Mapping in Dynamic Environments

Wolfram Burgard

University of Freiburg, Germany
Mapping is a Key Technology for Mobile Robots

- Robots can robustly navigate when they have a map.
- Robots have been shown to being able to deal with a substantial amount of dynamics given a map.
- An improvement can be expected once robot can dynamically update their maps.
Simultaneous Localization and Mapping (SLAM)

- SLAM enables a robot to simultaneously estimate its position and map.

- Typically the map is assumed to be static.

- How can we extend SLAM to dynamic environments?
A Graphical Model for SLAM in Static Environments
A Graphical Model for SLAM in Dynamic Environments
Probabilistic Formulation

\[ P(m_1,...,t, x_1,...,t \mid z_1,...,t, u_1,...,t) \]

- Scan matching, ...
- EKF, UKF, Particle Filters, ...
- Maximum likelihood estimation (GraphSLAM)
Mapping by Filtering

- **Idea**: Bayes filter over both robot positions and map.

\[
\text{Bel}(m_t, x_t) = \eta p(z_t \mid m_t, x_t) \int \int p(x_t \mid x_{t-1}, m_{t-1}, u_{t-1}) \text{Bel}(m_{t-1}, x_{t-1}) \, dm_{t-1} \, dx_{t-1}
\]

- Independence of \( m \) and \( x \):

\[
\text{Bel}(m_t, x_t) = \eta p(z_t \mid m_t, x_t) \int \int p(x_t \mid x_{t-1}, u_{t-1}) p(m_t \mid m_{t-1}) \text{Bel}(m_{t-1}, x_{t-1}) \, dm_{t-1} \, dx_{t-1}
\]

- **In static worlds**:

\[
\text{Bel}(m_t, x_t) = \eta p(z_t \mid m_t, x_t) \int \int p(x_t \mid x_{t-1}, u_{t-1}) \text{Bel}(m_{t-1}, x_{t-1}) \, dm_{t-1} \, dx_{t-1}
\]
Dynamic Environments

1. Generally, the map is not static while mapping:
   - People walking by,
   - moving objects.

2. Often the world changes over time:
   - Doors open or close,
   - objects are moved around,
   - plants grow ... 

3. Typical resulting problems:
   - Bad alignments (localization),
   - spurious objects (mapping)
Approaches to Deal with Dynamic Environments

1. Do nothing
2. Change the sensor model
3. Filtering
4. Modeling
Occupancy Grid Maps

- Introduced by Moravec and Elfes in 1985
- Represent environment by a grid.
- Estimate the probability that a location is occupied by an obstacle.

Key assumptions

- Occupancy of individual cells is independent

\[
Bel(m_t) = P(m_t \mid u_1, z_2, \ldots, u_{t-1}, z_t) = \prod_{x,y} Bel(m_t^{[xy]})
\]

- Robot positions are known!
Updating Occupancy Grid Maps

- Typically updated using inverse sensor model and odds ratio:

\[
Bel(m_t^{[xy]}) = 1 - \left( 1 + \frac{P(m_t^{[xy]} | z_t, x_t)}{1 - P(m_t^{[xy]} | z_t, x_t)} \frac{1 - P(m_t^{[xy]})}{P(m_t^{[xy]})} \frac{Bel(m_{t-1}^{[xy]})}{1 - Bel(m_{t-1}^{[xy]})} \right)^{-1}
\]

- Or log-odds ratio \( \overline{B} \):

\[
\overline{B}(m_t^{[xy]}) = \overline{B}(m_{t-1}^{[xy]}) + \ln P(m_t^{[xy]} | z_t, x_t) - \ln \left( 1 - P(m_t^{[xy]} | z_t, x_t) \right) + \ln \left( 1 - P(m_t^{[xy]}) \right) - \ln P(m_t^{[xy]})
\]
Using Bayes Rule in Dynamic Environments

[Avots et al., 2002]
Improvement: Multiple Levels of Maps

- Use a fixed map to represent the static aspects only.
- Use a map learned on the fly to represent the current state of the world.
- Combine both maps using a conservative strategy:

\[
Bel(m_t^{[x,y]}) = \max(Bel_{\text{static}}(m^{[x,y]}), Bel_{\text{dynamic}}(m_t^{[x,y]}))
\]
Works Well in Practice
Example: On-line Mapping with Rhino

[Burgard et al., 99]
Modeling Approach: Mapping in Populated Environments

Problem:
How can we build maps while people are walking through the scene?

Solution:
- Feature-based people tracking.
- Appropriately deal with the corresponding beams during localization and map updating.
Problem Description

- Key questions
  - How many people are there?
  - Where do they go?

- Requirements
  - Real time
  - Eventually no model of the environment
  - Robot in motion
Tracking with a Moving Robot

[Schulz et al., 01]
Removing People Detections

University of Bonn

Byzantine Museum, Athens
3D Maps in Populated Environments
A Fly-Through
Filtering Approach to Mapping in Dynamic Environments

**Problem:**
- Often models of non-stationary objects are not available.
- Often we cannot assume that there is a separation between non-stationary and static objects.

**Solution:**
- Using EM to learn beams reflected by dynamic objects.
The Measurement Model

1. pose at time $t$: $x_t$
2. beam $n$ of scan $t$: $z_{t,n}$
3. maximum range reading: $\zeta_{t,n} = 1$
4. beam reflected by dynamic object: $\zeta_{t,n} = 0$ and $c_{t,n} = 0$
5. beam reflected by static object: $\zeta_{t,n} = 0$ and $c_{t,n} = 1$

$$p(z_{t,n} \mid c_{t,n}, x_t, m) = \begin{cases} 
\prod_{k=0}^{z_{t,n}-1} (1 - m_f(x_t, n, k)) 
& \text{if } \zeta_{t,n} = 1 \\
\prod_{k=0}^{z_{t,n}} (1 - m_f(x_t, n, k)) 
& \text{if } \zeta_{t,n} = 0 \text{ and } c_{t,n} = 0 \\
\prod_{k=0}^{z_{t,n}-1} (1 - m_f(x_t, n, k)) 
& \text{if } \zeta_{t,n} = 0 \text{ and } c_{t,n} = 1 
\end{cases}$$
Application of EM

\[ E_c[\ln p(z, c \mid x, m) \mid z, x, m] = N \cdot \sum_{t=1}^{T} E_c[\ln p(c_t)] \]

\[ + \sum_{t=1}^{T} \sum_{n=1}^{N} \left( E_c[c_{t,n} \mid z_{t,n}, x_t, m] \cdot (1 - \zeta_{t,n}) \cdot \ln m_f(x_t, n, z_{t,n}) \right. \]

\[ + (1 - E_c[c_{t,n} \mid z_{t,n}, x_t, m]) \cdot (1 - \zeta_{t,n}) \cdot \ln(1 - m_f(x_t, n, z_{t,n})) + \sum_{k=0}^{z_{t,n} - 1} \ln(1 - m_f(x, n, k)) \]

E-Step:

\[ e_{t,n} = \begin{cases} \frac{p(c_{t,n} = 1 | \zeta_{t,n} = 1, x_t, m)}{p(c_{t,n} = 0 | \zeta_{t,n} = 1, x_t, m) + p(c_{t,n} = 1 | \zeta_{t,n} = 1, x_t, m)} & \zeta_{t,n} = 1 \\ p(c_{t,n} = 1) \cdot \frac{m_f(x_t, n, z_{t,n})}{m_f(x_t, n, z_{t,n}) + \left( \frac{1}{p(c_{t,n} = 1)} - 1 \right) \cdot (1 - m_f(x_t, n, z_{t,n}))} & \text{else} \end{cases} \]

M-Step:

Compute most likely map using Bayes rule by considering the expectations \( e_{t,n} \) during map updating and localization.
Computing the Most Likely Map

\[
\hat{m}^{[t]} = \arg \max_{m} \left[ \sum_{j=1}^{J} \sum_{t=1}^{T} \sum_{n=1}^{N} \left( I(f(x_{t}, n, z_{t, n}) = j) \cdot (1 - \varsigma_{t, n}) \cdot (e_{t, n} \ln m_{j} + (1 - e_{t, n}) \ln(1 - m_{j})) \right) + \sum_{k=0}^{z_{t, n} - 1} I(f(x_{t}, n, k) = j) \cdot \ln (1 - m_{j}) \right] \]

Suppose

\[
\alpha_{j} = \sum_{t=1}^{T} \sum_{n=1}^{N} I(f(x_{t}, n, z_{t, n}) = j) \cdot (1 - \varsigma_{t, n}) \cdot e_{t, n}
\]

\[
\beta_{j} = \sum_{t=1}^{T} \sum_{n=1}^{N} \left[ I(f(x_{t}, n, z_{t, n}) = j) \cdot (1 - \varsigma_{t, n}) \cdot (1 - e_{t, n}) + \sum_{k=0}^{z_{t, n} - 1} I(f(x_{t}, n, k) = j) \right]
\]
Computing the Most Likely Map

We assume that all cells \( m_j \) are independent:

\[
\hat{m}^{[t]} = \arg \max_m \left( \sum_{j=1}^{J} \alpha_j \ln m_j + \beta_j \ln(1 - m_j) \right)
\]

If we set

\[
\frac{\partial m}{\partial m_j} = \frac{\alpha_j}{m_j} - \frac{\beta_j}{1 - m_j} = 0
\]

we obtain

\[
m_j = \frac{\alpha_j}{\alpha_j + \beta_j}
\]

Computing the most likely map amounts to counting how often a cell has reflected a measurement.
EM-based Estimation of the Most Likely Map

**M-Step:**
Compute map $m$ based on the expectations $e_{tn}$

**E-Step:**
Compute the expectation $e_{tn}$ that beam $n$ in scan $t$ is reflected by a dynamic object given $m$

$e_{tn}$: Expectation that $z_{tn}$ is reflected by a dynamic object
Byzantine Museum, Athens
Wean Hall (Hallway)
Pittsburgh
Craig Street/Forbes Ave
Integrating Laser & Images

Wolfram
Integrating Laser & Images
Resulting Model
Modeling Low-Dynamics

- Environments are not static
  - Doors can be opened or closed
  - Some objects are moved regularly
- How to model low-dynamic aspects?
- How to use such knowledge to improve the capabilities of a mobile robot?
Door Example

- Possible states in a corridor environment
Approach

- Segment the map into sub-maps
- Estimate possible states for each sub-map
- Store the possible configurations in the map
- Use such a representation to improve, e.g., the robot’s localization abilities
Mapping Results
Mapping Results
Extended MCL

- Estimate the pose of the robot and the current state of the environment
- We only estimate the state of the current sub-map, the robot is in (sensor provides local information)
- This avoids a large state spaces of the robot (particles):

\[ \langle x, y, \theta, s \rangle \]

robot’s pose  sub-map configuration
Localization Accuracy Comparison
Dynamic Submaps

- What if the variety of the dynamics is too large to allow the clustering approach?
- The alternative is to re-map dynamically depending on the observed changes.
- This allows to localize relative to previously unseen objects
Localization in Semi-static Environments

- Temporary maps represent semi-static objects in the environment
- Observations are caused by both static and semi-static objects
Semi-static Maps

Local map representing semi-static objects as observed by the robot while navigating (*Memory of unexpected observations*)

Static map with three different semi-static maps
Semi-static Maps

- Represented as pose-graphs
- Constructed from consecutive measurements
- After loop closure we perform optimization
Localization with Temporary Maps

- If observation consistent with static map, use static map for localization
- else select semi-static map for localization.
- Mahalanobis distance for map selection
- If no map is found, we create one
Localization with Temporary Maps

- Semi-static maps are used if consistent with the observations.
- Otherwise they are discarded
Localization with Temporary Maps
Experiments

Localization in large open spaces

- 30 particles
- Laser max-range set to 20m

Standard particle filter  Temporary maps
Summary

- Different techniques for mapping in dynamic environments
- Object modeling
- Filtering
- Learning sub-map states
- Incrementally updating maps

- The problem is not solved and key for long-term autonomy