Simultaneous Localization and Mapping (SLAM)

Chapter 9:

Histogram Filter for Robot Localization

Saeed Saeedvand, Ph.D.

Contents

In this Chapter:

- ✓ Nonparametric Filters
- ✓ Histogram Filter for localization

Aim of this chapter:

✓ In this chapter, we explain Nonparametric Filters and see how histogram filter can be applied to the localization problem.

Posterior and prior probabilities

- ✓ A **posterior** probability is the probability of **assigning observations to groups given the data**.
- ✓ A **prior** probability is the probability that an **observation will fall into a group before you collect the data**.

Nonparametric Filters

Nonparametric filters

- ✓ A popular alternative to Gaussian techniques are **nonparametric** filters.

Do not rely on a **fixed functional form** of the posterior, **such as Gaussians**

- ✓ They **approximate** posteriors by a **finite number of values**, each roughly corresponding to a region in state space:

Nonparametric Filters

Nonparametric filters

✓ **What is the finite number of values**

- ✓ Some nonparametric Bayes filters rely on a **decomposition of the state space**
- ✓ Their **value corresponds to** the cumulative probability of the posterior density in a compact subregion of the state space.

- ✓ Some approximate the state space by **random samples** drawn from the posterior distribution

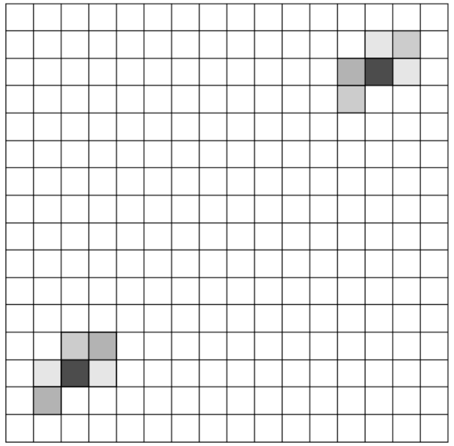
Note: The number of parameters used to approximate the posterior can be varied and change the quality of the approximation (higher better but costly).

Decomposition Techniques

✓ Decomposition techniques of continuous state spaces come into two basic approaches:

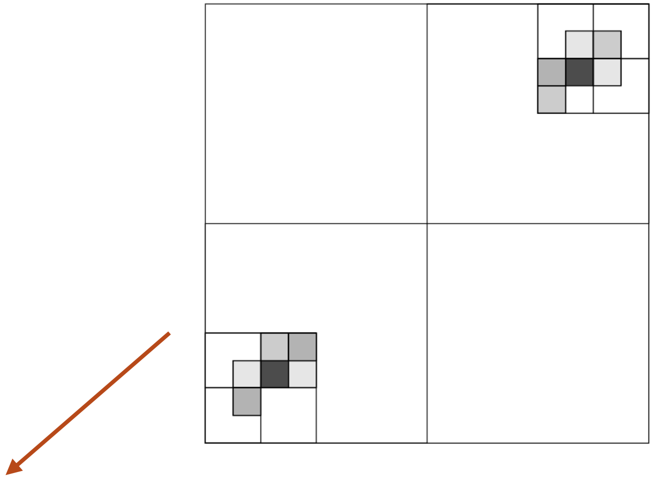
Static Techniques

✓ Rely on a fixed decomposition that is chosen in advance



Dynamic Techniques

✓ Adapt the decomposition to the specific shape of the posterior distribution (e.g. *density trees* approaches)



Density trees (Dynamic)

Density trees decompose the state space recursively, in ways that adapt the resolution to the posterior probability mass

Nonparametric Filters

Adaptive approach

- ✓ Techniques that can **adapt** the **number of parameters** to represent the posterior online

Resource-adaptive approach

- ✓ Techniques that **adapt based on the computational resources** available for belief computation
  - This approach play an important role in robotics
  - Enable robots to make decisions in real time, regardless of the computational resources available

Histogram Filter

- ✓ It decomposes the state space into finitely many regions, and represents the posterior by a **histogram**.

A histogram assigns a single cumulative probability to each region (a single probability value)

- ✓ It uses Bayes filters

In discrete spaces → *Discrete Bayes filters* We already discussed!

In continuous state spaces → *e.g. Histogram filters* → Discretization

Histogram Filter

- ✓ Histogram filters decompose a continuous state space into finitely many regions (new state space)

Idea: Thus, we can use discrete Bayesian algorithm to calculate

$$\text{range}(X_t) = x_{1,t} \cup x_{2,t} \cup \ldots x_{k,t}$$

$$x_{K,t} \bigcap x_{i,t} = \emptyset, i \neq k$$ No intersections between two regions

$$\bigcup_k x_{k,t} = \text{range}(X_t)$$ Union of all

- ✓ X_t: The state of the robot at time t
- ✓ range(X_t): A function representing the state space
- ✓ $X_{k,t}$: describes a region (that partitions state space)

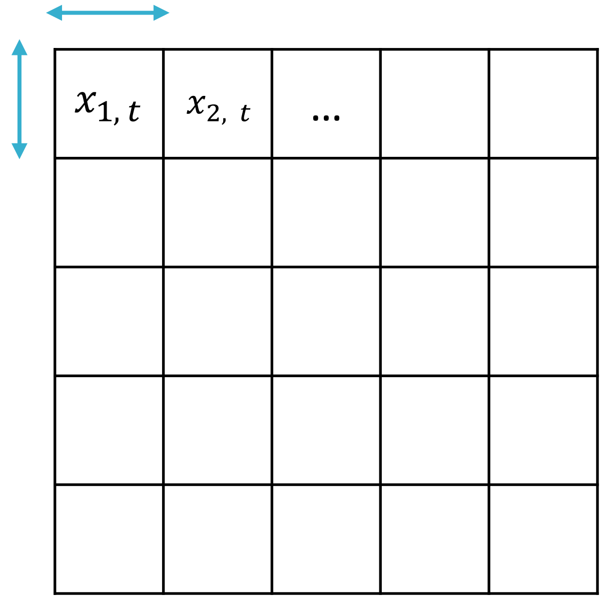
Histogram Filter

✓Decomposition of a continuous state space can be **multi-dimensional grid**, where each $X_{k,t}$ is a grid cell.

Concept:

Granularity of the decomposition:

- Can control the trade off between accuracy and computational efficiency



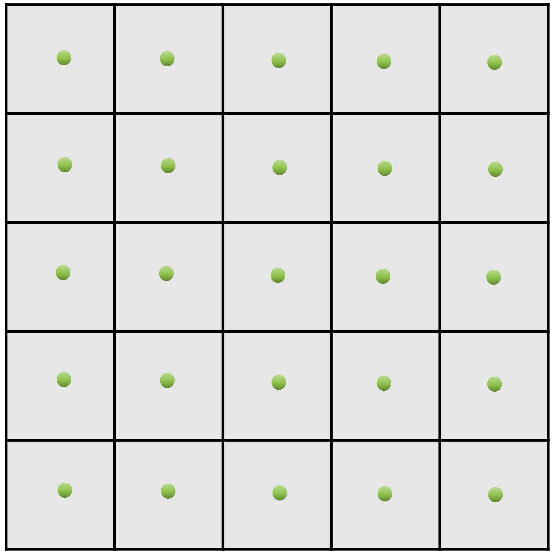
Histogram Filter

Steps

- ✓ Assign a uniform probability to each state x_t within each region $x_{k,t}$(initial belief)

$$p(x_t) = \frac{p_{k,t}}{|x_{k,t}|}$$

volume of the region $x_{k,t}$



- ✓ For each region $x_{k,t}$ a probability, $p_{k,t}$ can be assigned
- ✓ Within each region $X_{k,t}$, we can only maintain the weights for each region
- ✓ If we don’t know the pose the initial belief is equal for all locations

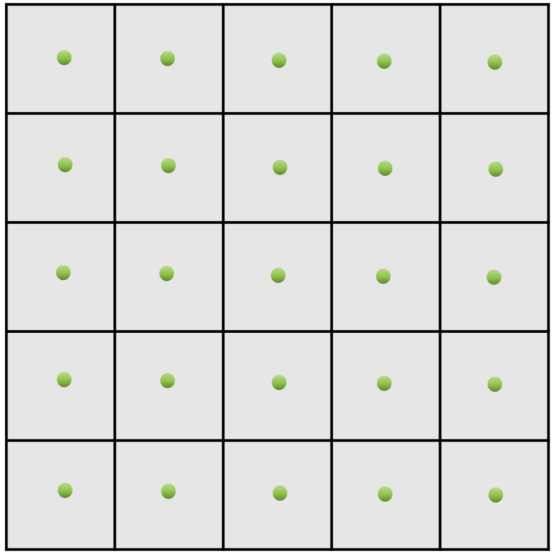
Histogram Filter

Example

$$p(x_t) = \frac{p_{k,t}}{|x_{k,t}|} = \frac{1}{25} = 0.04$$

volume of the region $x_{k,t}$

0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04
0.04	0.04	0.04	0.04	0.04



Bayesian filter (reminder)

Prediction

$$\overline{\text{bel}}(x_t) = \int p(x_t \mid x_{t-1}, u_t)\ \text{bel}(x_{t-1})\ dx$$

Correction

$$\text{bel}(x_t) = \eta\ p(o_t \mid x_t)\ \overline{\text{bel}}(x_t)$$

Note: $\int \text{bel}(x_t)\, dx_t = \sum_{i=1}^{N} p(x_t^i) = 1$

Algorithm Discrete_Bayes_filter($\{p_{k,t-1}\}, u_t, o_t$):
for all k do
$\bar{p}_{k,t} = \sum_i p(X_t = x_k \mid u_t, X_{t-1} = x_i)\ p_{i,t-1}$
$p_{k,t} = \eta\ p(o_t \mid X_t = x_k)\ \bar{p}_{k,t}$
endfor
return $\{p_{k,t}\}$

x_i, x_k denote individual states

Histogram Filter

Prediction

$$\overline{\mathrm{bel}}(x_t) = \int \mathrm{p}(x_t \mid x_{t-1}, u_t)\, \mathrm{bel}(x_{t-1})\, dx_{t-1}$$

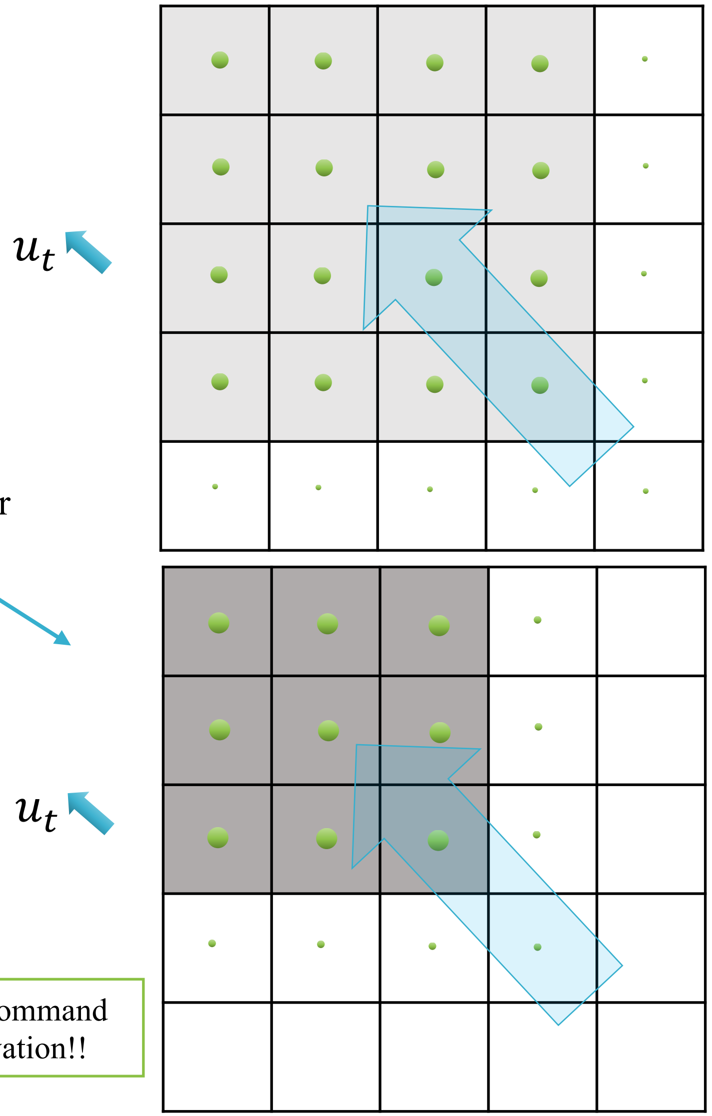
$$\overline{\mathrm{bel}}(x_t) = \sum_{i=1}^{N} \bar{w}_t^i\, \mathrm{bel}(x_{t-1})$$

$$\bar{w}_t^i = \sum_{j=1}^{N} w_{t-1}^j\, \mathrm{p}(x^i \mid x^j, u_t)$$

Predicted weight = Sum of the probability math * weights after motion command

$$\overline{\mathrm{bel}}(x_t) = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{t-1}^j\, \mathrm{p}(x^i \mid x^j, u_t)\, \mathrm{bel}(x_{t-1})$$

Next motion command with no observation!!



Map shift

Histogram Filter

Update

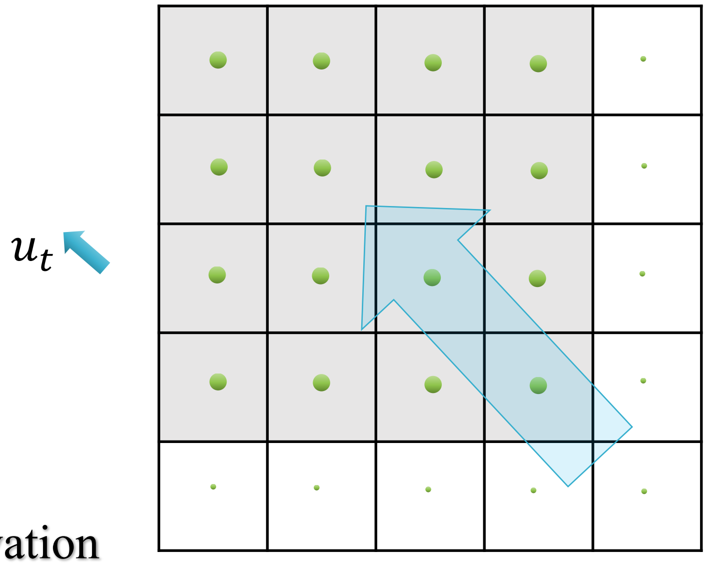
$$bel(x_t) = \eta \underbrace{p(o_t \mid x_t)} \overline{\text{bel}}(x_t)$$

$$= \sum_{i=1}^{N} \boxed{w_t^i} \overline{\text{bel}}(x_t)$$

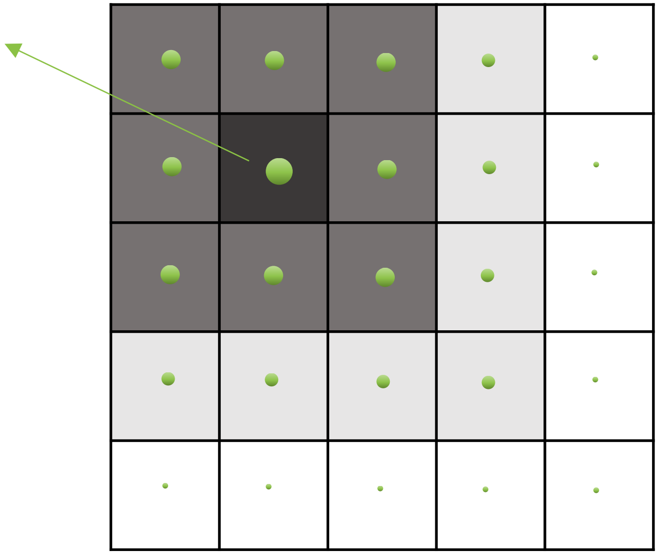
Prior weight * probability of measurement / sum of all (Normalize)

$$w_t^i = \frac{\bar{w}_t^i p(o_t \mid x^i)}{\sum_{j=1}^{N} \bar{w}_t^j \, p(o_t \mid x^j)}$$

$$bel(x_t) = \sum_{i=1}^{N} \frac{\bar{w}_t^i p(o_t \mid x^i)}{\sum_{j=1}^{N} \bar{w}_t^j \, p(o_t \mid x^j)} \overline{\text{bel}}(x_t)$$

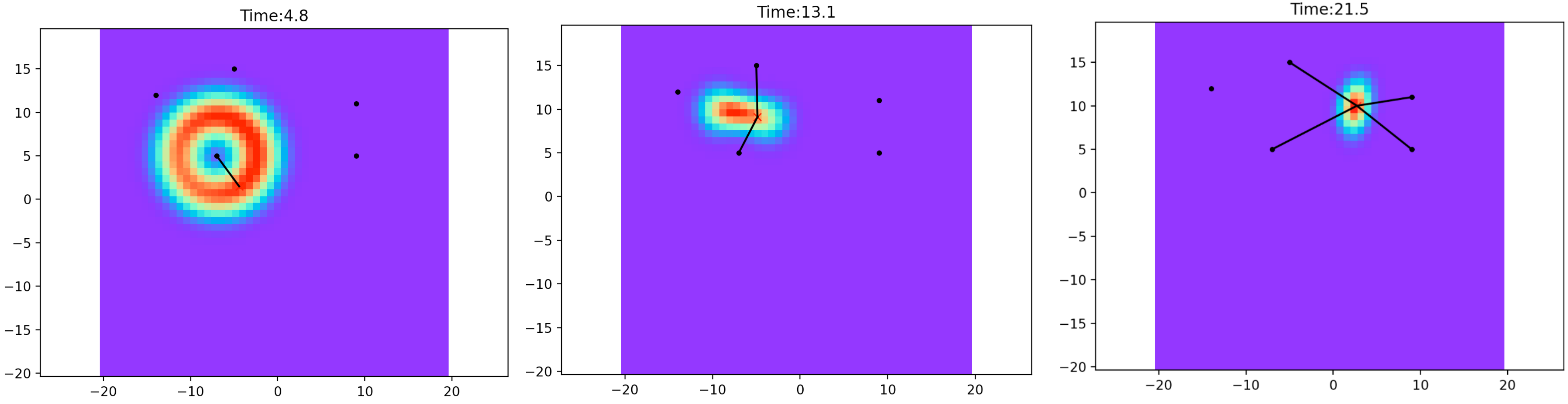


New Observation
o_t



Implementing Example

- ✓ Histogram Filter localization in different time stamps



Practice:

- ✓ Try with different sensor range and more obstacles first
- ✓ Extend Given Example code to construct a map of environment.

python robotics:

https://atsushisakai.github.io/PythonRobotics/modules/localization/histogram_filter_localization/histogram_filter_localization.html

https://github.com/AtsushiSakai/PythonRobotics/blob/master/Localization/histogram_filter/histogram_filter.py

Summary

- In this chapter we discussed
  - ✓ Nonparametric Filters
  - ✓ Decomposition Techniques
  - ✓ Histogram Filter for localization