State Estimation for Mobile Robots

Luca Iocchi
Cooperative Cognitive Robots Lab
Dept. of Computer and System Sciences
University of Rome “La Sapienza”, Italy

Where am I?
Many robotic tasks require an estimation of the robot location (position + orientation) in the environment

Where are important objects?
Many robotic tasks require an estimation of object location (position, orientation, speed, direction)

Problem Definition

\[ Z_t : \text{observations about robot location/object state at time } t \]

\[ X_t : \text{state of robot location/object state at time } t \]

Three different problems

\[ X_t = \text{algorithm } (Z_t) \]

\[ X_t = \text{algorithm } (Z_1, \ldots, Z_t) \]

\[ p(X_t) = \text{algorithm } (Z_1, \ldots, Z_t) \]
2D Robot Location

\[ X = (x, y, \theta) \]

\[ Z = \text{observations about lines of the field and goals} \]

Direct vs. Indirect Observations

**Direct observations** if sensors can directly measure state variables (e.g., GPS + compass)

**Indirect observations** if sensors can measure quantities related to state variables (e.g., field elements)

Example

Self-Localization through landmark perception

Problem Formulation

\[ Z_t : \text{observations about robot location/object state at time } t \]

\[ X_t : \text{state of robot location/object state at time } t \]

\[ p(X_t | X_{t-1}) \]

\[ p(Z_t | X_t) \]
Problem Formulation

Given DBN

![Diagram showing state transitions and observations](image)

Objective: estimate $p(X_t | Z_1, \ldots, Z_t)$

Probabilistic Solution

$$P(X_t | Z_{1:t}, Z_{1:t-1}) = P(X_t | Z_{1:t}) =$$

$$P(X_t | Z_{t}, Z_{1:t-1}) =$$

$$P(Z_t | X_t, Z_{1:t-1}) P(X_t | Z_{1:t-1}) / P(Z_t | Z_{1:t-1}) =$$

$$\alpha P(Z_t | X_t) P(X_t | Z_{1:t-1}) =$$

$$\alpha P(Z_t | X_t) \sum_{X_{t-1}} P(X_t | x_{t-1}) P(x_{t-1} | Z_{1:t-1})$$

Bayesian Filtering

$$P(X_t | Z_{1:t}) = \alpha P(Z_t | X_t) \sum_{X_{t-1}} P(X_t | x_{t-1}) P(x_{t-1} | Z_{1:t-1})$$

Prior

Transition (or motion) model

Likelihood (sensor model)

Problems

1) How to compute sensor and motion models?
2) How to compute the distribution in an efficient way?
Estimation of sensor and motion models

\[ P(Z_t \mid X_t) \quad P(X_t \mid X_{t-1}) \]

**Analytical models:** precise, but difficult to obtain

**Empirical models:** can be applied to any sensor/actuator, but precision depends on accuracy of experimental evaluation

Sensor models

\[ P(Z_t \mid X_t) \quad Z_t \text{ direct estimation of } X_t \]

**Transition (motion) models**

\[ P(X_t \mid X_{t-1}) \]

Analytical / experimental evaluation of probability distribution of being in the state \( X_t \), given that the previous state was \( X_{t-1} \)

- \( X_t \) robot location depends on \( X_{t-1} \) and command \( U_t \)
- \( X_t \) object position seen by the robot depends on \( X_{t-1} \) which include position, speed and direction
Kalman Filter vs. Particle Filter

- Efficiency
- Precision
- Robustness
- Easy of implementation
- Easy of parameters tuning

Kalman Filter: optimal for linear systems and normal distributions, very efficient

Monte Carlo (Particle) Filter: good for any distribution, can be computationally expensive

Examples

- **Self-localization (global vs. position tracking)**
  X = robot location, Z = landmark positions

- **SLAM**
  X = robot location+map, Z = landmark positions

- **Object tracking**
  X = object status (position, speed, direction,…
  Z = observation about object

PF Self-Localization in RoboCup
PF based SLAM

KF based boat tracking

References

