Bayes filter for mapping

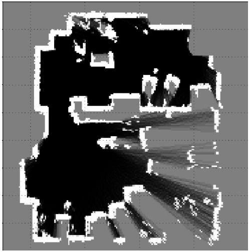
Miroslav Kulich

Intelligent and Mobile Robotics Group
Gerstner Laboratory for Intelligent Decision Making and Control
Czech Technical University in Prague

Tuesday 24/07/2012

Occupancy grid maps

- Introduced by Moravec and Elfes in 1985
- Because of intrinsic limitations in any sonar, it is important to compose a coherent world-model using information gained from multiple reading
- 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

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

    - Robot positions are known!

Updating occupancy grid maps

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

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

- Additional assumption: Map is static.

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

Occupancy grid cells

- The proposition $occ(i, j)$ means:
  - The cell C_{ij} is occupied.
- Probability: $p(occ(i, j))$ has range $[0, 1]$.
- Odds: $o(occ(i, j))$ has range $[0, +\infty)$.

$$o(A) = \frac{p(A)}{p(\neg A)}$$

- Log odds: $\log o(occ(i, j))$ has range $(-\infty, +\infty)$
- Each cell C_{ij} holds the value $\log o(occ(i, j))$

Probabilistic occupancy grids

- We will apply Bayes rule:

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)}$$

  - where A is $occ(i, j)$
  - and B is an observation $r = D$
- We can simplify this by using the log odds representation.

Bayes rule using odds

- Bayes rule:

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)}$$

- Likewise:

$$p(\neg A|B) = \frac{p(B|\neg A)p(\neg A)}{p(B)}$$

- so:

$$o(A|B) = \frac{p(A|B)}{p(\neg A|B)} = \frac{p(B|A)p(A)}{p(B|\neg A)p(\neg A)} = \lambda(B|A)o(A)$$

- where:

$$o(A|B) = \frac{p(A|B)}{p(\neg A|B)}$$

  and

$$\lambda(B|A) = \frac{p(B|A)}{p(B|\neg A)}$$

Easy update using Bayes

- Bayes rule can be written:

$$o(A|B) = \lambda(B|A)o(A)$$

- Take log odds to make multiplication into addition:

$$\log o(A|B) = \log \lambda(B|A) + \log o(A)$$

- Easy update for cell content.

Occupancy grid cell update

- Cell C_{ij} holds $\log o(occ(i,j))$.
- Evidence $r = D$ means sensor r returns D.
- For each cell C_{ij} accumulate evidence from each sensor reading:

$$\log o(A|B) = \log \lambda(B|A) + \log o(A)$$

$$\log o(occ(i,j)|r = D) = \log o(occ(i,j)) + log\,\lambda(r = D|occ(i,j))$$

Sensor model for a laser range-finder

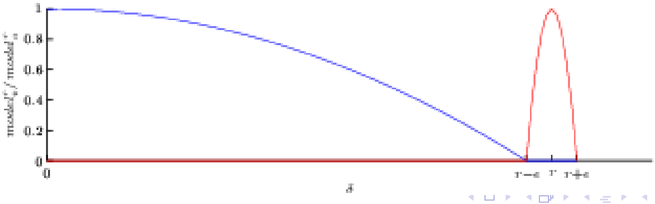
Probability density $p(z_t|m_t^{[xy]})$ is defined:

$$p(z_t|m_t^{[xy]}) = \frac{1 + model_O^{z_t}(\alpha, r) - model_V^{z_t}(\alpha, r)}{2},$$

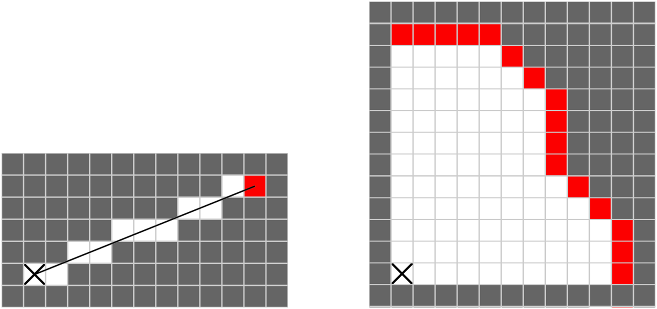
where (α, r) are polar coordinates of the cell $m_t^{[xy]}$ in sensor coordinate system and z_t is measured distance.

$$model_v^r(\delta) = \begin{cases} 1 - \left(\frac{\delta}{r-\epsilon}\right)^2, & \text{for } \delta \in < 0, r - \epsilon > \\ 0 & \text{otherwise} \end{cases}$$

$$model_o^r(\delta) = \begin{cases} 1 - \left(\frac{\delta - r}{\epsilon}\right)^2, & \text{for } r < X \wedge \delta \in < r - \epsilon, r + \epsilon > \\ 0 & \text{otherwise} \end{cases}$$



Laser model - a practical approach



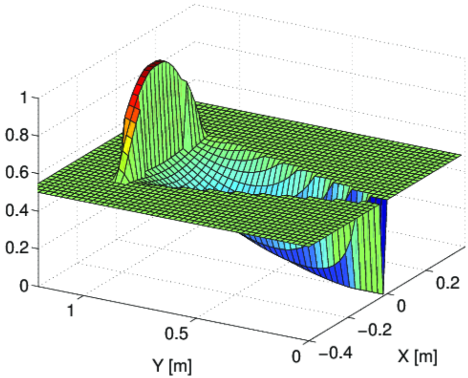
- Connect a cell corresponding the sensor position with the hit cell.
- Set all cells on the line as empty.
- Set the hit cell as occupied.
- Apply Bayes rule to update the grid.
- Use some line drawing algorithm (Bresenham).
- Improvement: use flood-fill algorithm to draw the whole scan.

Sensor model for sonar

Probability density $p(z_t|m_t^{[xy]})$ is defined:

$$p(z_t|m_t^{[xy]}) = \frac{1 + model_O^{z_t}(\alpha, r) - model_V^{z_t}(\alpha, r)}{2},$$

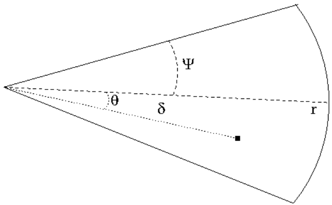
where (α, r) are polar coordinates of the cell $m_t^{[xy]}$ in sensor coordinate system and z_t is measured distance.



Sensor model for sonar (Elfes)

Model is defined by:

- width of the signal: Ψ
- precision of sensor measurement: ϵ



For measured distance r we get:

$$\begin{aligned} model_v^r(\delta,\phi) &= V_r(\delta)A_n(\phi) \\ model_o^r(\delta,\phi) &= O_r(\delta)A_n(\phi), \end{aligned}$$

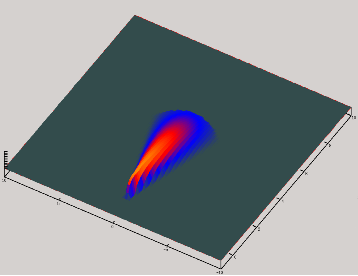
where

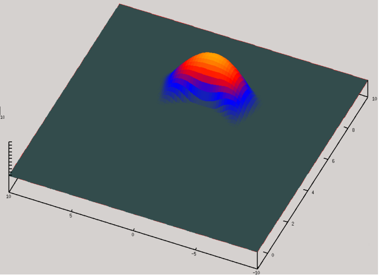
$$V_r(\delta) = \begin{cases} 1-\left(\frac{\delta}{r}\right)^2, & \textit{for} \quad \delta \in < 0, r-\epsilon > \\ 0 & \text{otherwise} \end{cases}$$

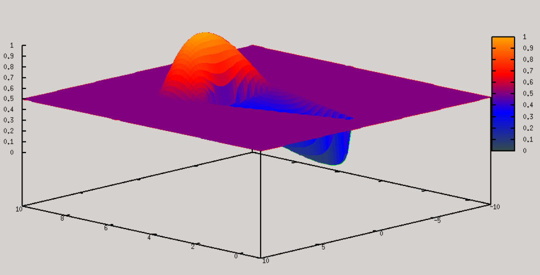
$$O_r(\delta) = \begin{cases} 1-\left(\frac{\delta-r}{\epsilon}\right)^2, & \textit{for} \quad \delta \in < r-\epsilon, r+\epsilon > \\ 0 & \text{otherwise} \end{cases}$$

$$A_n(\phi) = \begin{cases} 1-\left(\frac{2\phi}{\Psi}\right)^2, & \textit{for} \quad \phi \in \left\langle -\frac{\Psi}{2}, \frac{\Psi}{2} \right\rangle \\ 0 & \text{otherwise} \end{cases}$$

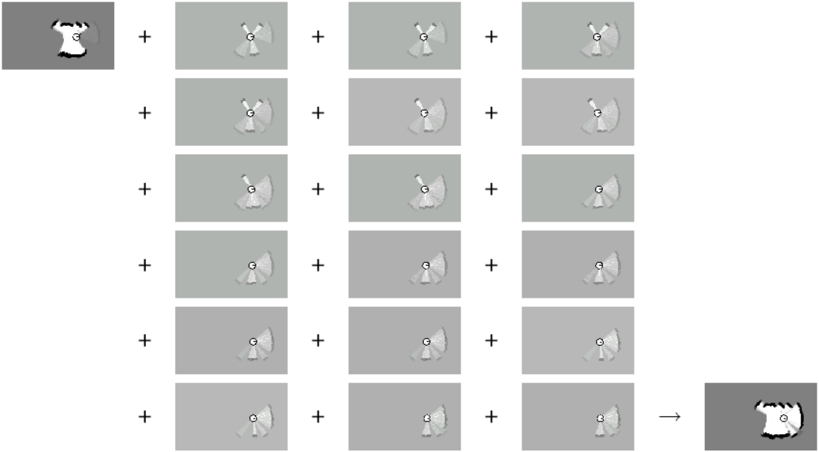
Sensor model for sonar



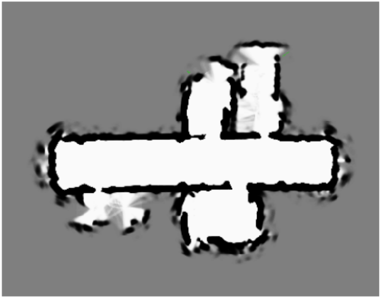


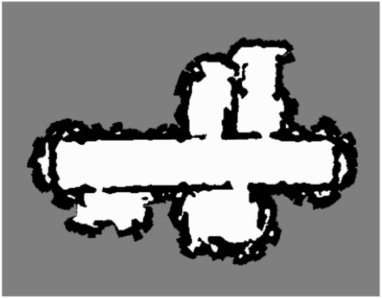


Example - incremental updating of occupancy grids



Example - map obtained with ultrasound sensors





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

Alternative: Simple counting

Reflection maps

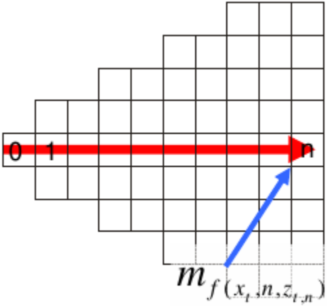
- For every cell count
  - *hits*(x, y): number of cases where a beam ended at $\langle x, y \rangle$
  - *misses*(x, y): number of cases where a beam passed through $\langle x, y \rangle$

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

- Value of interest: $p((reflects(x, y)))$

The measurement model

pose at time t: x_t
beam n of scan t: $z_{t,n}$
maximum range reading: $\zeta_{t,n} = 1$
beam reflected by an object: $\zeta_{t,n} = 0$



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

Computing the most likely mapping

- Compute values for m that maximize

$$m^* = \arg\max_m p(m|z_1, z_2, \ldots, z_t, x_1, x_2, \ldots, x_t)$$

- Assuming an uniform prior probability for $p(m)$, this is equivalent to maximizing (apply Bayes rule):

$$\begin{aligned} m^* &= \arg\max_m p(z_1, z_2, \ldots, z_t | m, x_1, x_2, \ldots, x_t) \\ &= \arg\max_m \prod_{t=1}^{T} p(z_t | m, x_t) \\ &= \arg\max_m \sum_{t=1}^{T} \ln p(z_t | m, x_t) \end{aligned}$$

Computing the most likely mapping

$$
\begin{aligned}
m^* &= \arg\max_m \left[\sum_{j=1}^{J} \sum_{t=1}^{T} \sum_{n=1}^{N} \left(I\left(f(x_t, n, z_{t,n}) = j\right)(1 - \zeta_{t,n}) \ln m_j \right.\right. \\
&+ \left.\left. \sum_{k=0}^{z_{t,n}-1} I\left(f(x_t, n, k) = j\right) \ln(1 - m_j) \right) \right]
\end{aligned}
$$

Suppose the number of times a beam

- that is not a maximum range beam ended in cell j ($hits(j)$).

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

- intercepted cell j without ending in it ($misses(j)$).

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

Computing the most likely mapping

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

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

we obtain

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

$$\Downarrow$$

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.

Comparison

Occupancy map $\times$ Reflection map

Grid maps - summary

- Occupancy 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.
- It stores the posterior probability that the corresponding area in the environment is occupied.
- Occupancy grid maps 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.
- We provided a sensor model for computing the likelihood of measurements and showed that the counting procedure underlying reflection maps yield the optimal map.