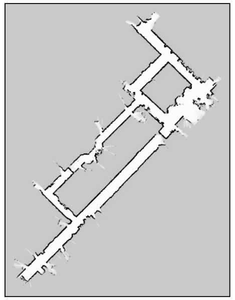
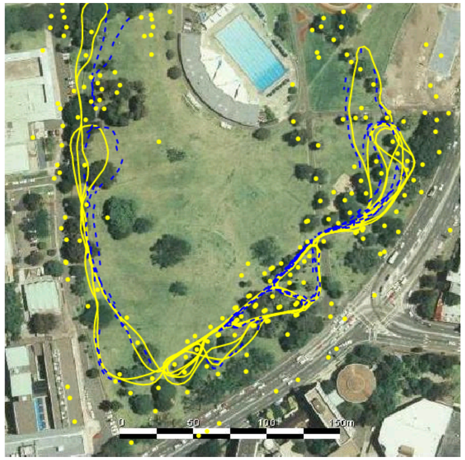
Robot Mapping

Grid Maps

Cyrill Stachniss

Features vs. Volumetric Maps





Courtesy by E. Nebot

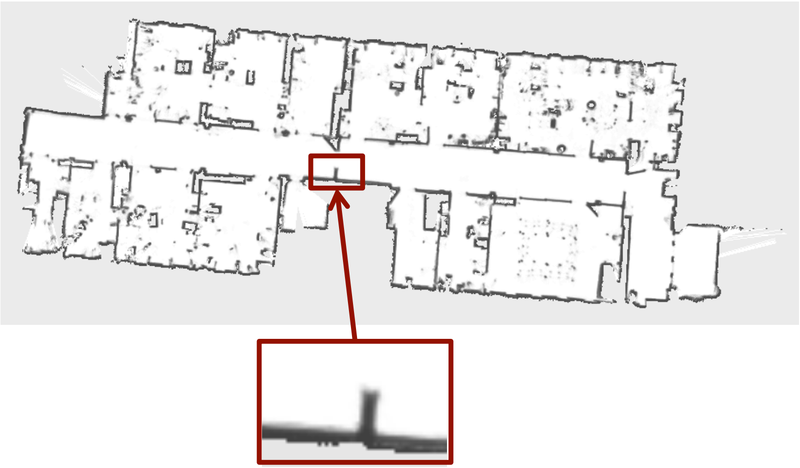
Features

- So far, we only used feature maps
- Natural choice for Kalman filter-based SLAM systems
- Compact representation
- Multiple feature observations improve the position estimate (EKF)

Grid Maps

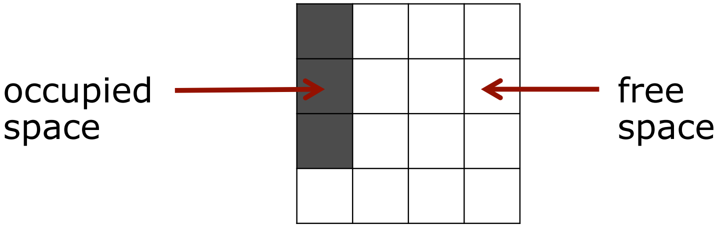
- Discretize the world into cells
- Grid structure is rigid
- Each cell is assumed to be occupied or free space
- Non-parametric model
- Require substantial memory resources
- Does not rely on a feature detector

Example



Assumption 1

- The area that corresponds to a cell is either completely free or occupied

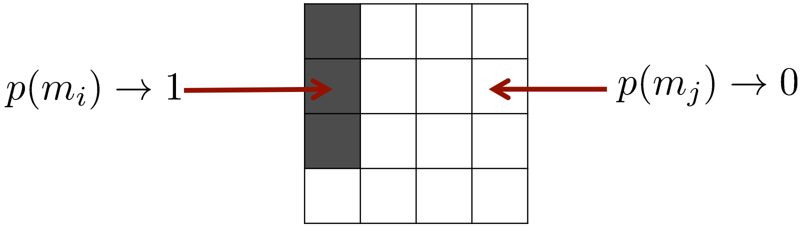


occupied space

free space

Representation

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



$p(m_i) \to 1$

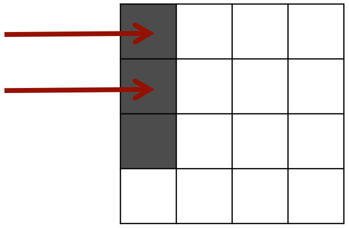
$p(m_j) \to 0$

Occupancy Probability

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

Assumption 2

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



no dependency
between the cells

Representation

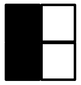
- The probability distribution of the map is given by the product over the cells

$$\underset{\substack{\uparrow \\ \text{map}}}{p(m)} = \prod_i \underset{\substack{\uparrow \\ \text{cell}}}{p(m_i)}$$

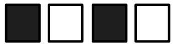
Representation

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

$$\underset{\uparrow}{p(m)} = \prod_i p(m_i)$$



example map
(4-dim vector)



4 individual cells

Estimating a Map From Data

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

$$p(m \mid z_{1:t}, x_{1:t}) = \prod_i \underset{\substack{\uparrow \\ \text{binary random variable}}}{p(m_i} \mid z_{1:t}, x_{1:t})$$

→ Binary Bayes filter
(for a static state)

Static State Binary Bayes Filter

$$p(m_i \mid z_{1:t}, x_{1:t}) \overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\; p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})}$$

Static State Binary Bayes Filter

$$\begin{aligned} p(m_i \mid z_{1:t}, x_{1:t}) &\overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\; p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\ &\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \end{aligned}$$

Static State Binary Bayes Filter

$$\begin{aligned} p(m_i \mid z_{1:t}, x_{1:t}) &\overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\; p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\ &\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \end{aligned}$$

$$p(z_t \mid m_i, x_t) \overset{\text{Bayes rule}}{=} \frac{p(m_i \mid z_t, x_t)\; p(z_t \mid x_t)}{p(m_i \mid x_t)}$$

Static State Binary Bayes Filter

$$\begin{aligned} p(m_i \mid z_{1:t}, x_{1:t}) &\overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\; p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\ &\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\ &\overset{\text{Bayes rule}}{=} \frac{p(m_i \mid z_t, x_t)\; p(z_t \mid x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i \mid x_t)\; p(z_t \mid z_{1:t-1}, x_{1:t})} \end{aligned}$$

Static State Binary Bayes Filter

$$
\begin{aligned}
p(m_i \mid z_{1:t}, x_{1:t}) &\overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\; p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Bayes rule}}{=} \frac{p(m_i \mid z_t, x_t)\; p(z_t \mid x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i \mid x_t)\; p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(m_i \mid z_t, x_t)\; p(z_t \mid x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i)\; p(z_t \mid z_{1:t-1}, x_{1:t})}
\end{aligned}
$$

Static State Binary Bayes Filter

$$
\begin{aligned}
p(m_i \mid z_{1:t}, x_{1:t}) &\overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\; p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Bayes rule}}{=} \frac{p(m_i \mid z_t, x_t)\; p(z_t \mid x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i \mid x_t)\; p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(m_i \mid z_t, x_t)\; p(z_t \mid x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i)\; p(z_t \mid z_{1:t-1}, x_{1:t})}
\end{aligned}
$$

Do exactly the same for the opposite event:

$$
p(\neg m_i \mid z_{1:t}, x_{1:t}) \overset{\text{the same}}{=} \frac{p(\neg m_i \mid z_t, x_t)\; p(z_t \mid x_t)\; p(\neg m_i \mid z_{1:t-1}, x_{1:t-1})}{p(\neg m_i)\; p(z_t \mid z_{1:t-1}, x_{1:t})}
$$

Static State Binary Bayes Filter

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

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

Static State Binary Bayes Filter

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

$$
\begin{aligned}
&\frac{p(m_i \mid z_{1:t}, x_{1:t})}{p(\neg m_i \mid z_{1:t}, x_{1:t})} \\
&= \frac{p(m_i \mid z_t, x_t)\; p(m_i \mid z_{1:t-1}, x_{1:t-1})\; p(\neg m_i)}{p(\neg m_i \mid z_t, x_t)\; p(\neg m_i \mid z_{1:t-1}, x_{1:t-1})\; p(m_i)} \\
&= \frac{p(m_i \mid z_t, x_t)}{1 - p(m_i \mid z_t, x_t)} \frac{p(m_i \mid z_{1:t-1}, x_{1:t-1})}{1 - p(m_i \mid z_{1:t-1}, x_{1:t-1})} \frac{1 - p(m_i)}{p(m_i)}
\end{aligned}
$$

Static State Binary Bayes Filter

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

$$\frac{p(m_i \mid z_{1:t}, x_{1:t})}{1 - p(m_i \mid z_{1:t}, x_{1:t})}$$

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

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

Log Odds Notation

- Log odds ratio is defined as

$$l(x) \quad = \quad \log \frac{p(x)}{1 - p(x)}$$

- and with the ability to retrieve $p(x)$

$$p(x) \quad = \quad 1 - \frac{1}{1 + \exp l(x)}$$

Occupancy Mapping in Log Odds Form

- The product turns into a sum

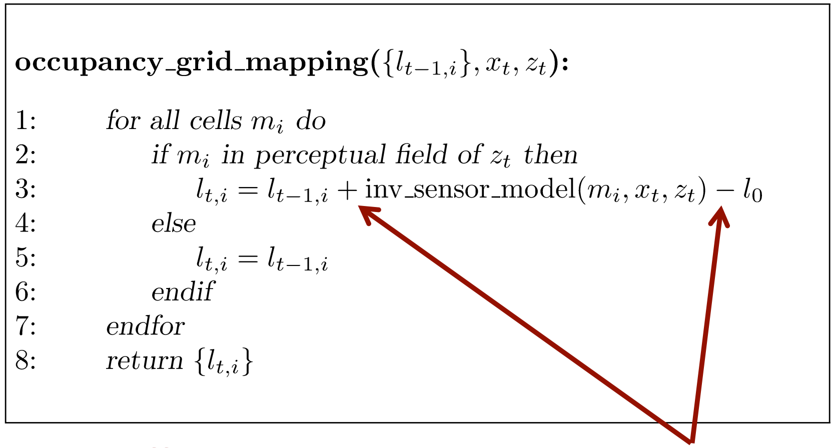
$$l(m_i \mid z_{1:t}, x_{1:t})$$

$$= \underbrace{l(m_i \mid z_t, x_t)}_{\text{inverse sensor model}} + \underbrace{l(m_i \mid z_{1:t-1}, x_{1:t-1})}_{\text{recursive term}} - \underbrace{l(m_i)}_{\text{prior}}$$

- or in short

$$l_{t,i} \quad = \quad \text{inv\_sensor\_model}(m_i, x_t, z_t) + l_{t-1,i} - l_0$$

Occupancy Mapping Algorithm



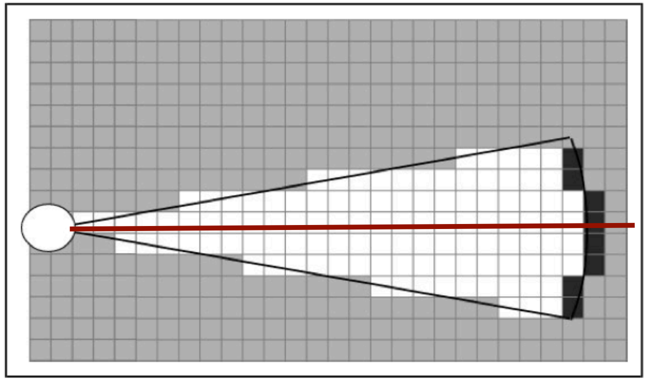
```
occupancy_grid_mapping($\{l_{t-1,i}\}, x_t, z_t$):

    for all cells $m_i$ do
        if $m_i$ in perceptual field of $z_t$ then
            $l_{t,i} = l_{t-1,i} + \text{inv\_sensor\_model}(m_i, x_t, z_t) - l_0$
        else
            $l_{t,i} = l_{t-1,i}$
        endif
    endfor
    return $\{l_{t,i}\}$
```

highly efficient, only requires to compute sums

Occupancy Grid Mapping

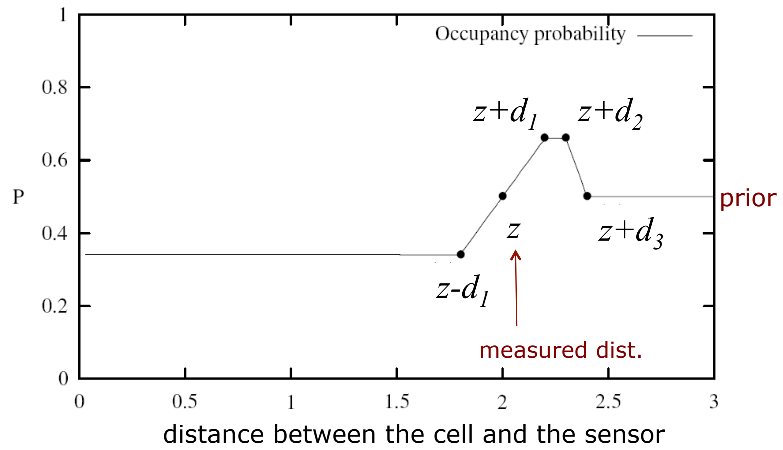
- Developed in the mid 80'ies by Moravec and Elfes
- Originally developed for noisy sonars
- Also called "mapping with know poses"

Inverse Sensor Model for Sonars Range Sensors

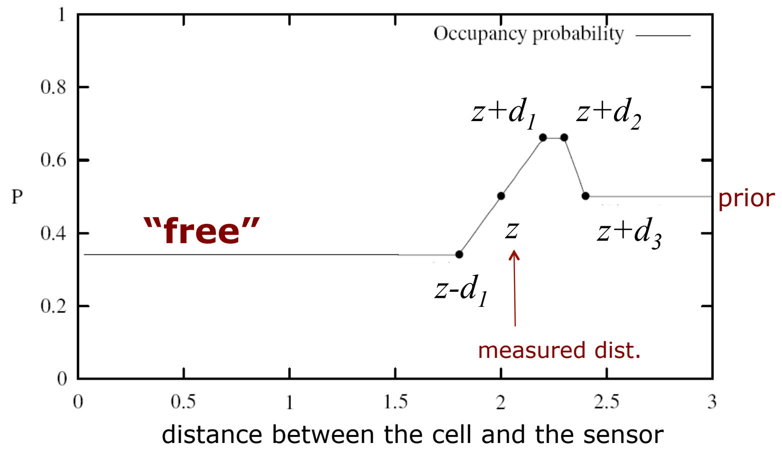


In the following, consider the cells along the optical axis (red line)

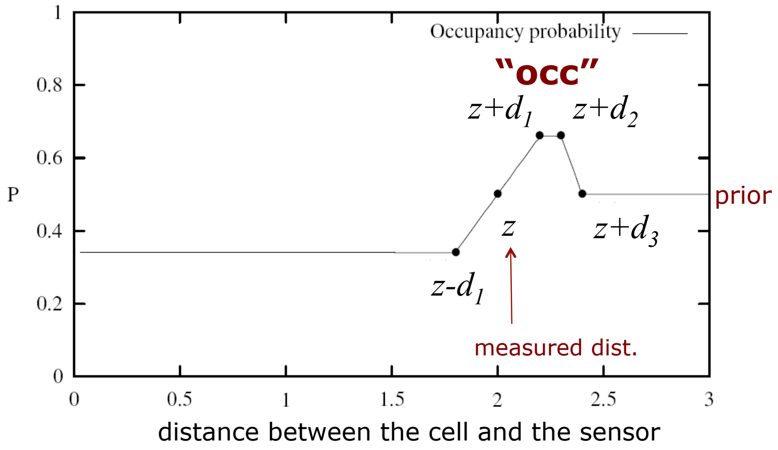
Occupancy Value Depending on the Measured Distance



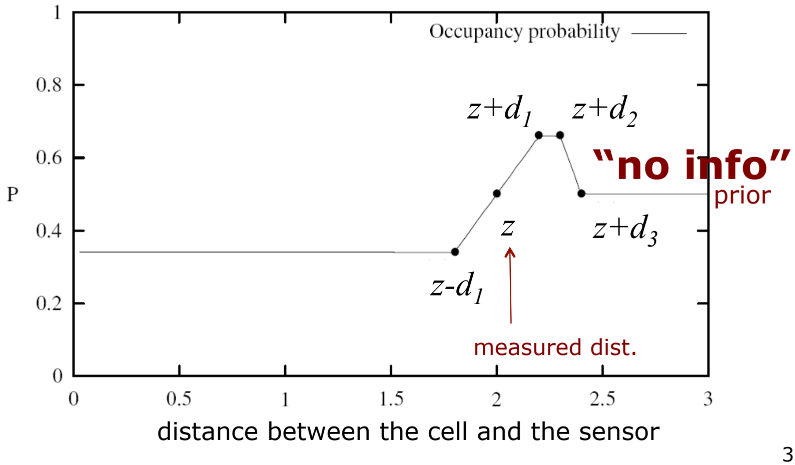
Occupancy Value Depending on the Measured Distance



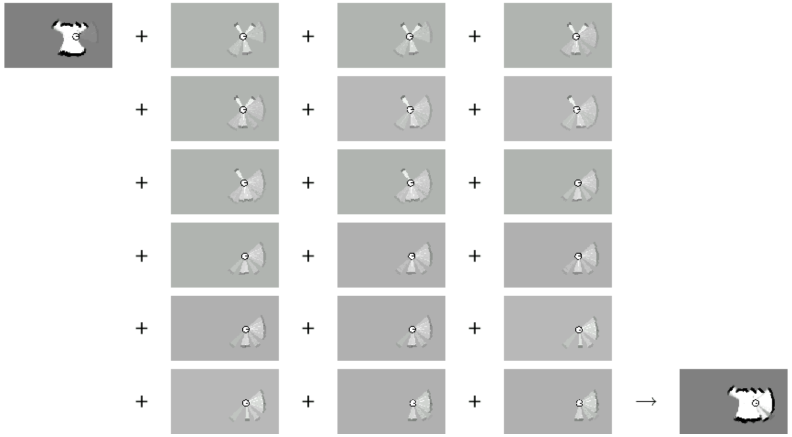
Occupancy Value Depending on the Measured Distance



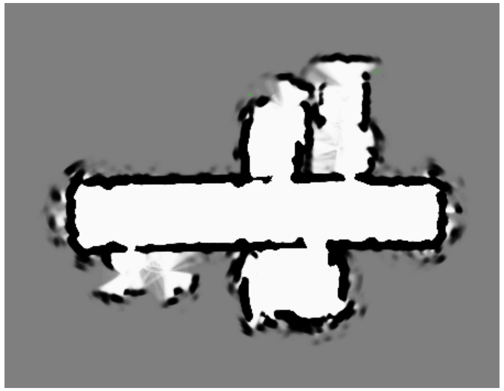
Occupancy Value Depending on the Measured Distance



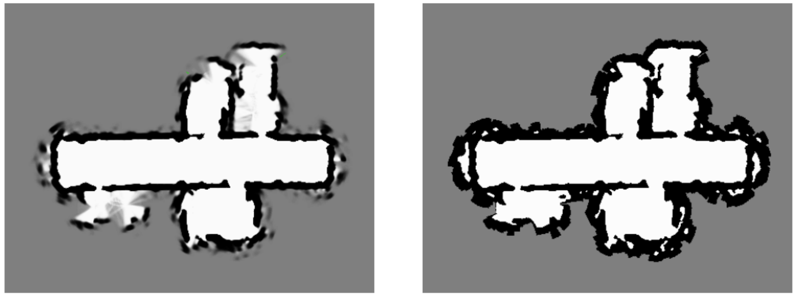
Example: Incremental Updating of Occupancy Grids



Resulting Map Obtained with Ultrasound Sensors

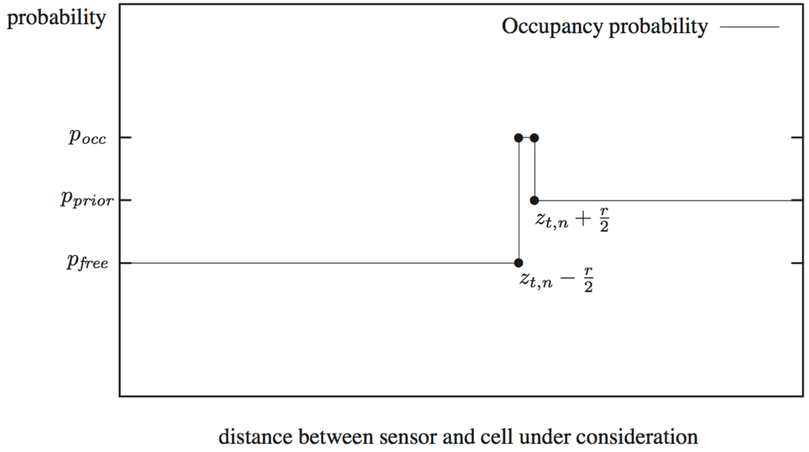


Resulting Occupancy and Maximum Likelihood Map

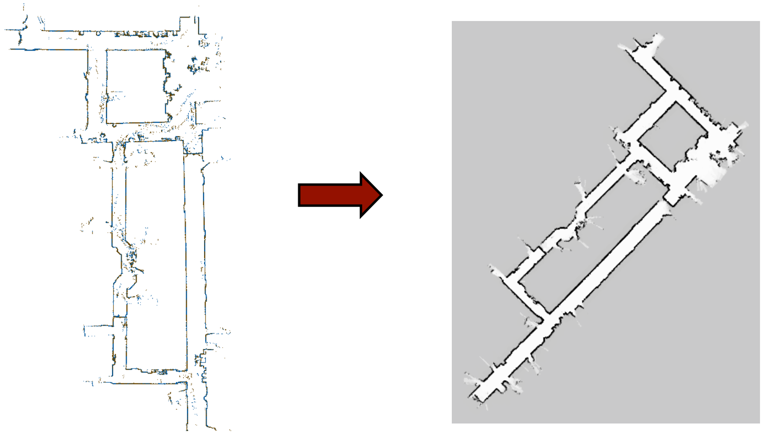


The maximum likelihood map is obtained by rounding the probability for each cell to 0 or 1.

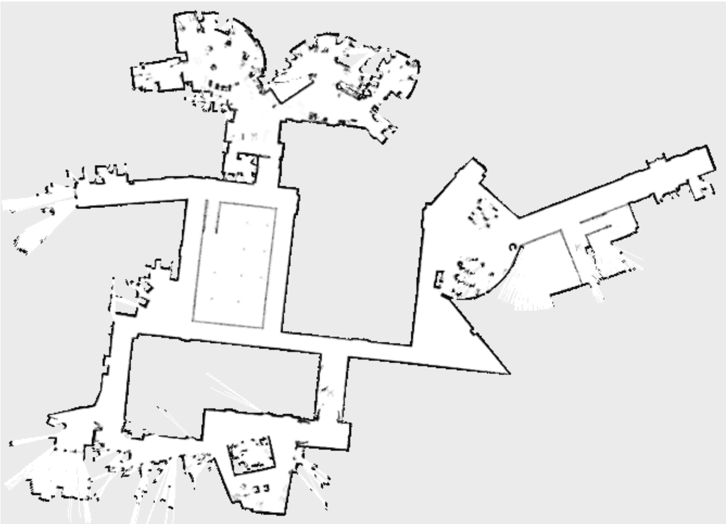
Inverse Sensor Model for Laser Range Finders



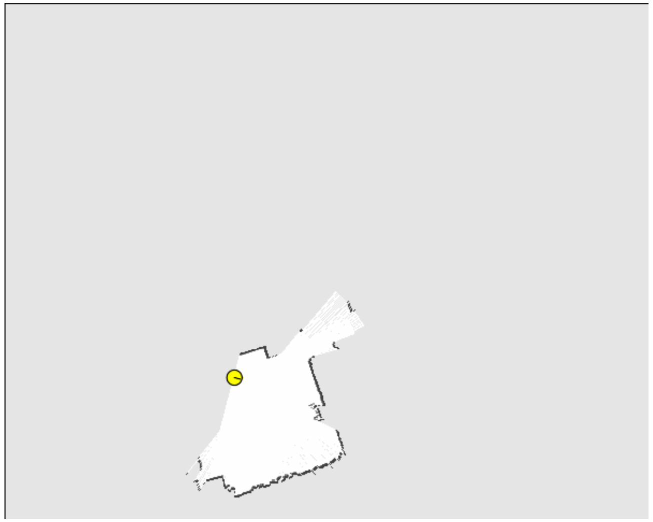
Occupancy Grids From Laser Scans to Maps



Example: MIT CSAIL 3rd Floor



Uni Freiburg Building 106



Summary

- Occupancy grid maps discretize the space into independent cells
- Each cell is a binary random variable estimating if the cell is occupied
- Static state binary Bayes filter per cell
- Mapping with known poses is easy
- Log odds model is fast to compute
- No need for predefined features

Literature

Static state binary Bayes filter

- Thrun et al.: “Probabilistic Robotics”, Chapter 4.2

Occupancy Grid Mapping

- Thrun et al.: “Probabilistic Robotics”, Chapter 9.1+9.2