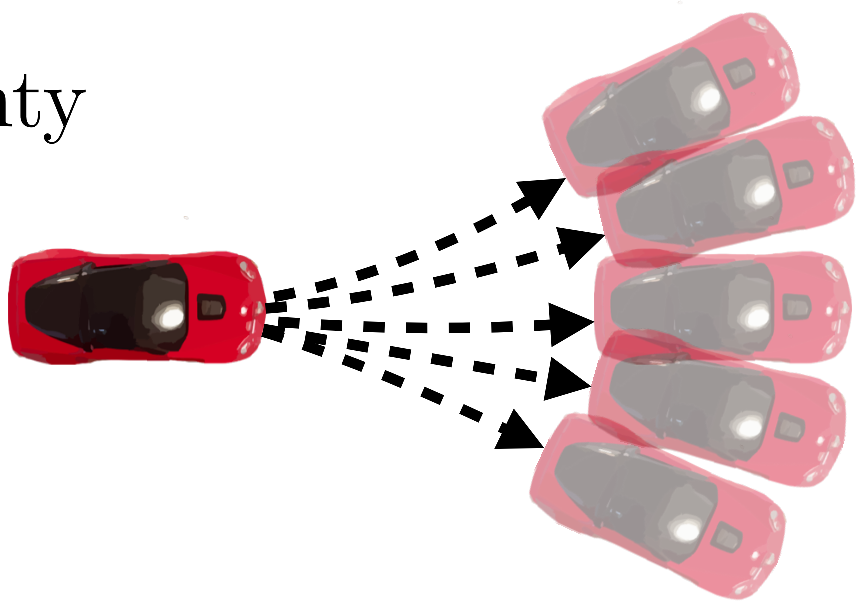
In practice - do we really need this?

1. We need something that is computationally cheap

(Bayes filter will sample repeatedly from this)

# What is the practical goal of modeling?

In theory - try to accurately model uncertainty



In practice - do we really need this?

1. We need something that is computationally cheap

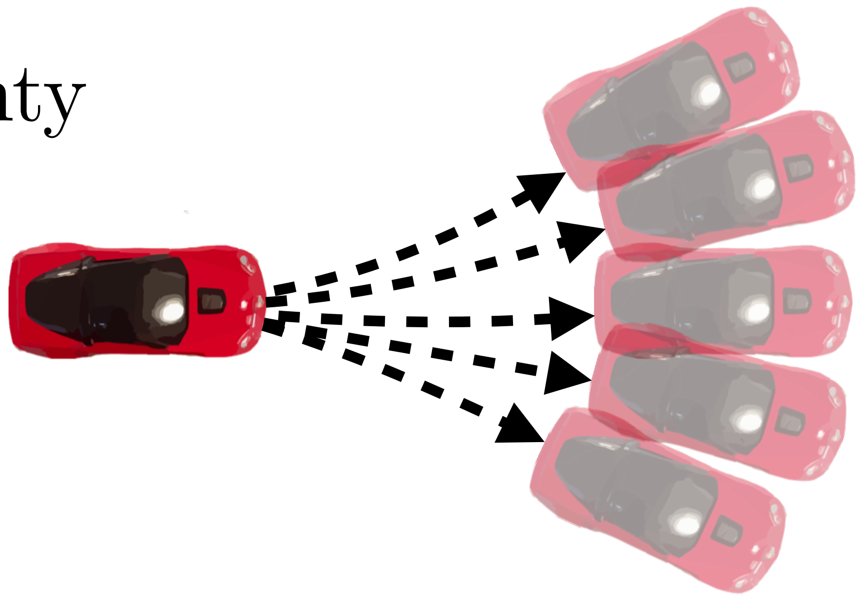
   (Bayes filter will sample repeatedly from this)

2. We need just enough stochasticity to explain any measurements we may see

   (bayes filter will use measurements to hone in on the right state)

# What is the practical goal of modeling?

In theory - try to accurately model uncertainty



In practice - do we really need this?

1. We need something that is computationally cheap

(Bayes filter will sample repeatedly from this)

2. We need just enough stochasticity to explain any measurements we may see

(bayes filter will use measurements to hone in on the right state)

3. We need a model that can deal with unknown unknown

(No matter what the model, we need to overestimate uncertainty)

Key Idea:

Simple model + Stochasticity

# Three questions you should ask

1. Why is the model probabilistic?

2. What defines a good model?

3. What model should I use for my robot?

# A simple model: Kinematic car

# A simple model: Kinematic car

Kinematic model governs how wheel speeds map to robot velocities

# A simple model: Kinematic car

Kinematic model governs how wheel speeds map to robot velocities

Dynamic model governs how wheel torques map to robot accelerations

# A simple model: Kinematic car

Kinematic model governs how wheel speeds map to robot velocities

Dynamic model governs how wheel torques map to robot accelerations

We will ignore the dynamics and focus on the kinematics
(assume we can set the speed directly)

# A simple model: Kinematic car

Kinematic model governs how wheel speeds map to robot velocities

Dynamic model governs how wheel torques map to robot accelerations

We will ignore the dynamics and focus on the kinematics
(assume we can set the speed directly)

Assume wheels rolls on hard, flat, horizontal ground without slipping

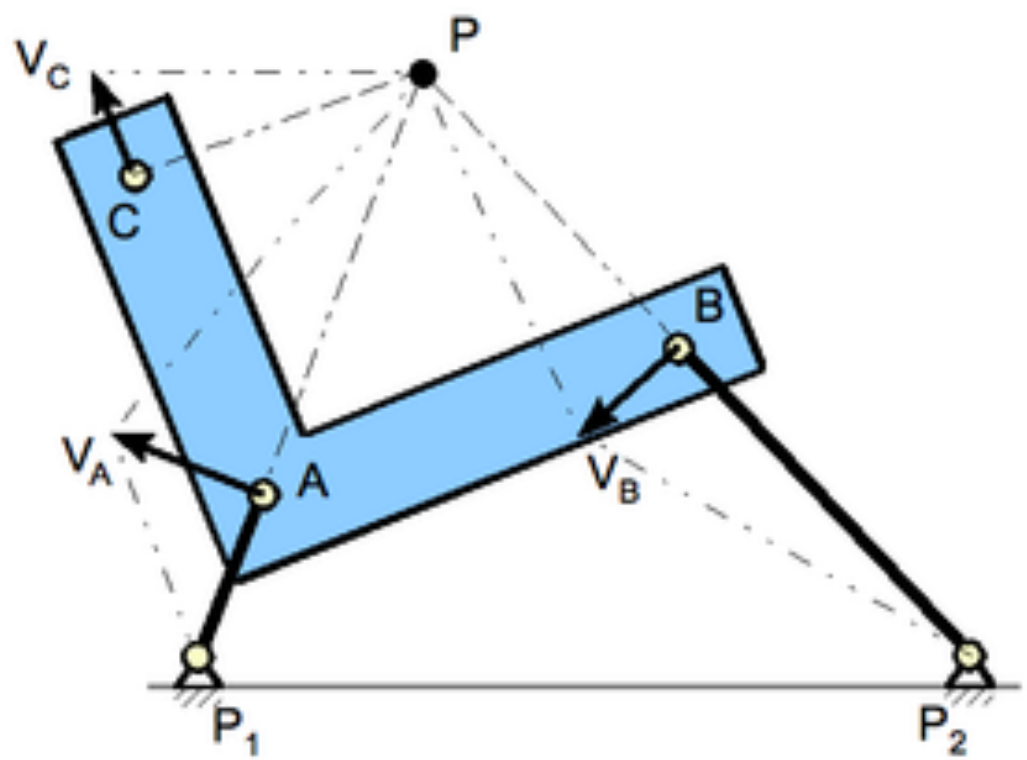
# Motion model

$$P(x_t | u_t, x_{t-1})$$

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \begin{matrix} \text{(x-coord)} \\ \text{(y-coord)} \\ \text{(heading)} \end{matrix}$$

$$\begin{bmatrix} V \\ \delta \end{bmatrix} \begin{matrix} \text{(speed)} \\ \text{(steering angle)} \end{matrix}$$

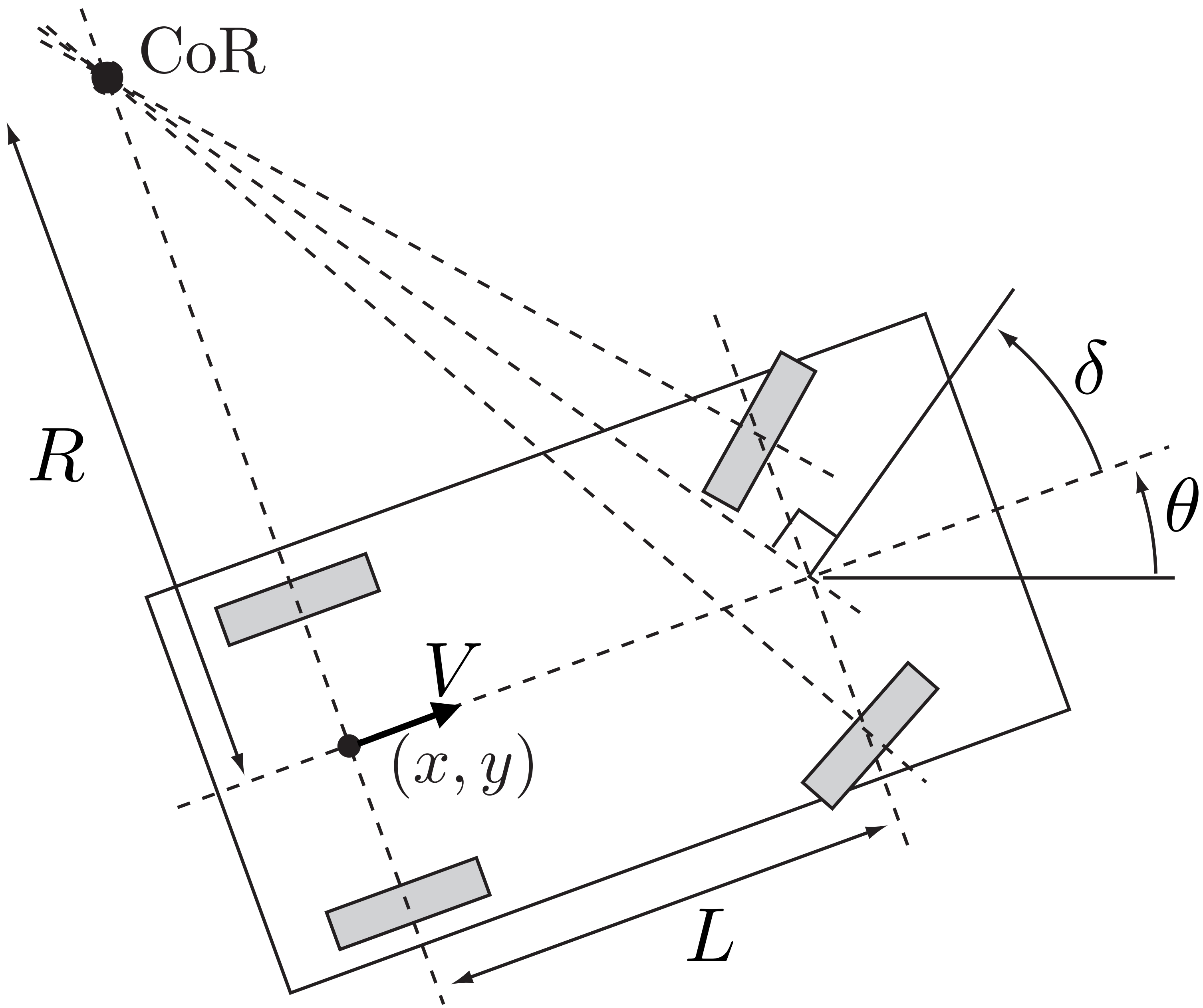
# Instant centre of rotation (CoR)



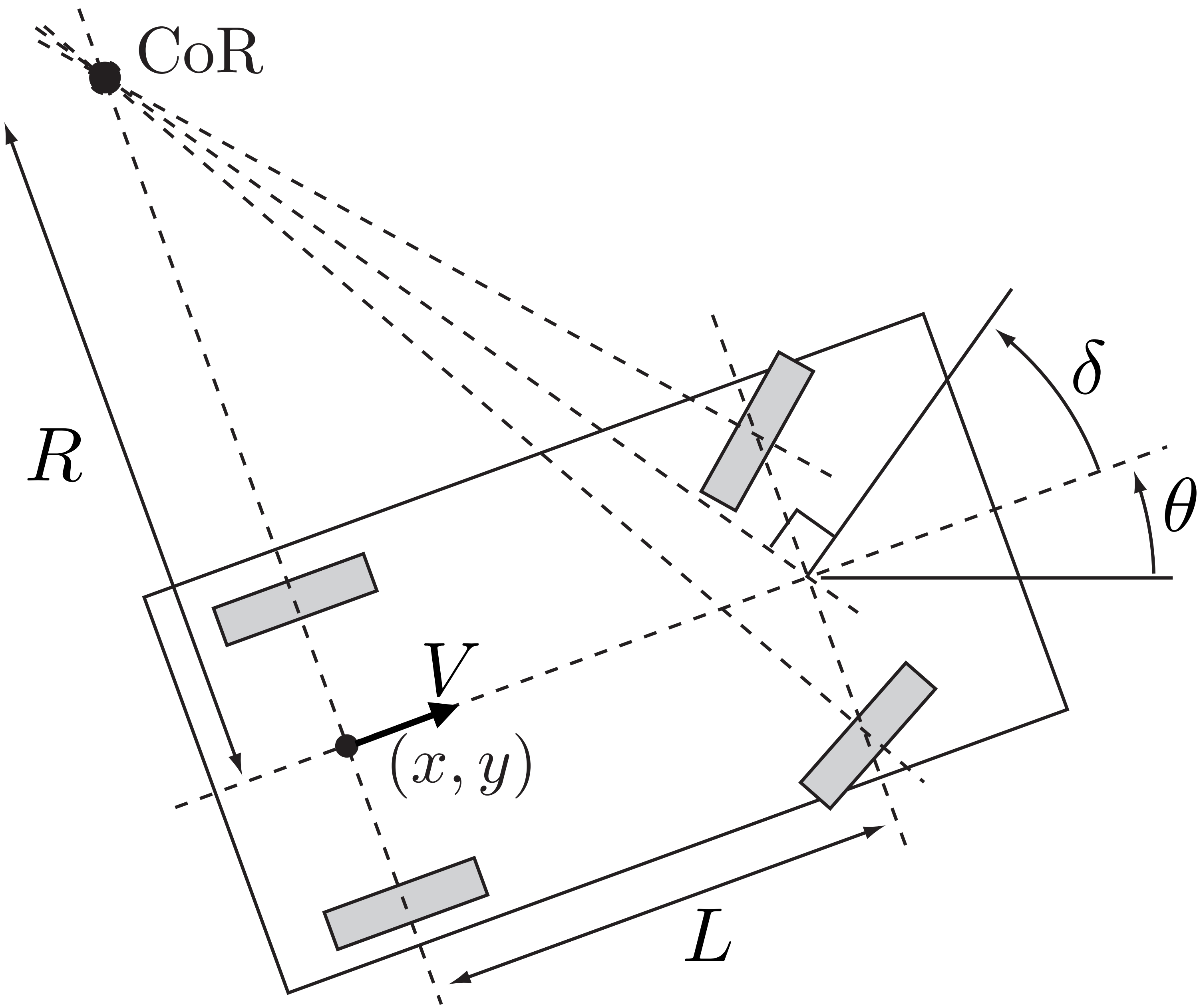

A rigid body undergoing rotation and translation can be viewed as pure rotation about a instant centre of rotation.

https://en.wikipedia.org/wiki/Instant_centre_of_rotation

# Equations of motion for rear axel



# Equations of motion for rear axel

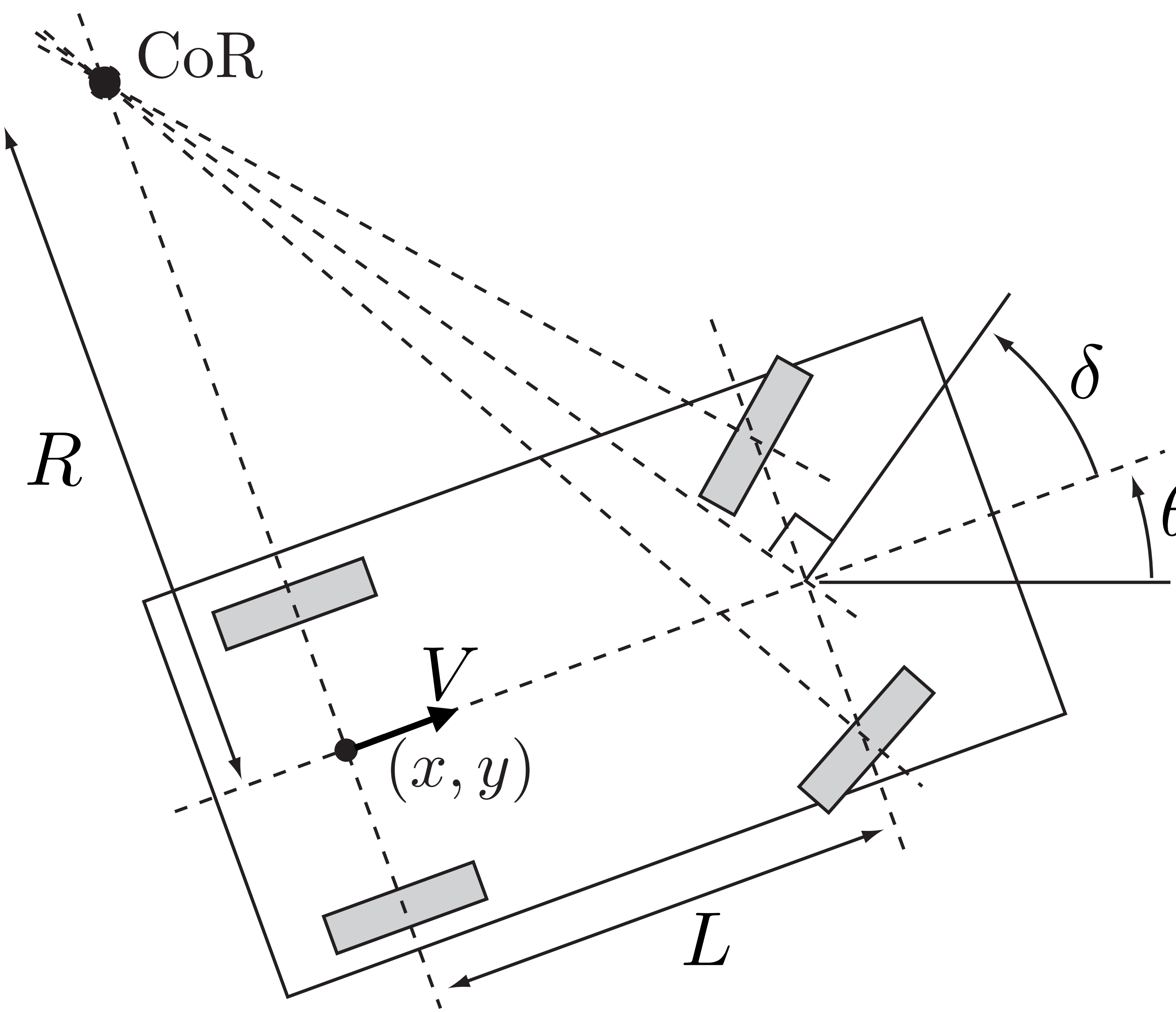



$$\tan\delta = \frac{L}{R}$$

$$R = \frac{L}{\tan\delta}$$

$$\omega = \frac{V}{R} = \frac{V\tan\delta}{L}$$

# Equations of motion for rear axel




$$\tan\delta = \frac{L}{R}$$

$$R = \frac{L}{\tan\delta}$$

$$\omega = \frac{V}{R} = \frac{V\tan\delta}{L}$$

$$\dot{x} = V\cos(\theta)$$

$$\dot{y} = V\sin(\theta)$$

$$\dot{\theta} = \omega = \frac{V\tan\delta}{L}$$

# Numerical integration

# Numerical integration

$$\dot{\theta} = \frac{V}{L} \tan \delta$$

Assume that steering angle is piece-wise constant from t to t+1

# Numerical integration

$$\dot{\theta} = \frac{V}{L} \tan \delta$$

Assume that steering angle is piece-wise constant from t to t+1

$$\theta_{t+1} - \theta_t = \int_t^{t+\Delta t} \dot{\theta} dt \quad \Rightarrow \quad \theta_{t+1} = \theta_t + \frac{V}{L} \tan \delta \Delta t$$

# Numerical integration

$$\dot{\theta} = \frac{V}{L} \tan \delta$$

Assume that steering angle is piece-wise constant from t to t+1

$$\theta_{t+1} - \theta_t = \int_t^{t+\Delta t} \dot{\theta} dt \quad \rightarrow \quad \theta_{t+1} = \theta_t + \frac{V}{L} \tan \delta \Delta t$$

$$x_{t+1} - x_t = \int_t^{t+\Delta t} \dot{x} dt = \int_t^{t+\Delta t} V \cos \theta(t) dt$$

# Numerical integration

$$\dot{\theta} = \frac{V}{L} \tan \delta$$

Assume that steering angle is piece-wise constant from t to t+1

$$\theta_{t+1} - \theta_t = \int_t^{t+\Delta t} \dot{\theta} dt \quad \Rightarrow \quad \theta_{t+1} = \theta_t + \frac{V}{L} \tan \delta \Delta t$$

$$x_{t+1} - x_t = \int_t^{t+\Delta t} \dot{x} dt = \int_t^{t+\Delta t} V \cos \theta(t) dt = \int_t^{t+\Delta t} \frac{V \cos \theta}{\dot{\theta}} \frac{d\theta}{dt} dt$$

# Numerical integration

$$\dot{\theta} = \frac{V}{L} \tan \delta$$

Assume that steering angle is piece-wise constant from t to t+1

$$\theta_{t+1} - \theta_t = \int_t^{t+\Delta t} \dot{\theta} dt \quad \Rightarrow \quad \theta_{t+1} = \theta_t + \frac{V}{L} \tan \delta \Delta t$$

$$x_{t+1} - x_t = \int_t^{t+\Delta t} \dot{x} dt = \int_t^{t+\Delta t} V \cos \theta(t) dt = \int_t^{t+\Delta t} \frac{V \cos \theta}{\dot{\theta}} \frac{d\theta}{dt} dt$$

$$= \frac{L}{\tan \delta} \int_t^{t+\Delta t} \cos \theta d\theta = \frac{L}{\tan \delta} (\sin \theta_{t+1} - \sin \theta_t)$$

# Numerical integration

$$\dot{\theta} = \frac{V}{L} \tan \delta$$

Assume that steering angle is piece-wise constant from t to t+1

$$\theta_{t+1} - \theta_t = \int_t^{t+\Delta t} \dot{\theta} dt \quad \rightarrow \quad \theta_{t+1} = \theta_t + \frac{V}{L} \tan \delta \Delta t$$

$$x_{t+1} - x_t = \int_t^{t+\Delta t} \dot{x} dt = \int_t^{t+\Delta t} V \cos \theta(t) dt = \int_t^{t+\Delta t} \frac{V \cos \theta}{\dot{\theta}} \frac{d\theta}{dt} dt$$

$$= \frac{L}{\tan \delta} \int_t^{t+\Delta t} \cos \theta d\theta = \frac{L}{\tan \delta} (\sin \theta_{t+1} - \sin \theta_t)$$

$$y_{t+1} - y_t = \frac{L}{\tan \delta} (-\cos \theta_{t+1} + \cos \theta_t)$$

# Why is the motion model probabilistic?

If we know how to write out equations of motion,
we should be able to exactly predict where a body ends up, right?

What are the sources of ~~noise~~ stochasticity?

| **Category** | **Example** |
|---|---|
| Control signal error | Voltage discretization, communication lag |
| Unknown physics parameters | Friction of carpet, tire pressure |
| Incorrect physics | Ignoring tire deformation, ignoring wheel slippage |

# Stochasticity

1. Control signal error

$$\hat{V} \sim \mathcal{N}(V, \sigma_v^2) \qquad \hat{\delta} \sim \mathcal{N}(\delta, \sigma_\delta^2)$$

2. Unknown physics parameters

$$\hat{L} \sim \mathcal{N}(L, \sigma_L^2)$$

3. Incorrect physics

$$\hat{x} \sim \mathcal{N}(x, \sigma_x^2) \qquad \hat{y} \sim \mathcal{N}(y, \sigma_y^2) \qquad \hat{\theta} \sim \mathcal{N}(\theta, \sigma_\theta^2)$$

# Questions

1. Can you derive the equations of motion for front axel?

2. Can you derive the equations of motion for centre of mass?