Introduction to Mobile Robotics

Simultaneous Localization and Mapping

Landmark-based FastSLAM

The SLAM Problem

- SLAM stands for simultaneous localization and mapping
- The task of building a map while estimating the pose of the robot relative to this map
- Why is SLAM hard?
  Chicken and egg problem: a map is needed to localize the robot and a pose estimate is needed to build a map

The SLAM Problem

A robot moving through an unknown, static environment

Given:
- The robot’s controls
- Observations of nearby features

Estimate:
- Map of features
- Path of the robot

Why is SLAM a hard problem?

SLAM: robot path and map are both unknown!

Robot path error correlates errors in the map
Why is SLAM a hard problem?

- In the real world, the mapping between observations and landmarks is unknown.
- Picking wrong data associations can have catastrophic consequences.
- Pose error correlates data associations.

Data Association Problem

- A data association is an assignment of observations to landmarks.
- In general, there are more than \( \binom{n}{m} \) possible associations.
- Also called “assignment problem”.

Particle Filters

- Represent belief by random samples.
- Estimation of non-Gaussian, nonlinear processes.
- Sampling Importance Resampling (SIR) principle:
  - Draw the new generation of particles.
  - Assign an importance weight to each particle.
  - Resampling.
- Typical application scenarios are tracking, localization, ...

Localization vs. SLAM

- A particle filter can be used to solve both problems.
- Localization: state space \( <x, y, \theta> \).
- SLAM: state space \( <x, y, \theta, map> \).
  - For landmark maps = \( <l_1, l_2, ..., l_m> \).
  - For grid maps = \( <c_{11}, c_{12}, ..., c_{1n}, c_{21}, ..., c_{nm}> \).
- Problem: The number of particles needed to represent a posterior grows exponentially with the dimension of the state space!
Dependencies

- Is there a dependency between the dimensions of the state space?
- If so, can we use the dependency to solve the problem more efficiently?

In the SLAM context
- The map depends on the poses of the robot.
- We know how to build a map given the position of the sensor is known.

Factored Posterior (Landmarks)

\[
p(x_{1:t}, l_{1:m} | z_{1:t}, u_{0:t-1}) = p(x_{1:t} | z_{1:t}, u_{0:t-1}) \cdot p(l_{1:m} | x_{1:t}, z_{1:t})
\]

Does this help to solve the problem?

Factorization first introduced by Murphy in 1999
Mapping using Landmarks

Knowledge of the robot’s true path renders landmark positions conditionally independent.

Factored Posterior

\[
p(x_{1:t}, l_{1:m} | z_{1:t}, u_{0:t-1}) = p(x_{1:t} | z_{1:t}, u_{0:t-1}) \cdot \prod_{i=1}^{M} p(l_i | x_{1:t}, z_{1:t})
\]

- This factorization is also called Rao-Blackwellization.
- Given that the second term can be computed efficiently, particle filtering becomes possible!

FastSLAM

- Rao-Blackwellized particle filtering based on landmarks [Montemerlo et al., 2002]
- Each landmark is represented by a 2x2 Extended Kalman Filter (EKF)
- Each particle therefore has to maintain \( M \) EKFs

<table>
<thead>
<tr>
<th>Particle</th>
<th>x, y, θ</th>
<th>Landmark 1</th>
<th>Landmark 2</th>
<th>Landmark M</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#N</td>
<td>x, y, θ</td>
<td>Landmark 1</td>
<td>Landmark 2</td>
<td>Landmark M</td>
</tr>
</tbody>
</table>
FastSLAM – Action Update

Particle #1

Particle #2

Particle #3

FastSLAM – Sensor Update

Particle #1

Particle #2

Particle #3

FastSLAM – Sensor Update

Particle #1

Particle #2

Particle #3

FastSLAM – Video
FastSLAM Complexity

- Update robot particles based on control \( u_{t-1} \) \( \mathcal{O}(N) \)
- Incorporate observation \( z_t \) into Kalman filters \( \mathcal{O}(N \cdot \log(M)) \)
- Resample particle set \( \mathcal{O}(N \cdot \log(M)) \)

\( N = \text{Number of particles} \)
\( M = \text{Number of map features} \)

Data Association Problem

- Which observation belongs to which landmark?

Multi-Hypothesis Data Association

- Data association is done on a per-particle basis
- Robot pose error is factored out of data association decisions

Per-Particle Data Association

- Two options for per-particle data association
  - Pick the most probable match
  - Pick a random association weighted by the observation likelihoods
- If the probability is too low, generate a new landmark

\[ P(\text{observation|red}) = 0.3 \quad P(\text{observation|blue}) = 0.7 \]
Results – Victoria Park

- 4 km traverse
- < 5 m RMS position error
- 100 particles

Blue = GPS
Yellow = FastSLAM

Dataset courtesy of University of Sydney

Results – Data Association

Comparison of FastSLAM and EKF Given Motion Ambiguity

Results – Accuracy

Accuracy of FastSLAM vs. the EKF on Simulated Data
FastSLAM Summary

- FastSLAM factors the SLAM posterior into low-dimensional estimation problems
  - Scales to problems with over 1 million features
- FastSLAM factors robot pose uncertainty out of the data association problem
  - Robust to significant ambiguity in data association
  - Allows data association decisions to be delayed until unambiguous evidence is collected
- Advantages compared to the classical EKF approach
- Complexity of $O(N \log M)$