



CS-417 INTRODUCTION TO ROBOTICS AND INTELLIGENT SYSTEMS

Ultrasonic Sensing and Mapping

Introduction to Mapping

- What the world looks like?
- Knowledge representation
 - Robotics, AI, Vision
- Who is the end-user?
 - Human or Machine
- Ease of Path Planning
- **Uncertainty!**



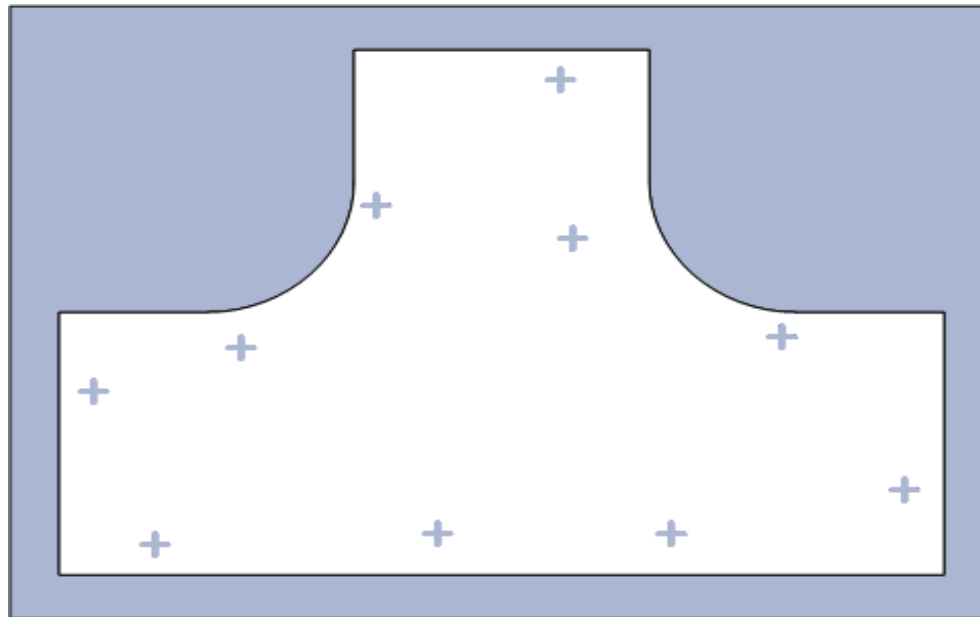
Simultaneous Localization And Mapping

SLAM is the process of building a map of an environment while, at the same time, using that map to maintain the location of the robot.

- Problems for SLAM in large scale environments:
 - Controlling growth of uncertainty and complexity
 - Achieving autonomous exploration



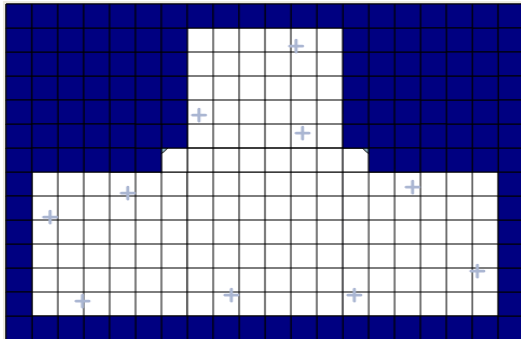
Consider this Environment:



Three Basic Map Types

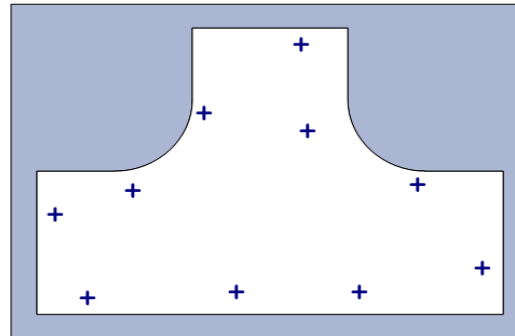
Grid-Based:

Collection of discretized obstacle/free-space pixels



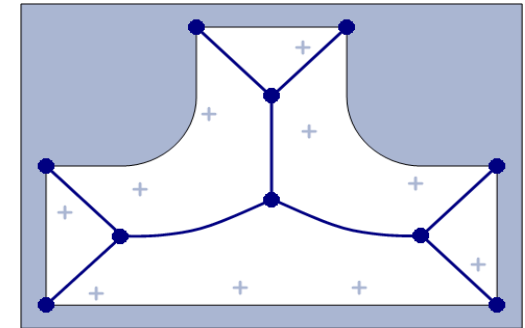
Feature-Based:

Collection of landmark locations and correlated uncertainty

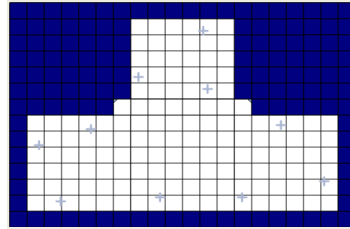


Topological:

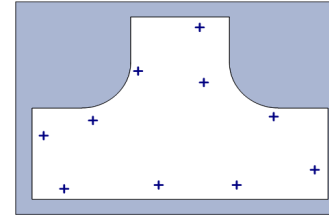
Collection of nodes and their interconnections



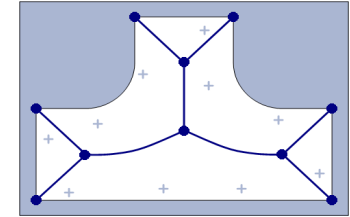
Three Basic Map Types



Grid-Based



Feature-Based



Topological

	Grid-Based	Feature-Based	Topological
Construction	Occupancy grids	Kalman Filter	Navigation control laws
Complexity	Grid size <i>and</i> resolution	Landmark covariance (N^3)	Minimal complexity
Obstacles	Discretized obstacles	Only structured obstacles	GVG defined by the safest path
Localization	Discrete localization	Arbitrary localization	Localize to nodes
Exploration	Frontier-based exploration	No inherent exploration	Graph exploration

Other Maps

	Appearance Based	Geometry Based	Mesh Based
Construction	Images	Lines, planes, etc	Mesh
Path Planning	N/A	Geometry based	Graph based
Localization	Arbitrary localization	Arbitrary localization	Arbitrary localization



Mapping

World

Robot

Map



Mapping

World

- Indoor/Outdoor
- 2D/2.5D/3D
- Static/Dynamic
- Known/Unknown
- Abstract (web)

Robot

Map



Mapping

World

Robot

- Mobile
 - Indoor/Outdoor
 - Walking/Flying/Swimming
- Manipulator
- Humanoid
- Abstract

Map



Mapping

World

Robot

Map

- Topological
- Metric
- Feature Based
- 1D, 2D, 2.5D, 3D



Mapping

World

- Indoor/Outdoor
- 2D/2.5D/3D
- Static/Dynamic
- Known/Unknown
- Abstract (web)

Robot

- Mobile
 - Indoor/Outdoor
 - Walking/Flying/Swimming
- Manipulator
- Humanoid
- Abstract

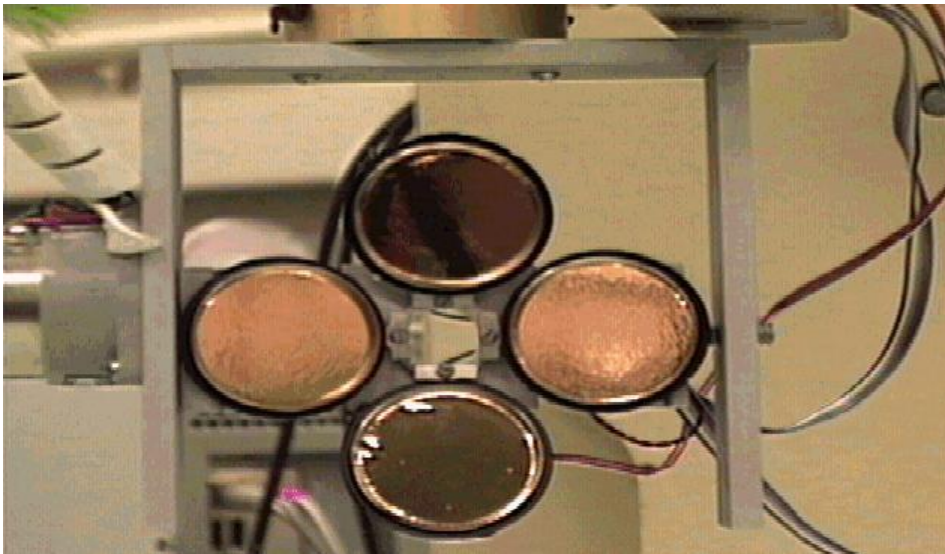
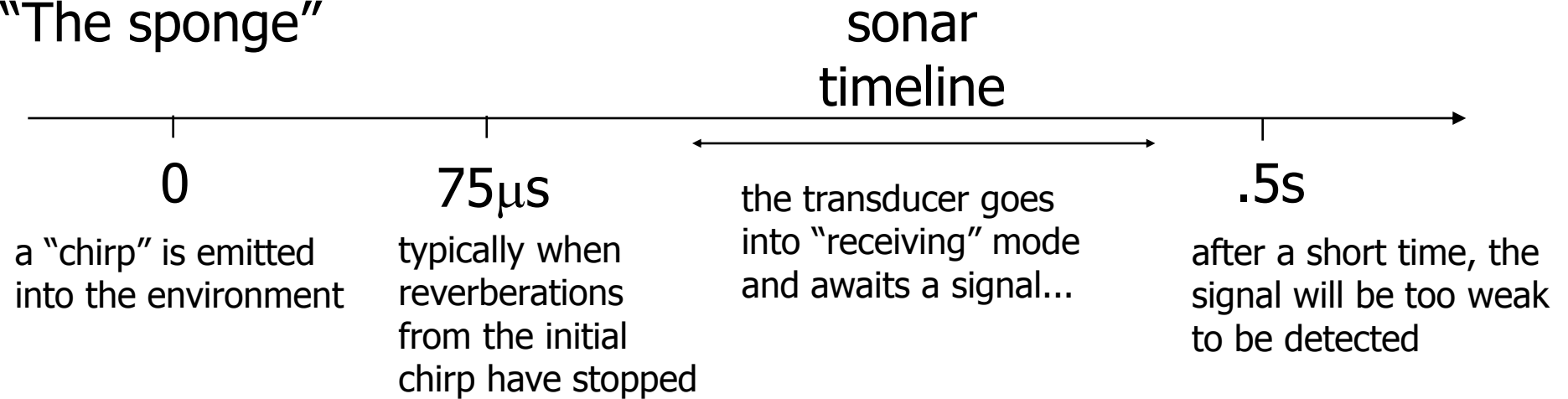
Map

- Topological
- Metric
- Feature Based
- 1D,2D,2.5D,3D



Sonar sensing

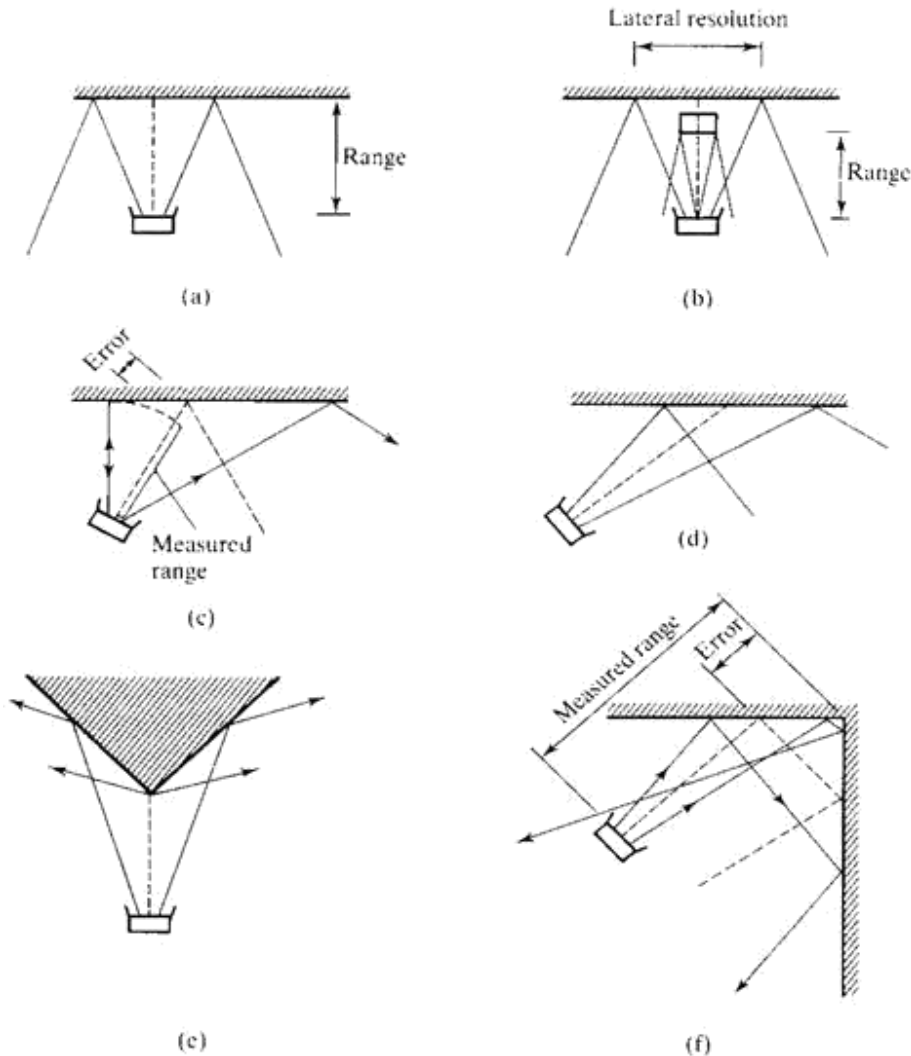
“The sponge”



Polaroid sonar emitter/receivers

Why is sonar sensing limited to between ~12 in. and ~25 feet ?

Sonar effects



(a) Sonar providing an accurate range measurement

(b-c) Lateral resolution is not very precise; the closest object in the beam's cone provides the response

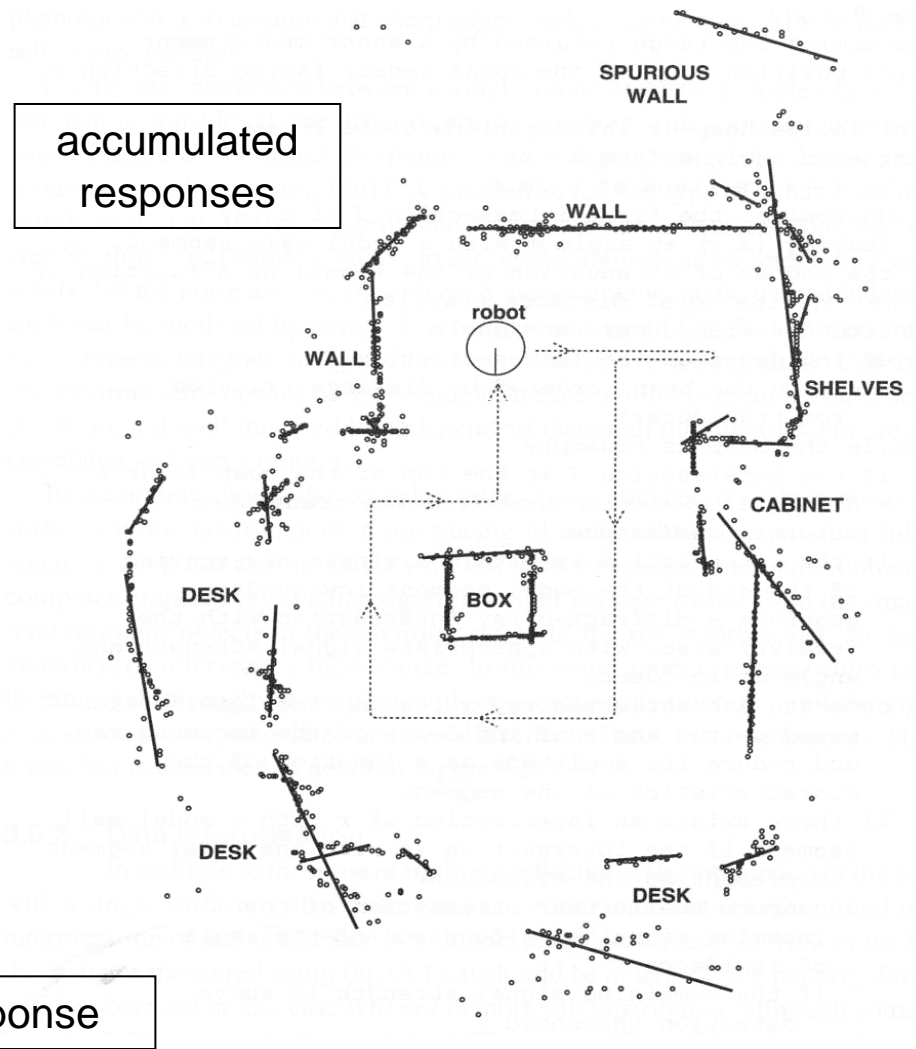
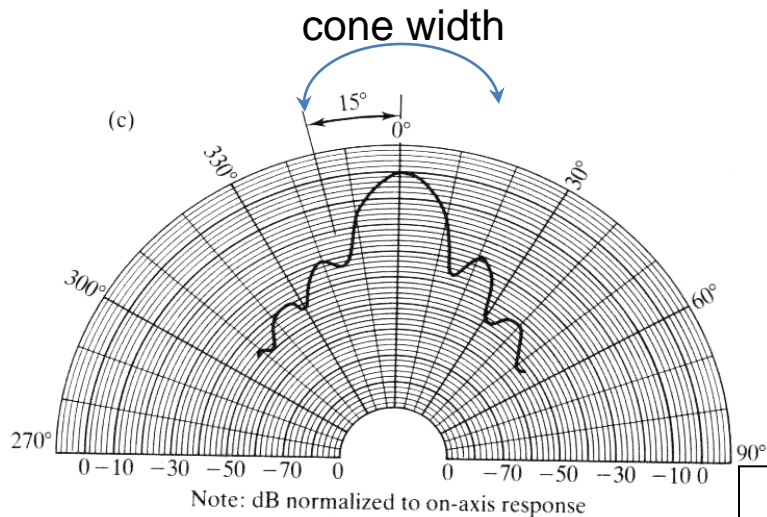
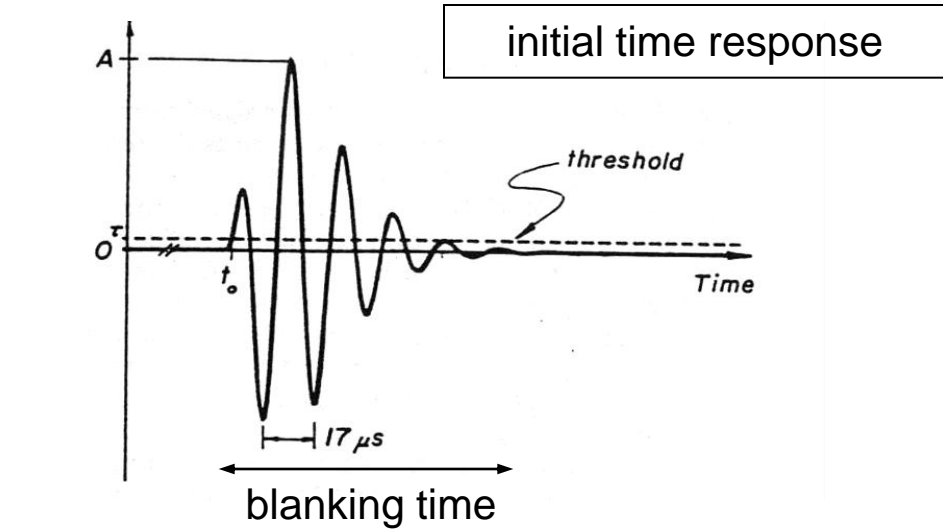
(d) Specular reflections cause walls to disappear

(e) Open corners produce a weak spherical wavefront

(f) Closed corners measure to the corner itself because of multiple reflections --> sonar ray tracing

resolution: time / space

Sonar modeling



spatial response

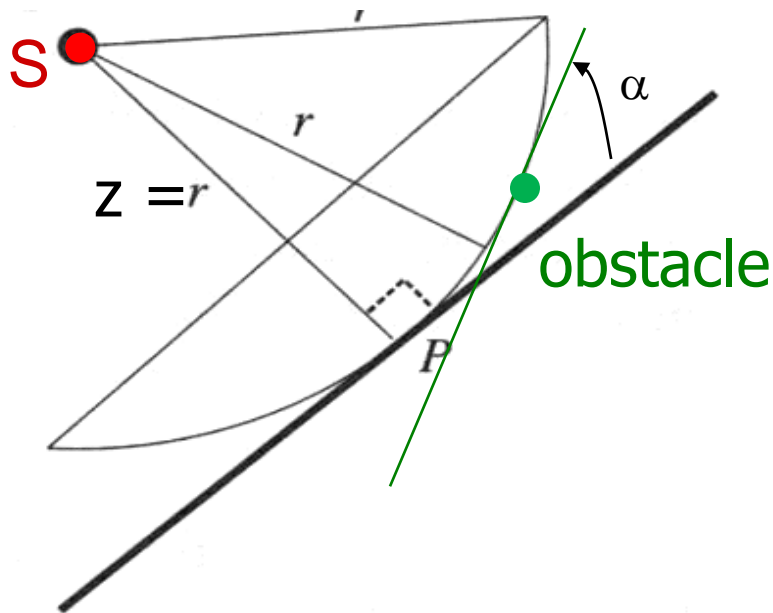


Sonar Modeling

response model (Kuc)

$$h_R(t, z, a, \alpha) = \frac{2c \cos \alpha}{\pi a \sin \alpha} \sqrt{1 - \frac{c^2(t - 2z/c)^2}{a^2 \sin^2 \alpha}}$$

sonar
reading



- Models the response, h_R , with:

c = speed of sound

a = diameter of sonar element

t = time

z = orthogonal distance

α = angle of environment surface

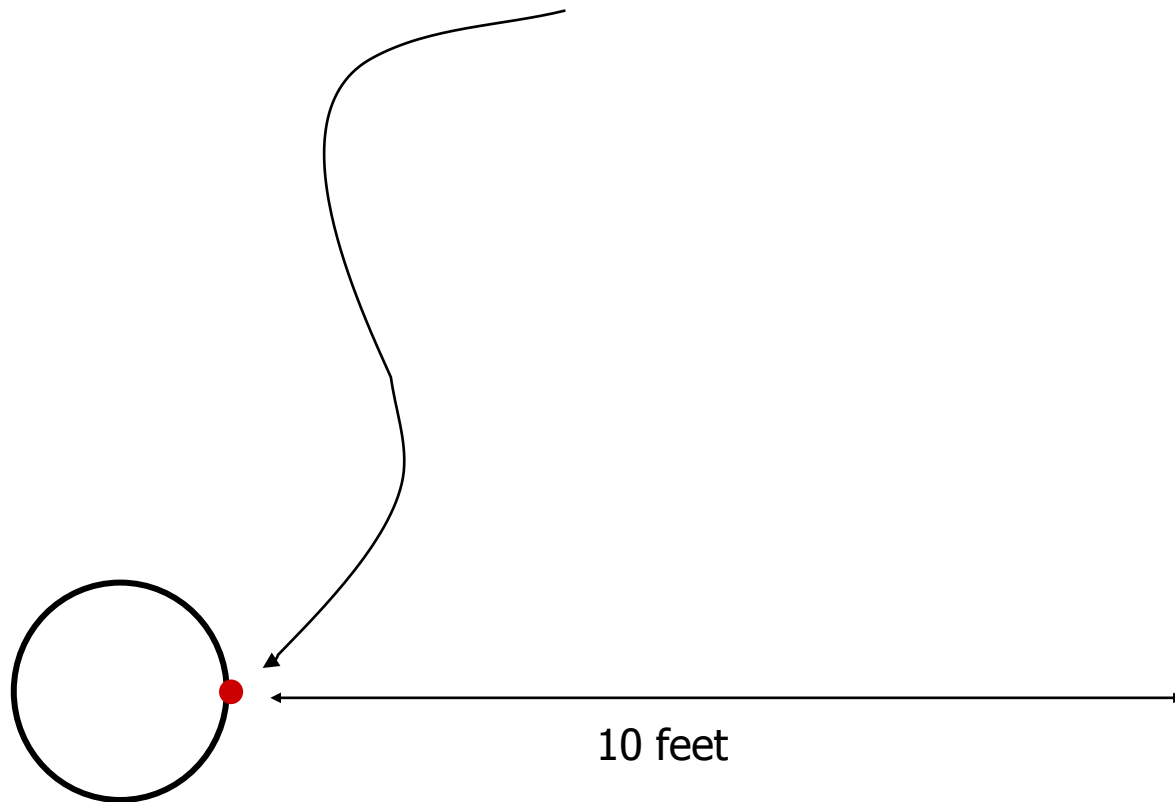
- Then, add noise to the model to obtain a probability: $p(S | o)$

chance that the sonar reading is S ,
given an obstacle at location O



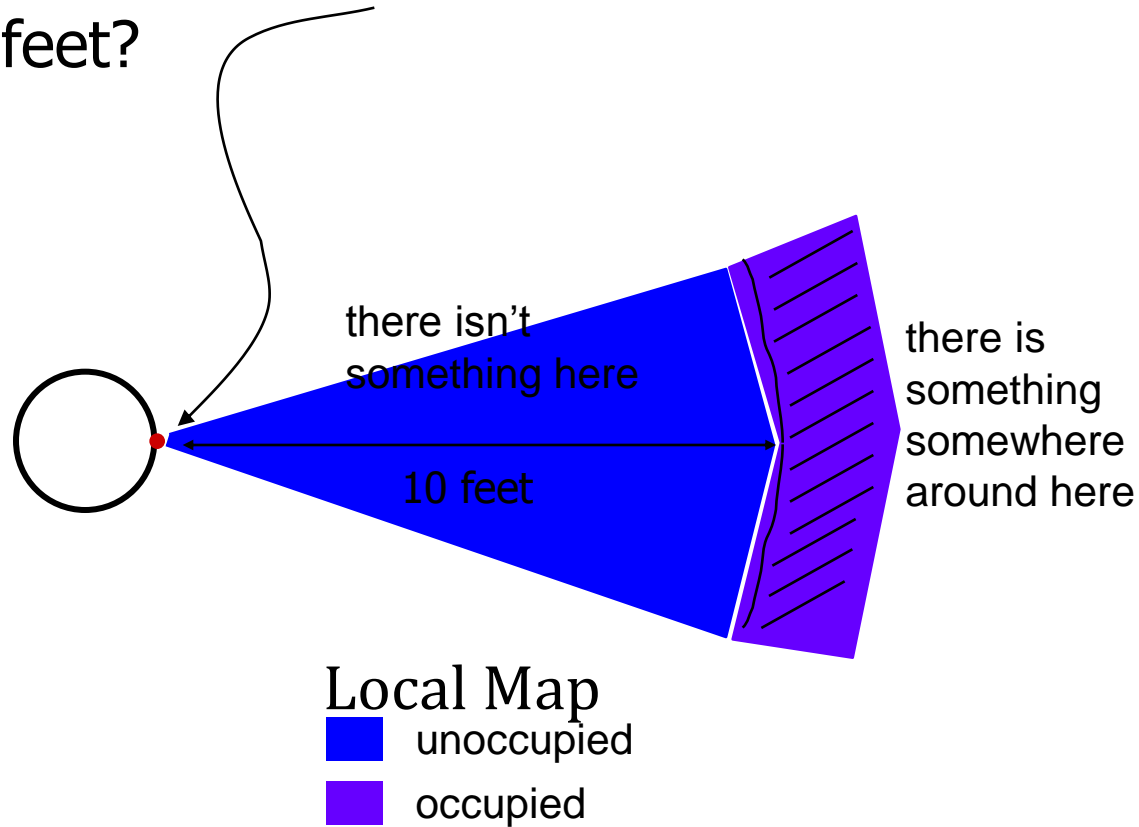
Using sonar to create maps

What should we conclude if this sonar reads 10 feet?



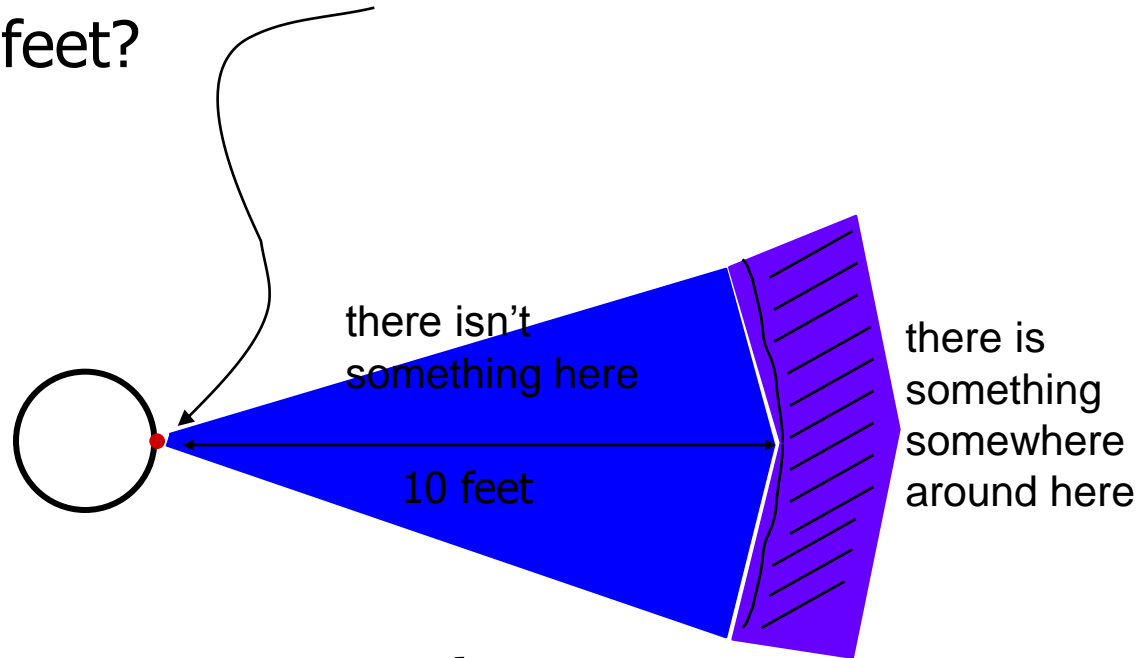
Using sonar to create maps

What should we conclude if this sonar reads 10 feet?






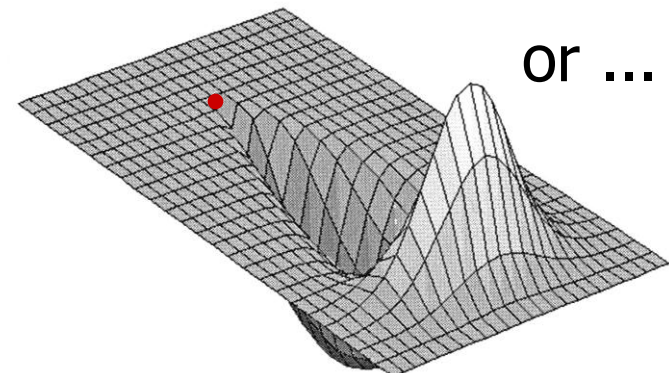
Using sonar to create maps

What should we conclude if this sonar reads 10 feet?



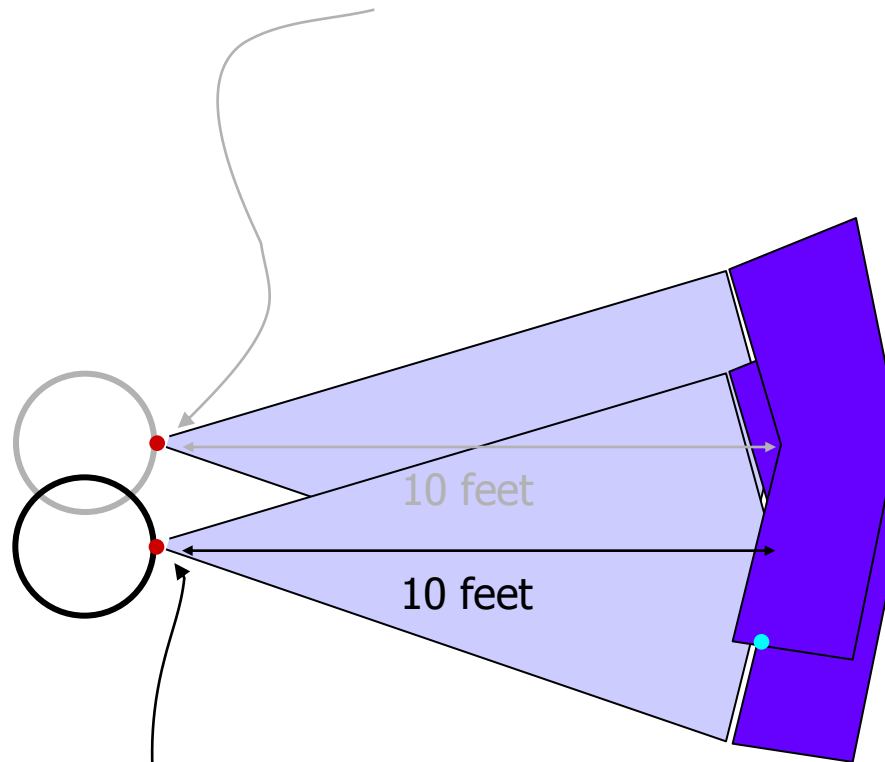
Local Map

-  unoccupied
-  no information
-  occupied



Using sonar to create maps

What should we conclude if this sonar reads 10 feet...

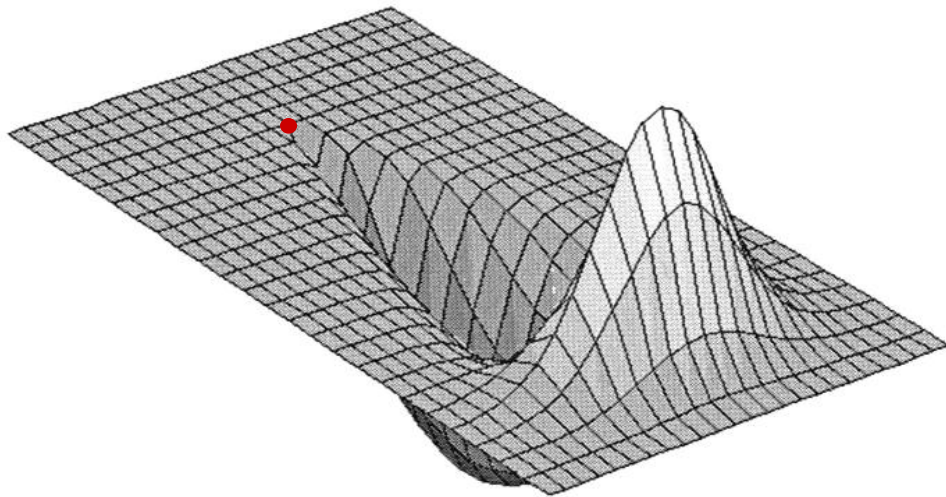


and how do we add the information that the next sonar reading (as the robot moves) reads 10 feet, too?



Combining sensor readings

- The key to making accurate maps is combining lots of data.
- But combining these numbers means we have to know what they are !



what is in each cell of this sonar model / map ?

What should our map contain ?

- small cells
- each represents a bit of the robot's environment
- larger values => obstacle
- smaller values => free

What is it a map of ?

Several answers to this question have been tried:

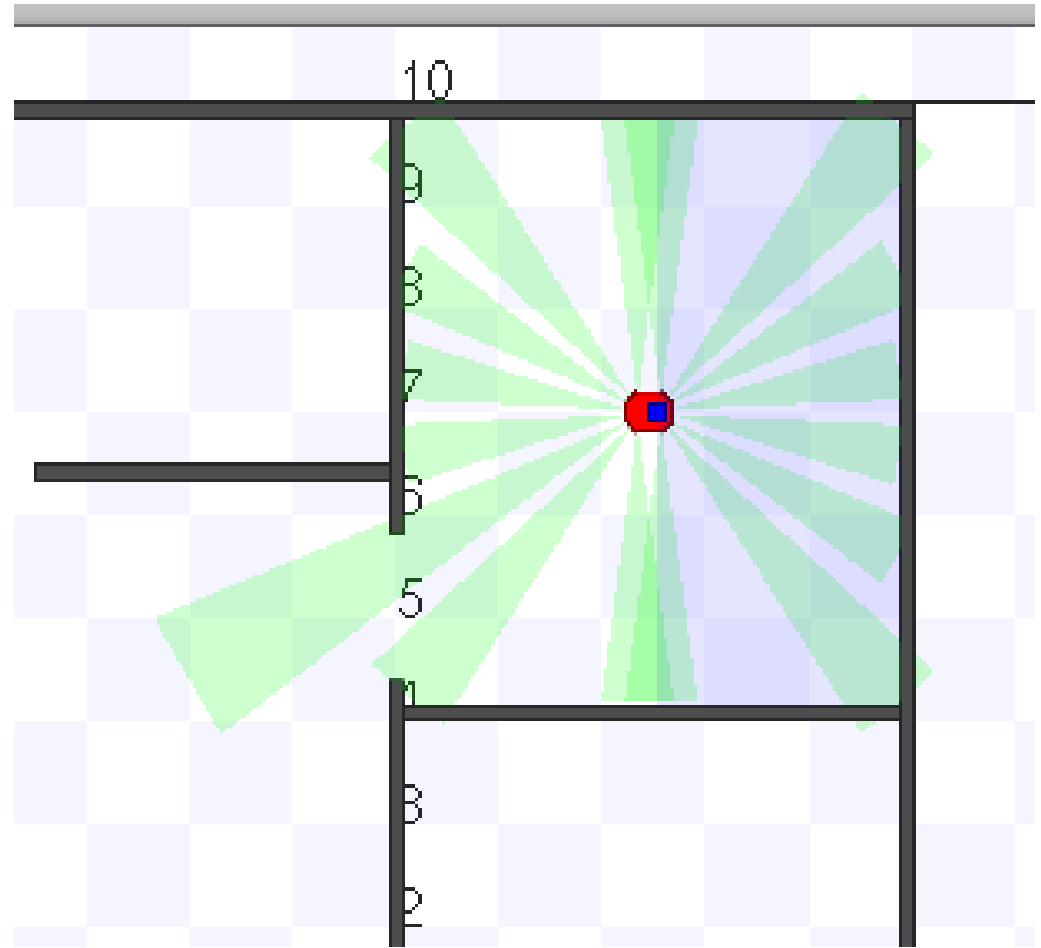
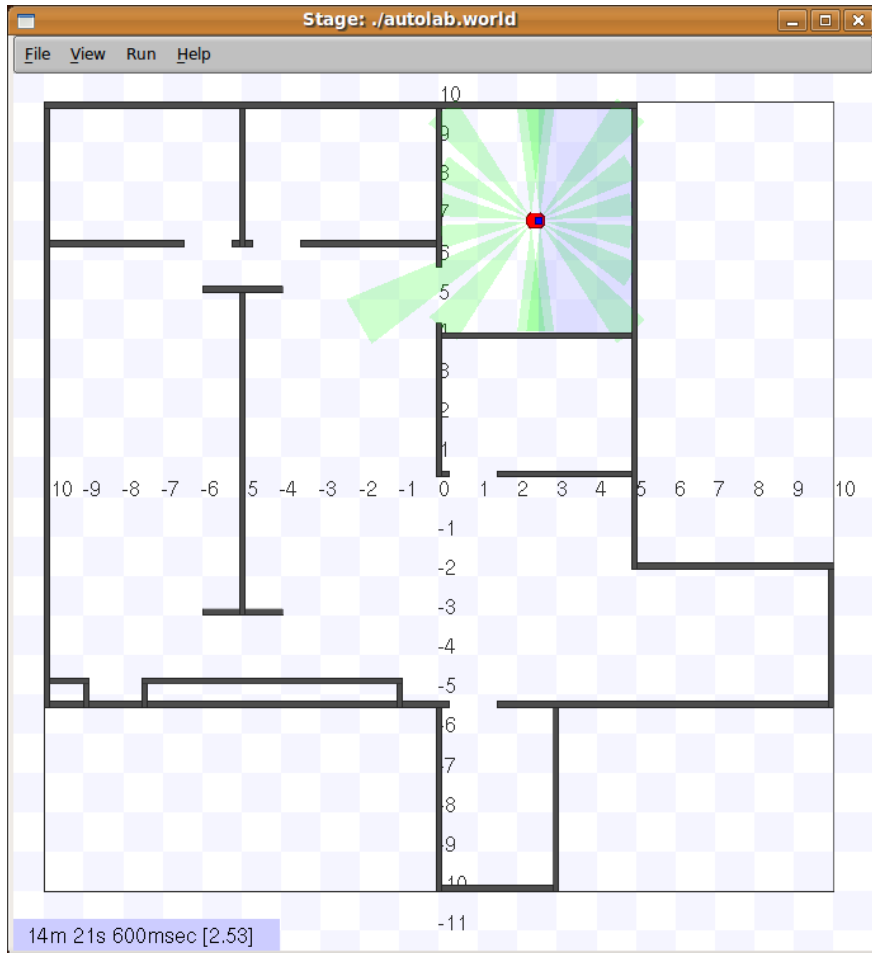
pre '83 It's a map of occupied cells. O_{xy} cell (x,y) is occupied \bar{O}_{xy} cell (x,y) is unoccupied

'83 - '88 It's a map of probabilities: $p(o | S_{1..i})$ The certainty that a cell is **occupied**, given the sensor readings S_1, S_2, \dots, S_i
 $p(\bar{o} | S_{1..i})$ The certainty that a cell is **unoccupied**, given the sensor readings S_1, S_2, \dots, S_i

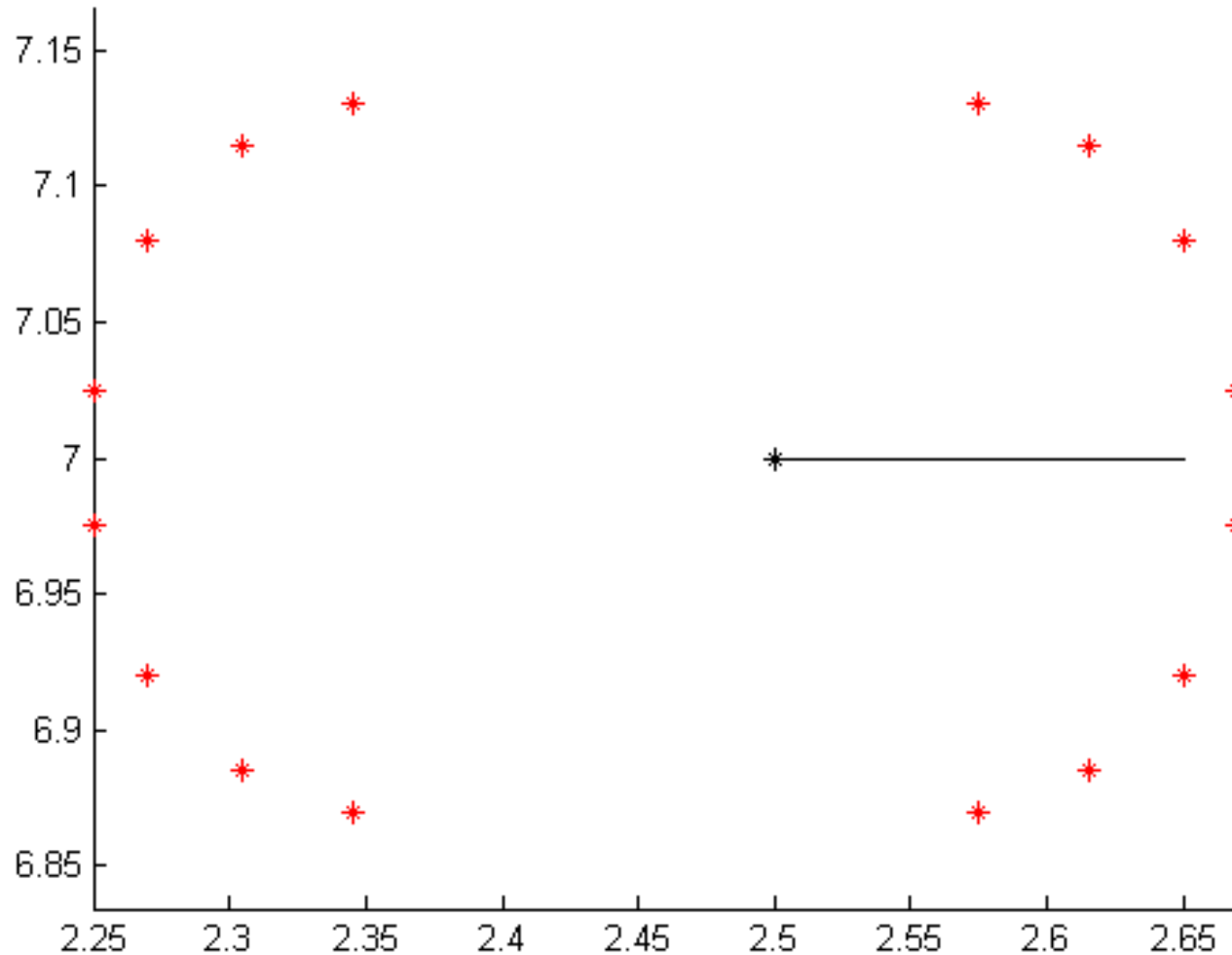
- maintaining related values separately?
- initialize all certainty values to zero
- contradictory information will lead to both values near 1
- combining them takes some work...



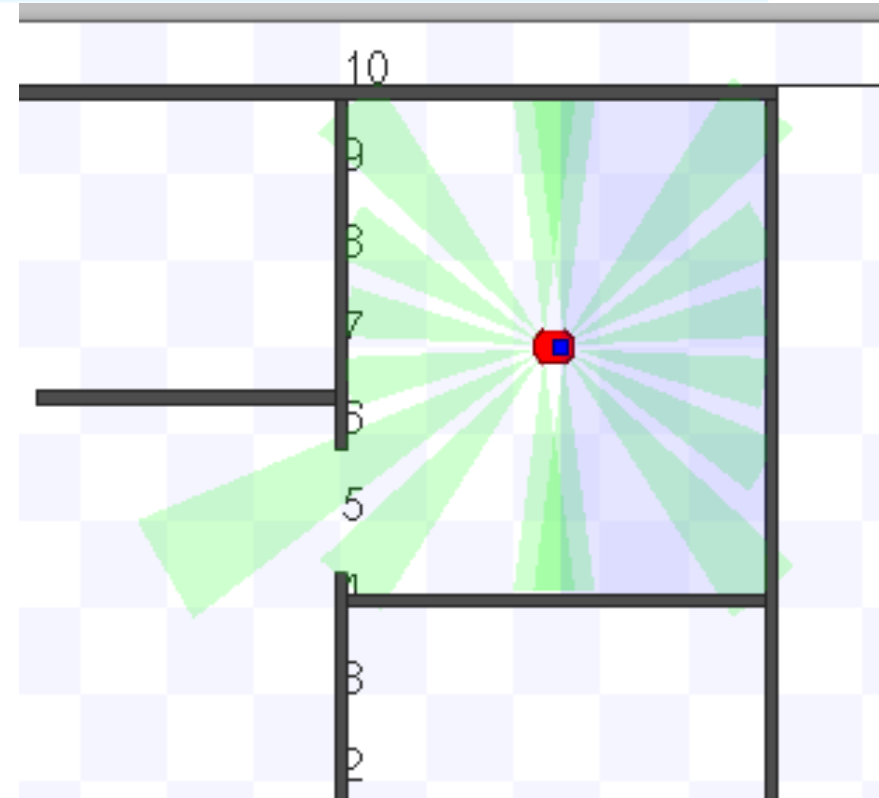
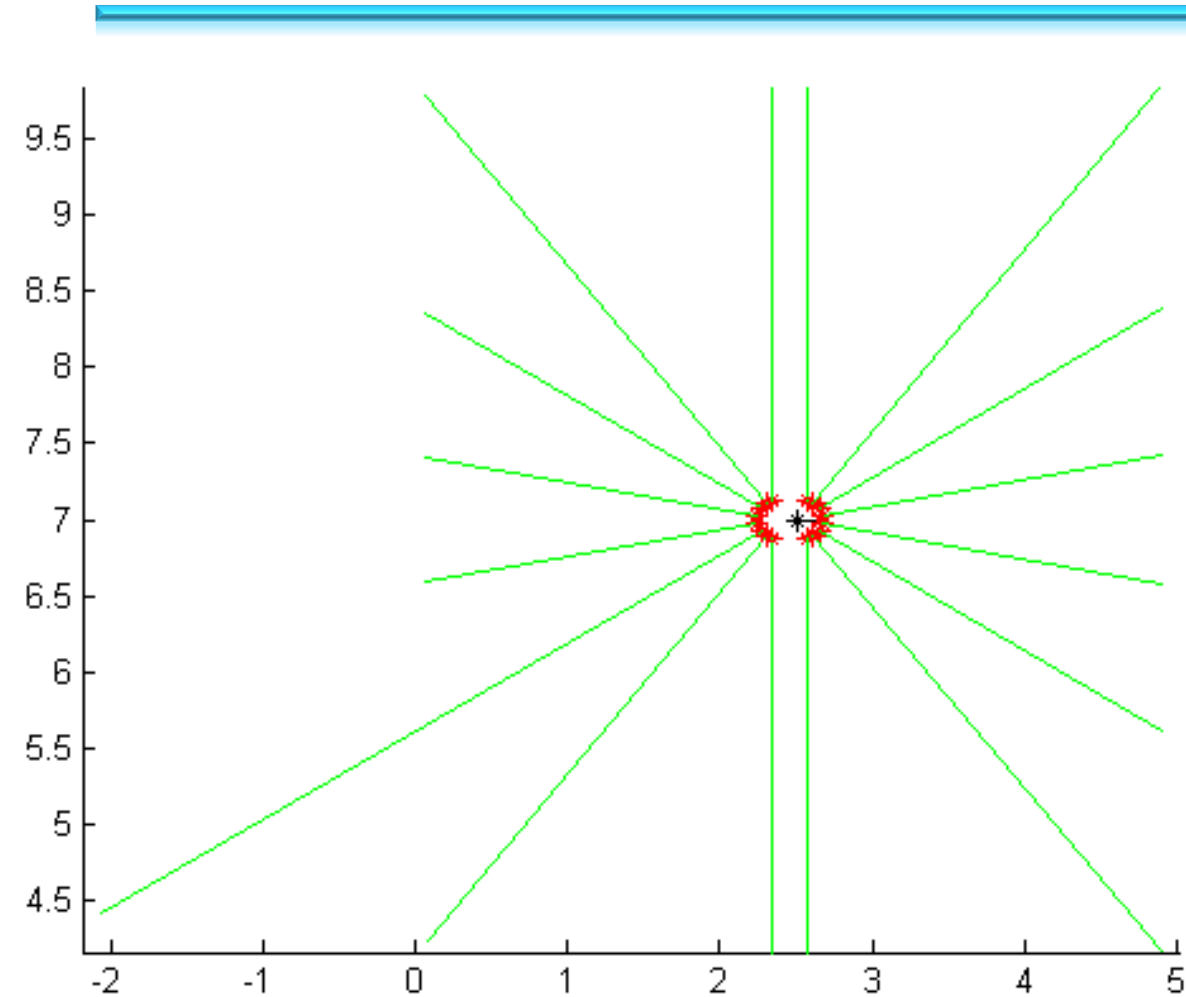
Sonars from P/S



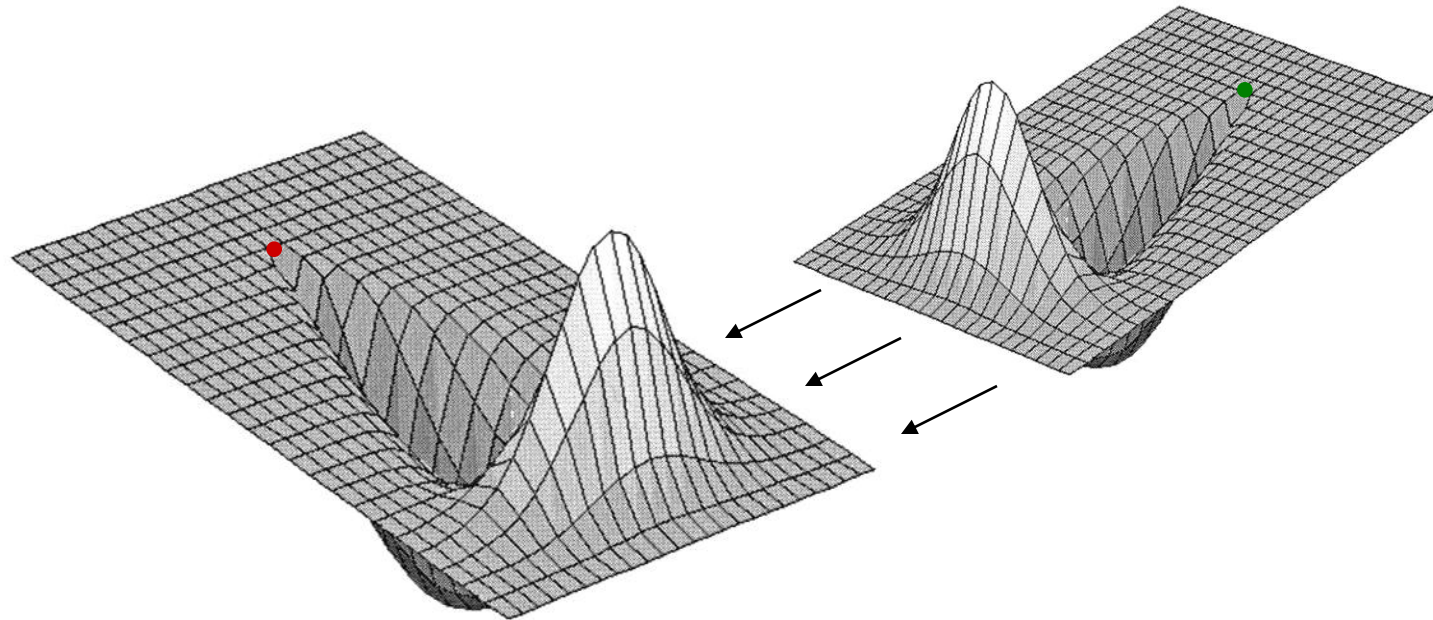
Sonar Locations Pioneer 3DX Robot



Sonar Data Calculation



Combining probabilities



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



What is it a map of ?

Several answers to this question have been tried:

pre '83 It's a map of occupied cells. o_{xy} cell (x,y) is occupied \bar{o}_{xy} cell (x,y) is unoccupied

'83 - '88 It's a map of probabilities: $p(o | S_{1..i})$ The certainty that a cell is **occupied**, given the sensor readings S_1, S_2, \dots, S_i

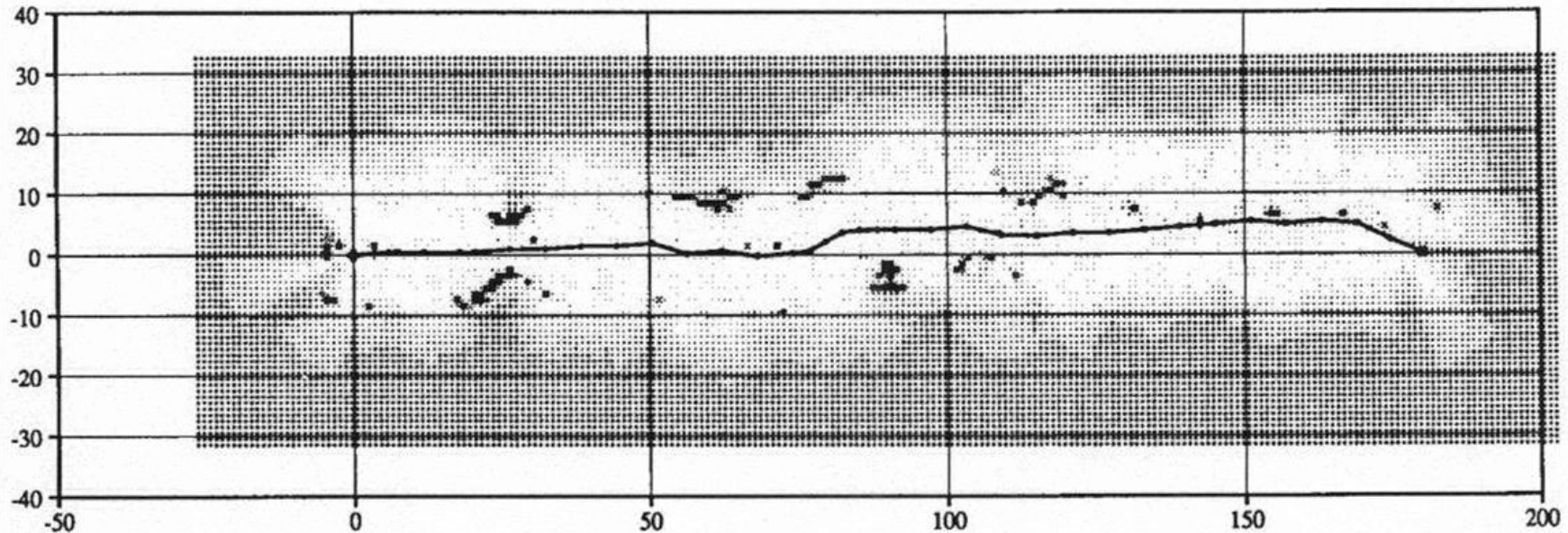
$p(\bar{o} | S_{1..i})$ The certainty that a cell is **unoccupied**, given the sensor readings S_1, S_2, \dots, S_i

★ It's a map of *odds*. The odds of an event are expressed *relative to the complement* of that event.

The odds that a cell is **occupied**, given the sensor readings S_1, S_2, \dots, S_i $\overset{\text{probabilities}}{\text{odds}(o | S_{1..i})} = \frac{p(o | S_{1..i})}{p(\bar{o} | S_{1..i})}$



An example map



units: feet

Evidence grid of a tree-lined outdoor path

- lighter areas: *lower* odds of obstacles being present
- darker areas: *higher* odds of obstacles being present

how to combine them?



Conditional probability

Some intuition...

$$p(o | S) = \frac{\text{The probability of event } o, \text{ given event } S.}{\text{The probability that a certain cell } o \text{ is occupied, given that the robot sees the sensor reading } S.}$$

$$p(S | o) = \frac{\text{The probability of event } S, \text{ given event } o.}{\text{The probability that the robot sees the sensor reading } S, \text{ given that a certain cell } o \text{ is occupied.}}$$

- What is really meant by conditional probability ?
- How are these two probabilities related?



Bayes Rule

- Conditional probabilities

$$p(o \wedge S) = p(o | S)p(S)$$



Bayes Rule

- Conditional probabilities

$$p(o \wedge S) = p(o | S)p(S)$$

- Bayes rule relates conditional probabilities

$$p(o | S) = \frac{p(S | o)p(o)}{p(S)}$$

Bayes rule



Bayes Rule

- Conditional probabilities

$$p(o \wedge S) = p(o | S)p(S)$$

- Bayes rule relates conditional probabilities

$$p(o | S) = \frac{p(S | o)p(o)}{p(S)}$$

Bayes rule

- So, what does this say about $\text{odds}(o | S_2 \wedge S_1)$?

Can we update easily ?



Combining evidence

So, how do we combine evidence to create a map?

What we want --

$$\text{odds}(o \mid S_2 \wedge S_1)$$

the new value of a cell in the map
after the sonar reading S_2

What we know --

$$\text{odds}(o \mid S_1)$$

the old value of a cell in the map
(before sonar reading S_2)

$$p(S_i \mid o) \ \& \ p(S_i \mid \bar{o})$$

the probabilities that a certain obstacle
causes the sonar reading S_i



Combining evidence

$$\text{odds}(o | S_2 \wedge S_1) = \frac{p(o | S_2 \wedge S_1)}{p(\bar{o} | S_2 \wedge S_1)}$$



Combining evidence

$$\begin{aligned} \text{odds}(o | S_2 \wedge S_1) &= \frac{p(o | S_2 \wedge S_1)}{p(\bar{o} | S_2 \wedge S_1)} \\ &= \frac{p(S_2 \wedge S_1 | o) p(o)}{p(S_2 \wedge S_1 | \bar{o}) p(\bar{o})} \end{aligned}$$

definition of odds



Combining evidence

$$\text{odds}(o | S_2 \wedge S_1) = \frac{p(o | S_2 \wedge S_1)}{p(\bar{o} | S_2 \wedge S_1)}$$

definition of odds

$$= \frac{p(S_2 \wedge S_1 | o) p(o)}{p(S_2 \wedge S_1 | \bar{o}) p(\bar{o})}$$

Bayes' rule (+)

$$= \frac{p(S_2 | o) p(S_1 | o) p(o)}{p(S_2 | \bar{o}) p(S_1 | \bar{o}) p(\bar{o})}$$



Combining evidence

$$\text{odds}(o | S_2 \wedge S_1) = \frac{p(o | S_2 \wedge S_1)}{p(\bar{o} | S_2 \wedge S_1)}$$

definition of odds

$$= \frac{p(S_2 \wedge S_1 | o) p(\bar{o})}{p(S_2 \wedge S_1 | \bar{o}) p(o)}$$

Bayes' rule (+)

$$= \frac{p(S_2 | o) p(S_1 | o) p(\bar{o})}{p(S_2 | \bar{o}) p(S_1 | \bar{o}) p(o)}$$

conditional
independence of
 S_1 and S_2

$$= \frac{p(S_2 | o) p(o | S_1)}{p(S_2 | \bar{o}) p(\bar{o} | S_1)}$$

Bayes' rule (+)



Combining evidence

$$\text{odds}(o | S_2 \wedge S_1) = \frac{p(o | S_2 \wedge S_1)}{p(\bar{o} | S_2 \wedge S_1)}$$

definition of odds

$$= \frac{p(S_2 \wedge S_1 | o) p(o)}{p(S_2 \wedge S_1 | \bar{o}) p(\bar{o})}$$

Bayes' rule (+)

$$= \frac{p(S_2 | o) p(S_1 | o) p(o)}{p(S_2 | \bar{o}) p(S_1 | \bar{o}) p(\bar{o})}$$

conditional independence of S_1 and S_2

$$= \frac{p(S_2 | o) p(o | S_1)}{p(S_2 | \bar{o}) p(\bar{o} | S_1)}$$

Bayes' rule (+)

precomputed values

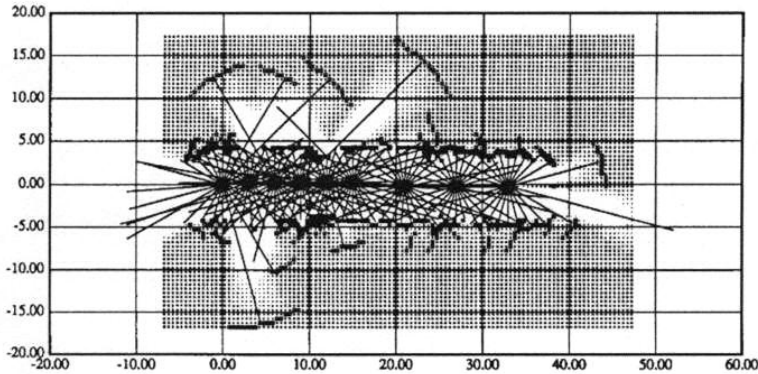
previous odds

the sensor model

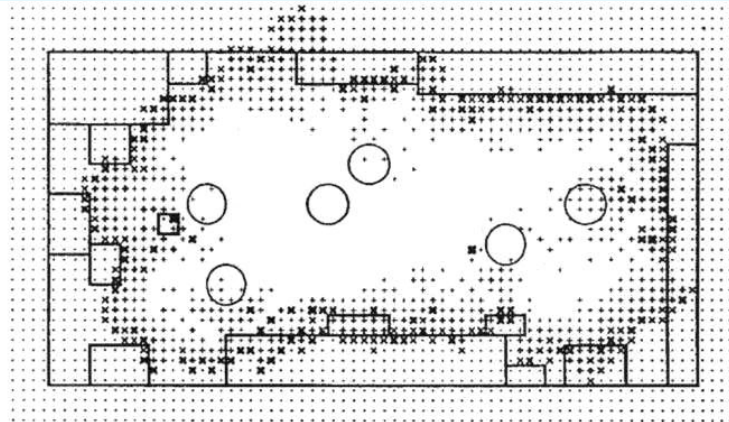
Update step = multiplying the previous odds by a precomputed weight.



Evidence grids

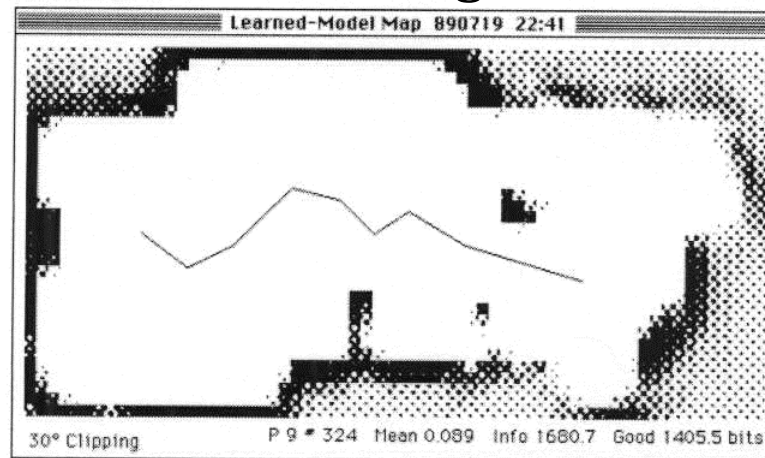
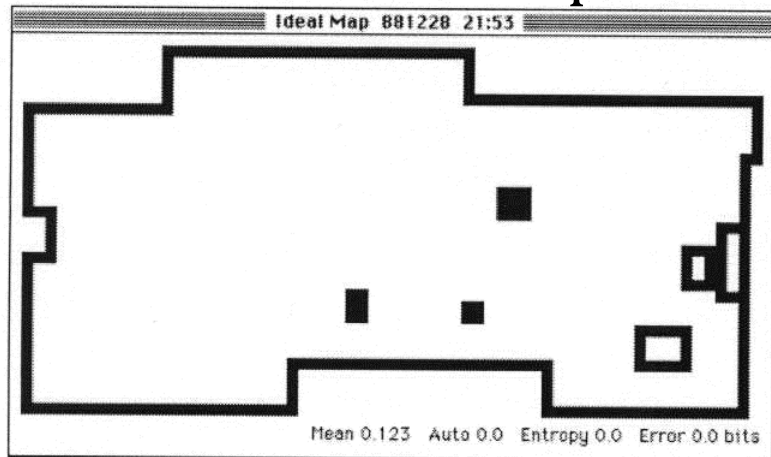


hallway with some open doors



lab space

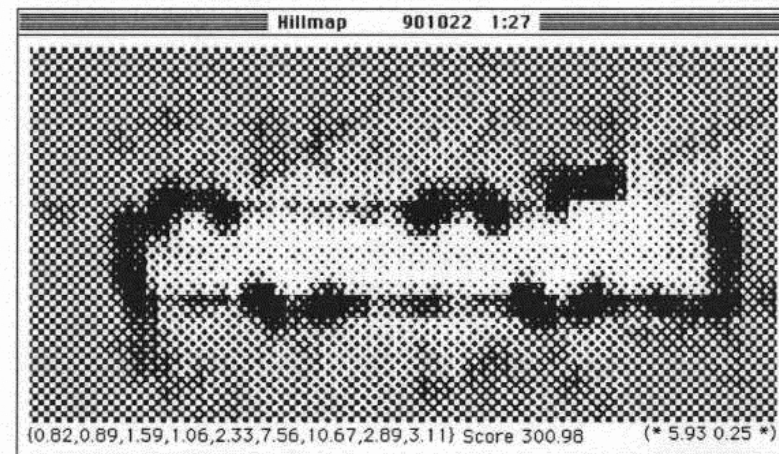
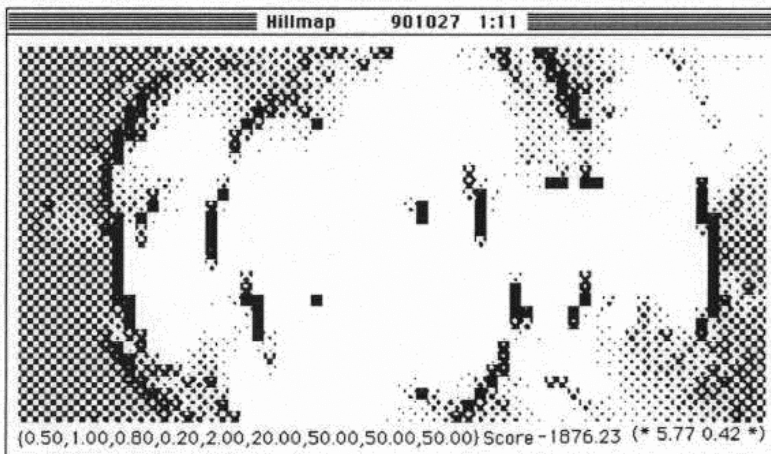
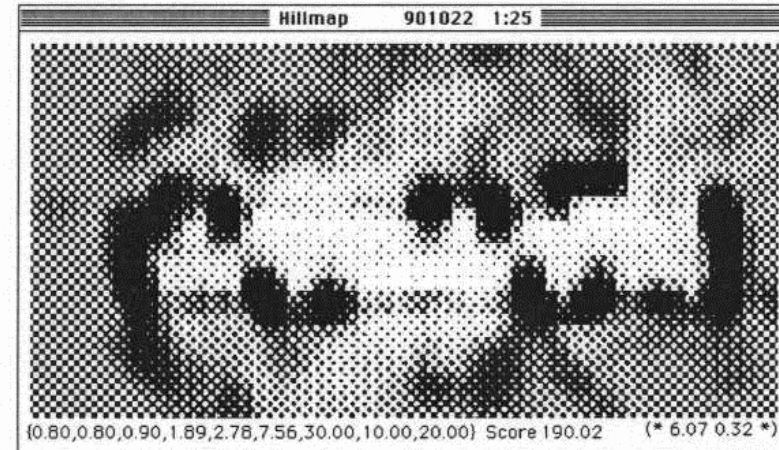
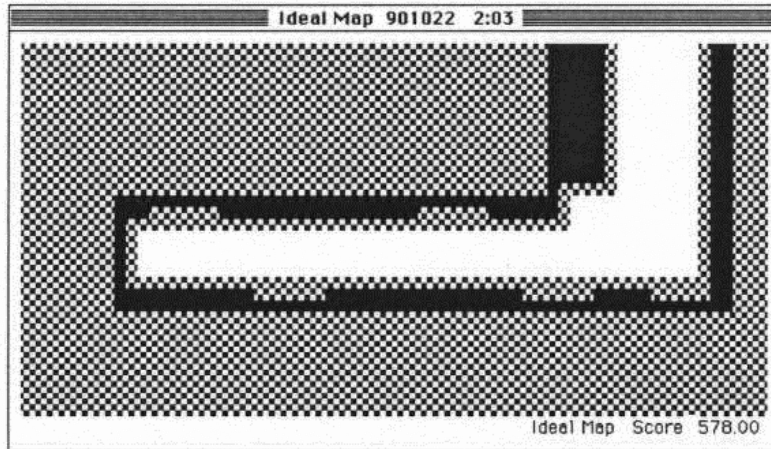
known map and estimated evidence grid



Learning the Sensor Model

The sonar model depends dramatically on the environment
-- we'd like to *learn* an appropriate sensor model

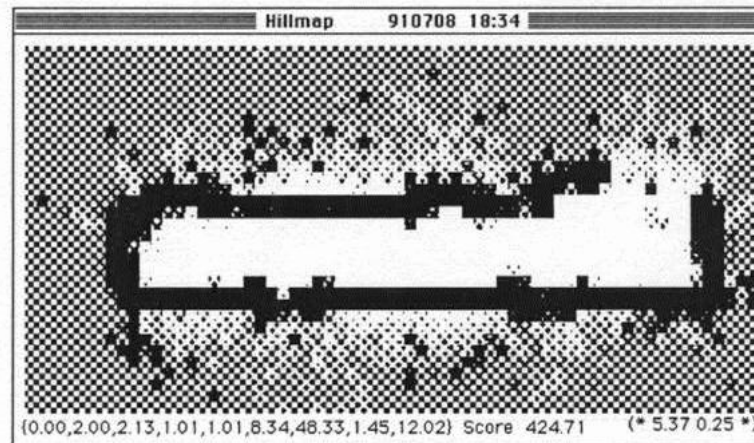
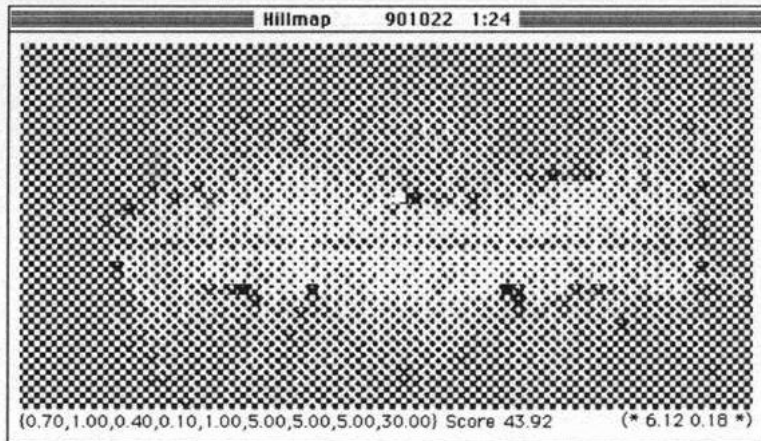
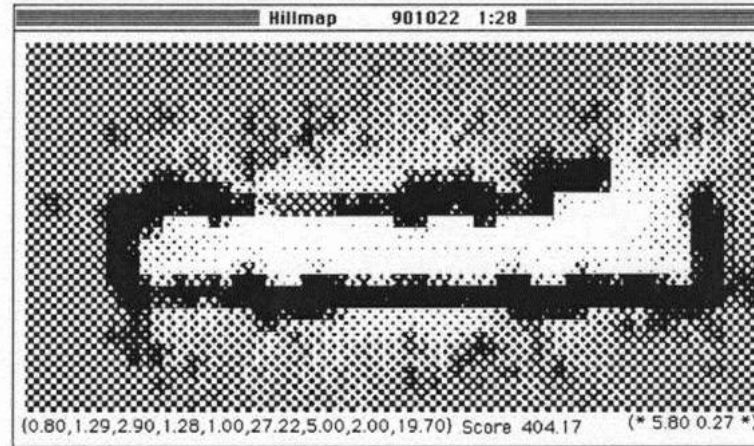
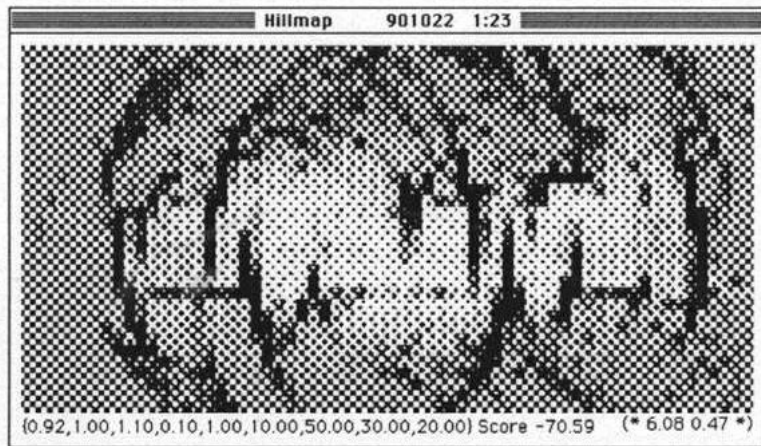
rather than hire Roman Kuc
to develop another one...



Learning the Sensor Model

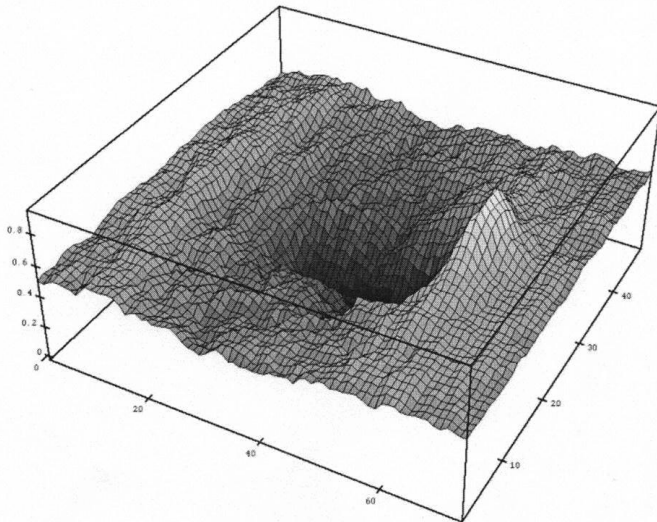
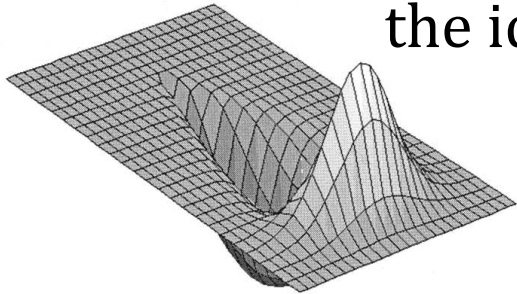
The sonar model depends dramatically on the environment
-- we'd like to *learn* an appropriate sensor model

rather than hire Roman Kuc
to develop another one...

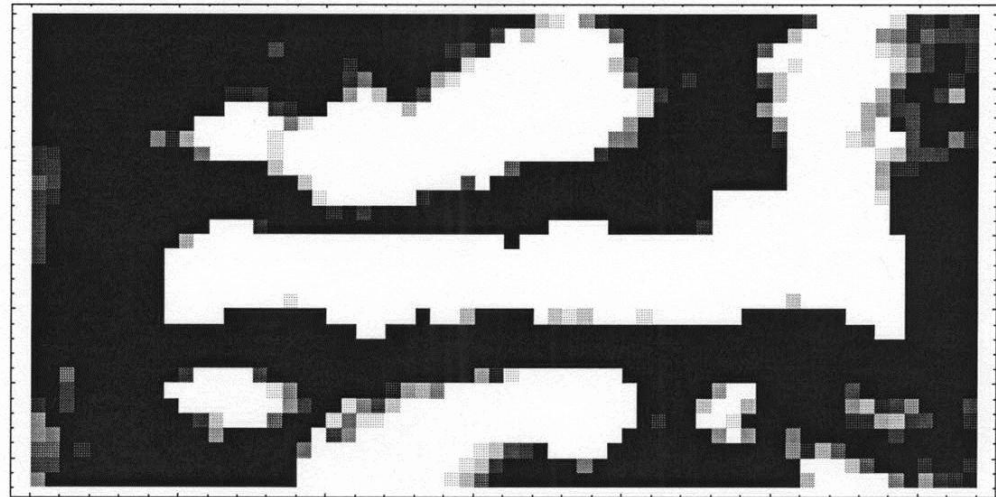


Learning the Sensor Model

the idealized model



part of the learned model



the mapping results of a model that had an even better match score (against the ideal map)

Sensor fusion

Incorporating data from other sensors -- e.g., IR rangefinders and stereo vision...

- (1) create another sensor model
- (2) update along with the sonar

