Robot Localization

Embedded Motion Control 2013

Jos Elfring

TU/e Technische Universiteit
Eindhoven
University of Technology

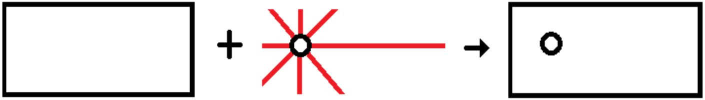
September 11, 2013

Where innovation starts

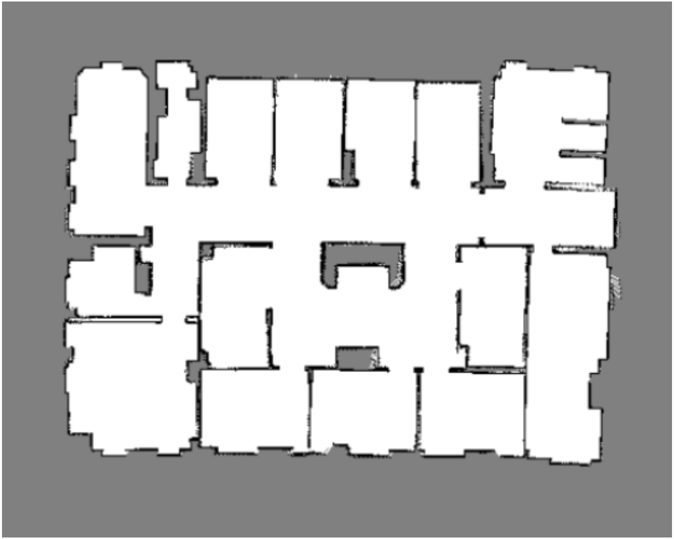
Motivation

Many tasks require a localized robot

- Map is assumed to be available
- Measurements are performed
- Position (x, y, θ) of the robot in map is calculated



Goal: localization



Today's Topics

1. Probability theory
2. Gaussian distributions
3. Robot localization using a particle filter

Random variable

A random variable is a variable that can take different values, each with its own probability:

- Side after a coin flip
- Number of pips (dots) after rolling a dice
- Rank and suit of card taken from a deck
- Position of a robot in a room
- ...

A random variable is represented by an uppercase symbol, its value by a lowercase symbol:

- X takes a value *heads* or *tails*
- $X = \mathit{heads}$ or $X = \mathit{tails}$
- $X \in \{\mathit{heads}, \mathit{tails}\}$

Probabilities

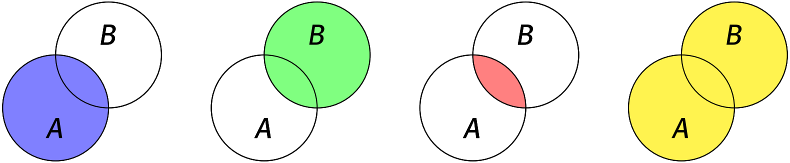
Each value a random variable might take is associated with a probability:

- $p(X = heads) = 0.5$
- $p(X = tails) = 0.5$
- $p(heads) = p(tails) = 0.5$

Definitions and properties:

- Set of all possible outcomes is called event space, *e.g.*, $\Omega = \{heads, tails\}$
- $0 \leq p(E = e_1) \leq 1$
- $p(E = e_1) \in \mathbb{R}$
- Probability that *some* event in the event space occurs is one

Notation



A | B | $A \cap B$ | $A \cup B$

Dependence versus independence (1/2)

Pick one card from a deck of cards:

- $p(hearts) = \frac{1}{4}$
- $p(red) = p(hearts \cup diamonds) = p(hearts) + p(diamonds) = \frac{1}{2}$
  Note: hearts and diamonds are mutually exclusive, *i.e.*,
  $hearts \cap diamonds = \emptyset$ hence $p(hearts \cap diamonds) = 0$
- Ace *and* spades:

$$p(ace \cap spades) = p(ace \mid spades) \cdot p(spades) = \frac{1}{13} \cdot \frac{1}{4} = \frac{1}{52}$$

Rank and suit of a card are (statistically) independent:

- Two random variables are independent if the probability of one does not affect the probability of the other.

TU/e Technische Universiteit Eindhoven University of Technology

Dependence versus independence (2/2)

Two random variables: color $C \in \{green, blue\}$ and bowl $B \in \{b_1, b_2\}$.

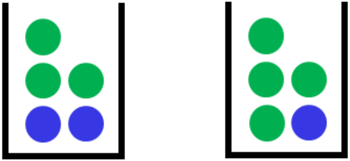
- $p(b_1) = p(b_2) = \frac{1}{2}$

Dependent event:

- Probability green ball given the bowl identity:

$$p(green \mid b_1) = \frac{p(green, b_1)}{p(b_1)} = \frac{3/10}{1/2} = \frac{3}{5}$$

$$p(green \mid b_2) = \frac{p(green, b_2)}{p(b_2)} = \frac{4/10}{1/2} = \frac{4}{5}$$



Rules of probability

Sum rule:

$$p(X) = \sum_{Y} p(X, Y),$$

Example: $p(green) = p(green, b_1) + p(green, b_2) = \frac{3}{10} + \frac{4}{10} = \frac{7}{10}.$

Product rule:

$$p(X, Y) = p(Y \mid X)p(X),$$

Example: $p(green, b_1) = p(green \mid b_1) \cdot p(b_1) = \frac{3}{5} \cdot \frac{1}{2} = \frac{3}{10}.$

- $p(X, Y)$ is the joint probability, the probability of X *and* Y
- $p(X \mid Y)$ is the conditional probability, the probability of X given Y
- $p(X)$ is the marginal probability, the probability of X

Bayes' theorem

Combining the sum rule and the product rule leads to Bayes' theorem:

$$p(X \mid Y) = \frac{p(Y \mid X)p(X)}{p(Y)},$$

In words:

$$\text{posterior} \propto \text{likelihood} \times \text{prior},$$

the denominator is usually referred to as normalizing constant.

Bayes' theorem: example

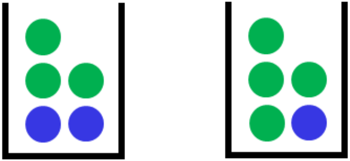
Assume a blue ball is observed. What is the probability of the ball being picked from bowl one?

Using Bayes' theorem:

$$p(b_1 \mid blue) = \frac{p(blue \mid b_1)p(b_1)}{p(blue)} = \frac{2/5 \cdot 1/2}{3/10} = \frac{2}{3},$$

where the normalizing constant equals:

$$p(blue) = p(blue \mid b_1)p(b_1) + p(blue \mid b_2)p(b_2) = \frac{2}{5} \cdot \frac{1}{2} + \frac{1}{5} \cdot \frac{1}{2} = \frac{3}{10}.$$

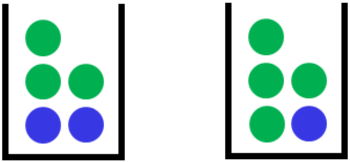


Bayes' theorem: conclusion

Before the observing the color of the ball, the prior probability of b_1 is $\frac{1}{2}$.

Once told the ball is blue, the posterior probability of b_1 is $\frac{2}{3}$.

The evidence is used to update the belief about the identity of the box that was selected.



Probability Density Functions

A probability density function f:

- Integral describes the probability of a random variable taking a value in a given interval:

$$p(a \leq X \leq b) = \int_a^b f_X(x) dx.$$

- Is nonnegative everywhere
- Can have an arbitrary number of dimensions
- The integral over the entire event space Ω equals one:

$$\int_\Omega f(x) dx = 1.$$

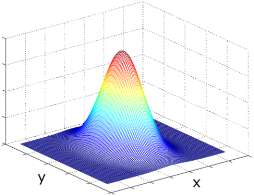
Gaussian distribution properties

Continuous probability density function.

- Characterized by a mean vector $\boldsymbol{\mu}$ and a covariance matrix $\boldsymbol{\Sigma}$:

$$\mathcal{N}\left(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}\right)=\frac{1}{\sqrt{(2\pi)^n|\boldsymbol{\Sigma}|}}\exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}(\boldsymbol{x}-\boldsymbol{\mu})\right),$$

  where $\boldsymbol{\mu}\in\mathbb{R}^n$, $\boldsymbol{\Sigma}\in\mathbb{R}^{n\times n}$ is positive-definite and $|\cdot|$ represents the determinant.
- Gaussian is conjugate to itself: if likelihood and prior are Gaussian distributions, the posterior will be Gaussian too
- Gaussian distribution is unimodal

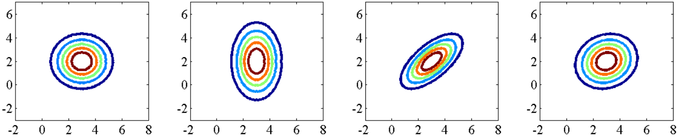


Covariance matrix

- Diagonal elements: variance in each of the dimensions
- Off-diagonal elements: how do two dimensions vary together
- Symmetric

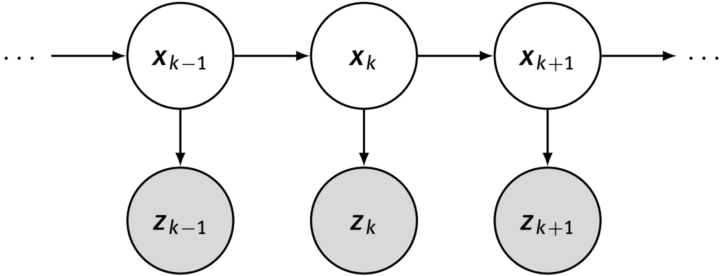
$$\mathbf{\Sigma}_1 = \begin{bmatrix} 1.5 & 0 \\ 0 & 1.5 \end{bmatrix}, \quad \mathbf{\Sigma}_2 = \begin{bmatrix} 1.5 & 0 \\ 0 & 3.0 \end{bmatrix},$$

$$\mathbf{\Sigma}_3 = \begin{bmatrix} 1.5 & 1.0 \\ 1.0 & 1.5 \end{bmatrix}, \quad \mathbf{\Sigma}_4 = \begin{bmatrix} 1.5 & 0.2 \\ 0.2 & 1.5 \end{bmatrix}.$$



General State Estimation problem

Estimate a hidden state $\boldsymbol{x}_k$ using measurements $\boldsymbol{z}_k$ related to this state.



In general:

- A state is a random variable that evolves over time (at discrete time steps)
- A motion model or process model describes how a state evolves over time
- Markov assumption: the history of state $\boldsymbol{x}_k$ is adequately summarized by the previous state only, *i.e.*, $p(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1}, \ldots, \boldsymbol{x}_0) = p(\boldsymbol{x}_k \mid \boldsymbol{x}_{k-1})$

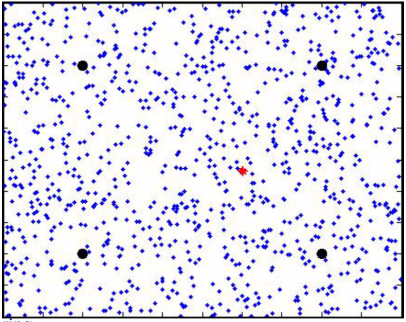
During the remainder of this presentation:

- The state contains the robot position
- The motion model represents the robot's motion

Particle Filter

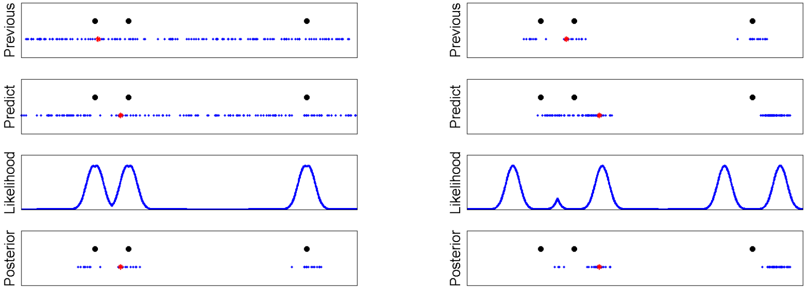
Idea: Approximate posterior by many guesses, density of guesses is posterior distribution

- Each guess is named a particle
- Map is assumed to be available
- Prediction step: apply robot movements to set of particles
- Up-date step: determine consistency measurement and particle (and keep consistent particles, remove inconsistent particles)



Technische Universiteit Eindhoven University of Technology

Particle Filter: 1D example



- Resampling avoids degeneracy: very few particles with high weight
- Noise in measurement and motion models avoids impoverishment

Technische Universiteit Eindhoven University of Technology

Particle Filter: implementation

- Each particle is a ‹weight,state› pair
- Collection of n particles represented by the set S

function ParticleFilter(S, u, z)
    $S' = \emptyset, \eta = 0$
    for $i \leftarrow 1$ to n **do**
        Sample index $j \sim \{w\}$ with replacement
        Sample possible successor state: $x' \sim p(x' \mid u, s_j)$
        Compute new weight: $w' = p(z \mid x')$
        Add new particle: $S' = S' \cup \{< x', w' >\}$
        Update auxiliary variable: $\eta = \eta + w'$
    end for
    for $m \leftarrow 1$ to n **do**
        Normalize weight: $w_m = \frac{1}{\eta} w_m$
    end for
end function

Technische Universiteit Eindhoven University of Technology

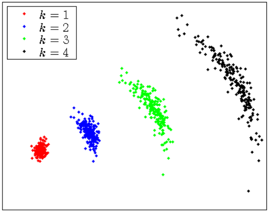
Particle Filters: pros and cons

Advantages:

- Allows for complex and multimodal distributions and non-linear models
- Easy to implement
- Computational effort 'where needed'

Disadvantages:

- Problems when number of particles is low
- Does not scale well to high dimensional spaces



Particle Filters: good to know

- Many particle filter variants exist, main difference: resampling approach
- Algorithm introduced today has many different names:
  - Sampling Importance Resampling filter
  - Bootstrap filter
  - Monte Carlo filter
  - CONDENSATION algorithm

Particle filters are applied successfully in many, many applications!

Technische Universiteit
Eindhoven
University of Technology

Further Reading

Introduction to particle filters:

- David Salmond and Neil Gordon, *An introduction to particle filters* (includes Matlab code example): http://dip.sun.ac.za/~herbst/MachineLearning/ExtraNotes/ParticleFilters.pdf

Basic probability theory, particle filters and much more:

- Sebastian Thrun, Wolfram Burgard, Dieter Fox, *Probabilistic Robotics*, MIT Press, 2005

Random variables, Bayes' theorem, Gaussian distribution and more:

- http://en.wikipedia.org/wiki/Random_variable
- http://en.wikipedia.org/wiki/Bayes'_theorem
- http://en.wikipedia.org/wiki/Multivariate_normal_distribution