Mapping with Known Poses

Pieter Abbeel
UC Berkeley EECS

Many slides adapted from Thrun, Burgard and Fox, Probabilistic Robotics

Why Mapping?

- Learning maps is one of the fundamental problems in mobile robotics
- Successful robot systems rely on maps for localization, path planning, activity planning etc.

The General Problem of Mapping

- Formally, mapping involves, given the control inputs and sensor data,

$$d = \{u_1, z_1, u_2, z_2, \ldots, u_n, z_n\}$$

to calculate the most likely map

$$m^* = \arg\max_m P(m \mid d)$$

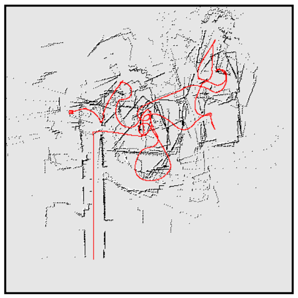
Mapping as a Chicken and Egg Problem

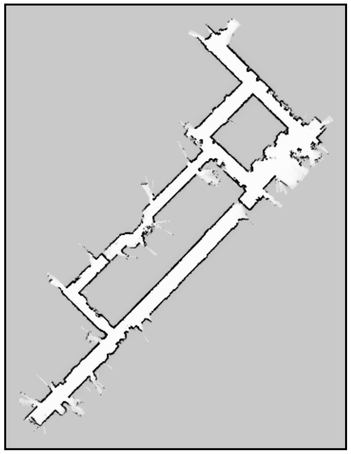
- So far we learned how to estimate the pose of a robot given the data and the map.
- Mapping, however, involves to simultaneously estimate the pose of the vehicle and the map.
- The general problem is therefore denoted as the simultaneous localization and mapping problem (SLAM).
- **Throughout this set of slides we will describe how to calculate a map given we know the pose of the robot.**
  - We'll build on top of this to achieve SLAM.

Types of SLAM-Problems

This Lecture

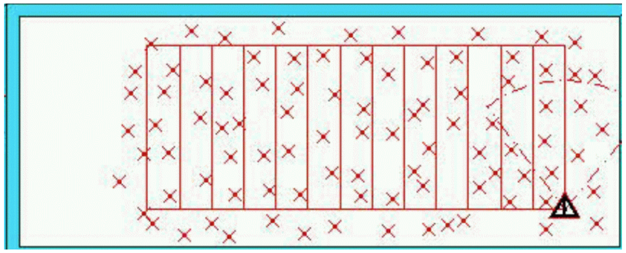
- Grid maps or scans

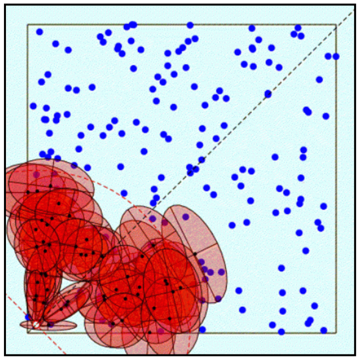


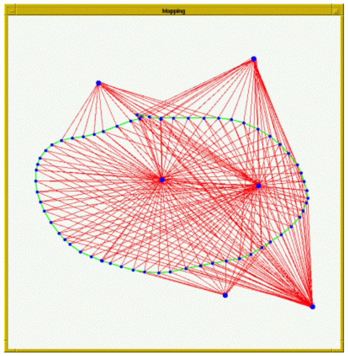


[Lu & Milios, 97; Gutmann, 98: Thrun 98; Burgard, 99; Konolige & Gutmann, 00; Thrun, 00; Arras, 99; Haehnel, 01;…]

- Landmark-based







[Leonard et al., 98; Castelanos et al., 99: Dissanayake et al., 2001; Montemerlo et al., 2002;…

Grid Maps

- Occupancy grid maps
  - For each grid cell represent whether occupied or not
- Reflection maps
  - For each grid cell represent probability of reflecting a sensor beam

Occupancy Grid Maps

- Introduced by Moravec and Elfes in 1985
- Represent environment by a grid. Each cell can be empty or occupied.
- E.g., 10m by 20m space, 5cm resolution → 80,000 cells → $2^{80,000}$ possible maps.
- → Can't efficiently compute with general posterior over maps
- Key assumption:
  - Occupancy of individual cells (m[xy]) is independent

$$Bel(m_t) = P(m_t \mid u_1, z_2 \ldots, u_{t-1}, z_t) = \prod_{x,y} Bel(m_t^{[xy]})$$

Updating Occupancy Grid Maps

- Idea: Update each individual cell using a binary Bayes filter.

$$Bel(m_t^{[xy]}) = \eta \, p(z_t \mid m_t^{[xy]}) \int p(m_t^{[xy]} \mid m_{t-1}^{[xy]}, u_{t-1}) Bel(m_{t-1}^{[xy]}) \, dm_{t-1}^{[xy]}$$

- Additional assumption: Map is static.

$$Bel(m_t^{[xy]}) = \eta \, p(z_t \mid m_t^{[xy]}) Bel(m_{t-1}^{[xy]})$$

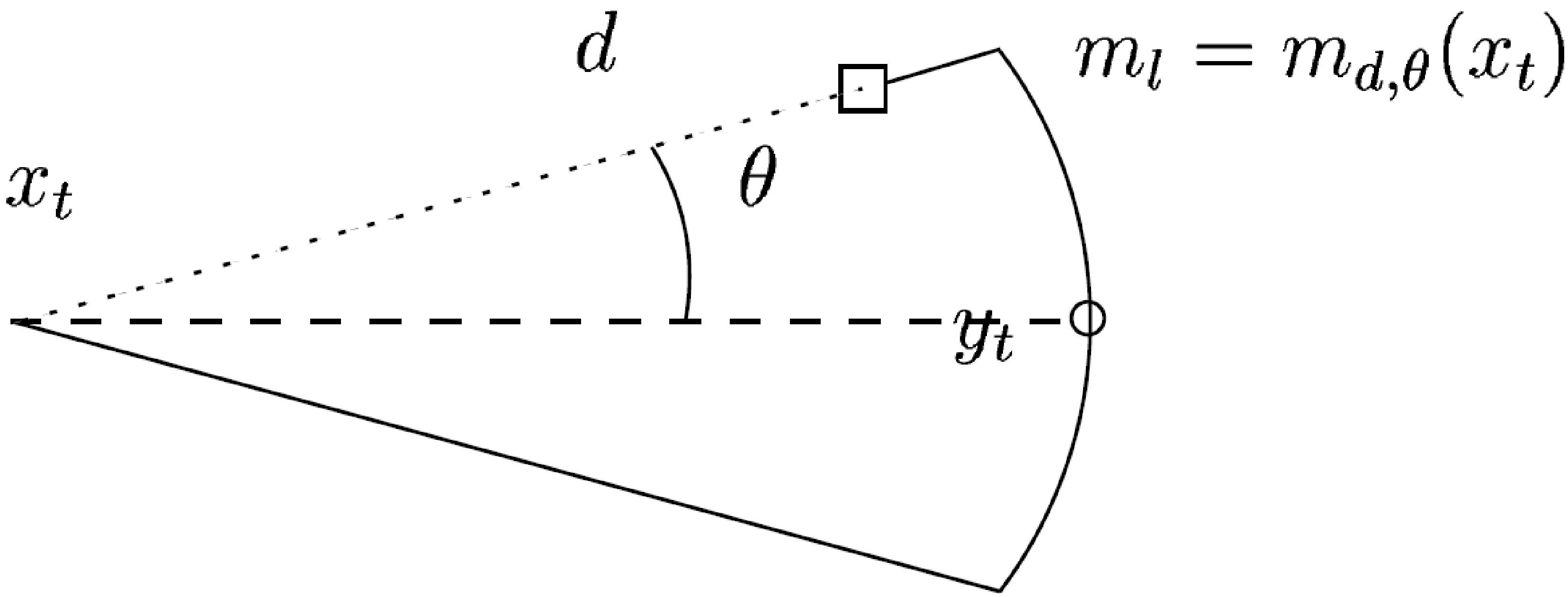
Updating Occupancy Grid Maps

- Per grid-cell update:

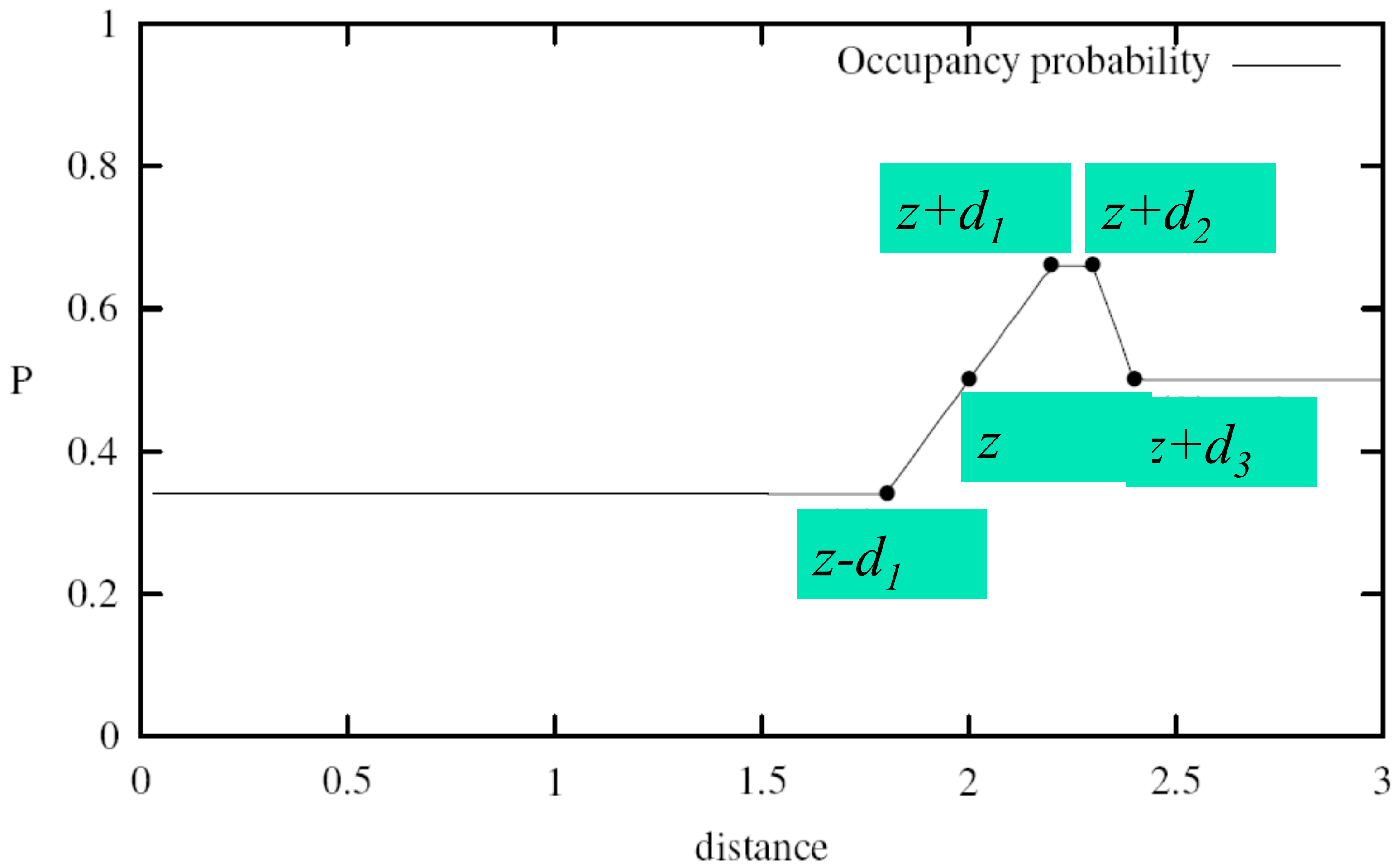
$$Bel(m_t^{[xy]} = v) = \eta \; p(z_t \mid m_t^{[xy]} = v) Bel(m_{t-1}^{[xy]})$$

- BUT: how to obtain $p(z_t \mid m_t^{[xy]} = v)$?
  - = $\displaystyle\sum_{m:m^{[xy]}=v} p(z_t|m)Bel(m)$
  - This would require summation over all maps
  - → Heuristic approximation to update that works well in practice

Key Parameters of the Model



Occupancy Value Depending on the Measured Distance



Recursive Update

Assumption: measurements independent

We will focus on the recursive update for a single grid cell. Let $p(\text{occ}|z)$ be the probability of that grid cell being occupied given a single measurement z. This function is shown on the previous slide.

We will keep track of the ratio of the probability of the cell being occupied over the probability of the cell not being occupied. I.e., we are working with:

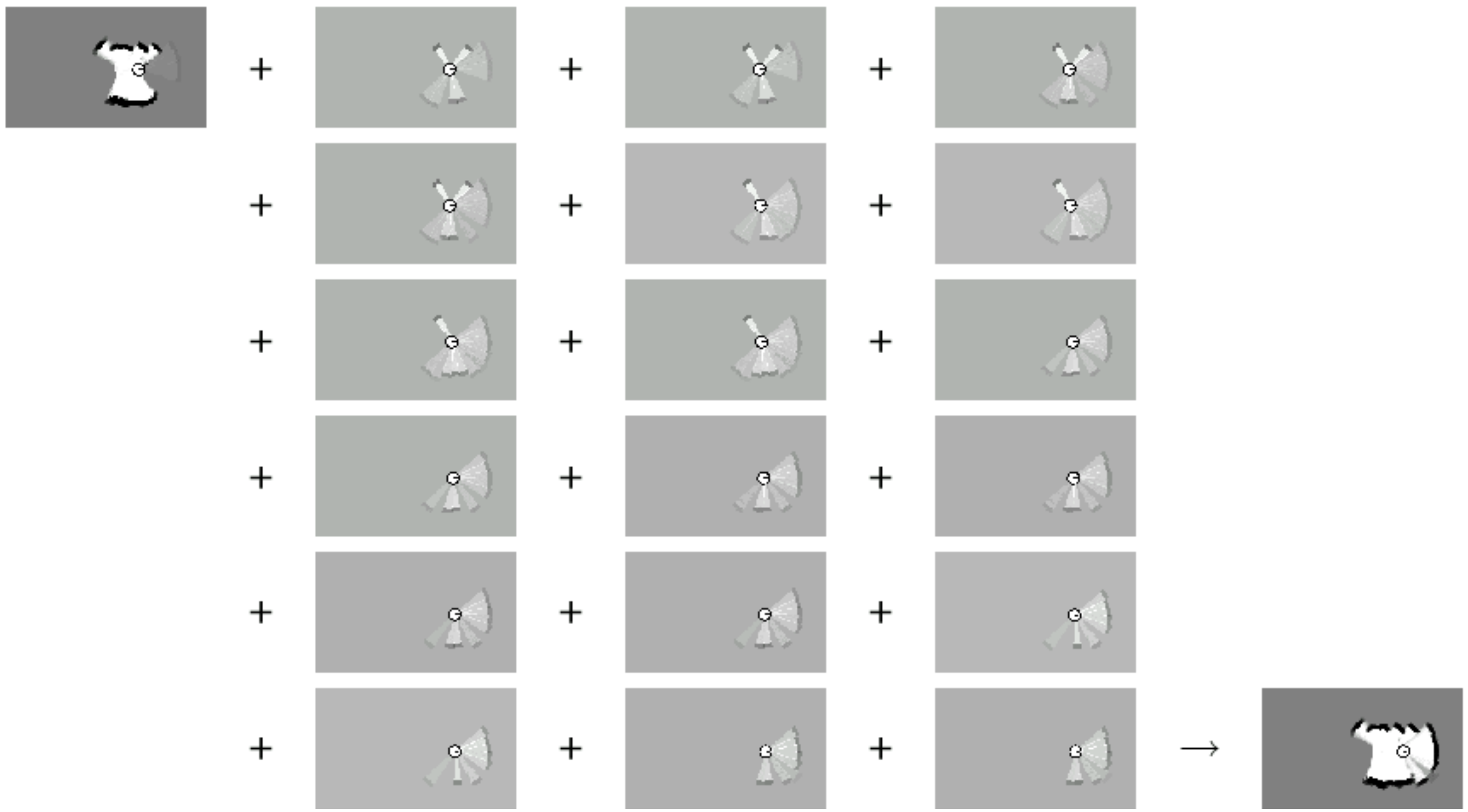
$$
\begin{aligned}
\frac{p(\text{occ}|z_1,\ldots,z_t)}{p(\neg\text{occ}|z_1,\ldots,z_t)} &= \frac{p(\text{occ}|z_1,\ldots,z_t)p(z_1,\ldots,z_t)}{p(\neg\text{occ}|z_1,\ldots,z_t)p(z_1,\ldots,z_t)} \\
&= \frac{p(\text{occ},z_1,\ldots,z_t)}{p(\neg\text{occ},z_1,\ldots,z_t)} \\
&= \frac{p(\text{occ})\prod_{s=1}^{t} p(z_s|\text{occ})}{p(\neg\text{occ})\prod_{s=1}^{t} p(z_s|\text{occ})}
\end{aligned}
$$

To perform a recursive update, we need to compute:

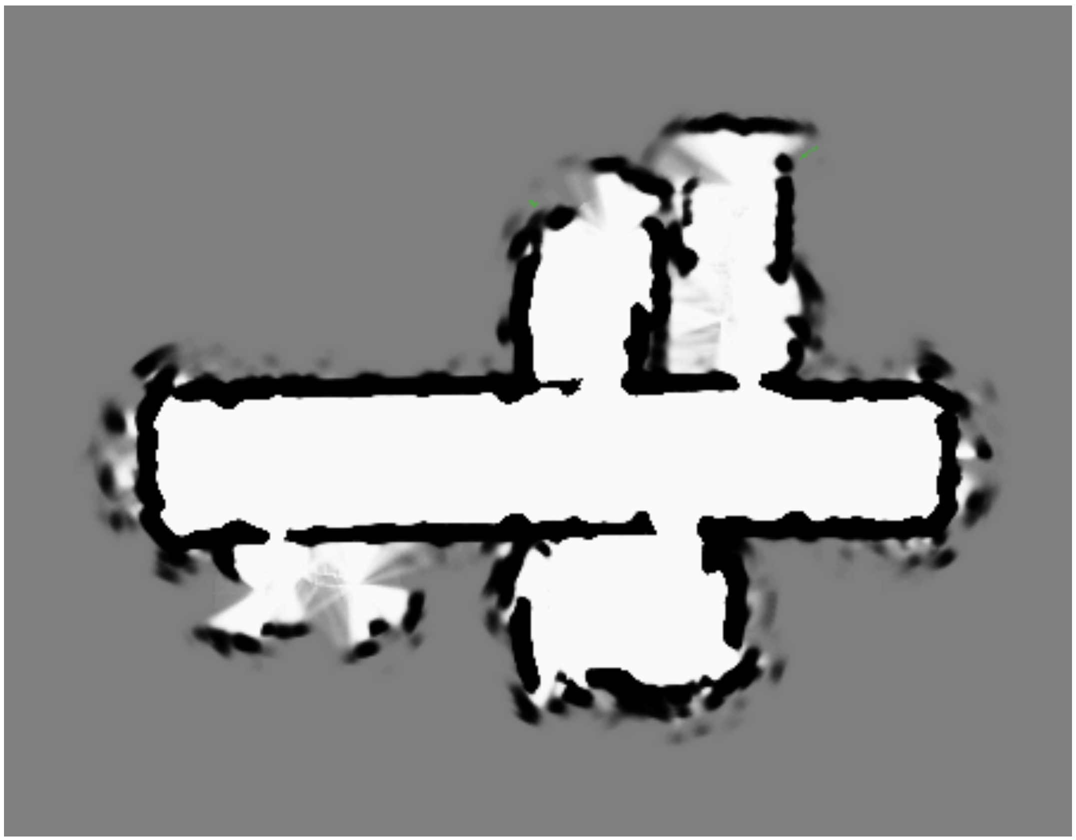
$$
\begin{aligned}
\frac{p(z_t|\text{occ})}{p(z_t|\neg\text{occ})} &= \frac{p(\text{occ}|z_t)p(z_t)/p(\text{occ})}{p(\neg\text{occ}|z_t)p(z_t)/p(\neg\text{occ})} \\
&= \frac{p(\text{occ}|z_t)}{p(\neg\text{occ}|z_t)}\,\frac{1/p(\text{occ})}{1/p(\neg\text{occ})}
\end{aligned}
$$

Hence we found a way to perform the update by simply having access to $p(\text{occ}|z_t)$, which we have from the previous slide, $p(\neg\text{occ}|z_t) = 1 - p(\text{occ}|z_t)$, which is readily derived from the previous slide, and the prior $p(\text{occ})$, $p(\neg\text{occ}) = 1 - p(\text{occ})$.

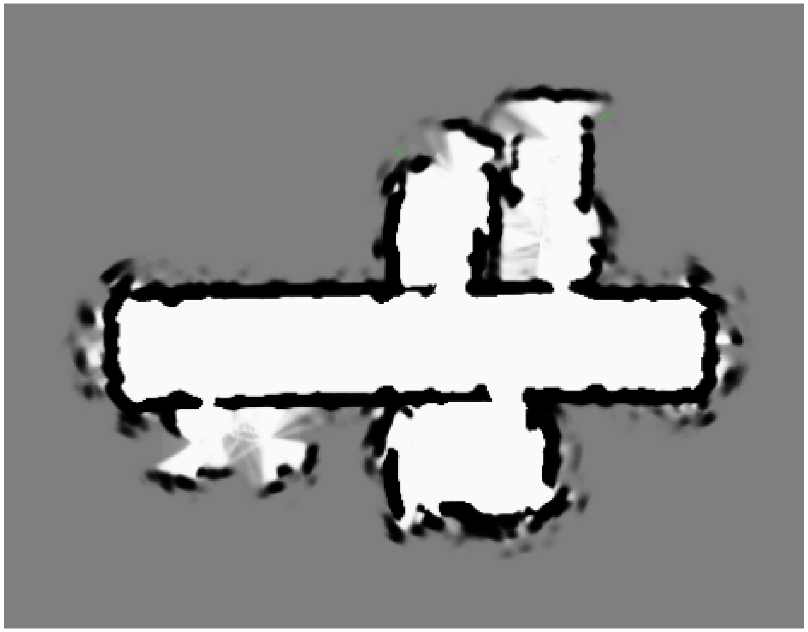
Incremental Updating of Occupancy Grids (Example)

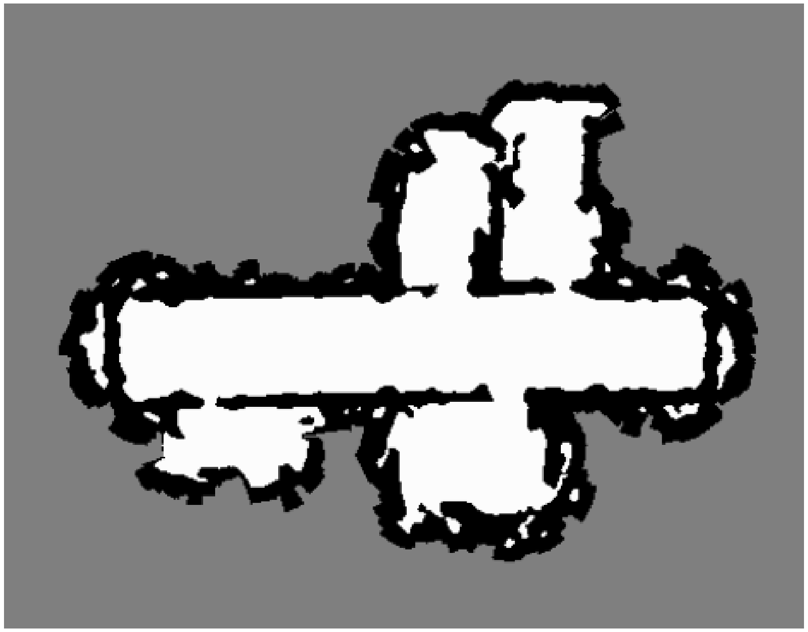


Resulting Map Obtainedwith Ultrasound Sensors



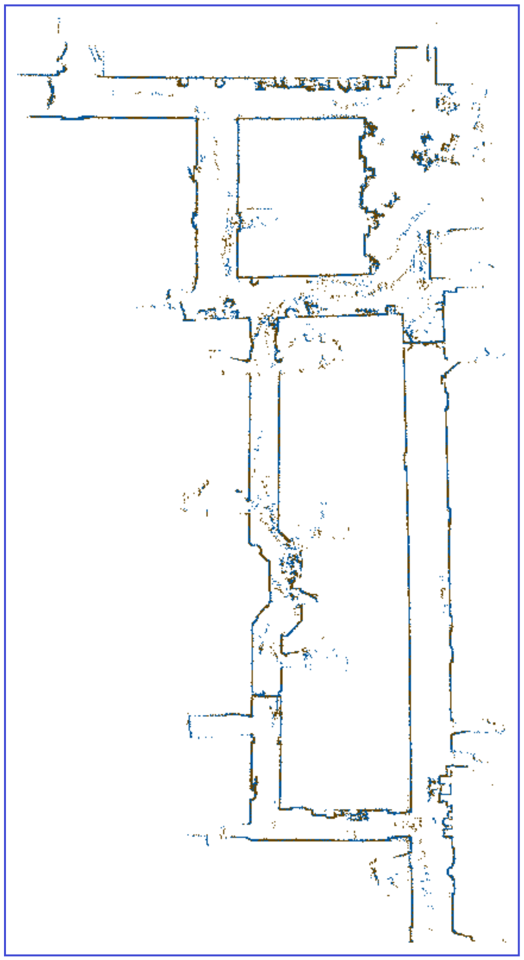
Resulting Occupancy and Maximum Likelihood Map

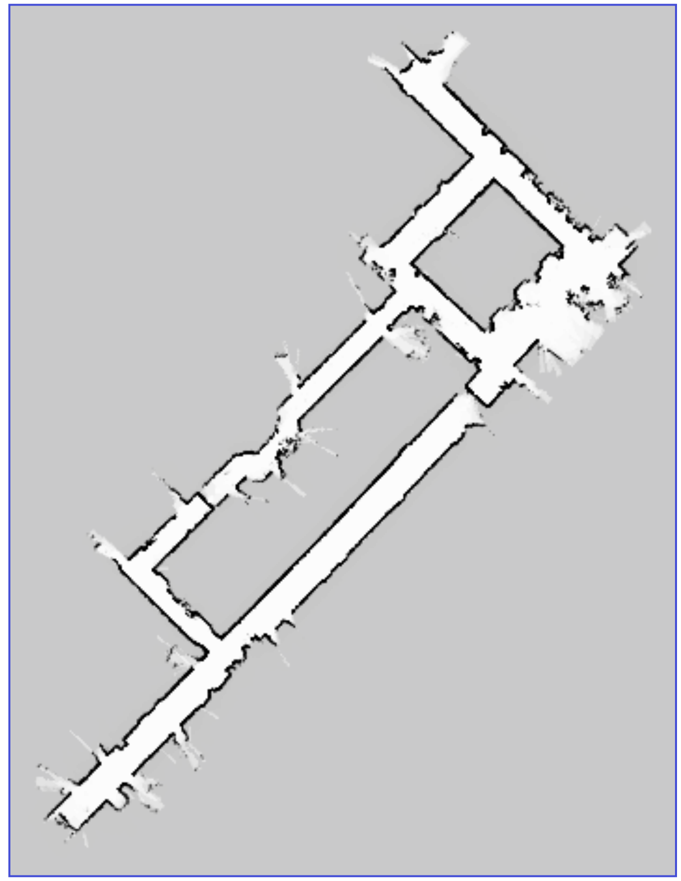




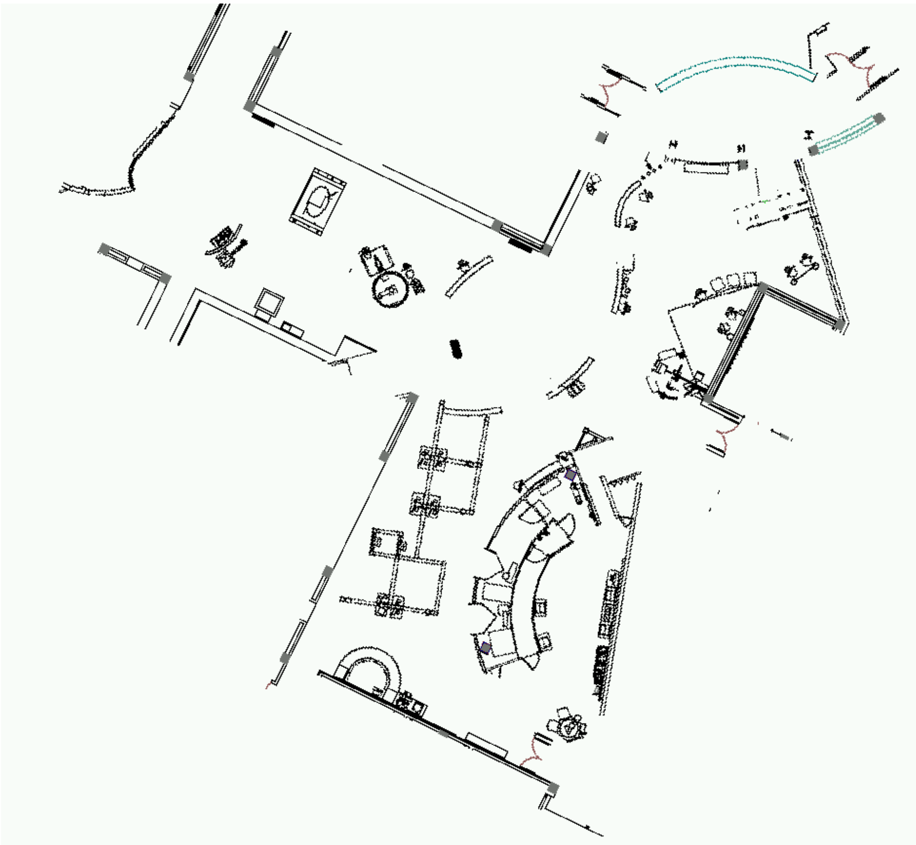
The maximum likelihood map is obtained by clipping the occupancy grid map at a threshold of 0.5

Occupancy Grids: From scans to maps

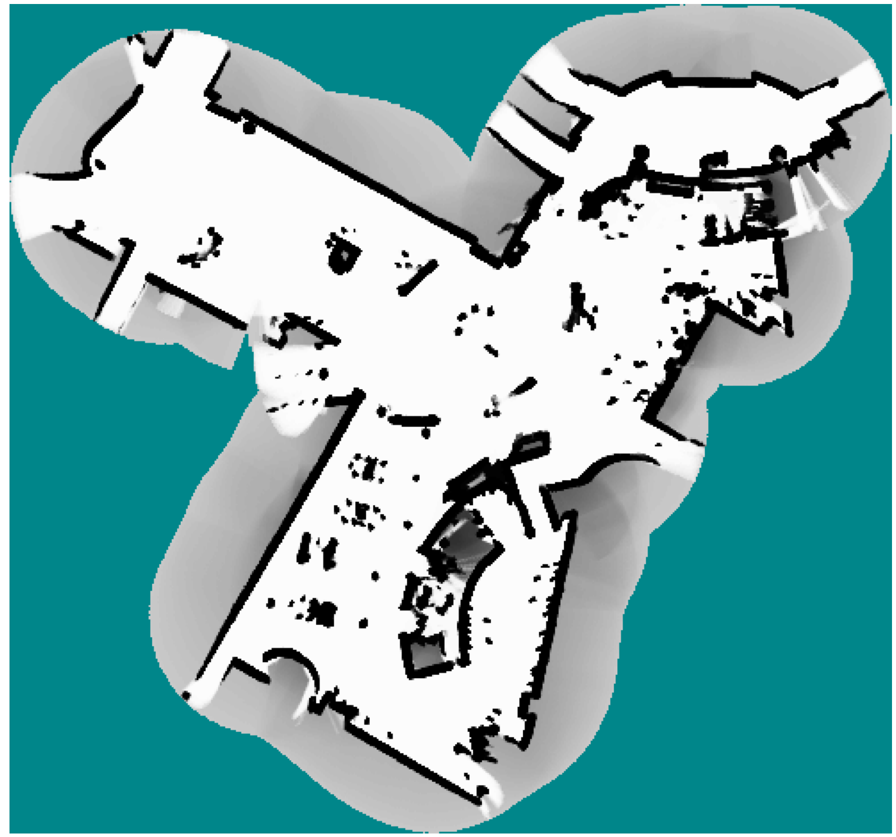




Tech Museum, San Jose



CAD map



occupancy grid map

Grid Maps

- Occupancy grid maps
  - For each grid cell represent whether occupied or not
- Reflection maps
  - For each grid cell represent probability of reflecting a sensor beam

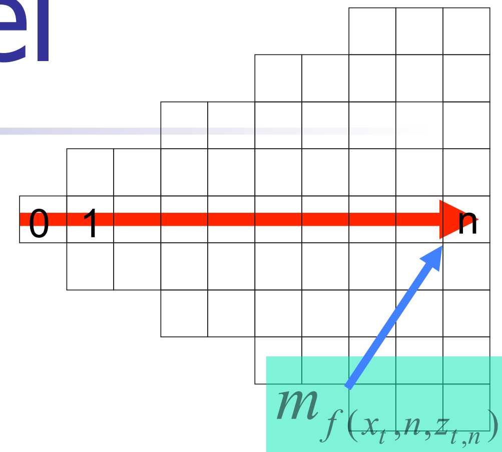
Reflection Maps: Simple Counting

- 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>

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

- Value of interest: $P(reflects(x,y))$
- Turns out we can give a formal Bayesian justification for this counting approach

The Measurement Model



1. pose at time *t*: x_t
2. beam *n* of scan *t*: $z_{t,n}$
3. maximum range reading: $\varsigma_{t,n} = 1$
4. beam reflected by an object: $\varsigma_{t,n} = 0$

$$p(z_{t,n} \mid x_t, m) = \begin{cases} \displaystyle\prod_{k=0}^{z_{t,n}-1} (1 - m_{f(x_t,n,k)}) & \text{if } \varsigma_{t,n} = 1 \\ \displaystyle m_{f(x_t,n,z_{t,n})} \prod_{k=0}^{z_{t,n}-1} (1 - m_{f(x_t,n,k)}) & \text{if } \varsigma_{t,n} = 0 \end{cases}$$

Computing the Most Likely Map

- Compute values for m that maximize

$$m^* = \arg\max_m P(m \mid z_1, \ldots, z_t, x_1, \ldots, x_t)$$

- Assuming a uniform prior probability for p(m), this is equivalent to maximizing (applic. of Bayes rule)

$$m^* = \arg\max_m P(z_1, \ldots, z_t \mid m, x_1, \ldots, x_t)$$

$$= \arg\max_m \prod_{t=1}^{T} P(z_t \mid m, x_t)$$

$$= \arg\max_m \sum_{t=1}^{T} \ln P(z_t \mid m, x_t)$$

Computing the Most Likely Map

$$m^* = \arg\max_m \left[\sum_{j=1}^{J} \sum_{t=1}^{T} \sum_{n=1}^{N} \Big(I(f(x_t, n, z_{t,n}) = j) \cdot (1 - \varsigma_{t,n}) \cdot \ln m_j \right.$$

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

Suppose

$$\alpha_j = \sum_{t=1}^{T} \sum_{n=1}^{N} I(f(x_t, n, z_{t,n}) = j) \cdot (1 - \varsigma_{t,n})$$

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

Meaning of α_j and β_j

$$\alpha_j = \sum_{t=1}^{T} \sum_{n=1}^{N} I(f(x_t, n, z_{t,n}) = j) \cdot (1 - \varsigma_{t,n})$$

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

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

corresponds to the number of times a beam intercepted cell j without ending in it (*misses(j)*).

Computing the Most Likely Reflection Map

We assume that all cells m_j are independent:

$$m^* = \arg\max_m \left(\sum_{j=1}^{J} \alpha_j \ln m_j + \beta_j \ln(1 - m_j) \right)$$

If we set

$$\frac{\partial m}{\partial m_j} = \frac{\alpha_j}{m_j} - \frac{\beta_j}{1 - m_j} = 0$$

we obtain

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

Computing the most likely map amounts to counting how often a cell has reflected a measurement and how often it was intercepted.

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.

Example Occupancy Map

Example Reflection Map

glass panes

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 cell is intercepted by a beam that does not end in it.
- Accordingly, after n measurements we will have

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

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

Summary

- Grid maps are a popular approach to represent the environment of a mobile robot given known poses.
- In this approach each cell is considered independently from all others.
- Occupancy grid maps
  - store the posterior probability that the corresponding area in the environment is occupied.
  - can be estimated efficiently using a probabilistic approach.
- Reflection maps are an alternative representation.
  - store in each cell the probability that a beam is reflected by this cell.
  - the counting procedure underlying reflection maps yield the optimal reflection map.

Inverse_range_sensor_model(m_i, x_t, z_t)

1. Let x_i, y_i be the center-of-mass of grid-cell m_i.
2. Let $x_t = (x, y, \theta)$
3. $r = \sqrt{(x_i - x)^2 + (y_i - y)^2}$
4. $\phi = \text{atan2}(y_i - y, x_i - x) - \theta$
5. $k = \arg\min_j |\phi - \theta_{j,\text{sens}}|$
6. If $r > \min(z_{\max}, z_t^k + \alpha/2)$ or $|\phi - \theta_{k,\text{sens}}| > \beta/2$ then
7. // no new information obtained about m_i
8. Else if $z_t^k < z_{\max}$ and $|r - z_t^k| < \alpha/2$
9. // we measured m_i as occupied
10. Else if $r \leq z_t^k$
11. // we measured m_i as free

- α: thickness of obstacles
- β: width of the sensor beam

[Probabilistic Robotics, Table 9.2]