Robot Mapping

Short Introduction to Particle Filters and Monte Carlo Localization

Cyrill Stachniss
Gaussian Filters

- The Kalman filter and its variants can only model **Gaussian distributions**

\[
p(x) = \text{det}(2\pi \Sigma)^{-\frac{1}{2}} \exp \left( -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right)
\]
Motivation

- Goal: approach for dealing with arbitrary distributions
Key Idea: Samples

- Use **multiple samples** to represent arbitrary distributions
Particle Set

- Set of weighted samples

\[ \mathcal{X} = \left\{ \langle x[i], w[i] \rangle \right\}_{i=1,\ldots,N} \]

- The samples represent the posterior

\[ p(x) = \sum_{i=1}^{N} w[i] \delta_{x[i]}(x) \]
Particles for Approximation

- Particles for function approximation

- The more particles fall into an interval, the higher its probability density

How to obtain such samples?
Importance Sampling Principle

- We can use a different distribution $g$ to generate samples from $f$
- Account for the “differences between $g$ and $f$” using a weight $w = f/g$
- target $f$
- proposal $g$
- Pre-condition: $f(x)>0 \Rightarrow g(x)>0$
Importance Sampling Principle
Particle Filter

- Recursive Bayes filter
- Non-parametric approach
- Models the distribution by samples
- Prediction: draw from the proposal
- Correction: weighting by the ratio of target and proposal

The more samples we use, the better is the estimate!
Particle Filter Algorithm

1. Sample the particles using the proposal distribution

\[ x_t^{[i]} \sim \pi(x_t \mid \ldots) \]

2. Compute the importance weights

\[ w_t^{[i]} = \frac{\text{target}(x_t^{[i]})}{\text{proposal}(x_t^{[i]})} \]

3. Resampling: “Replace unlikely samples by more likely ones”
Particle Filter Algorithm

Particle\_filter(\mathcal{X}_{t-1}, u_t, z_t):
1: \quad \tilde{\mathcal{X}}_t = \mathcal{X}_t = \emptyset
2: \quad \text{for } m = 1 \text{ to } M \text{ do}
3: \quad \text{sample } x_t^{[m]} \sim \pi(x_t)
4: \quad w_t^{[m]} = \frac{p(x_t^{[m]})}{\pi(x_t^{[m]})}
5: \quad \tilde{\mathcal{X}}_t = \tilde{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle
6: \quad \text{endfor}
7: \quad \text{for } m = 1 \text{ to } M \text{ do}
8: \quad \text{draw } i \text{ with probability } \propto w_t^{[i]}
9: \quad \text{add } x_t^{[i]} \text{ to } \mathcal{X}_t
10: \quad \text{endfor}
11: \quad \text{return } \mathcal{X}_t
Monte Carlo Localization

- Each particle is a pose hypothesis
- Proposal is the motion model

\[ x_t^{[i]} \sim p(x_t | x_{t-1}, u_t) \]

- Correction via the observation model

\[ \omega_t^{[i]} = \frac{\text{target}}{\text{proposal}} \propto p(z_t | x_t, m) \]
Particle Filter for Localization

\textbf{Particle\_filter}(\mathcal{X}_{t-1}, u_t, z_t):

1: \quad \tilde{\mathcal{X}}_t = \mathcal{X}_t = \emptyset
2: \quad \text{for } m = 1 \text{ to } M \text{ do}
3: \quad \text{sample } x_t^{[m]} \sim p(x_t | u_t, x_{t-1}^{[m]})
4: \quad w_t^{[m]} = p(z_t | x_t^{[m]})
5: \quad \tilde{\mathcal{X}}_t = \tilde{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle
6: \quad \text{endfor}
7: \quad \text{for } m = 1 \text{ to } M \text{ do}
8: \quad \text{draw } i \text{ with probability } \propto w_t^{[i]}
9: \quad \text{add } x_t^{[i]} \text{ to } \mathcal{X}_t
10: \quad \text{endfor}
11: \quad \text{return } \mathcal{X}_t
Application: Particle Filter for Localization (Known Map)
Resampling

- Survival of the fittest: “Replace unlikely samples by more likely ones”
- “Trick” to avoid that many samples cover unlikely states
- Needed as we have a limited number of samples
Resampling

- Roulette wheel
- Binary search
- $O(n \log n)$

- Stochastic universal sampling
- Low variance
- $O(n)$
Low Variance Resampling

\texttt{Low\_variance\_resampling}(\mathcal{X}_t, \mathcal{W}_t):
1: \quad \bar{\mathcal{X}}_t = \emptyset
2: \quad r = \text{rand}(0; M^{-1})
3: \quad c = w_t^{[1]}
4: \quad i = 1
5: \quad \text{for } m = 1 \text{ to } M \text{ do}
6: \quad \quad U = r + (m - 1) \cdot M^{-1}
7: \quad \quad \text{while } U > c
8: \quad \quad \quad i = i + 1
9: \quad \quad \quad c = c + w_t^{[i]}
10: \quad \quad \text{endwhile}
11: \quad \quad \text{add } x_t^{[i]} \text{ to } \bar{\mathcal{X}}_t
12: \quad \text{endfor}
13: \quad \text{return } \bar{\mathcal{X}}_t
initialization
observation
resampling
motion update
measurement
weight update
resampling
measurement
weight update
resampling
motion update
measurement
weight update
resampling
motion update
Summary – Particle Filters

- Particle filters are non-parametric, recursive Bayes filters
- Posterior is represented by a set of weighted samples
- Not limited to Gaussians
- Proposal to draw new samples
- Weight to account for the differences between the proposal and the target
- Work well in low-dimensional spaces
Summary – PF Localization

- Particles are propagated according to the motion model
- They are weighted according to the likelihood of the observation
- Called: Monte-Carlo localization (MCL)
- MCL is the gold standard for mobile robot localization today