University of Applied Sciences
HOCHSCHULE EMDEN·LEER

Mobile Robotics

Prof. Dr.-Ing. Gavin Kane

Mapping

Why Mapping?

- Learning maps is one of the fundamental problems in mobile robotics
- Maps allow robots to efficiently carry out their tasks, allow localization ...
- Successful robot systems rely on maps for localization, path planning, activity planning etc.

Mapping

The General Problem of Mapping

Formally, mapping involves, given the sensor data

$$d = \{u_1, z_1, u_2, z_2, \cdots u_t, z_t\} \tag{1}$$

to calculate the most likely map

$$m^* = argmax_m P(m|d) \tag{2}$$

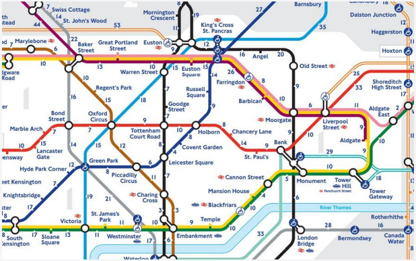
Mapping

Mapping as a Chicken and the Egg Problem

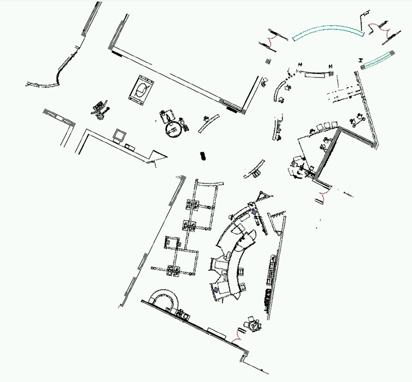
- So far we learned how to estimate the pose of the vehicle given the data and markers (known points on a map)
- Mapping, however, involves simultaneously estimating the pose of the vehicle and updating our map with markers we can find
- The general problem is therefore denoted as the simultaneous localization and mapping problem (SLAM)
- Throughout this section we will describe how to calculate a map given we know the pose of the vehicle

Mapping

Features / Landmark Map



Volumetric Map



Mapping

Problems in Mapping

- Sensor interpretation
  - How do we extract relevant information from raw sensor data?
  - How do we represent and integrate this information over time?
- Robot locations have to be estimated
  - How can we identify that we are at a previously visited place?
  - This problem is the so-called data association problem.

Mapping

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 $m^{[xy]}$ is independent

$$
\begin{aligned}
Bel(m_t) &= P(m_t | u_1, z_1, \cdots, u_t, z_t) && (3) \\
&= \prod_{x,y} Bel(m_t^{[xy]}) && (4)
\end{aligned}
$$

  - Robot positions are known!

Mapping

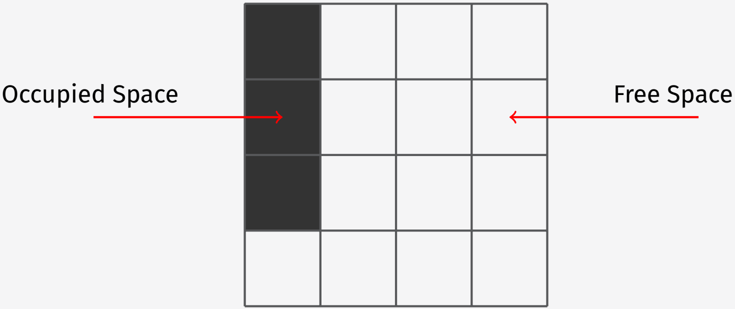
Grid Maps

- We discretize the world into cells
- The grid structure is rigid
- Each cell is assumed to be occupied or free
- It is a non-parametric model
- It requires substantial memory resources
- It does not rely on a feature detector

Mapping

Assumption 1

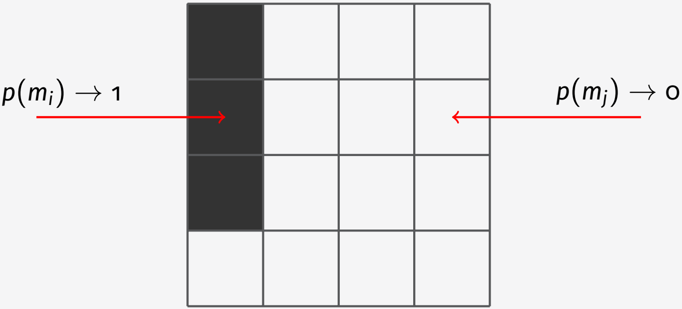
- The area that compromises a cell is either completely free, or occupied



Mapping

Representation

- Each cell is a binary random variable that models the occupancy



Mapping

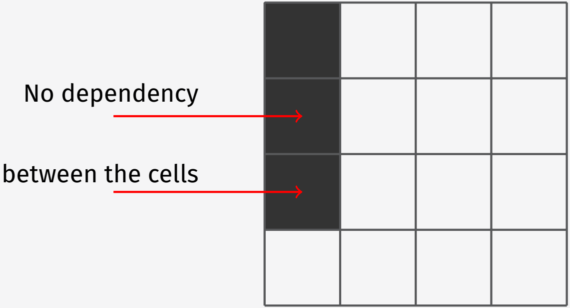
Representation

- Each cell is a binary random variable that models the occupancy
- Cell is occupied $p(m_i) \to 1$
- Cell is not occupied $p(m_i) \to 0$
- No information $p(m_i) \to 0.5$
- The environment is assumed to be static

Mapping

Assumption 2

- The cells (the random variables) are independent of each other



Mapping

Representation

- The probability distribution of the map is given by the product of the probability distributions of the individual cells

$$p(m) = \prod_i p(m_i)$$

$\uparrow$ map — $\uparrow$ cell

Matrix of Probabilities

Indepentant cell probabilities

Mapping

Estimating a Map from Data

- Given sensor data $z_{1:t}$and the poses of the sensor $x_{1:t}$, estimate the map

$$p(m|z_{1:t}, x_{1:t}) \quad = \quad \prod_i p(m_i|z_{1:t}, x_{1:t})$$

binary random variable

Binary Bayes filter

(for a static state)

Mapping

Static State Binary Bayes Filter

$$p(m_i | z_{1:t}, x_{1:t}) \overset{\text{Bayes Rule}}{=} \frac{p(z_t | m_i, z_{1:t-1}, x_{1:t}) \cdot p(m_i | z_{1:t-1}, x_{1:t})}{p(z_t | z_{1:t-1}, x_{1:t})}$$

Mapping

Static State Binary Bayes Filter

$$p(m_i | z_{1:t}, x_{1:t}) \overset{\text{Bayes Rule}}{=} \frac{p(z_t | m_i, z_{1:t-1}, x_{1:t}) \cdot p(m_i | z_{1:t-1}, x_{1:t})}{p(z_t | z_{1:t-1}, x_{1:t})}$$

$$\overset{\text{Markov}}{=} \frac{p(z_t | m_i, x_{1:t}) \cdot p(m_i | z_{1:t-1}, x_{1:t-1})}{p(z_t | x_t)}$$

University of Applied Sciences
HOCHSCHULE EMDEN·LEER

Mapping

Static State Binary Bayes Filter

$$p(m_i|z_{1:t}, x_{1:t}) \overset{\text{Bayes Rule}}{=} \frac{p(z_t|m_i, z_{1:t-1}, x_{1:t}) \cdot p(m_i|z_{1:t-1}, x_{1:t})}{p(z_t|z_{1:t-1}, x_{1:t})}$$

$$\overset{\text{Markov}}{=} \frac{p(z_t|m_i, x_{1:t}) \cdot p(m_i|z_{1:t-1}, x_{1:t-1})}{p(z_t|x_t)}$$

$$p(z_t|m_i, x_{1:t}) \overset{\text{Bayes Rule}}{=} \frac{p(m_t|z_t, x_t) p(z_t|x_t)}{p(m_i|x_t)}$$

Mapping

Static State Binary Bayes Filter

$$p(m_i|z_{1:t},x_{1:t}) \overset{\text{Bayes Rule}}{=} \frac{p(z_t|m_i,z_{1:t-1},x_{1:t})\cdot p(m_i|z_{1:t-1},x_{1:t})}{p(z_t|z_{1:t-1},x_{1:t})}$$

$$\overset{\text{Markov}}{=} \frac{p(z_t|m_i,x_{1:t})\cdot p(m_i|z_{1:t-1},x_{1:t-1})}{p(z_t|z_{1:t-1},x_{1:t})}$$

$$\overset{\text{Bayes Rule}}{=} \frac{p(m_t|z_t,x_t)p(z_t|x_t)p(m_i|z_{1:t-1},x_{1:t-1})}{p(m_i|x_t)p(z_t|x_t)}$$

Mapping

Static State Binary Bayes Filter

$$
\begin{aligned}
p(m_i|z_{1:t}, x_{1:t}) &\overset{\text{Bayes Rule}}{=} \frac{p(z_t|m_i, z_{1:t-1}, x_{1:t}) \cdot p(m_i|z_{1:t-1}, x_{1:t})}{p(z_t|z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(z_t|m_i, x_{1:t}) \cdot p(m_i|z_{1:t-1}, x_{1:t-1})}{p(z_t|z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Bayes Rule}}{=} \frac{p(m_t|z_t, x_t)p(z_t|x_t)p(m_i|z_{1:t-1}, x_{1:t-1})}{p(m_i|x_t)p(z_t|x_t)} \\
&\overset{\text{Bayes Rule}}{=} \frac{p(m_t|z_t, x_t)p(z_t|x_t)p(m_i|z_{1:t-1}, x_{1:t-1})}{p(m_i)p(z_t|x_t)}
\end{aligned}
$$

Mapping

Static State Binary Bayes Filter

$$
\begin{aligned}
p(m_i | z_{1:t}, x_{1:t}) &\overset{\text{Bayes Rule}}{=} \frac{p(z_t | m_i, z_{1:t-1}, x_{1:t}) \cdot p(m_i | z_{1:t-1}, x_{1:t})}{p(z_t | z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(z_t | m_i, x_{1:t}) \cdot p(m_i | z_{1:t-1}, x_{1:t-1})}{p(z_t | z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Bayes Rule}}{=} \frac{p(m_t | z_t, x_t) p(z_t | x_t) p(m_i | z_{1:t-1}, x_{1:t-1})}{p(m_i | x_t) p(z_t | x_t)} \\
&\overset{\text{Bayes Rule}}{=} \frac{p(m_t | z_t, x_t) p(z_t | x_t) p(m_i | z_{1:t-1}, x_{1:t-1})}{p(m_i) p(z_t | x_t)}
\end{aligned}
$$

Do exactly the same for the opposite Event:

$$
p(\neg m_i | z_{1:t}, x_{1:t}) = \frac{p(\neg m_t | z_t, x_t) p(z_t | x_t) p(\neg m_i | z_{1:t-1}, x_{1:t-1})}{p(\neg m_i) p(z_t | x_t)}
$$

Mapping

Static State Binary Bayes Filter

- By computing the ratio of both probabilities, we obtain:

$$\frac{p(m_i|z_{1:t},x_{1:t})}{p(\neg m_i|z_{1:t},x_{1:t})} = \frac{\frac{p(m_t|z_t,x_t)p(z_t|x_t)p(m_i|z_{1:t-1},x_{1:t-1})}{p(m_i)p(z_t|x_t)}}{\frac{p(\neg m_t|z_t,x_t)p(z_t|x_t)p(\neg m_i|z_{1:t-1},x_{1:t-1})}{p(\neg m_i)p(z_t|x_t)}}$$

University of Applied Sciences
HOCHSCHULE EMDEN·LEER

Mapping

Static State Binary Bayes Filter

- By computing the ratio of both probabilities, we obtain:

$$\frac{p(m_i|z_{1:t},x_{1:t})}{p(\neg m_i|z_{1:t},x_{1:t})} = \frac{\dfrac{p(m_t|z_t,x_t)\cancel{p(z_t|x_t)}p(m_i|z_{1:t-1},x_{1:t-1})}{p(m_i)\cancel{p(z_t|x_t)}}}{\dfrac{p(\neg m_t|z_t,x_t)\cancel{p(z_t|x_t)}p(\neg m_i|z_{1:t-1},x_{1:t-1})}{p(\neg m_i)\cancel{p(z_t|x_t)}}}$$

$$\frac{p(m_i|z_{1:t},x_{1:t})}{1-p(m_i|z_{1:t},x_{1:t})} = \frac{p(m_t|z_t,x_t)p(m_i|z_{1:t-1},x_{1:t-1})p(\neg m_i)}{p(\neg m_t|z_t,x_t)p(\neg m_i|z_{1:t-1},x_{1:t-1})p(m_i)}$$

$$= \underbrace{\frac{p(m_t|z_t,x_t)}{1-p(m_t|z_t,x_t)}}_{\text{uses } z_t}\underbrace{\frac{p(m_i|z_{1:t-1},x_{1:t-1})}{1-p(m_i|z_{1:t-1},x_{1:t-1})}}_{\text{recursive Term}}\underbrace{\frac{1-p(m_i)}{p(m_i)}}_{\text{prior}}$$

Mapping

Updating Occupancy Grid Map

- Recursive rule:

$$\frac{p(m_i|z_{1:t},x_{1:t})}{1-p(m_i|z_{1:t},x_{1:t})} = \underbrace{\frac{p(m_t|z_t,x_t)}{1-p(m_t|z_t,x_t)}}_{\text{uses } z_t} \underbrace{\frac{p(m_i|z_{1:t-1},x_{1:t-1})}{1-p(m_i|z_{1:t-1},x_{1:t-1})}}_{\text{recursive Term}} \underbrace{\frac{1-p(m_i)}{p(m_i)}}_{\text{prior}}$$

- Often written as a belief using the inverse sensor model:

$$Bel(m_t^{[xy]}) = \left[1 + \frac{1-p(m_t^{[xy]}|z_t,u_{t-1})}{p(m_t^{[xy]}|z_t,u_{t-1})} \cdot \frac{p(m_t^{[xy]})}{1-p(m_t^{[xy]})} \cdot \frac{1-Bel(m_{t-1}^{[xy]})}{Bel(m_{t-1}^{[xy]})}\right]^{-1}$$

Mapping

Updating the Occupancy Grid Maps

Or using the logit representation where logit, also known as log*odds*, calculated the logarithm of the odds. i.e. the Ratio

$$logit(x) = log\left(\frac{x}{1-x}\right) \tag{6}$$

$$logodds(x) = log\left(\frac{P(x)}{1-P(x)}\right) \tag{7}$$

$$\overline{B}\left(m_t^{[xy]}\right) = logodds\left(m_t^{[xy]}\right) \tag{8}$$

and so:

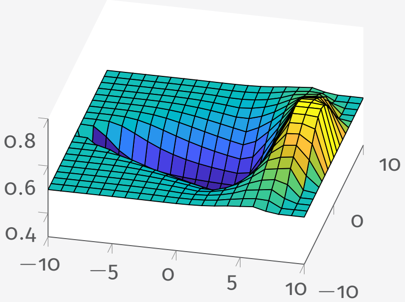
$$logodds(m_i|z_{1:t}, x_{1:t}) =$$

$$\underbrace{logodds(m_i|z_t, x_t)}_{\text{inverse Sensor Model}} + \underbrace{logodds(m_i|z_{1:t-1}, x_{1:t-1})}_{\text{recursive Term}} - \underbrace{logodds(m_i)}_{\text{Prior}}$$

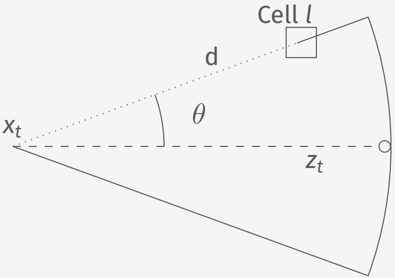
Mapping

Typical Sensor Model for Occupancy Grid Maps (Sonar)

Combination of a linear function and a Gaussian:



Key Parameters of the Model

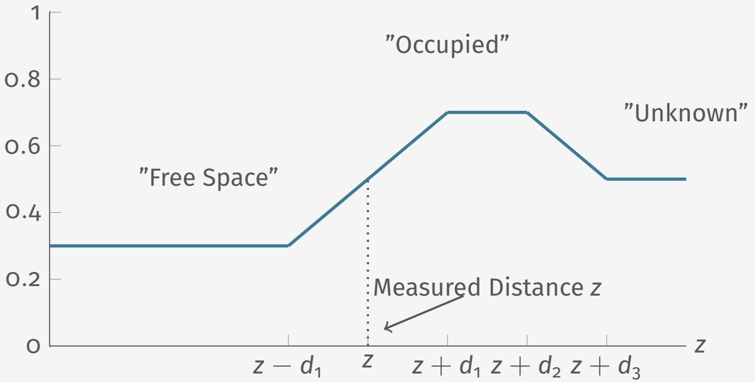


Linear in z_t

Gaussian in θ

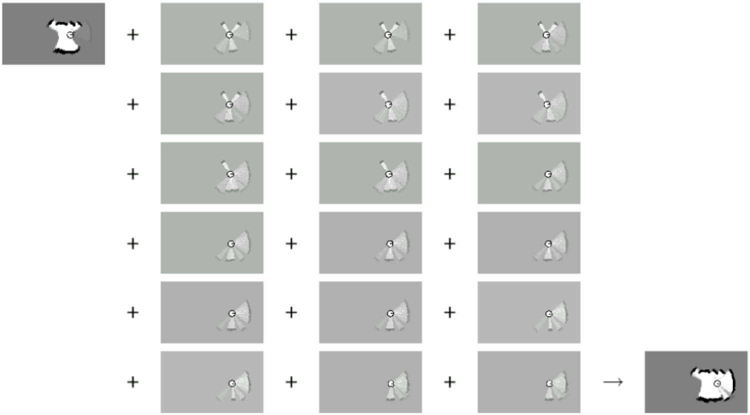
Mapping

Occupancy Probability Depending on the Measured Distance



Mapping

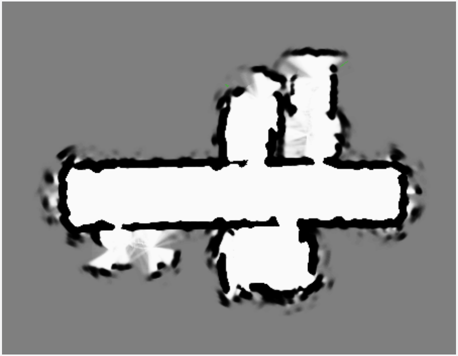
Incremental Updating of Occupancy Grids (Example)



Source: Uniu Freiburg, Wolfram

Mapping

Incremental Updating of Occupancy Grids (Example)

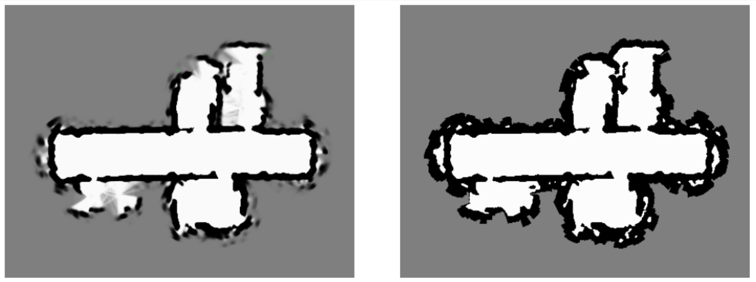


Source: Uni Freiburg, Wolfram

Mapping

Resulting Occupancy and Maximum Likelihood Map

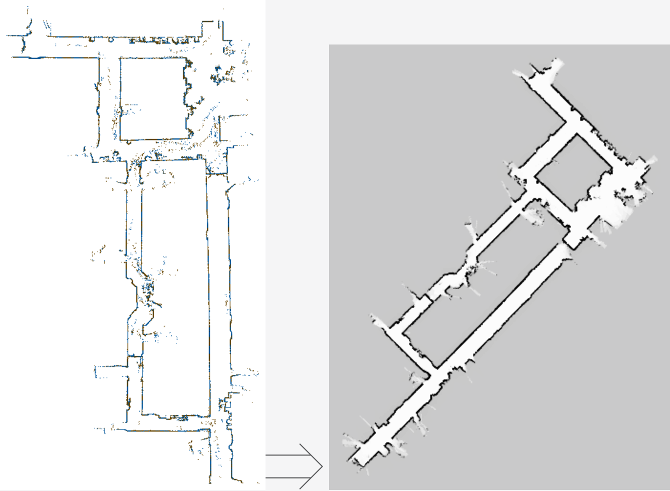
The maximum likelihood map is obtained by clipping the occupancy grid map at a threshold of 0.5



Source: Uni Freiburg, Wolfram

Mapping

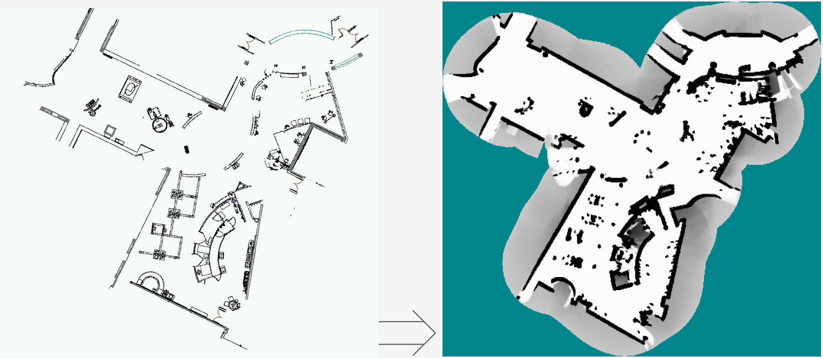
Occupancy Grids: From Scans to Maps (Laser)



Source: Uni Freiburg, Wolfram

Mapping

Tech Museum, San Jose



Source: Uni Freiburg, Wolfram

Mapping

Alternative: Counting Model

- For every cell count
  - hits(x,y): number of cases where a beam ended at <x,y>
  - misses(x,y): number of cases where a beam passed through <x,y>

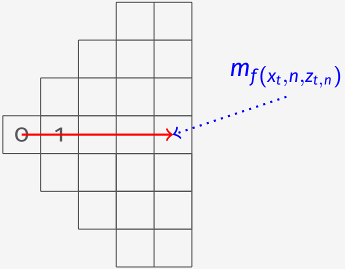
$$Bel\left(m^{[xy]}\right) = \frac{hits(x,y)}{hits(x,y) + misses(x,y)} \tag{9}$$

- Value of interest: P(reflects(x,y))

Mapping

The Measurement Model

- Pose at time t: x_t
- Beam n of scan at time t: $z_{t,n}$
- Maximum range reading: $\zeta_{t,n} = 1$
- Beam reflected by an object: $\zeta_{t,n} = 0$



$$p\left(z_{t,n} \mid x_t, m\right) = \begin{cases} \prod\limits_{k=0}^{z_{t,n-1}} \left(1 - m_{f(x_t,n,k)}\right) & \text{if } \zeta_{t,n} = 1 \\ m_{f(x_t,n,z_{t,n})} \prod\limits_{k=0}^{z_{t,n-1}} \left(1 - m_{f(x_t,n,k)}\right) & \text{if } \zeta_{t,n} = 0 \end{cases} \tag{10}$$

Mapping

Computing the Most Likely Map

- Compute values for m that maximize

$$m^* = argmax_m P\left(m|z_1, \cdots, z_t, x_1, \cdots x_t\right) \tag{11}$$

- Assuming a uniform prior probability for P(m), this is equivalent to maximizing (Bayes' rule)

$$m^* = argmax_m P\left(z_1, \cdots, z_t|m, x_1, \cdots x_t\right) \tag{12}$$

$$argmax_m \prod_{t=1}^{T} P\left(z_t|m, x_t\right) \tag{13}$$

$$argmax_m \sum_{t=1}^{T} lnP\left(z_t|m, x_t\right) \tag{14}$$

$$\tag{15}$$

Mapping

Computing the Most Likely Map

$$m^* = \operatorname{argmax}_m \sum_{j=1}^{J} \sum_{t=1}^{T} \sum_{n=1}^{N} I\left(f\left(x_t, n, z_{t,n}\right) = j\right) \cdot \left(1 - \zeta_{t,n}\right) \cdot \ln m_j$$

$$+ \sum_{k=0}^{z_{t,n}-1} I\left(f\left(x_t, n, k\right) = j\right) \cdot \ln\left(1 - m_j\right) \tag{16}$$

- Define:

$$\alpha_j = \sum_{t=1}^{T} \sum_{n=1}^{N} I\left(f\left(x_t, n, z_{t,n}\right) = j\right) \cdot \left(1 - \zeta_{t,n}\right) \tag{17}$$

$$\beta_j = \sum_{t=1}^{T} \sum_{n=1}^{N} \sum_{k=0}^{z_{t,n}-1} I\left(f\left(x_t, n, z_{t,n}\right) = j\right) \tag{18}$$

Mapping

Meaning of α und β

$$\alpha_j = \sum_{t=1}^{T} \sum_{n=1}^{N} I\left(f\left(x_t, n, z_{t,n}\right) = j\right) \cdot \left(1 - \zeta_{t,n}\right) \tag{19}$$

Corresponds to the number of times a beam that is not a maximum range beam ends in cell j (*hits(j)*)

$$\beta_j = \sum_{t=1}^{T} \sum_{n=1}^{N} \sum_{k=0}^{z_{t,n}-1} I\left(f\left(x_t, n, z_{t,n}\right) = j\right) \tag{20}$$

Corresponds to the number of times a beam traversed cell without ending in it (*misses(j)*)

Mapping

Computing the most likely map

Accordingly we get:

$$m^* = argmax_m \sum_{j=1}^{J} (\alpha_j ln m_j + \beta_j ln(1 - m_j)) \tag{21}$$

If we set

$$\frac{\delta}{\delta m_j} = \frac{\alpha_j}{m_j} - \frac{\beta_j}{1 - m_j} = 0 \tag{22}$$

We obtain

$$m_j = \frac{\alpha_j}{\alpha_j + \beta_j} \tag{23}$$

Computing the most likely map amounts to counting how often a cell has reflected a measurement and how often the cell was traversed by a beam.

Mapping

Difference between Occupancy Grid Maps and Counting

- The counting model determines how often a cell reflects a beam.
- The occupancy model represents whether or not a cell is occupied by an object.
- Although a cell might be occupied by an object, the reflection probability of this object might be very small.

Mapping

Example Occupancy Map

Example Reflection Map

Source: Uni Freiburg, Wolfram

Mapping

Example

- Out of 1000 beams only 60% are reflected from a cell and 40% intercept it without ending in it.
- Accordingly, the reflection probability will be 0.6.
- Suppose $p(occ|z) = 0.55$ when a beam ends in a cell and $p(occ|z) = 0.45$ when a beam traverses a cell without ending in it.
- Accordingly, after n measurements we will have

$$\left(\frac{0.55}{0.45}\right)^{n*0.6} \cdot \left(\frac{0.45}{0.55}\right)^{n*0.4} = \left(\frac{11}{9}\right)^{n*0.6} \cdot \left(\frac{11}{9}\right)^{-n*0.4} = \left(\frac{11}{9}\right)^{n*0.2} \tag{24}$$

- Whereas the reflection map yields a value of 0.6, the occupancy grid value converges to 1.

Mapping

Summary

- Occupancy grid maps are a popular approach to represent the environment given known poses.
- Each cell is considered independently from all others.
- Occupancy grids store the probability that the corresponding area in the environment is occupied.
- They can be learned efficiently using a probabilistic approach.
- Reflection maps are an alternative representation.
- They store in each cell the probability that a beam is reflected by this cell.
- The counting procedure underlying reflection maps yield the optimal map given the proposed sensor model.