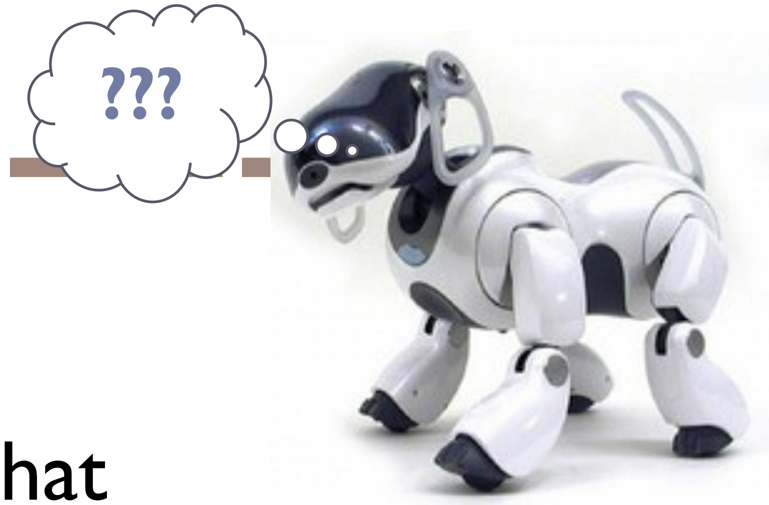
Bayesian Estimation: Localization

Gert Kootstra

Lecture Overview

- Robotic localization and the Bayesian filter
- Modeling noise
- Two examples of filters
  - Kalman Filter
  - Particle Filter
- Introduction to Simultaneous Localization and Mapping

Localiztion



- Determining the location of the robot in the world
- Essential ability for a robot that needs to navigate in complex environments
- Popular method: Bayesian Filters

$$P(x_t \mid z_{0\ldots t}, u_{0\ldots t})$$

  - Based on the sensor readings z_t and actions u_t
  - What is the probability to be at position x_t ?

Bayesian Filter

- Bayes' rule: $\boxed{P(x_t \mid z_t)}$

$$P(x \mid z) = \frac{P(z \mid x) \cdot P(x)}{P(z)}$$

Normaling factor

$$\boxed{P(x \mid z) = \alpha \cdot P(z \mid x) \cdot P(x)}$$

Markov Assumption

- The current situation depends only on the previous:

$$P(x_t \mid x_{o:t-1}) = P(x_t \mid x_{t-1})$$

Transition model

- The observation only depends on the current situation:

$$P(z_t \mid x_{0:t}, z_{0:t-1}) = P(z_t \mid x_t)$$

Sensor model

- Benefits:
  - Constant computation and memory demand
  - Usable in real-time situations: robotics

Bayesian Filter

- Iterative process for robot localization
- Transition model:

$$P(x_{t+1} \mid z_t) = \int_{-\infty}^{\infty} P(x_{t+1} \mid x_t) P(x_t \mid z_t) dx_t$$

- Sensor model:

$$P(x_{t+1} \mid z_{t+1}) = \alpha P(z_{t+1} \mid x_{t+1}) P(x_{t+1} \mid z_t)$$

Lecture Overview

- Robotic localization and the Bayesian filter
- **Modeling noise**
- Two examples of filters
  - Kalman Filter
  - Particle Filter
- Introduction to Simultaneous Localization and Mapping

Transition Model

- Transition model

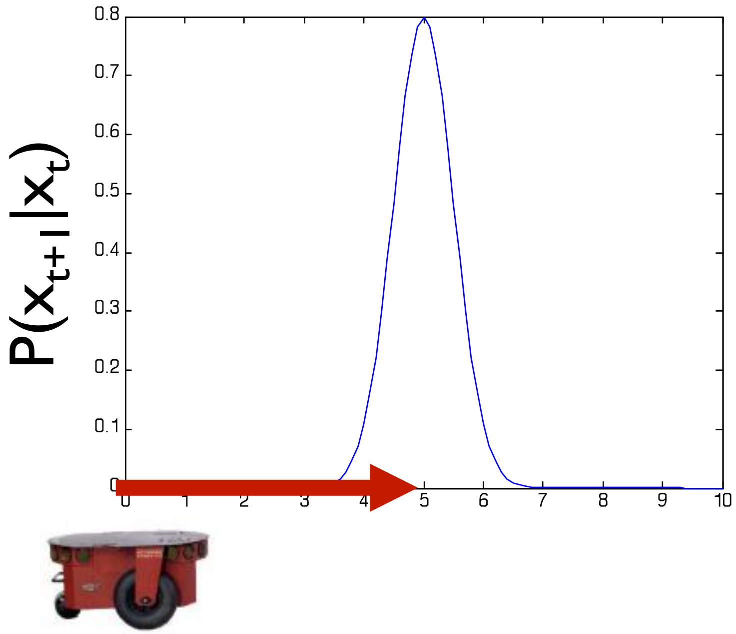
$$\boxed{P(x_{t+1} \mid x_t)}$$

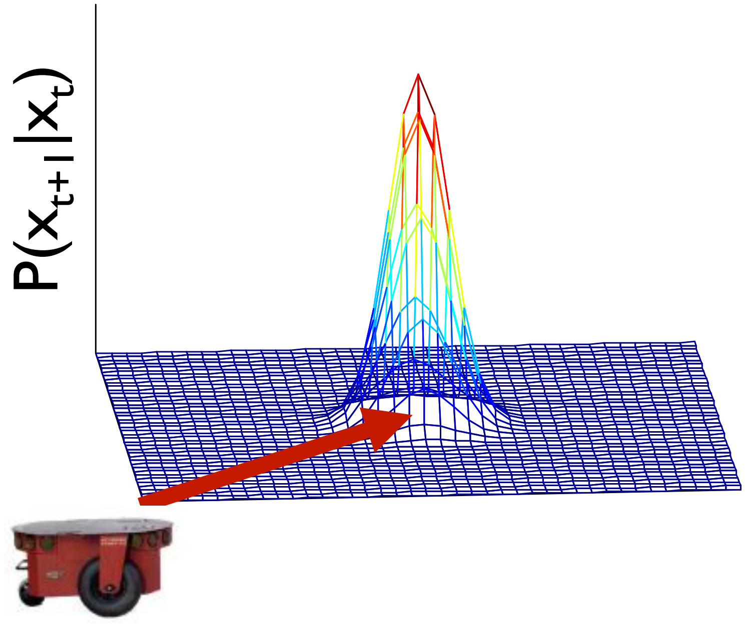
  - Given the estimate of the current robot position x_t
  - What is the probability that the robot is at x_{t+1} in the next time step?
  - The probability models the noise and uncertainty in the transition of robot.

Transition Model

- Simple Gaussian model

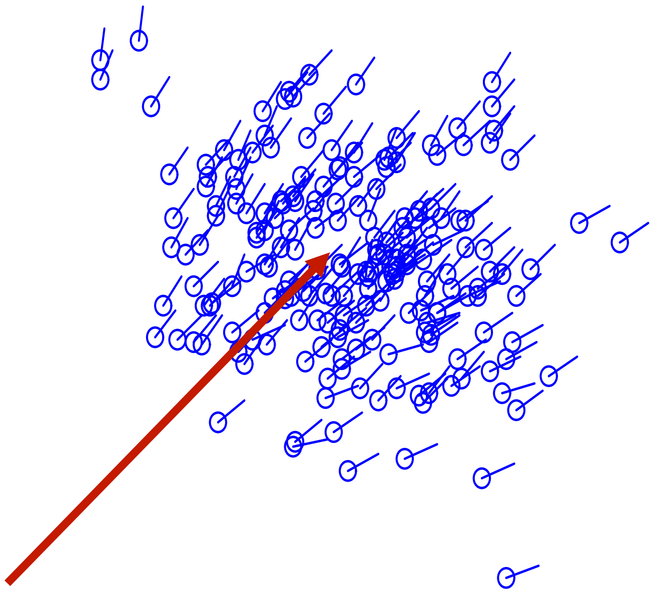
$$P(x_{t+1} \mid x_t)$$

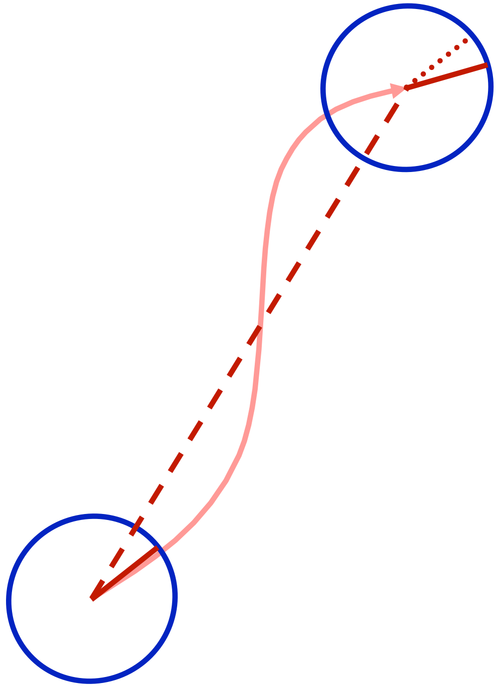




Transition Model

- More complex model (in particle filter)
  - Noise in transition
  - Noise in rotation
  - Noise in orientation





$$P(x_{t+1} \mid x_t)$$

Sensor Model

- Sensor model

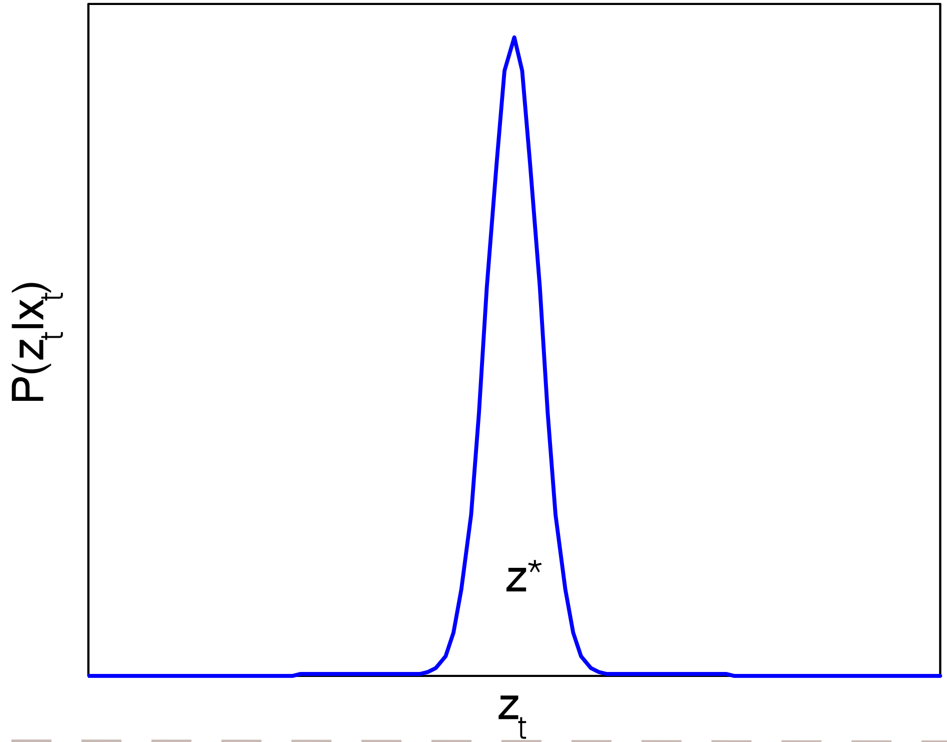
$$\boxed{P(z_t \mid x_t)}$$

  - Given the estimate of the current robot position x_t
  - What is the probability that the robot makes the observation z_t ?
  - A map of the environment needs to be known to get this probability
  - Noise of the robot's sensors are incorporated in this probability

Sensor Model

$P(z_t \mid x_t)$

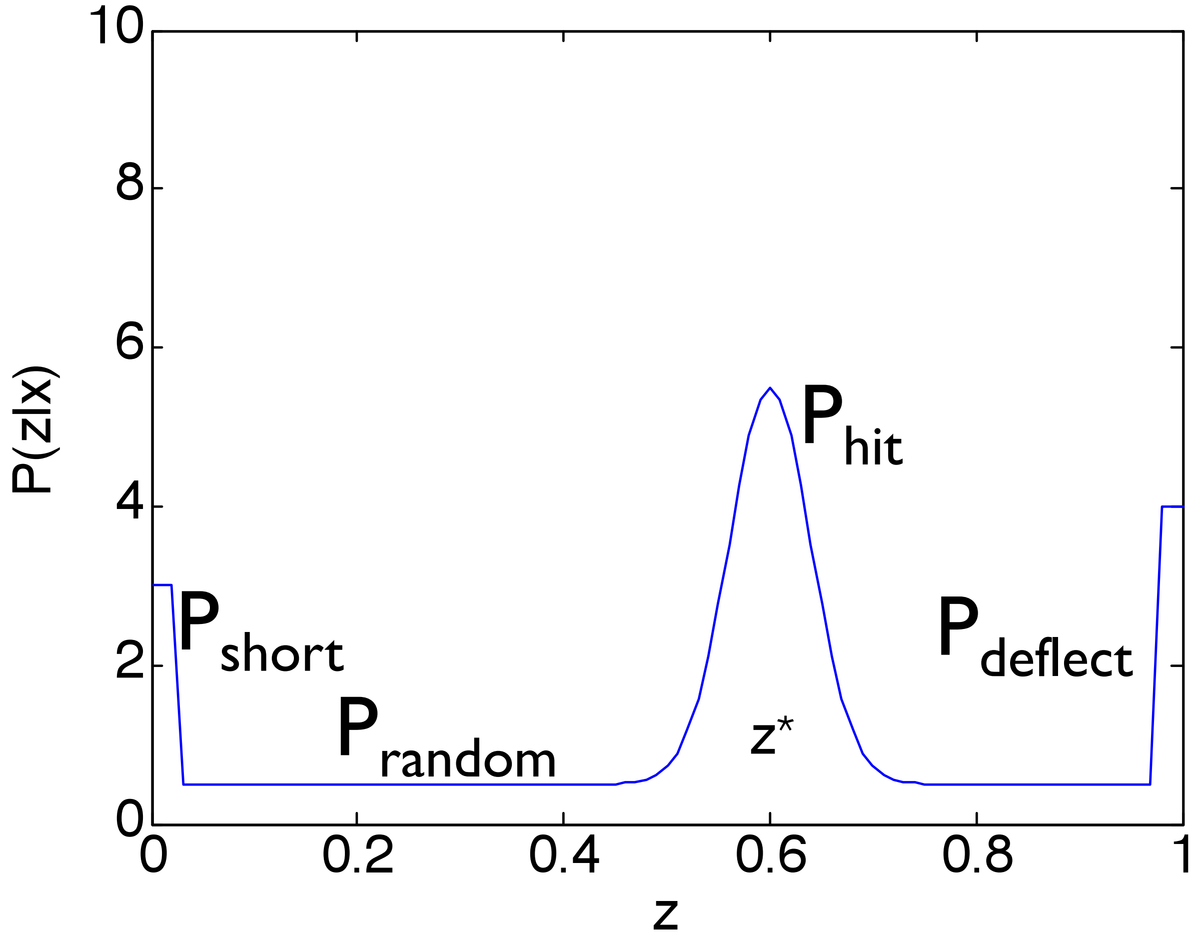
- Simple sensor model
  - z^* is the expected observation given the estimated position in the map x_t
  - Model of the sensor noise



Sensor Model

- More complex sensor model

$$P(z_t \mid x_t)$$



Lecture Overview

- Robotic localization and the Bayesian filter
- Modeling noise
- Two examples of filters
  - Kalman Filter
  - Particle Filter
- Introduction to Simultaneous Localization and Mapping

Bayesian Filters for Localization

- Two well-known filters
  - (Extended) Kalman Filter
  - Particle Filter (Monte-Carl Localization)

- Location of the robot is estimated based on
  - The action that the robot performs
  - The observation that the robot makes
  - A known map of the environment

Lecture Overview

- Robotic localization and the Bayesian filter
- Modeling noise
- Two examples of filters
  - Kalman Filter
  - Particle Filter
- Introduction to Simultaneous Localization and Mapping

Kalman Filter

- The transition and the sensor observation are both modeled with Gaussian distributions
- Two steps
  - Prediction step: The transition model

$$\boxed{P(x_{t+1} \mid z_t) = \int_{-\infty}^{\infty} P(x_{t+1} \mid x_t)\, P(x_t \mid z_t)\, dx_t}$$

  - Update step: The sensor model

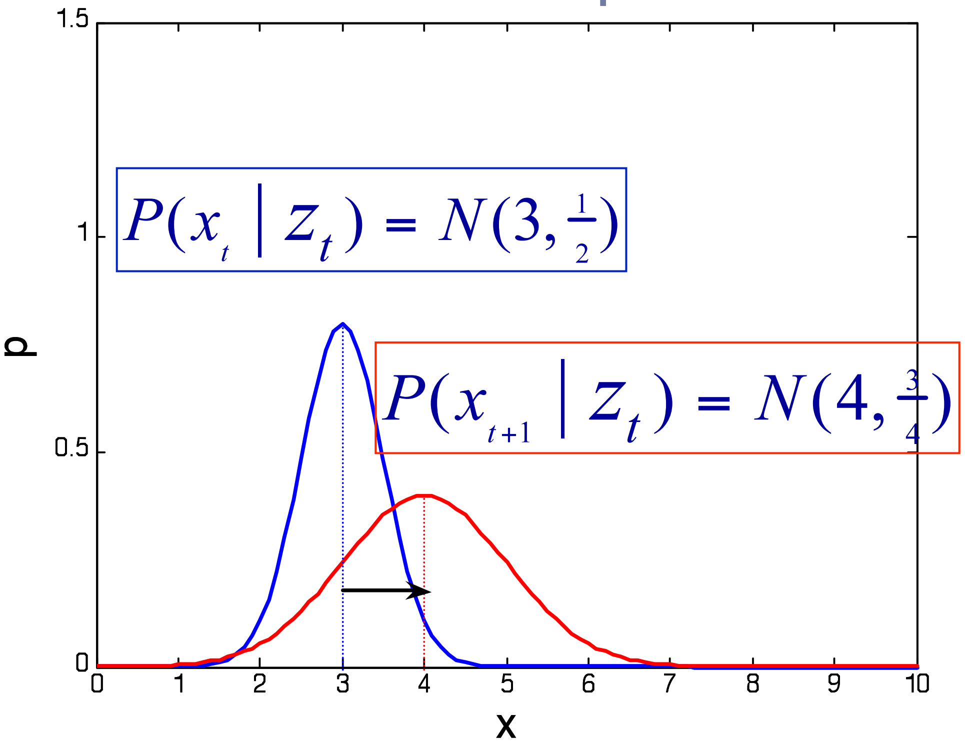
$$\boxed{P(x_{t+1} \mid z_{t+1}) = \alpha\, P(z_{t+1} \mid x_{t+1})\, P(x_{t+1} \mid z_t)}$$

Kalman Filter: Prediction Step

- Applying the transition model

$$\boxed{N(x_t, \sigma_t^2)} \longrightarrow \boxed{N(x_{t'}, \sigma_{t'}^2)}$$

The estimated position



$$P(x_t \mid z_t) = N(3, \tfrac{1}{2})$$

$$P(x_{t+1} \mid z_t) = N(4, \tfrac{3}{4})$$

Odometrie: $\Delta x, \sigma_x^2$

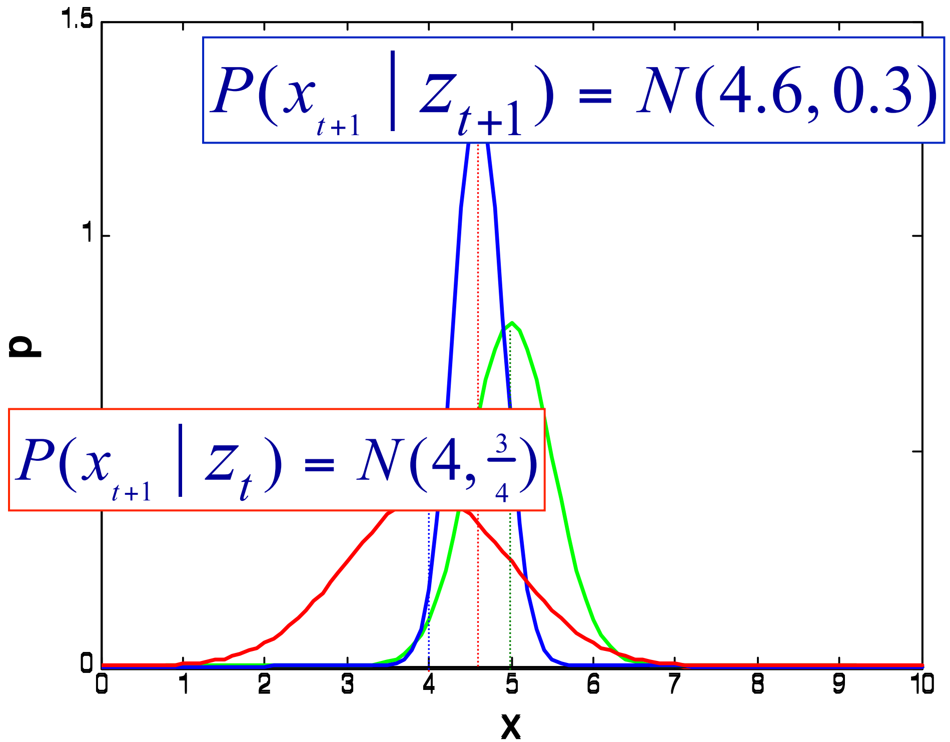
$$\boxed{x_{t'} = x_t + \Delta x}$$

$$\boxed{\sigma_{t'}^2 = \sigma_t^2 + \sigma_x^2}$$

Kalman Filter: Update Step

- Applying the sensor model

$$N(x_{t'}, \sigma_{t'}^2) \rightarrow N(x_{t+1}, \sigma_{t+1}^2)$$



$$\boxed{x_{t+1} = \frac{\sigma_z^2 \cdot x_{t'} + \sigma_{t'}^2 \cdot z_{t+1}}{\sigma_z^2 + \sigma_{t'}^2}}$$

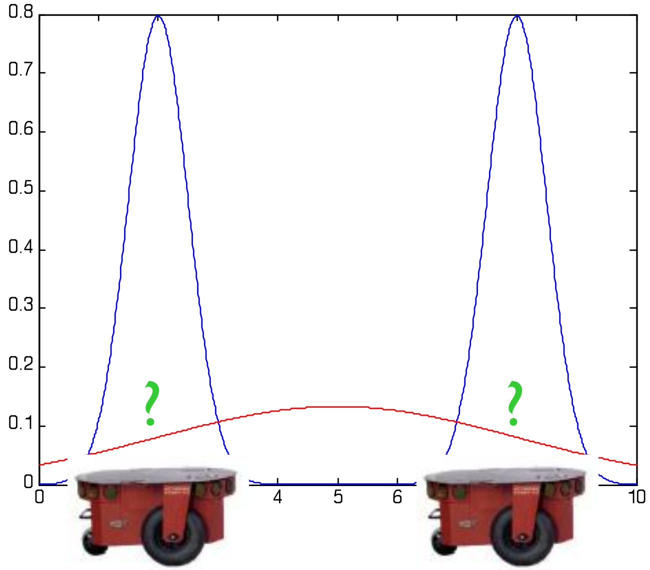
Gewogen gemiddelde

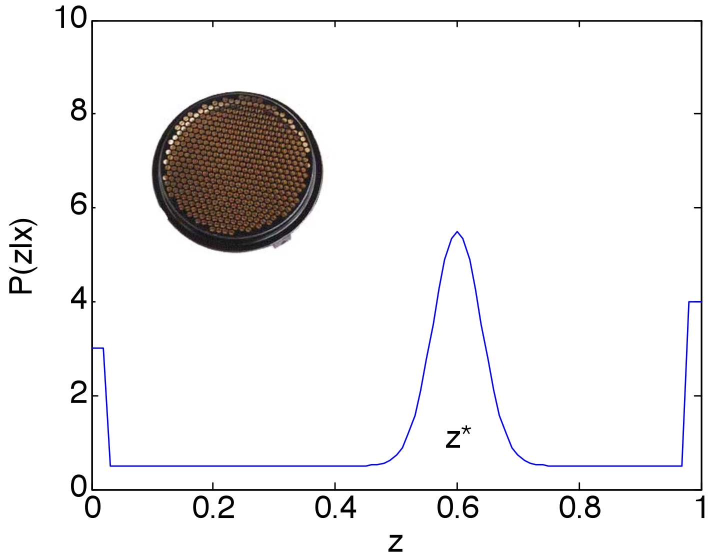
$$\boxed{\frac{1}{\sigma_{t+1}^2} = \frac{1}{\sigma_{t'}^2} + \frac{1}{\sigma_z^2}}$$

$$\sigma_{t+1}^2 < \min\left(\sigma_{t'}^2, \sigma_z^2\right)$$

Kalman Filter

- Efficient integration of the motion and the observation of the robot for localization
- Disadvantage:
  - Only unimodal distributions can be modeled



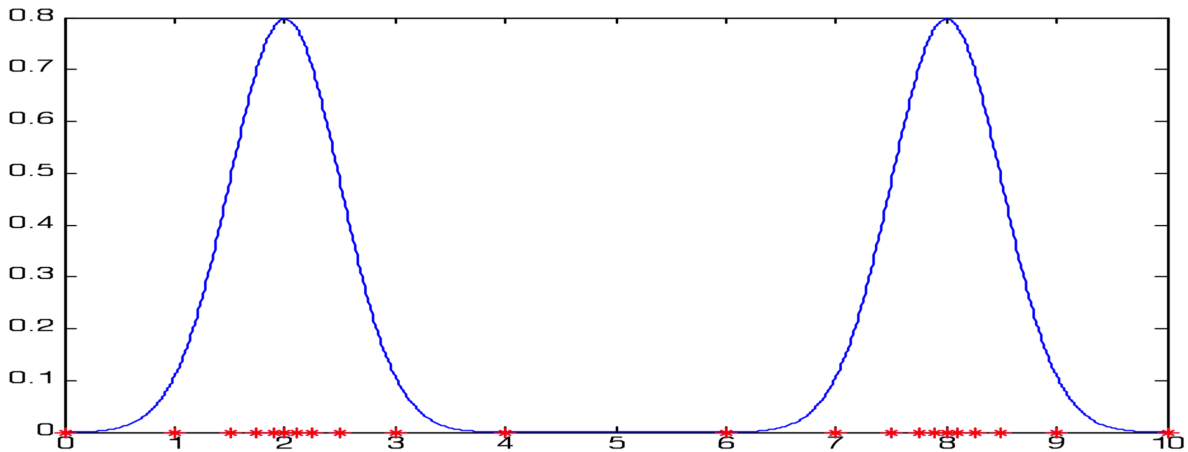


Lecture Overview

- Robotic localization and the Bayesian filter
- Modeling noise
- Two examples of filters
  - Kalman Filter
  - Particle Filter
- Introduction to Simultaneous Localization and Mapping

Particle Filters

- Representation using a set of particles
- The density of particles represents the distribution



- Distribution can take any form
- Enough particles need to be used

Particle Filters for Localization

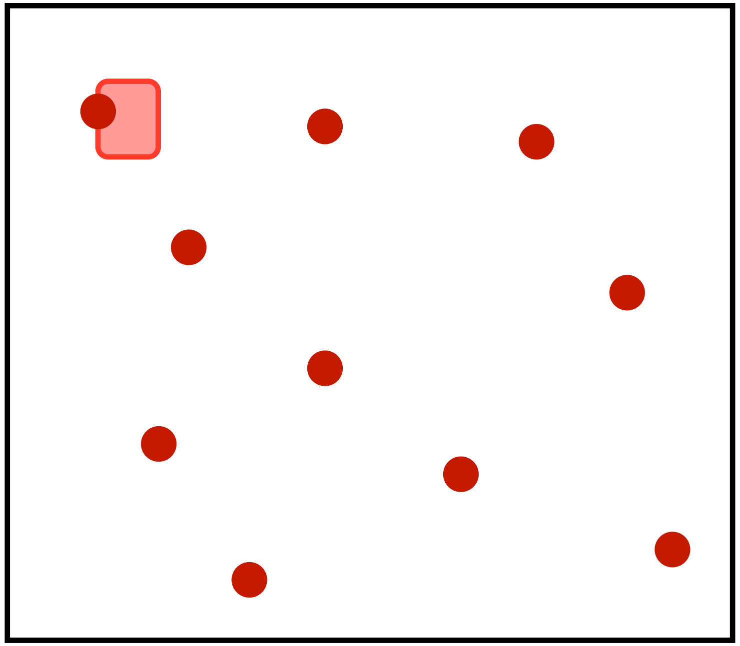
- The task of the particle filter is to distribute the particles to represented

$$P(x_t \mid x_{t-1}, z_t, u_t)$$

- By incorporating
  - The robot's observation
  - The robot's action
- Termed: Monte-Carlo Localization

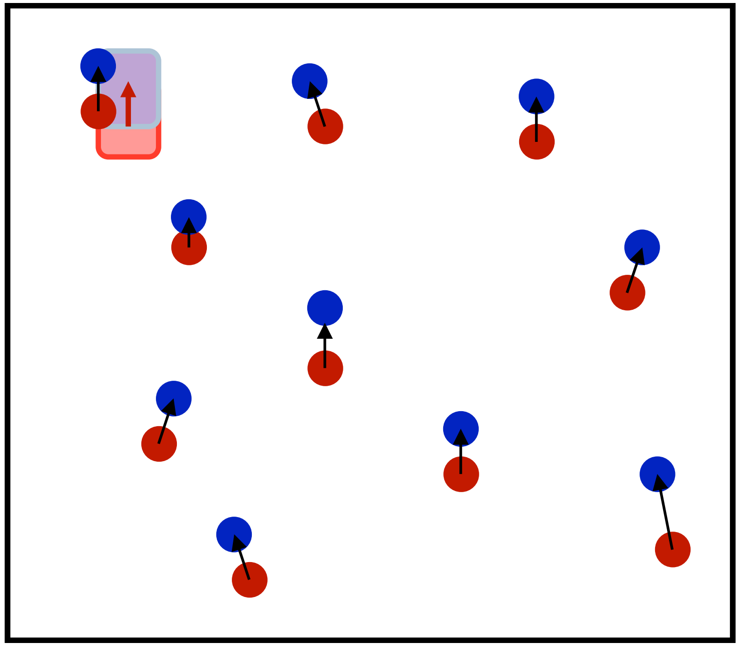
Monte-Carlo Localization

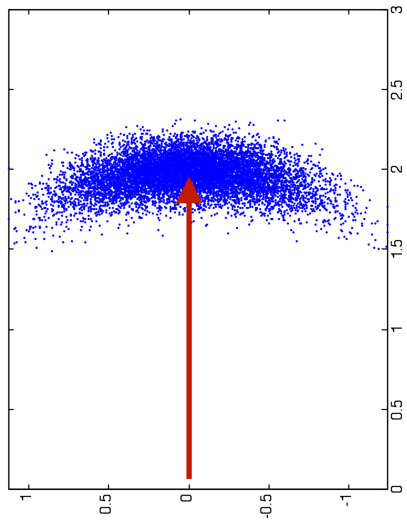
- Initially the particles are randomly distributed
  - No prior information about the robot's location



Monte-Carlo Localization

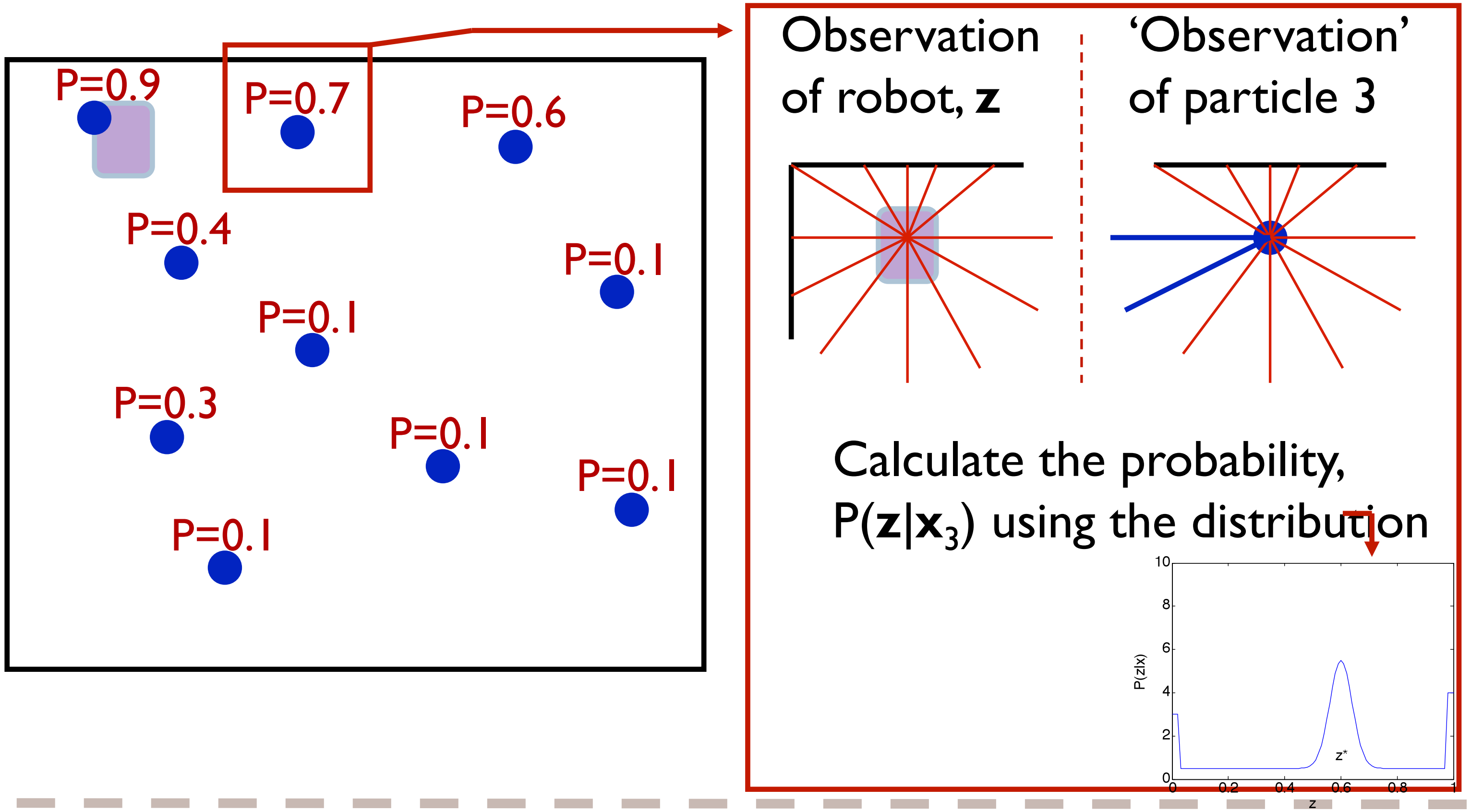
- Step 1: Applying the transition model
  - Based on odometry of the robot





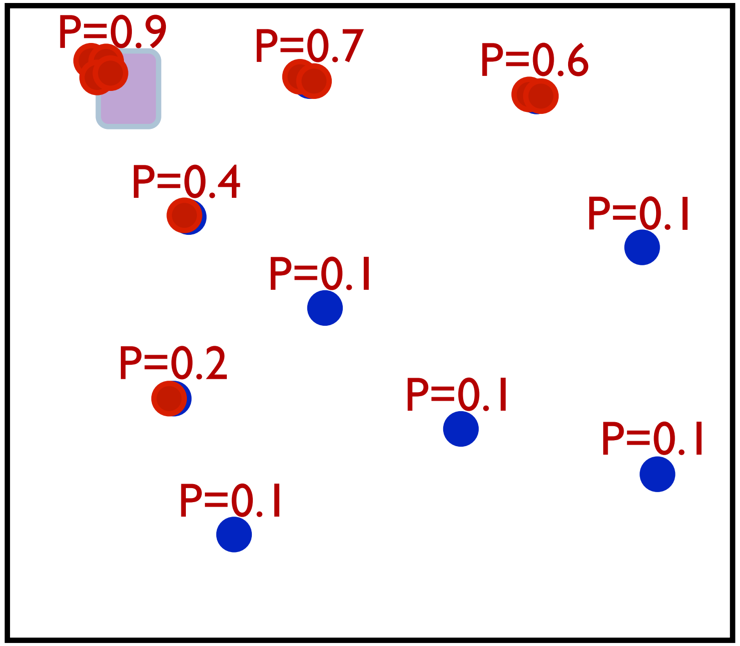
Monte-Carlo Localization

- Step 2: Applying the sensor model
  - Probability of particle based on observation



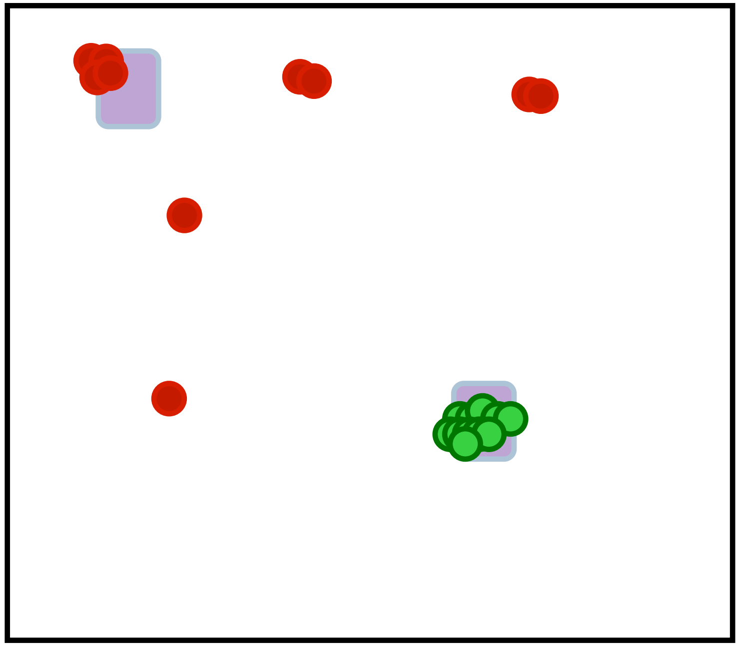
Monte-Carlo Localization

- Step 3: Resample the particle population
  - Sample proportional to the probability of particles

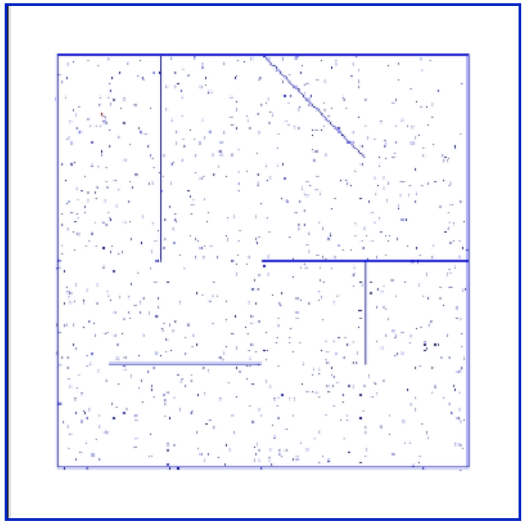


Monte-Carlo Localization

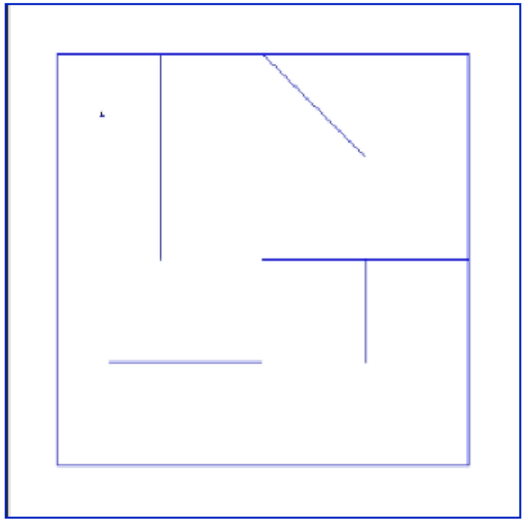
- By iteratively applying step 1-3, the particle population converges to the true distribution



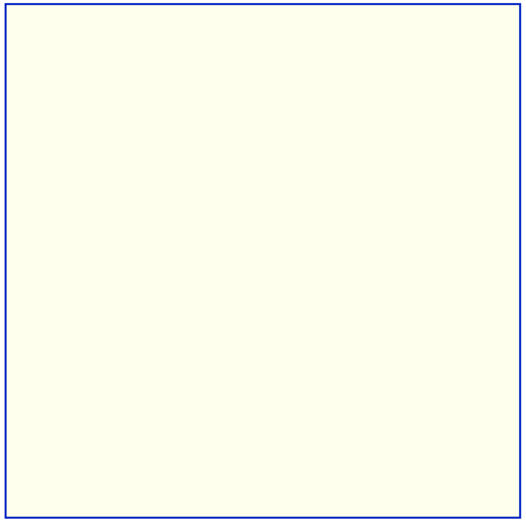
MCL Example



Many particle,
initial position unknown



Few particle,
initial position known



Few particles,
initial position unknown

Kalman Filter vs Particle Filter

- Kalman Filter
  - Efficient update, low memory
  - Can only model unimodal Gaussian distributions
- Particle Filter
  - Transition and sensor distributions can take any form
  - Needs sufficient number of particles, depending on the size of the environment
  - Less efficient

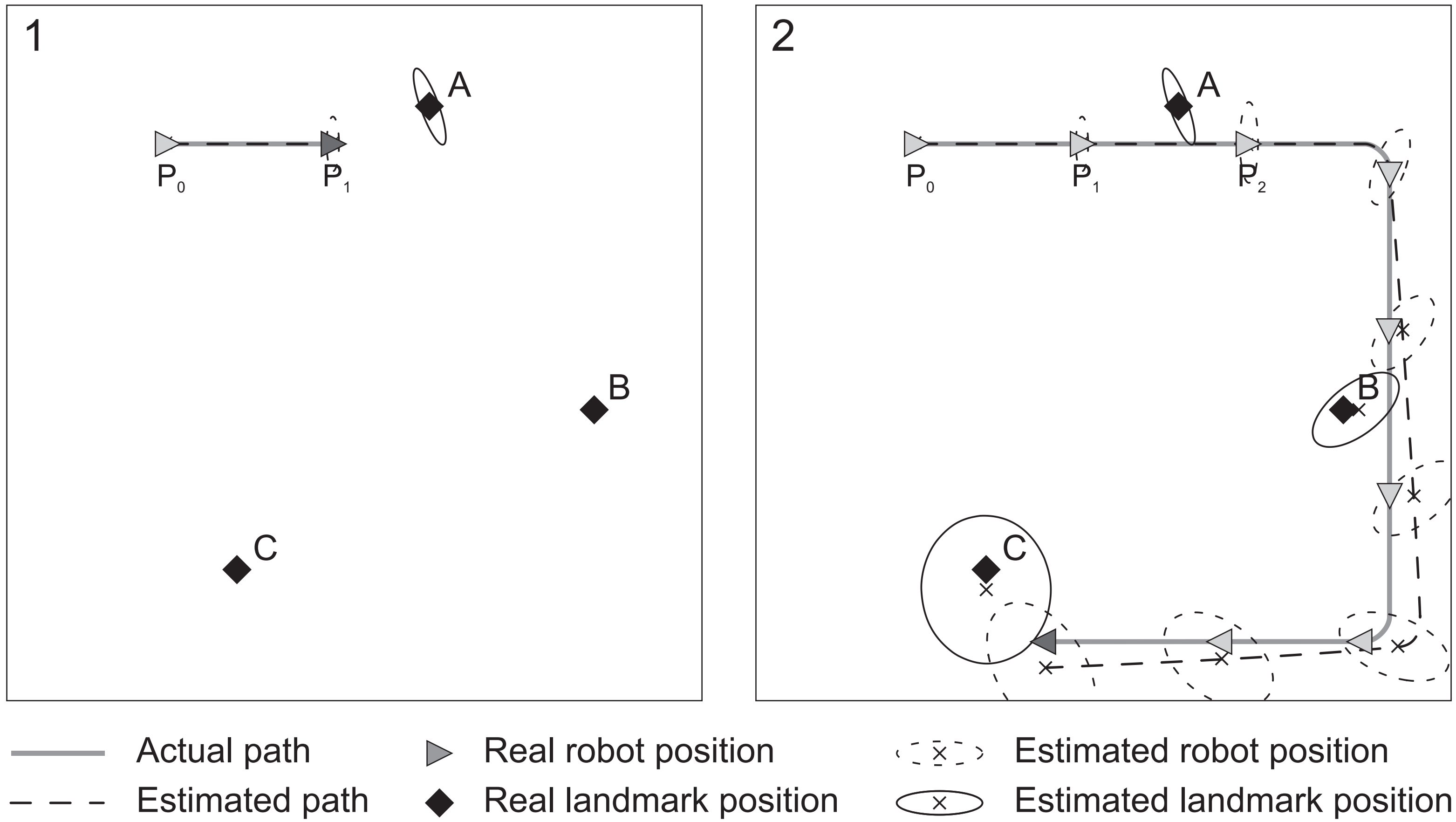
Lecture Overview

- Robotic localization and the Bayesian filter
- Modeling noise
- Two examples of filters
  - Kalman Filter
  - Particle Filter
- Introduction to Simultaneous Localization and Mapping

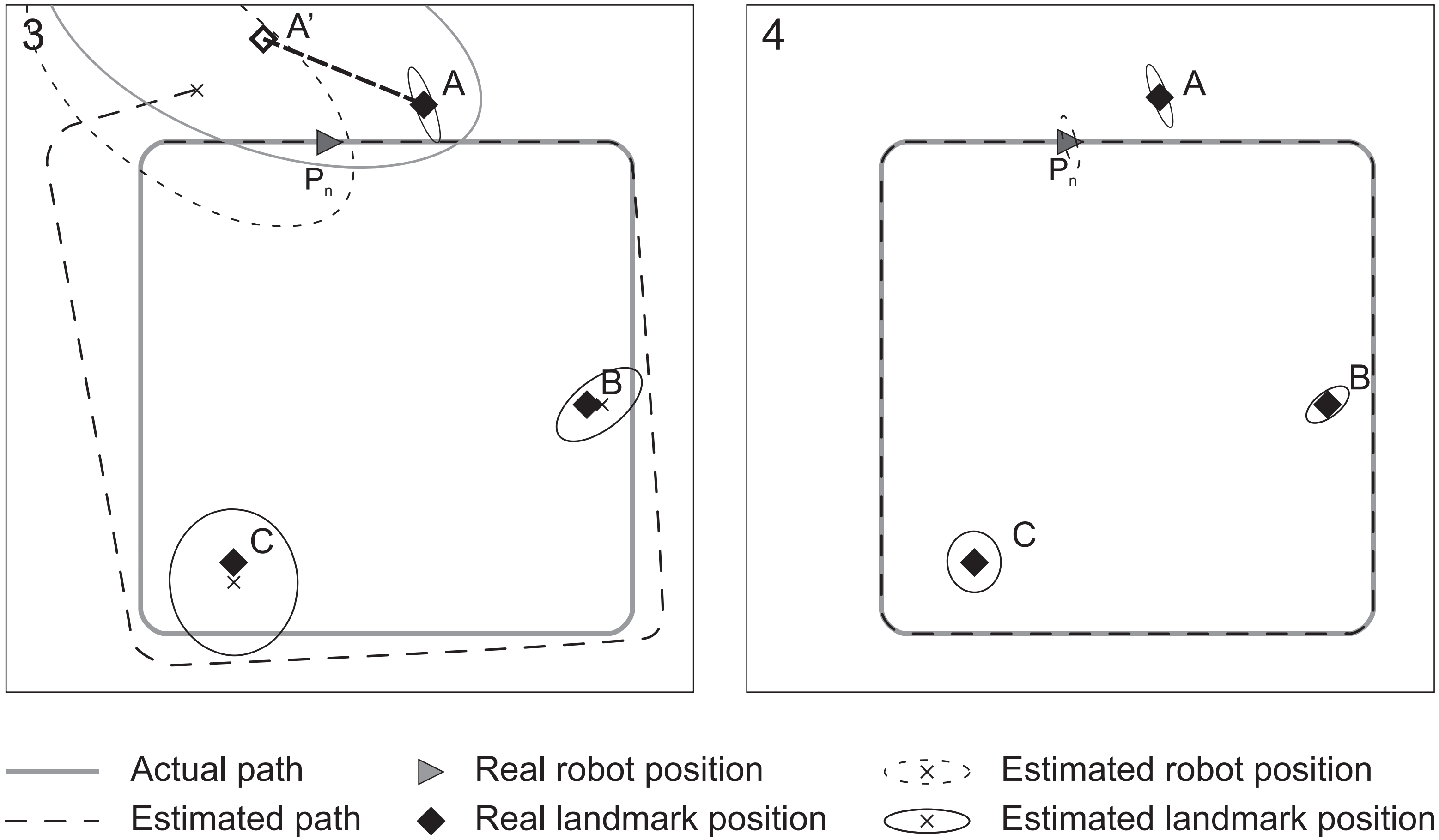
SLAM

- Simultaneous Localization and Mapping
  - Map of the environment is unknown
  - Build the map, and…
  - … use the map to localize
- Chicken-egg problem
  - Map is needed for localization
  - Location needs to be know to create map
- Problem
  - Uncertainty of robot's position grows

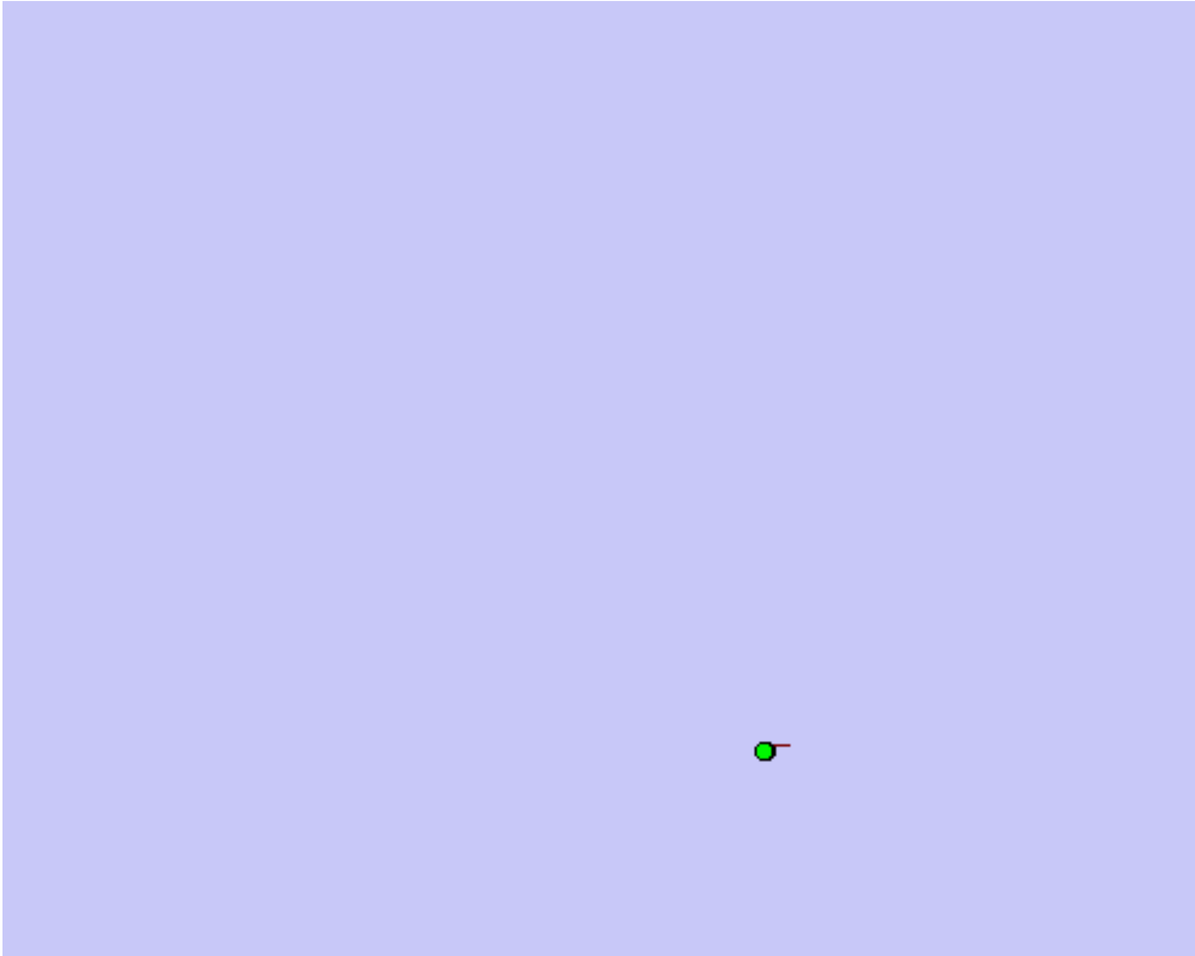
SLAM: Growing Uncertainty



SLAM: Loop Closing



SLAM: Example



Dieter Fox, Sebastiaan Thrun en Wolfram Burgard

This Lecture

- Robotic localization and the Bayesian filter
- Modeling noise
- Two examples of filters
  - Kalman Filter
  - Particle Filter
- Introduction to Simultaneous Localization and Mapping