Mapping
Occupancy Grid Mapping

Maria Isabel Ribeiro
Pedro Lima
with revisions introduced by Rodrigo Ventura
Instituto Superior Técnico/Instituto de Sistemas e Robótica

October 2008

Course Handouts

References

- S. Thrun, W.Burgard, D. Fox, “Probabilistic Robotics”, MIT Press, 2005
  - Occupancy Grid (Chap. 10)

- S. Thrun, Learning occupancy grid maps with forward sensor models. Autonomous robots, 15(2), 111-127, 2003.

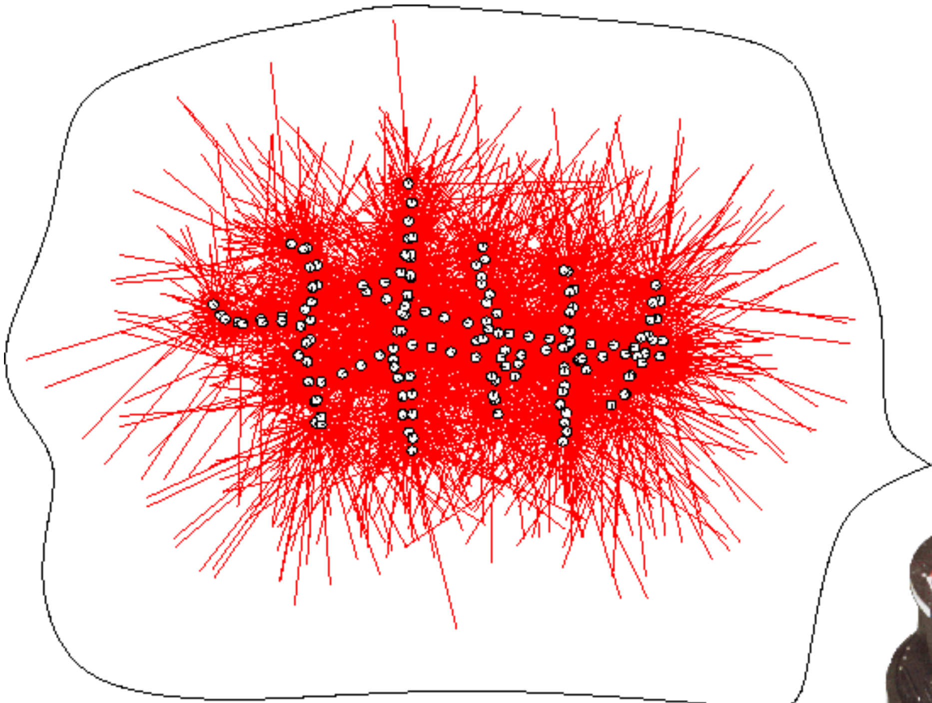
Mapping

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.

From handouts associated with
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

The General Problem of Mapping



What does the environment look like?

From handouts associated with
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

The General Problem of Mapping

- Formally, mapping involves, given the sensor data,

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

to calculate the most likely map

u_i - actions

z_i - observations

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

m – the map

From handouts associated with
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

Mapping as a Chicken and Egg Problem

- So far we learned how to estimate the pose of the vehicle 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 section we will describe how to calculate a map given we know the pose of the vehicle.

From handouts associated with
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

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.

From handouts associated with
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

Difficulty of the Mapping Problem

Range data acquired by a laser scanner collected in a large indoor environment

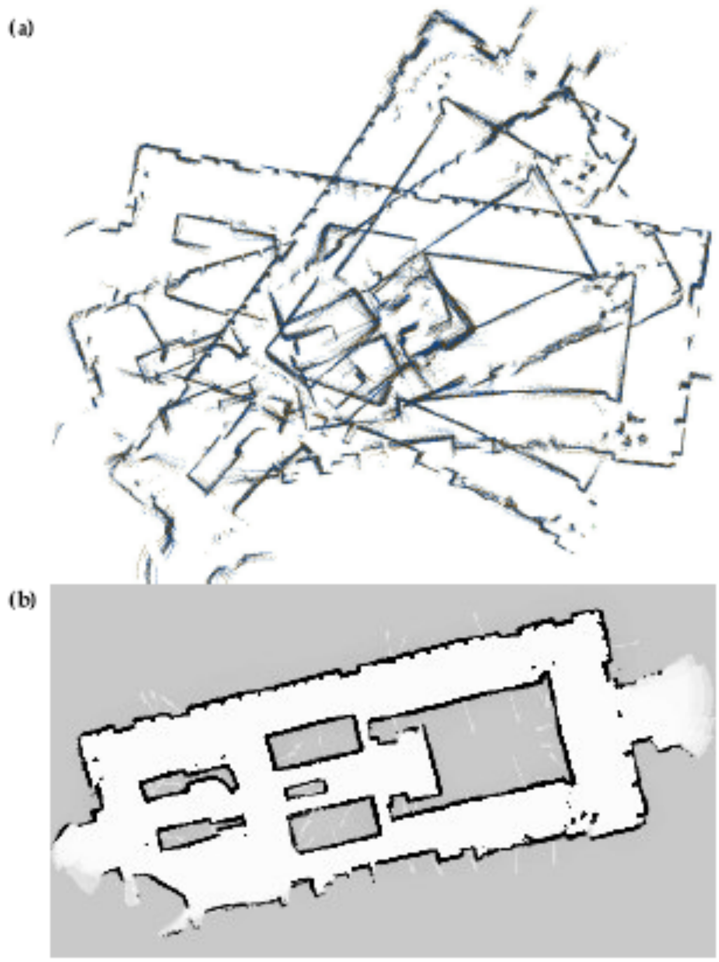


Figure 9.1 (a) Raw range data, position indexed by odometry. (b) Occupancy grid map.

Raw data. Figure generated using the robot's odometry information

Occupancy grid map

Figure from
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

Mapping with known poses

- Build the map, assuming that the robot(sensor) pose is known with no error.

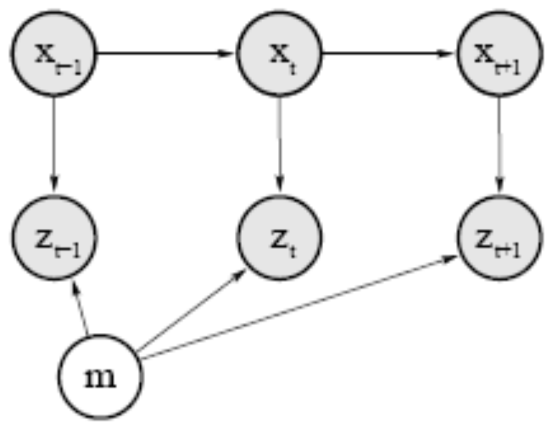


Figure 9.2 Graphical model of mapping with known poses. The shaded variables (poses x and measurements z) are known. The goal of mapping is to recover the map m.

Figure from
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

Occupancy Grid Mapping

- Map is a fine-grained grid defined over the continuous space of locations
  - The most common is a 2D grid map, that corresponds to a 2D slice of a 3D world.
- An occupancy grid map partitions the space into finitely many grid cells

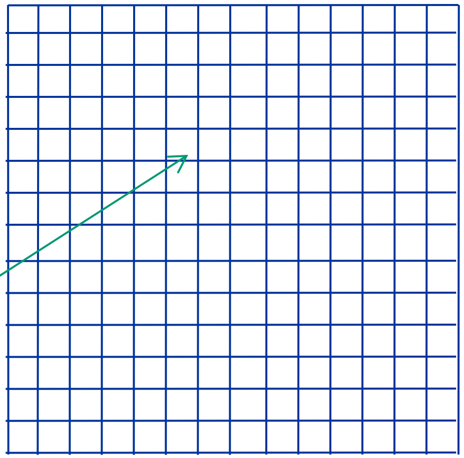
- m = the map
- $z_{1:t}$ = set of all measurements up to time t
- $x_{1:t}$ = the path of the robot defined through the sequence of poses

Mapping = estimate

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

$$m = \{ m_i \}$$

$$i = 1, \dots, M$$



m_i

grid cell with index i

Occupancy Grid Mapping

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

The number of grid cells might very large
The calculation of the posterior probability might be intractable

Ignoring dependencies among neighboring cells

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

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

to be solved by a Binary Bayes Filter

Recursive in t

- Each m_i has attached to it a binary occupancy value
- 1 for occupied, 0 for free
- $p(m_i)=p(m_i=1)$ is the probability that the grid cell m_i is occupied

Occupancy Grid Mapping

$p(m_i \mid z_{1:t}, x_{1:t})$ takes values in [0, 1]

to avoid numerical instabilities for probabilities near zero, the occupancy grid mapping uses the *log odds* representation of occupancy

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

$$\boxed{\begin{aligned} \mathrm{odd}(y) &= \frac{p(y)}{1-p(y)} = \frac{p(y)}{p(\neg y)} \\ \ell(y) &:= \log \frac{p(y)}{1-p(y)} \end{aligned}}$$

with initial condition

$$\ell_{0,i} = \log \frac{p(m_i = 1)}{p(m_i = 0)} = \log \frac{p(m_i)}{1 - p(m_i)}$$

Occupancy Grid Mapping

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

$$\ell_{t,i} = \ell_{t-1,i} + \log \frac{p(m_i \mid z_t, x_t)}{1-p(m_i \mid z_t, x_t)} - \ell_0$$

$$\boxed{\ell_{t,i} = \ell_{t-1,i} + InverseSensorModel(m_i, x_t, z_t) - \ell_0}$$

actual probabilities recovered as

$$p(m_i \mid z_{1:t}, x_{1:t}) = 1 - \frac{1}{1+\exp\{\ell_{t,i}\}}$$

Binary filter with static state

See the proof in Section 4.2 of Probabilistic Robotics

Inverse range-sensor model

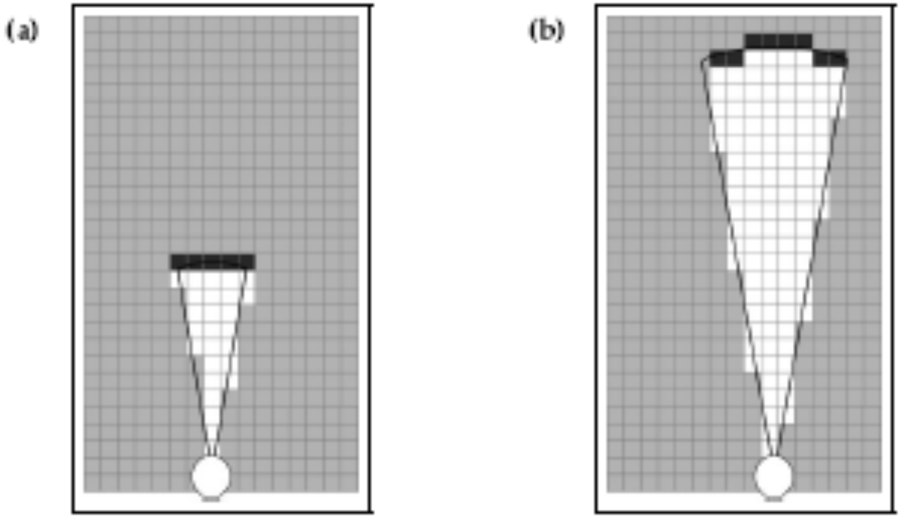
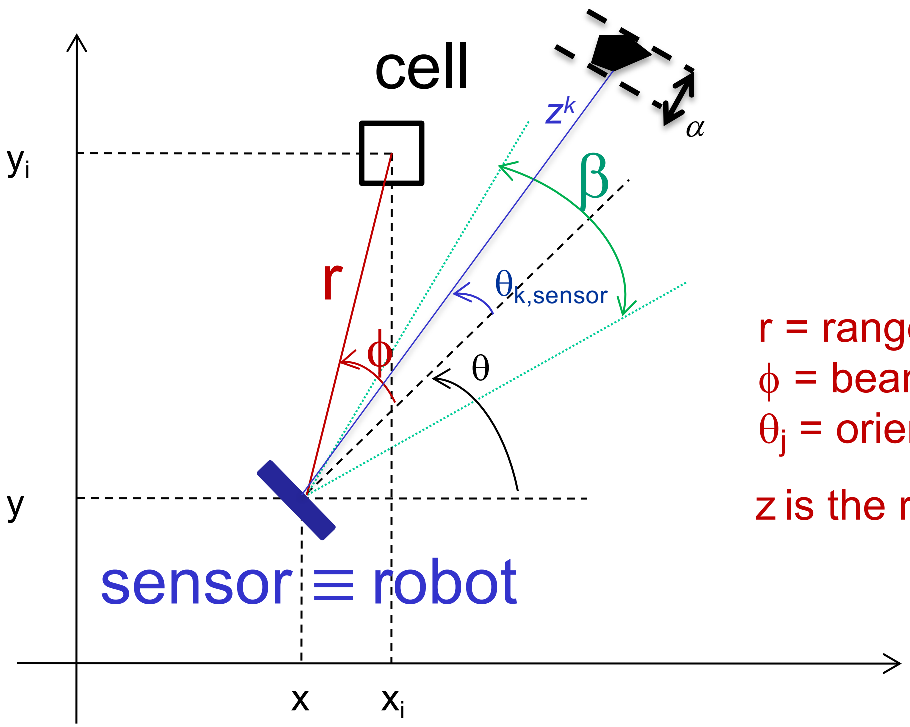


Figure 9.3 Two examples of an inverse measurement model **inverse_range_sensor_model** for two different measurement ranges. The darkness of each grid cell corresponds to the likelihood of occupancy. This model is somewhat simplistic; in contemporary implementations the occupancy probabilities are usually weaker at the border of the measurement cone.

Inverse range-sensor model notation



α = thickness of obstacles

β = angular width of the sensor beam

(x,y,θ) is the robot pose

r = range for the center of mass of cell m_i

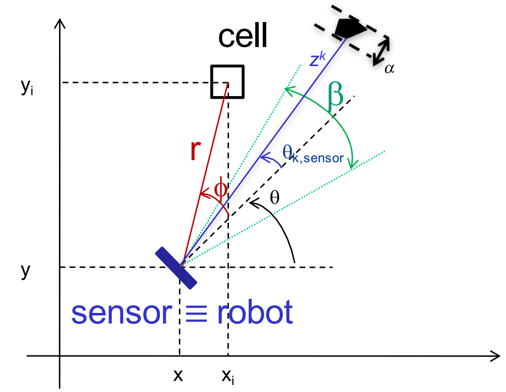
ϕ = bearing for the center of mass of cell m_i

θ_j = orientation of sensor *j*

z is the range measurement

Assuming that the sensor *j* is at the same location (position and orientation) as the robot ($\theta = \theta_{j,sensor}$)

Inverse range-sensor model



Algorithm inverse_range_sensor_model($\mathbf{m}_i, x_t, z_t$):

Let x_i, y_i be the center-of-mass of $\mathbf{m}_i$

$r = \sqrt{(x_i - x)^2 + (y_i - y)^2}$

$\phi = \text{atan2}(y_i - y, x_i - x) - \theta$

$k = \text{argmin}_j \, |\phi - \theta_{j,\text{sens}}|$

if $r > \min(z_{\max}, z_t^k + \alpha/2)$ *or* $|\phi - \theta_{k,\text{sens}}| > \beta/2$ *then*

return l_0 — if cell is out of the sensor cone, mark it with default l_0

if $z_t^k < z_{\max}$ *and* $|r - z_t^k| < \alpha/2$

return l_{occ} — if range measurement is less than max and within object thickness, mark cell as occupied ($l_{occ} > l_0$)

if $r \leq z_t^k$

return l_{free} — if range measurement is larger than distance to cell, mark cell as free ($l_{free} < l_0$)

endif

α = thickness of obstacles
β = angular width of the sensor beam
$x_t = \langle x, y, \theta \rangle$

r = range for the center of mass of cell m_i
ϕ = bearing for the center of mass of cell m_i
k = beam index for the beam bearing closest to ϕ
$\theta_{k,sens}$ = bearing measurement for beam *k*
z_t^k = the range measurement for beam *k*

Occupancy Grid Mapping

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

    for all cells $\mathbf{m}_i$ *do*

        if $\mathbf{m}_i$ *in perceptual field of* z_t *then*

            $l_{t,i} = l_{t-1,i} + \boxed{\textbf{inverse\_sensor\_model}(\mathbf{m}_i, x_t, z_t)} - l_0$

        else

            $l_{t,i} = l_{t-1,i}$

        endif

    endfor

    return $\{l_{t,i}\}$

implements the inverse range sensor model

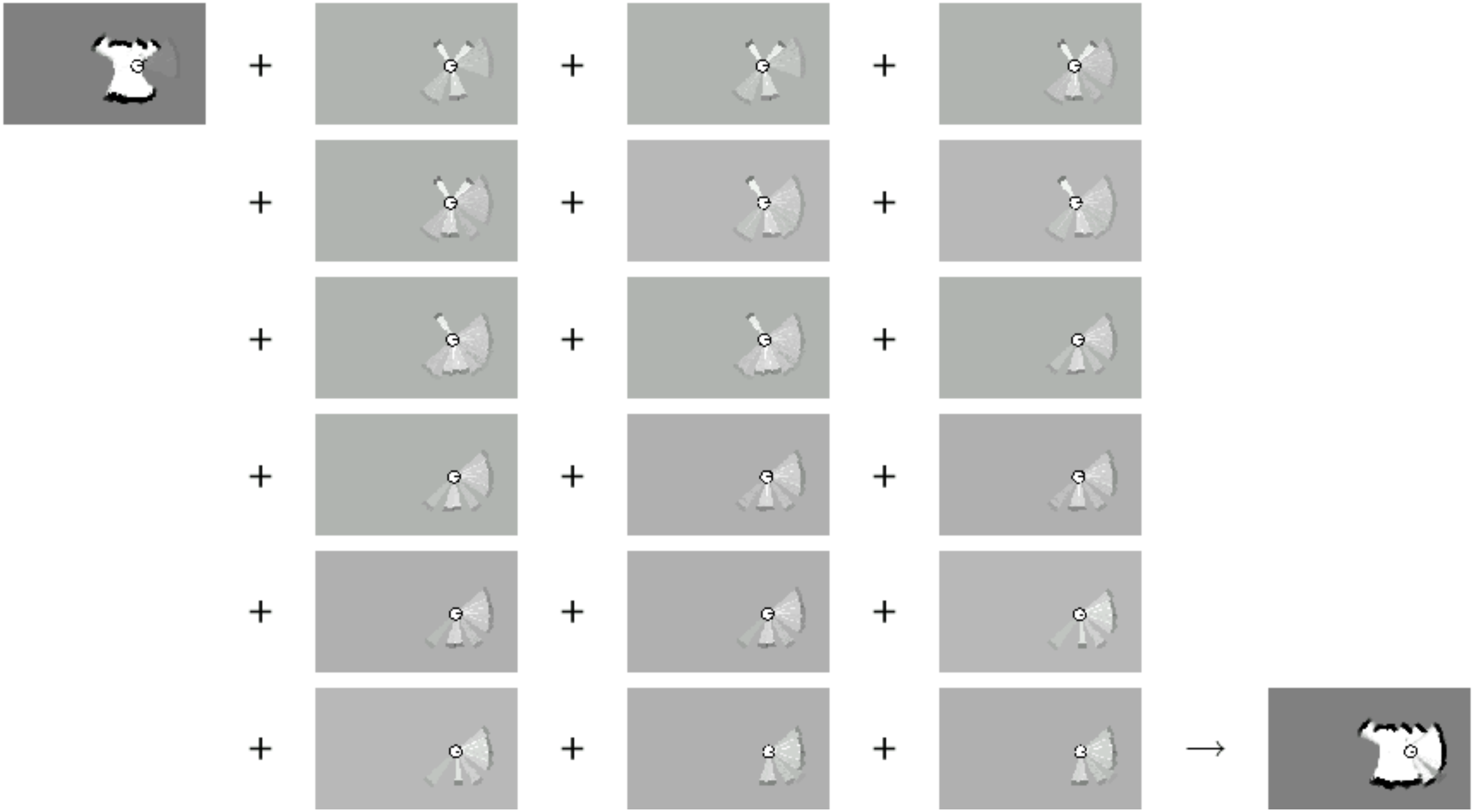
$$p(m_i \mid z_t, x_t)$$

in its log odds form

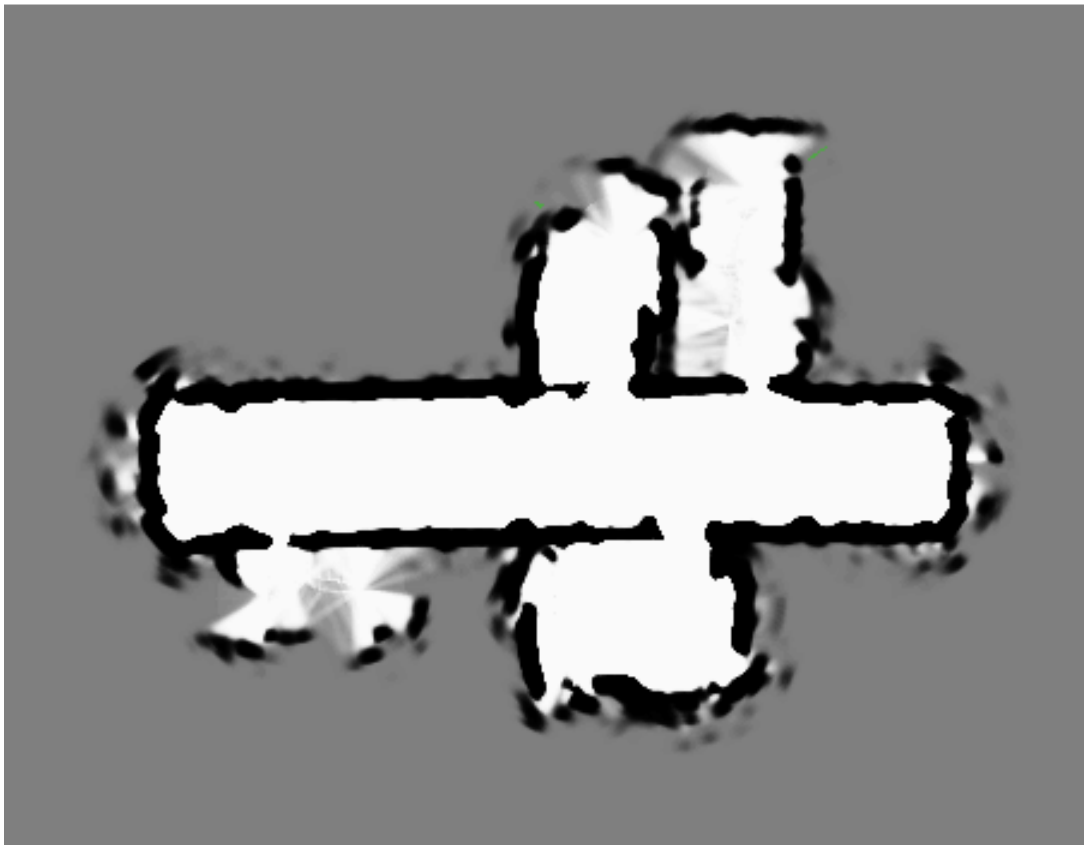
$$p(m_i \mid z_{1:t}, x_{1:t}) = 1 - \frac{1}{1 + \exp\{\ell_{t,i}\}}$$

$$inverse\_sensor\_model(m_i, x_t, z_t) = \log \frac{p(m_i \mid z_t, x_t)}{1 - p(m_i \mid z_t, x_t)}$$

Incremental Updating of Occupancy Grids (Example)



Resulting Map Obtained with Ultrasound Sensors



From handouts associated with
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

Occupancy Grid Mapping (Examples)

Figure 9.5 Occupancy probability map of an office environment built from sonar measurements. Courtesy of Cyrill Stachniss, University of Freiburg.

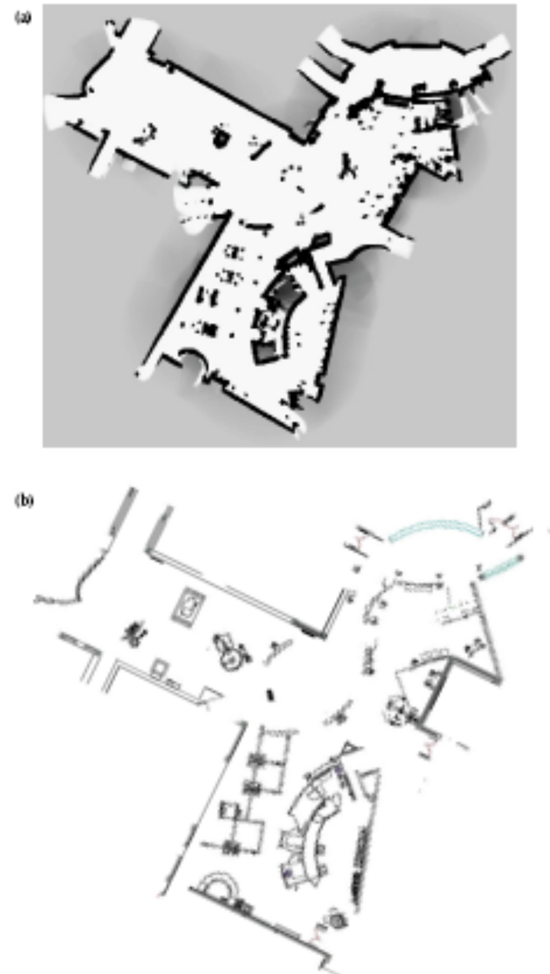
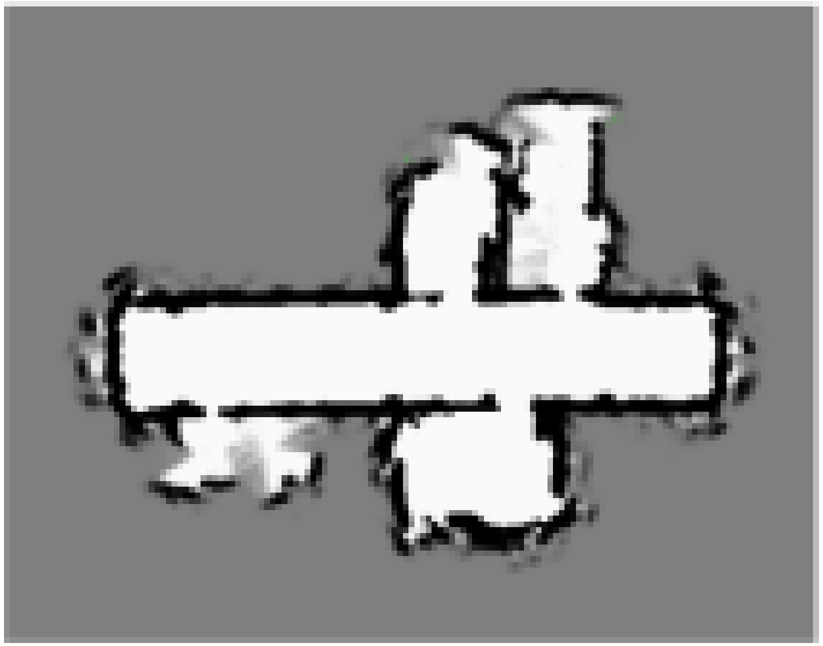
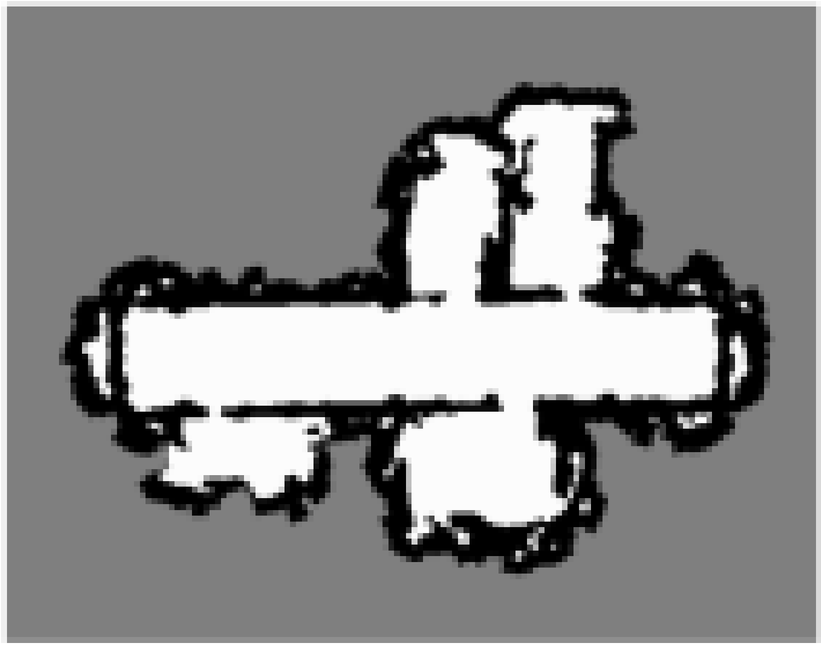


Figure 9.6 (a) Occupancy grid map and (b) architectural blue-print of a large open exhibit space. Notice that the blue-print is inaccurate in certain places.

Resulting Occupancy and Maximum Likelihood Map





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

From handouts associated with
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox

Mapping with forward sensor models

- Mapping using <u>inverse</u> sensor models:
  find map *M* maximizes $P(M|Z,X)$ given $P(m_i|Z,X)$
  where *Z* and *X* are all measurements and poses
- Mapping using <u>forward</u> sensor models:
  find map *M* that maximizes $P(Z|M,X) = \prod_i P(z_i|M,X)$
- Correspondences are latent variables c_i are introduced modeling the cause of a measurement
  - $c_{i,*} = 1$, iff spurious random measurement (e.g., reflection)
  - $c_{i,0} = 1$, iff no obstacle detected and the sensor returned max range
  - $c_{i,k} = 1$, iff hit from the k-th obstacle in the sensor cone
- Map *M* is discrete (occupied or free)

Mapping with forward sensor models

- Algorithm uses a Expectation-Maximization (EM) approach:
  1. **Initialization**: empty map *M*
  2. **E-step**: obtain the expected values of the correspondences c_i given map *M*, measurements *Z* and poses *X*
  3. **M-step**: finds the map M maximizing likelihood $P(Z, C|M, X)$, where *C* is all the correspondences, by flipping map cells that increase likelihood until local minimum
  4. go to step (2) until convergence, i.e., map does not change after one iteration

Mapping with forward sensor models

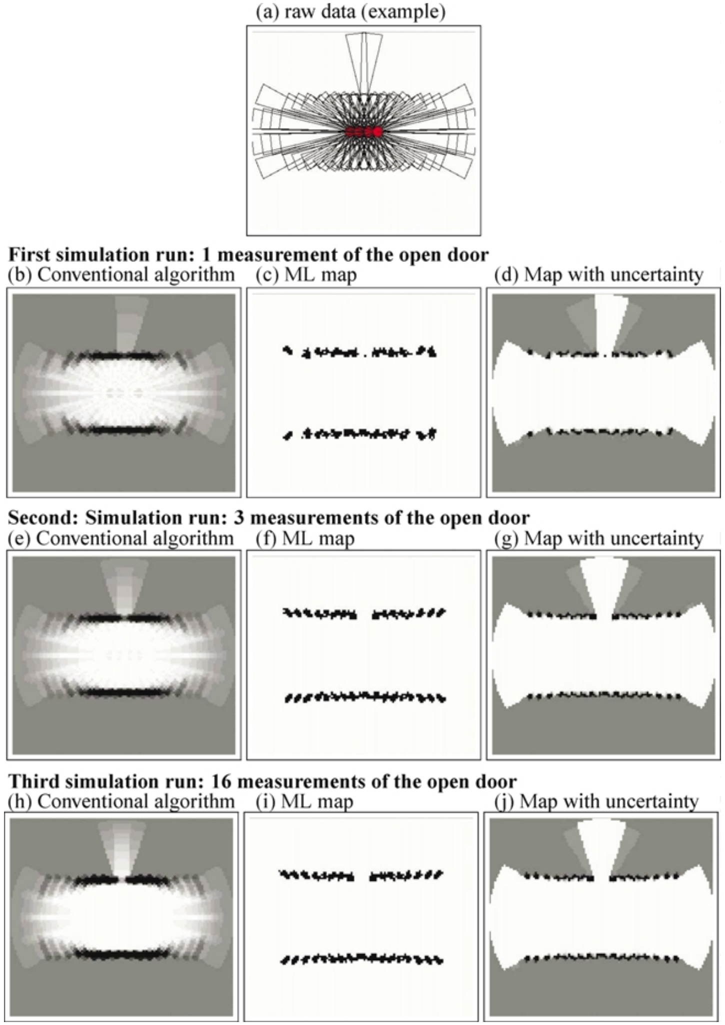
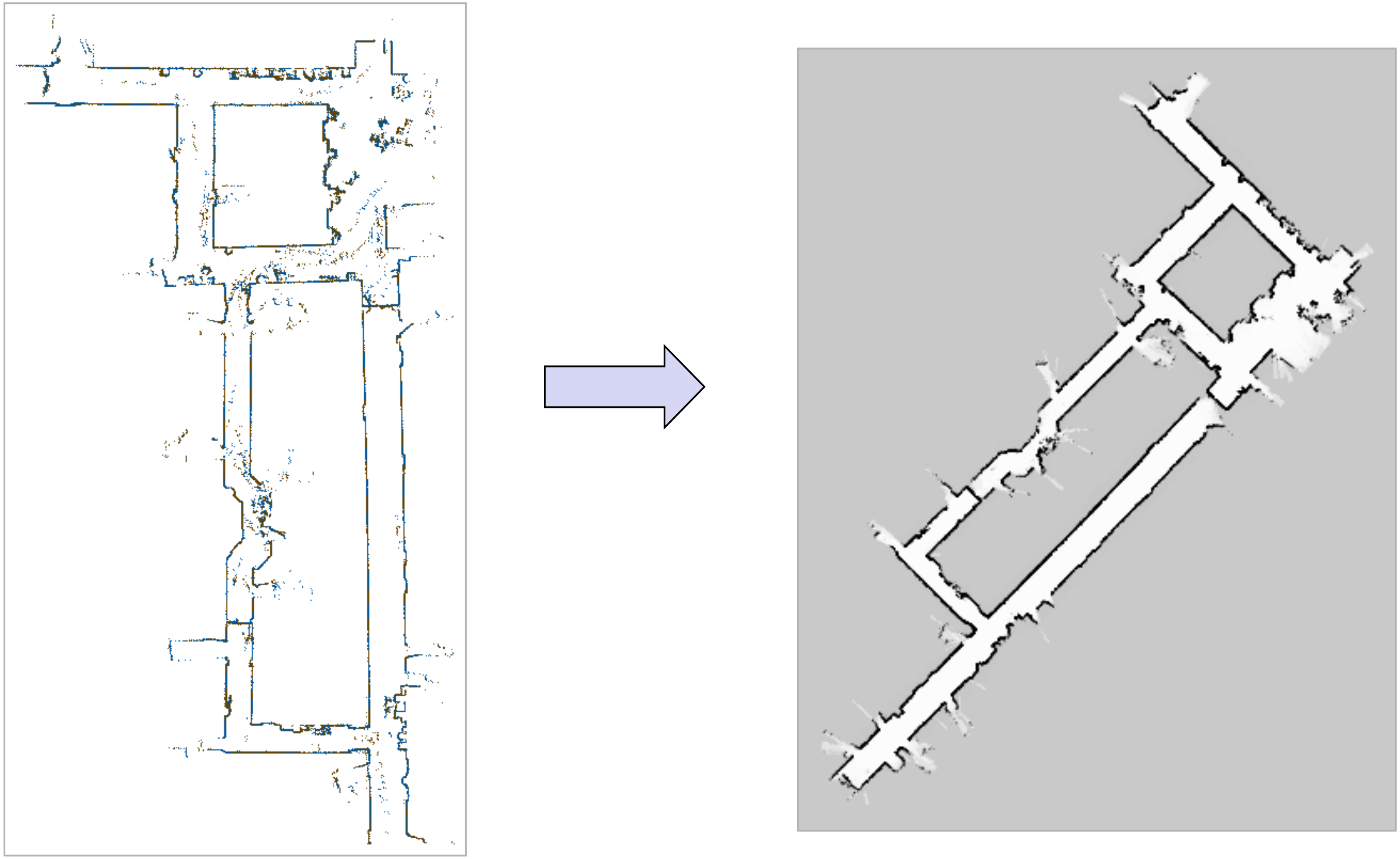


Figure 4. Simulation results: The robot travels along a corridor with an open door. When only a single measurement detects the open door (diagrams (b) through (d)), conventional occupancy grid maps do not show the open door (diagram (b)), whereas our approach does (diagram (d)). The same is the case for a data set where the open door was detected three times (diagrams (e) through (g)). With 16 measurements of the open door, the regular occupancy map show an open door (diagram (h)). Each grid consists of 122 by 107 grid with 10 centimeter resolution.

Occupancy Grids: From scans to maps



From handouts associated with
Probabilistic Robotics
S. Thrun, W. Burgard, D.Fox