UTS:CAS
CENTRE FOR AUTONOMOUS SYSTEMS

49274 ADVANCED ROBOTICS MAPPING

TERESA VIDAL CALLEJA

UNIVERSITY OF TECHNOLOGY SYDNEY

MAPPING FOR AUTONOMOUS SYSTEMS

A map is a spatial model of a robot's environment

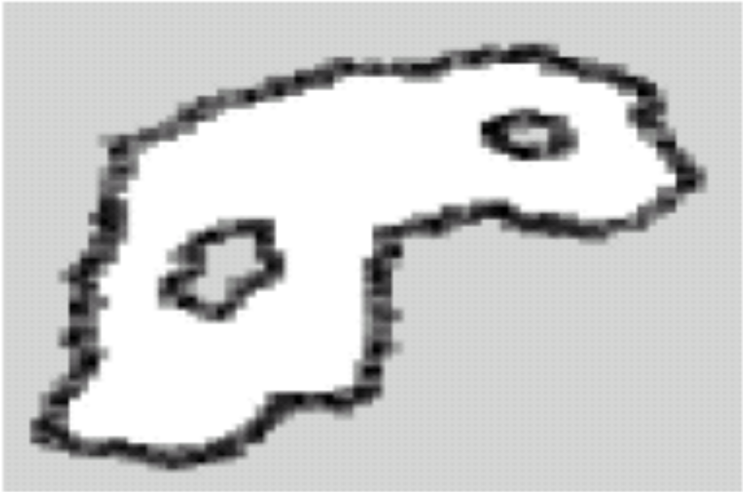
How do robots understand the world?

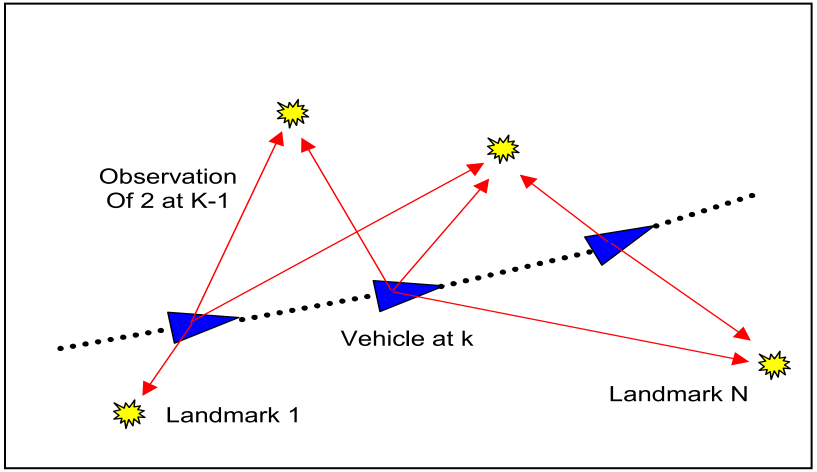
Environment representations

- Metric
- Topological
- Semantic

METRIC MAPS

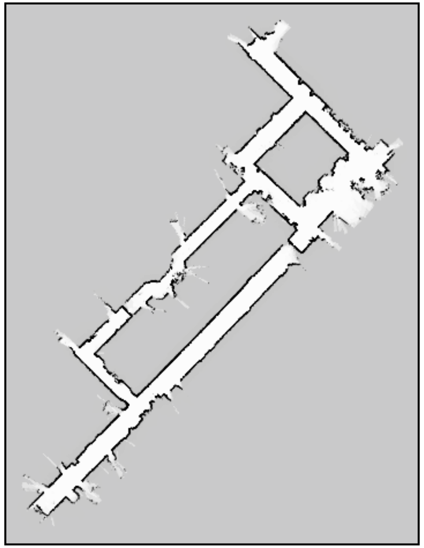
- Occupancy Grids – good for navigation
- Feature Maps --- good for localization

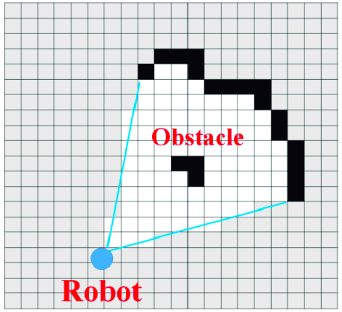




OCCUPANCY GRIDS

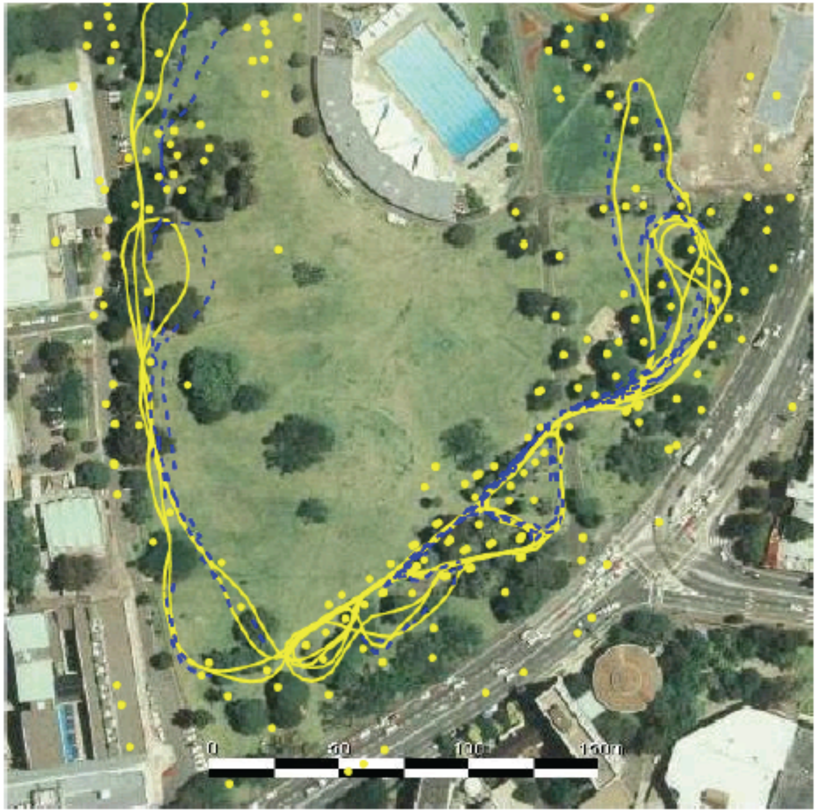
- Discretise the world into cells (2D) or voxels (3D)
- Grid structure is rigid
- Each cell is assumed to be occupied or free space
- Non-parametric model
- Require substantial memory resources
- Does not rely on a feature detector





FEATURE MAPS

- Landmark Based
- Kalman filter or Optimisation based systems
- Compact/parametric representation
- Multiple feature observations to improve position estimates



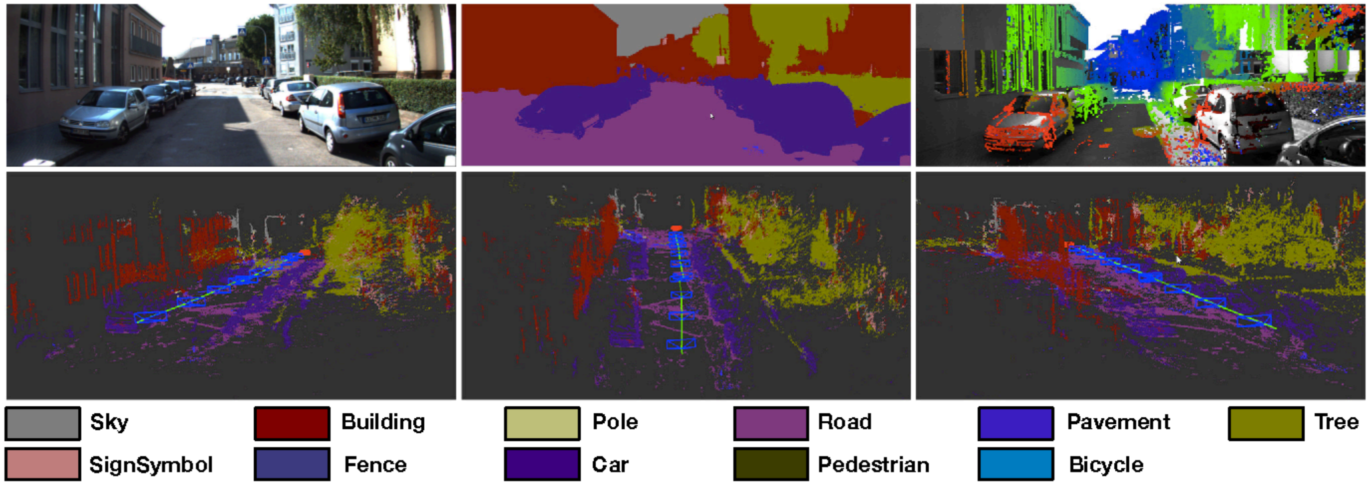
Courtesy by E. Nebot

TOPOLOGICAL MAPS

- Suitable for special environments
- Place descriptions
- Path descriptions

SEMANTIC MAPS

- Place descriptions
- High level planning
- Human-Robot interaction



MAPPING

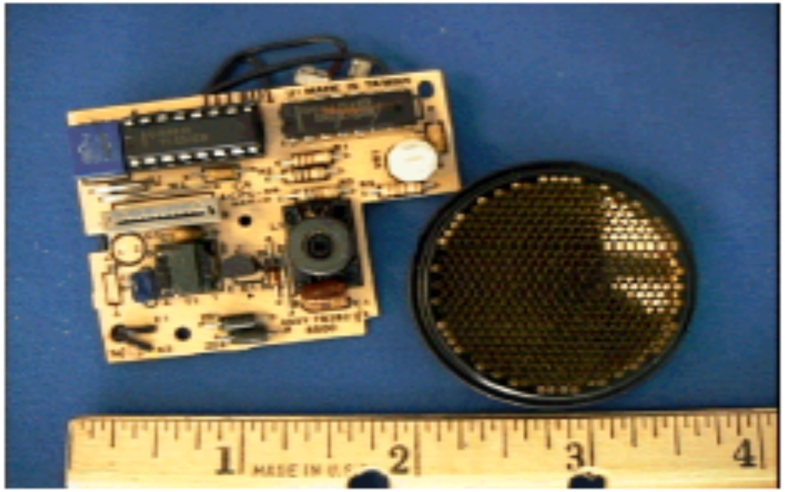
Why is it challenging:

- Perception problem: Relying on sensors to understand the world
- Noisy sensors
- Local coordinates to Global coordinates
- Motion involved
- Can change over time?

OCCUPANCY GRIDS MAP

Robot location is known all the time

- Developed in the mide 80'ies by Moravec and Elfes
- Originally developed for noisy sensors (sonar)

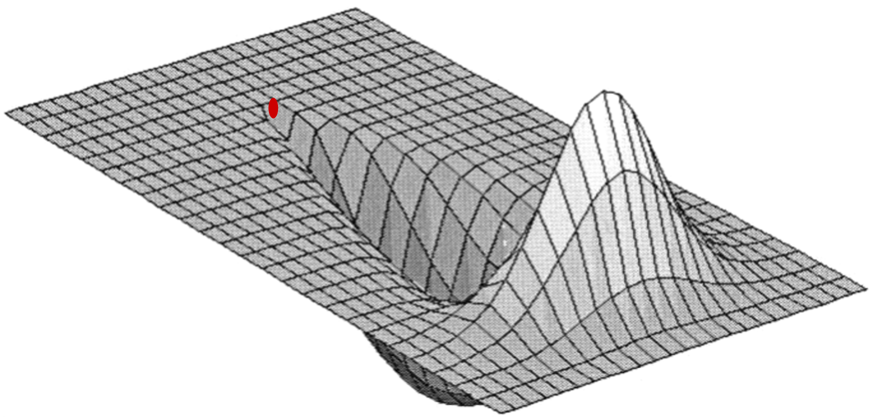


- Also called mapping with known poses

OCCUPANCY MAPS

Occupancy Probability for cells in Grid Maps

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



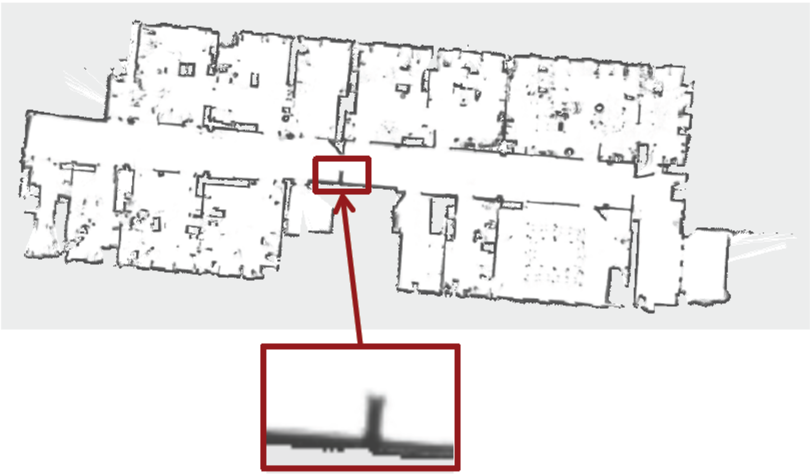
OCCUPANCY MAPPING

Estimating a Map from the Data

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

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

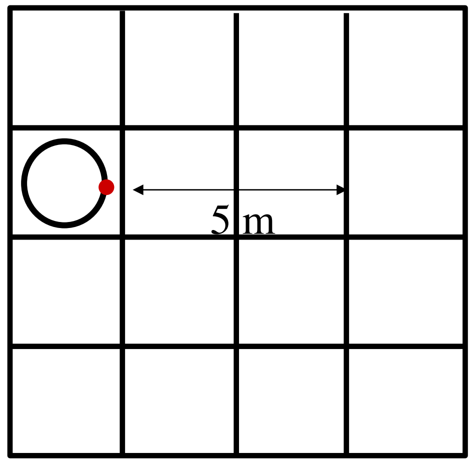
binary random variable



→ Binary Bayes filter (for a static state)

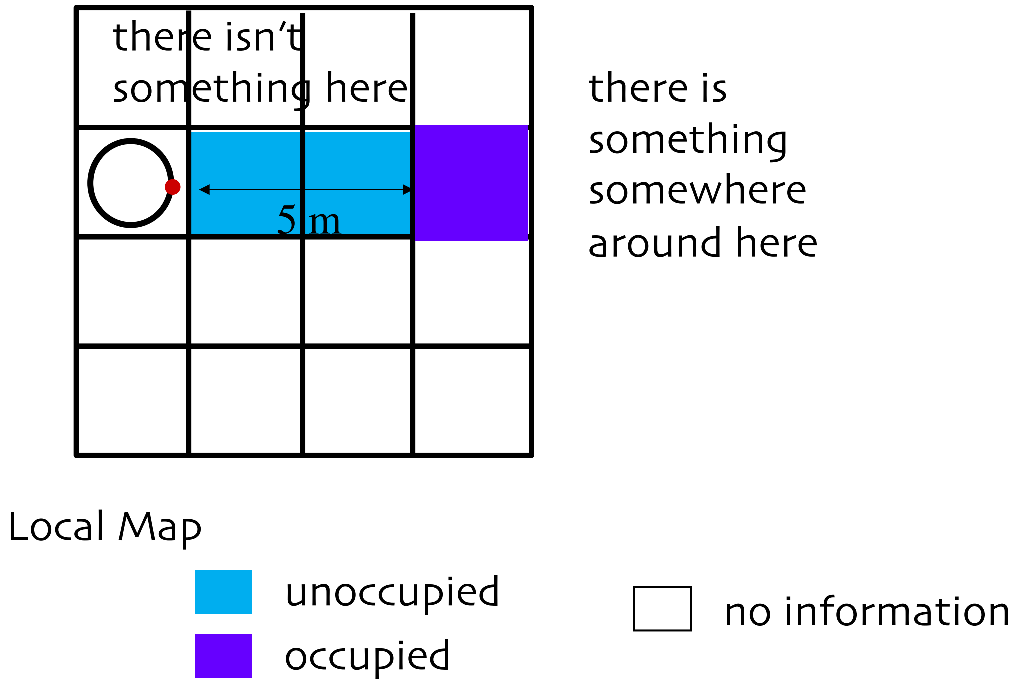
OCCUPANCY GRIDS MAP

What can we tell if this range sensor reads 5 m?



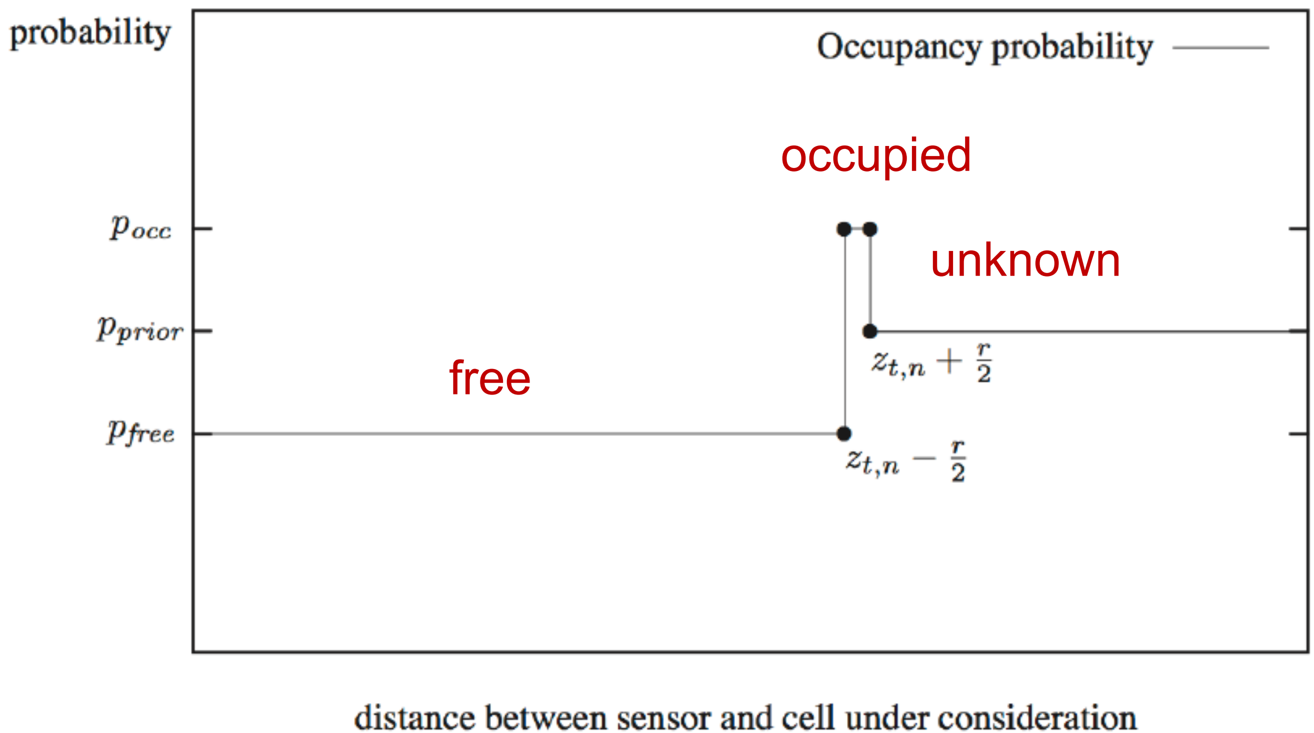
OCCUPANCY GRIDS MAP

What can we tell if this range sensor reads 5 m?



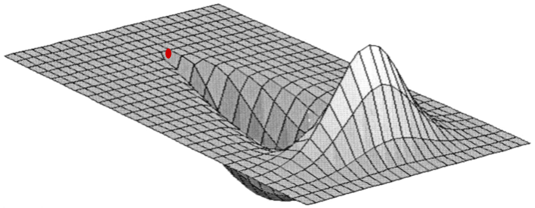
OCCUPANCY VALUE DEPENDING ON THE MEASURED DISTANCE

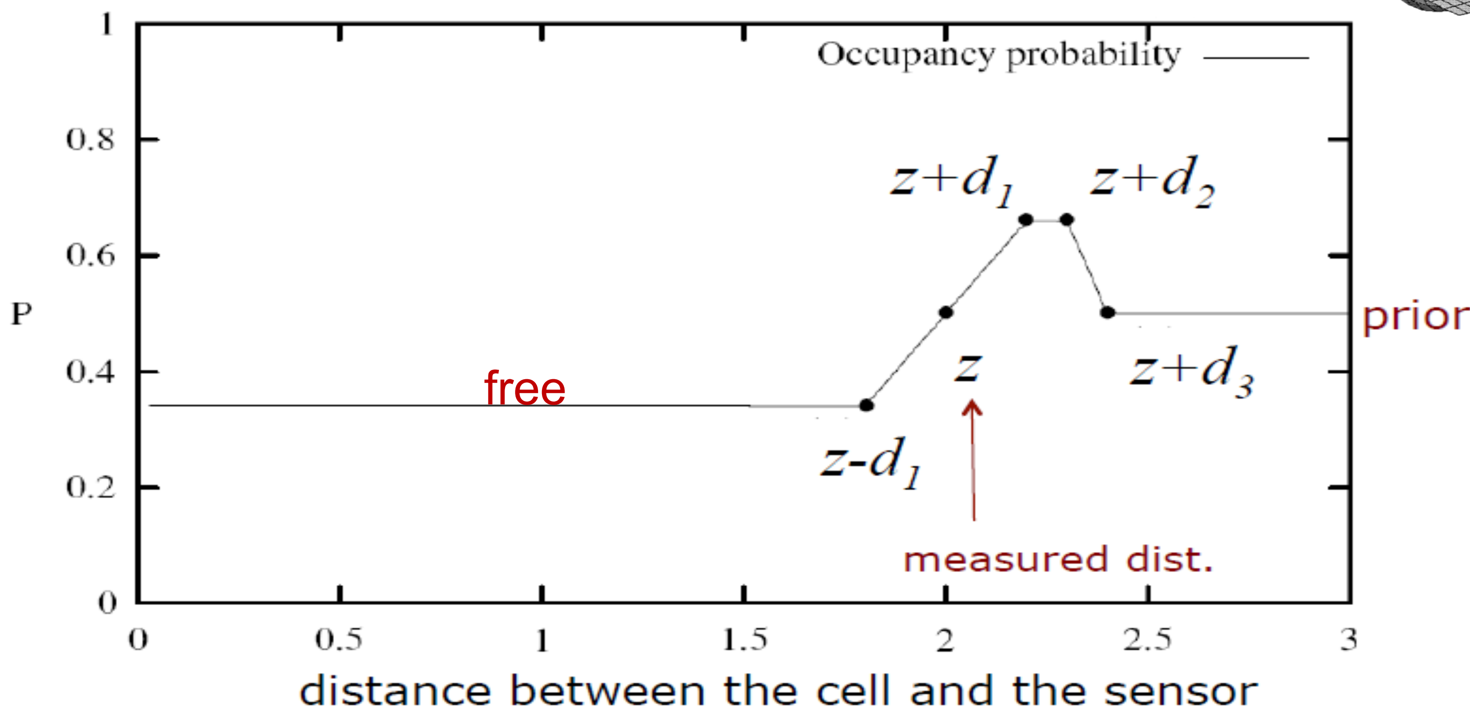
Inverse sensor model for a lidar



OCCUPANCY VALUE DEPENDING ON THE MEASURED DISTANCE

Inverse sensor model for a sonar

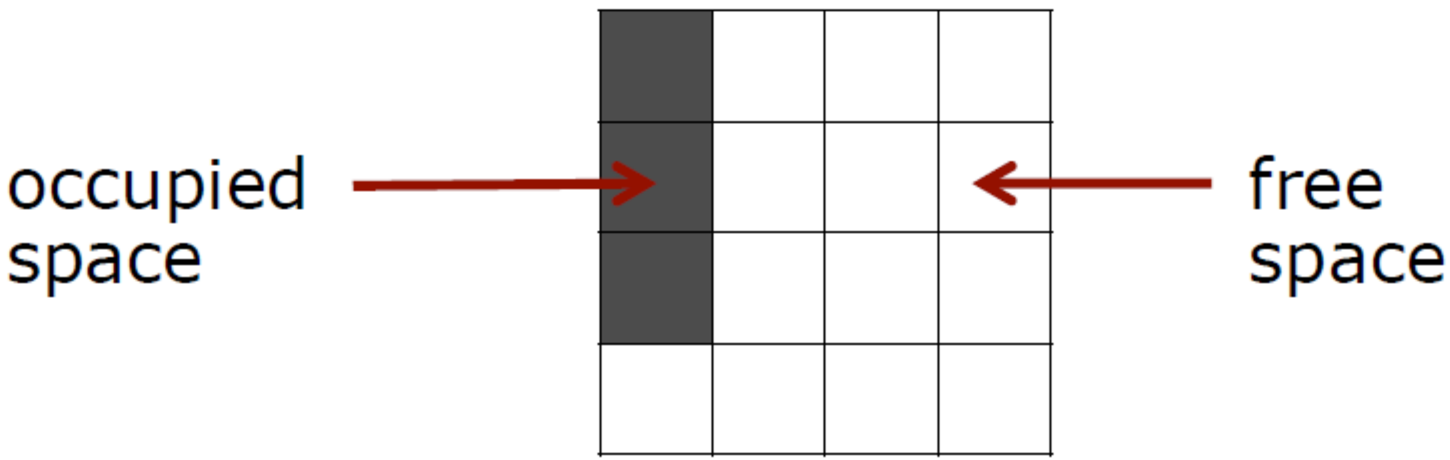




OCCUPANCY MAPS

Assumption 1

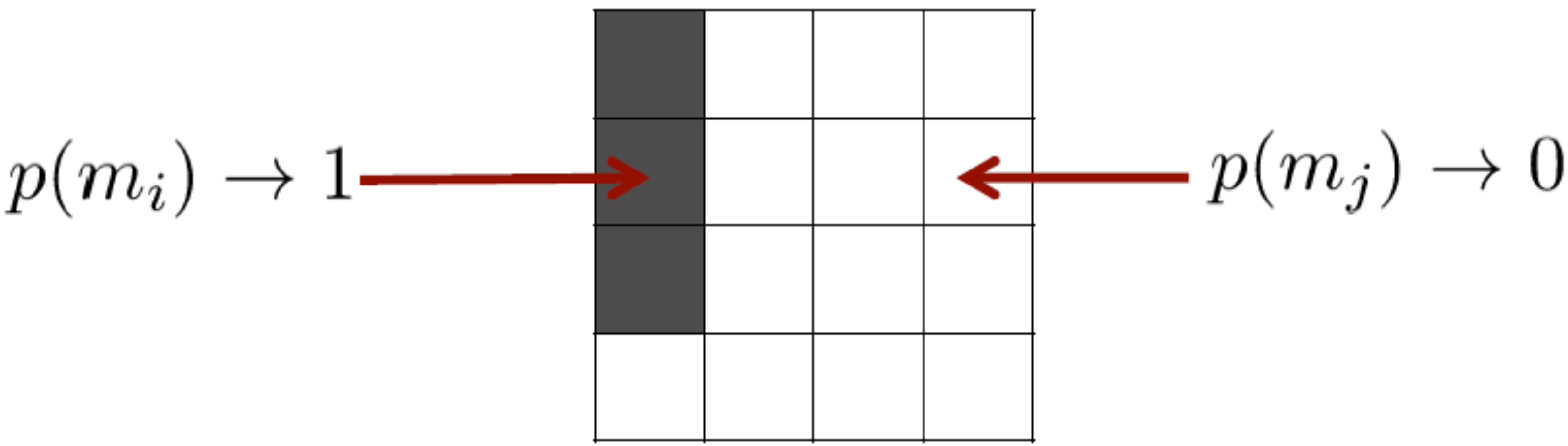
- The area that corresponds to a cell is either completely free or occupied



OCCUPANCY MAPS

Variables for Grid representation

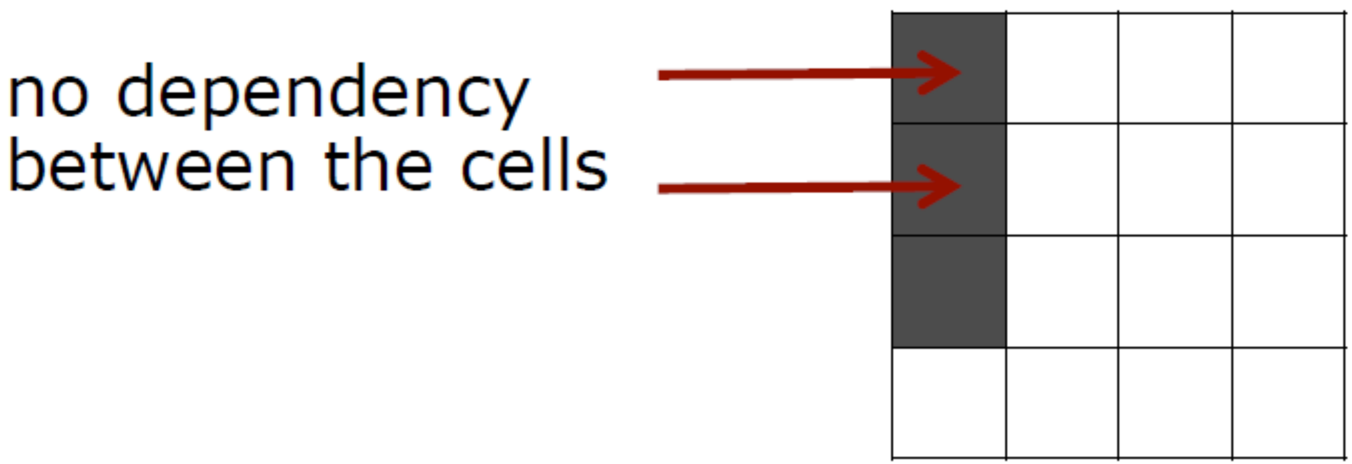
- Each cell is a binary random variable that models occupancy



OCCUPANCY MAPS

Assumption 2 for cells in Grid Map

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



OCCUPANCY MAPS

Probability of cells occupancy in Grid Maps

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

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

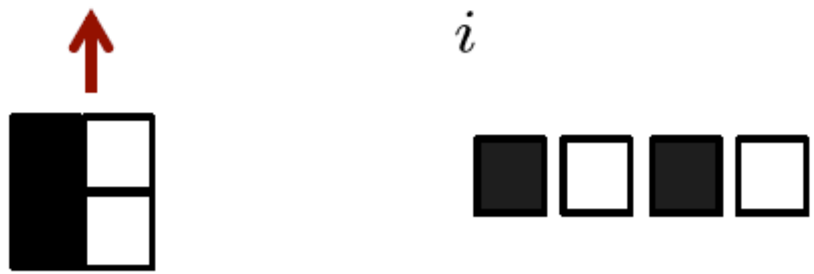
map

cell

OCCUPANCY MAPS

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

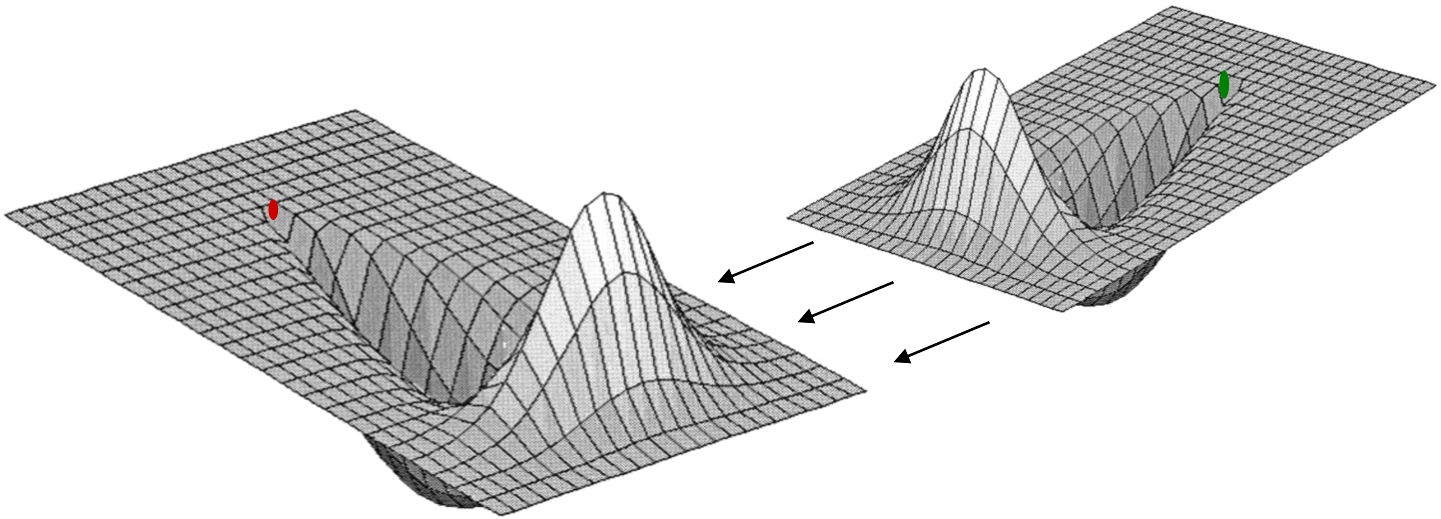
$$p(m) = \prod_i p(m_i)$$



example map (4-dim vector)

4 individual cells

COMBINING PROBABILITIES



How to combine two sets of probabilities into a single map ?

CONDITIONAL PROBABILITIES

Some intuition...

$p(m \mid z) =$ The probability of event m, given event z

The probability that a certain cell is occupied, given that the robot gets the sensor reading z

---

$p(z \mid m) =$ The probability of event z, given event m

The probability that the robot gets the sensor reading z, given that a certain cell is occupied.

- What is really meant by conditional probability ?

$$p(m \mid z) \neq p(z \mid m) \quad ??$$

- How are these two probabilities related?

BAYES RULE

- Joint probabilities/Conditional probabilities

$$p(m, z) = p(m \mid z)\, p(z)$$

(the probability of both m and z are true)

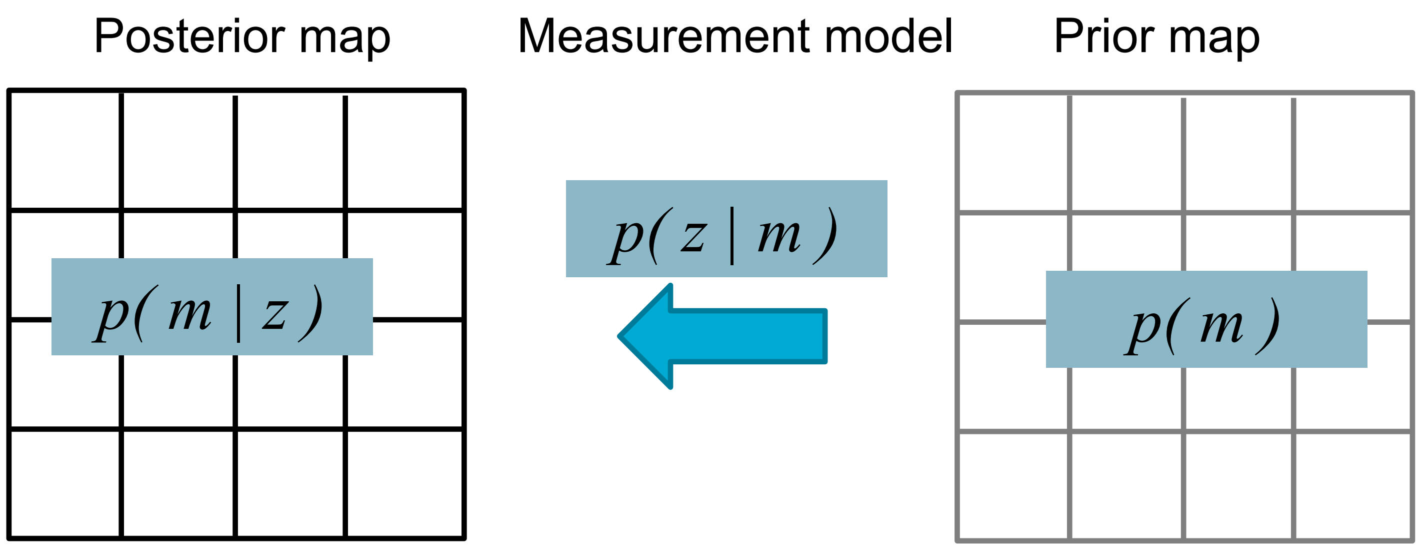
- Bayes rule relates conditional probabilities

$$\boxed{p(m \mid z) = \frac{p(z \mid m)\, p(m)}{p(z)}}$$ Bayes rule

- So, what does this say about $odds(m \mid z_2, z_1)$?

Can we update the odds easily ?

BAYES RULE



$$p(m \mid z) = \frac{p(z \mid m)\, p(m)}{p(z)}$$

Bayes rule

LOG ODDS INSTEAD OF PROBABILITIES

$$Odds := \frac{X\ happens}{X\ not\ happens} = \frac{P(X)}{P(\bar{X})}$$

Using Odds is more convenient

$$odds(m \mid z) = \frac{p(m=1 \mid z)}{p(m=0 \mid z)} = \frac{p(m \mid z)}{p(\bar{m} \mid z)}$$

LOG ODDS INSTEAD OF PROBABILITIES

Bayes rule

$$p(m=1 \mid z) = \frac{p(z \mid m=1)\, p(m=1)}{p(z)}$$

$$odds(m \mid z) = \frac{p(m=1 \mid z)}{p(m=0 \mid z)} = \frac{p(z \mid m=1)\, p(m=1)/p(z)}{p(m=0 \mid z)}$$

LOG ODDS INSTEAD OF PROBABILITIES

$$odds(m \mid z) = \frac{p(m=1 \mid z)}{p(m=0 \mid z)} = \frac{p(z \mid m=1)\, p(m=1)/\cancel{p(z)}}{p(z \mid m=0)\, p(m=0)/\cancel{p(z)}}$$

Bayes rule

$$p(m=0 \mid z) = \frac{p(z \mid m=0)\, p(m=0)}{p(z)}$$

LOG ODDS INSTEAD OF PROBABILITIES

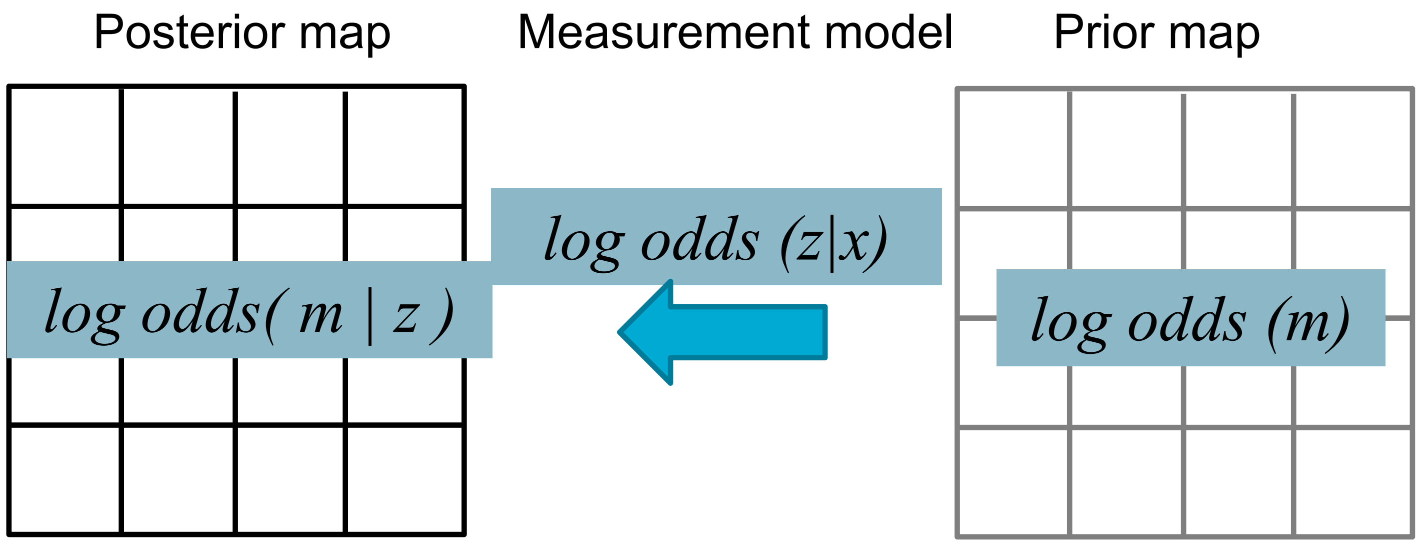
$$odds(m \mid z) = \frac{p(m=1 \mid z)}{p(m=0 \mid z)} = \frac{p(z \mid m=1)\, p(m=1)}{p(z \mid m=0)\, p(m=0)}$$

Take the log

$$\log odds(m \mid z) = \log \frac{p(m=1 \mid z)}{p(m=0 \mid z)} = \log \frac{p(z \mid m=1)\, p(m=1)}{p(z \mid m=0)\, p(m=0)}$$

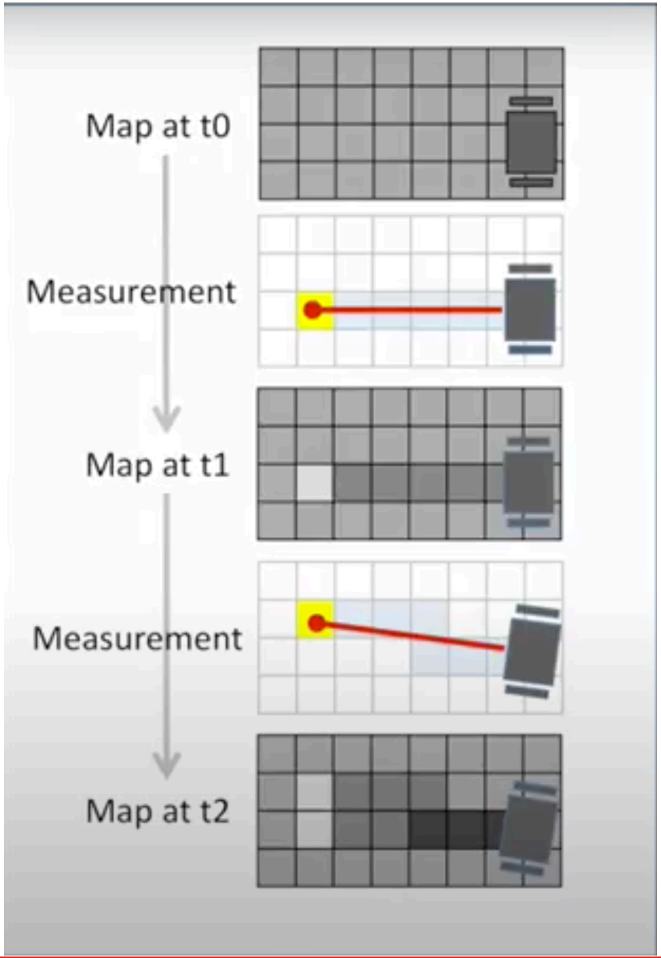
$$= \log \frac{p(z \mid m=1)}{p(z \mid m=0)} + \log \frac{p(m=1)}{p(m=0)}$$

LOG ODDS INSTEAD OF PROBABILITIES



$$\text{log odds}(\, m \mid z\,) = \text{log odds}\,(z|x) + \text{log odd}\,(m)$$

OCCUPANCY MAPPING



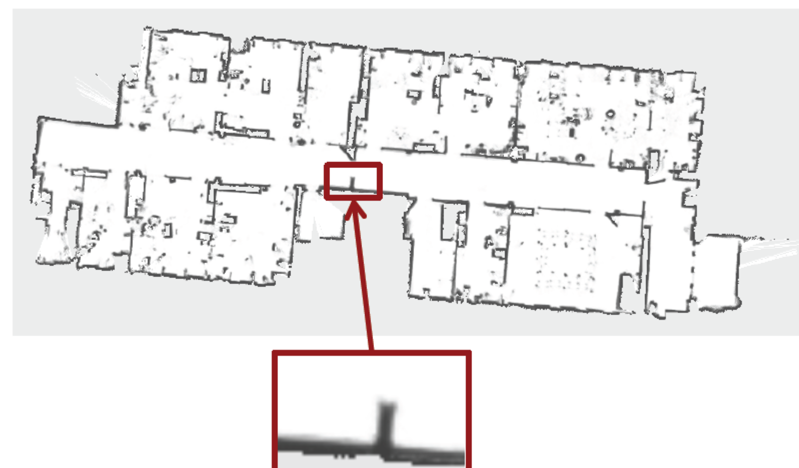
OCCUPANCY MAPPING

Estimating a Map from the Data

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

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

binary random variable



→ Binary Bayes filter (for a static state)

STATIC STATE BINARY FILTER

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

STATIC STATE BINARY FILTER

Markov Assumption

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

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

STATIC STATE BINARY FILTER

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

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

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

STATIC STATE BINARY FILTER

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

STATIC STATE BINARY FILTER

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

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

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

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

STATIC STATE BINARY FILTER

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

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

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

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

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 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 FILTER

- By computing the ratio of both probabilities, we obtain

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

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

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

STATIC STATE BINARY 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

The odds that a cell is **occupied**, given the sensor readings $z_{1..i}$

Log odds ratio is defined as

log odds (x) = $l(x) = \log \frac{p(x)}{1 - p(x)}$ $= \log p(x) / p(\bar{x})$

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

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

OCCUPANCY MAPPING IN LOG ODDS

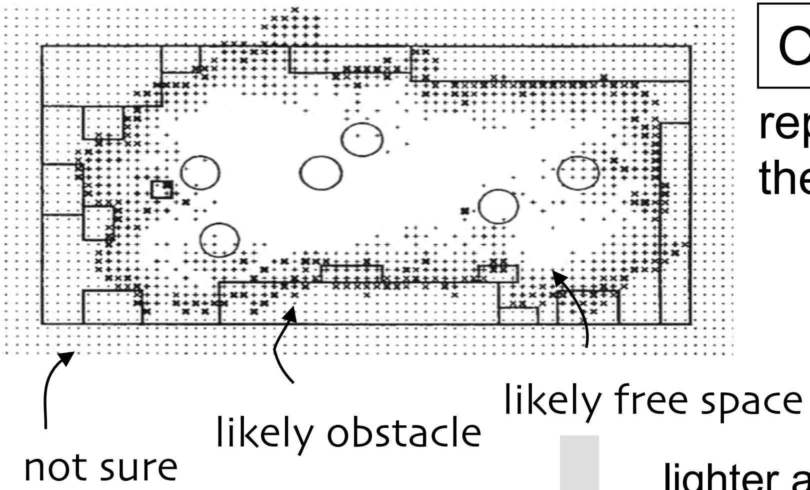
The product turns into a sum

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

or in short

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

OCCUPANCY MAPPING SUMMARY



Occupancy Map...

represent space as a collection of cells, each with the odds (or probability) that it contains an obstacle

evidence = log odds

lighter areas: *lower* odds of obstacles being present

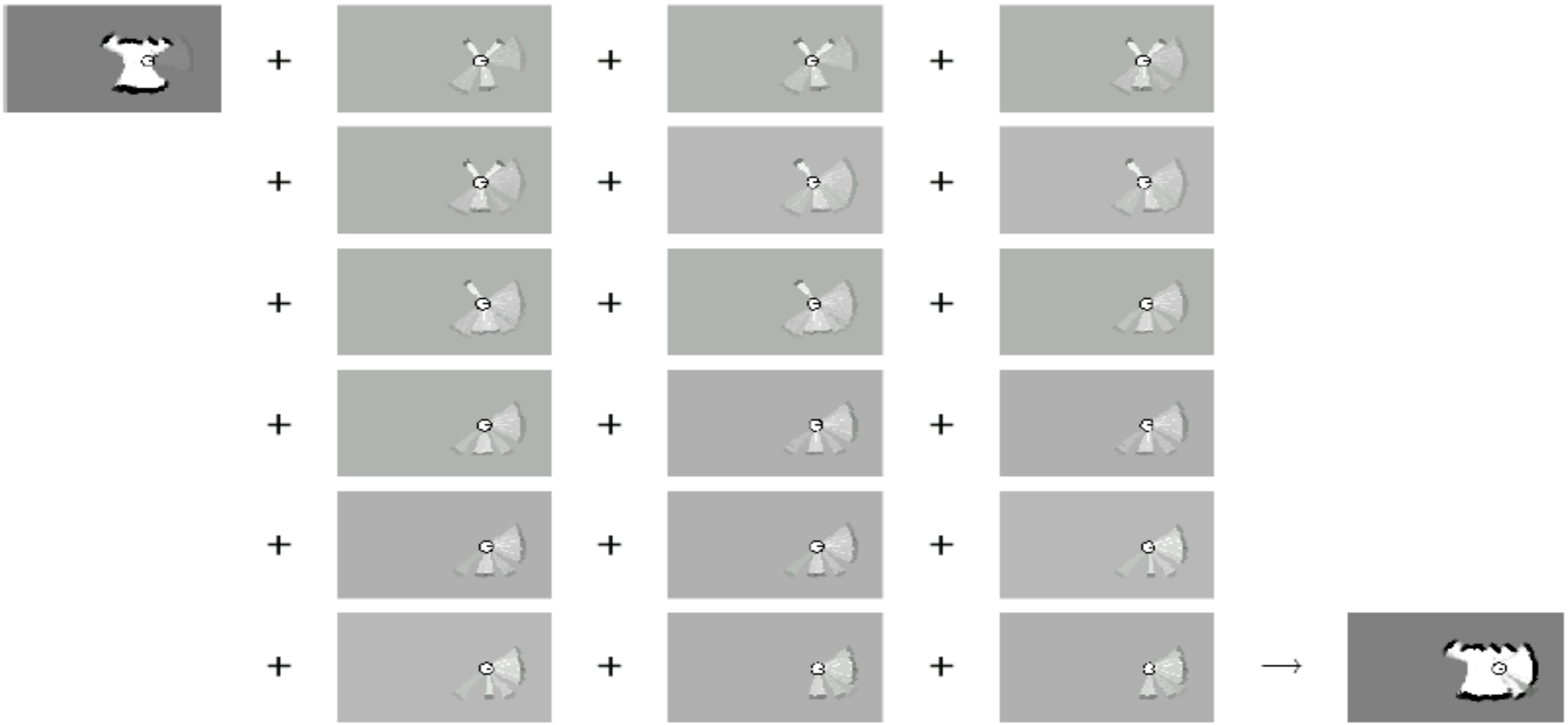
darker areas: *higher* odds of obstacles being present

- The locations of the robot are assumed to be known.

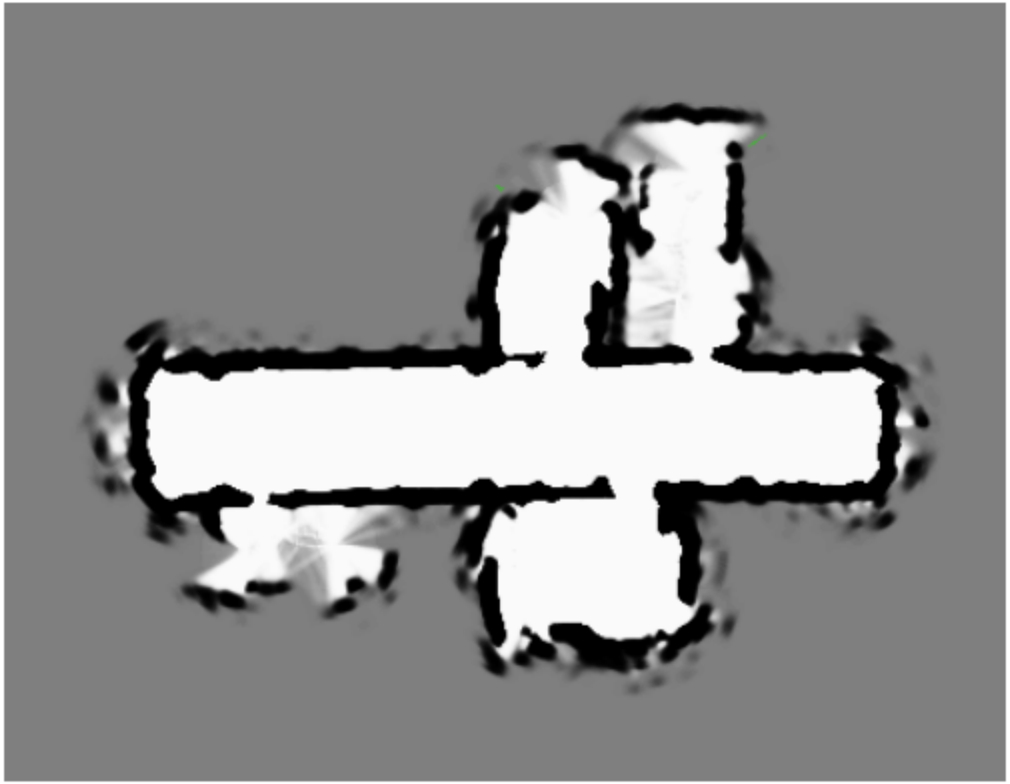
(It is important that the robot odometry is correct --- is this practical?)

- It is assumed that the multiple sensor reading are independent – sometimes not true

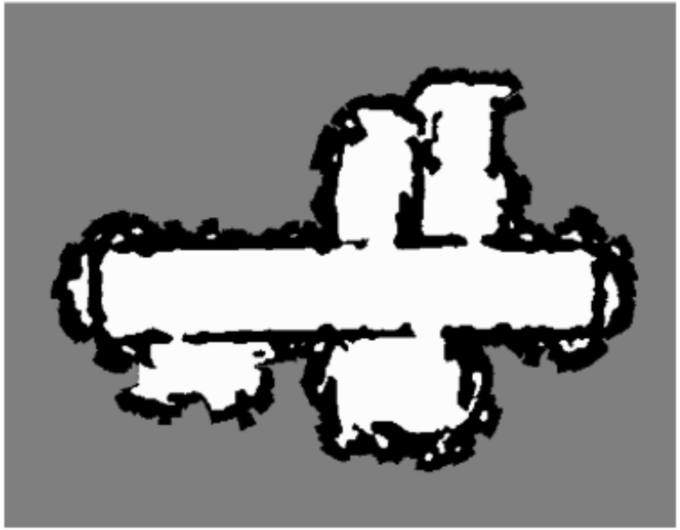
INCREMENTAL UPDATING OF OCCUPANCY GRIDS



OCCUPANCY MAP FROM SONAR

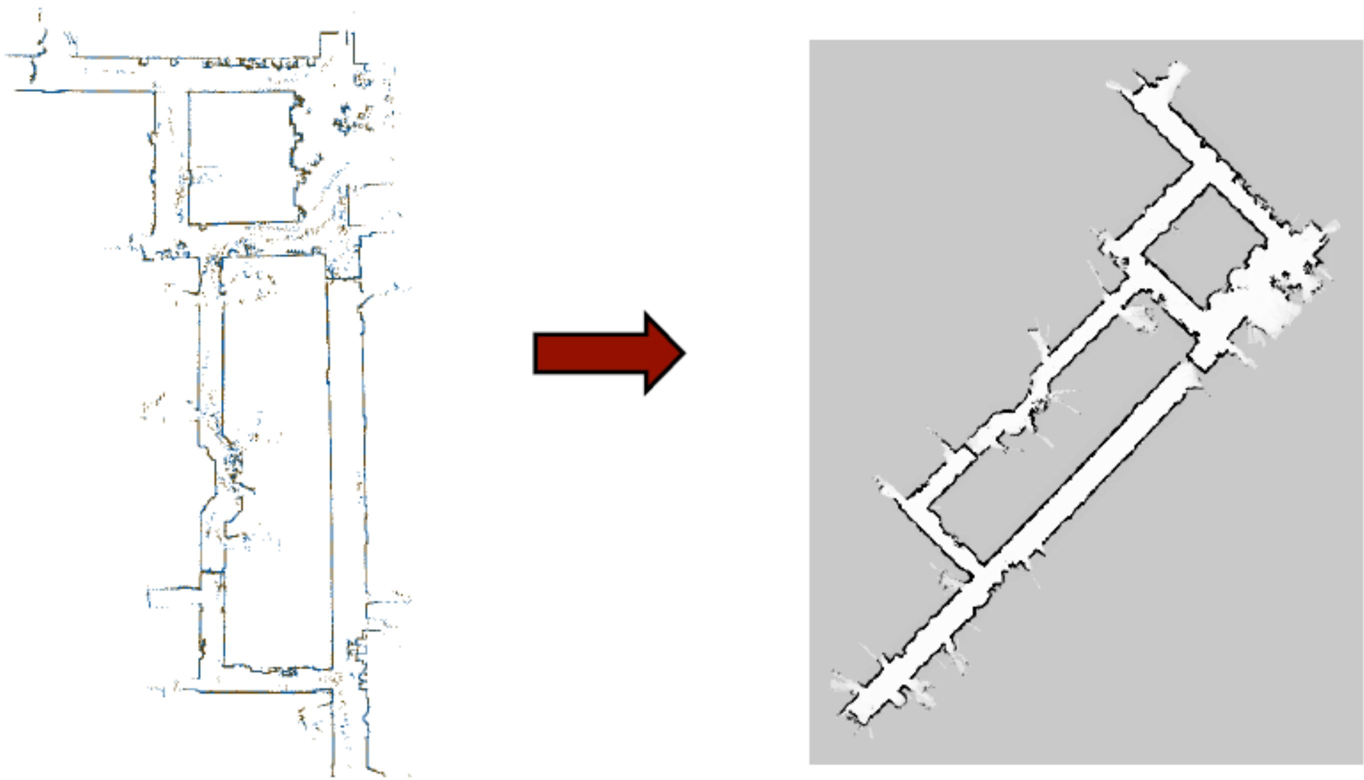


OCCUPANCY AND MAXIMUM LIKELIHOOD MAP



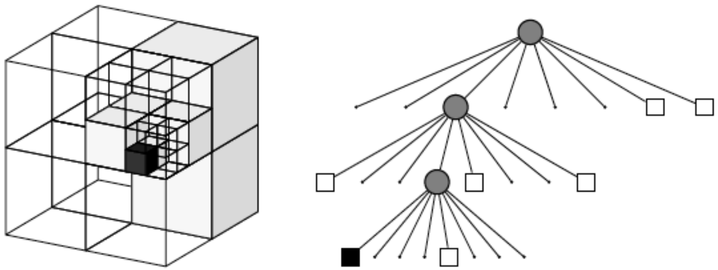
The maximum likelihood map is obtained by rounding the probability for each cell to 0 or 1.

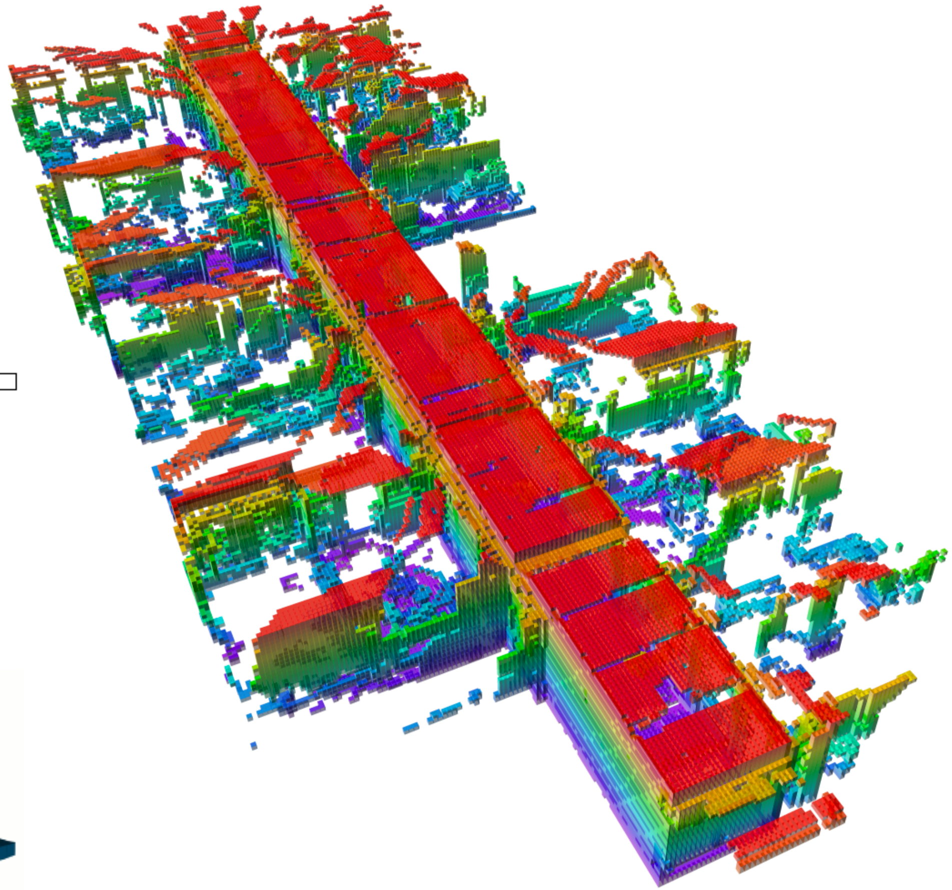
OCCUPANCY MAP FROM LIDAR SCAN TO MAP



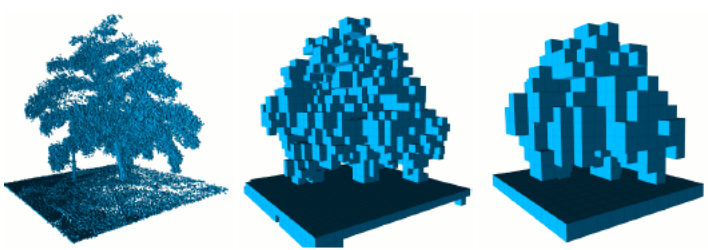
OCTOMAPS

- 3D Maps
- Based on Octrees and Voxels





- Multiresolution



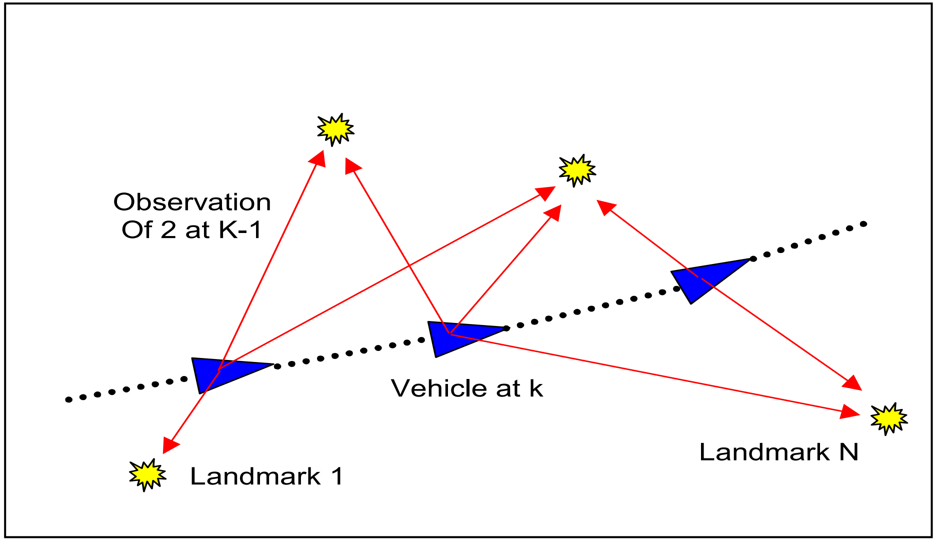
SUMMARY

- Occupancy grid maps discretise 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

FEATURE MAPS

Assume robot location is known all the time

- The feature positions can be estimated one by one using the sensor readings (ranges and/or bearings)
- e.g. applying EKF (what is the state vector?)
- No prediction step if feature is stationary
- real variable



FEATURE MAPPING EXAMPLE

FEATURE MAPS

Assume robot location is known all the time

- The feature positions can be estimated one by one using
- **EKF (state vector is the feature position)**

The observation model at time $k+1$ is given by

$$\mathbf{z}_{k+1} = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_{k+1}.$$

Update using observation:

$$\hat{\mathbf{x}}_{k+1} = \mathbf{x}_k + K(\mathbf{z}_{k+1} - \mathbf{h}(\bar{\mathbf{x}}_k))$$

$$P_{k+1} = P_k - KSK^T,$$

where:

$$S = \nabla \mathbf{h} P_k \nabla \mathbf{h}^T + R$$

$$K = P_k \nabla \mathbf{h}^T S^{-1}.$$

The uncertainty of the feature keeps reducing

SIMULTANEOUS LOCALIZATION AND MAPPING (SLAM)

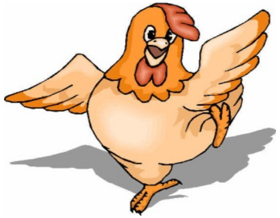
If we have a map:

We can localize!

If we can localize:

We can make a map!

CIRCULAR ERROR PROBLEM



If we have a map:

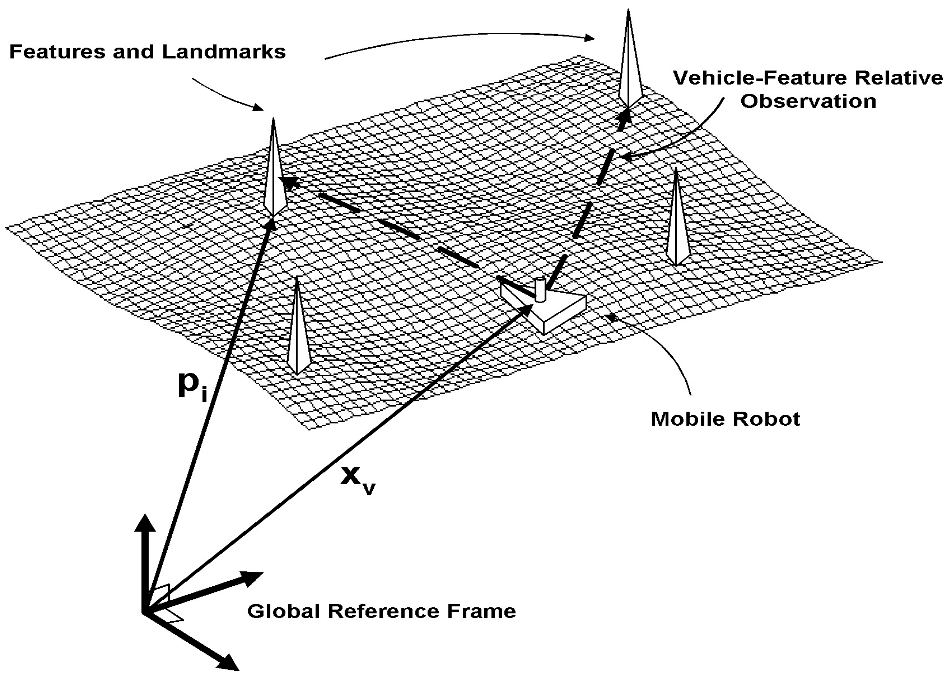
We can localize!

NOT THAT SIMPLE!

If we can localize:

We can make a map!

SLAM



- Robot moves around and observe landmarks
- No a priori knowledge of landmark locations
- From relative observations of landmarks, simultaneously compute an estimate of robot location and an estimate of landmark locations
- While continuing in motion, build a complete map of landmarks and use these to provide continuous estimates of robot location

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