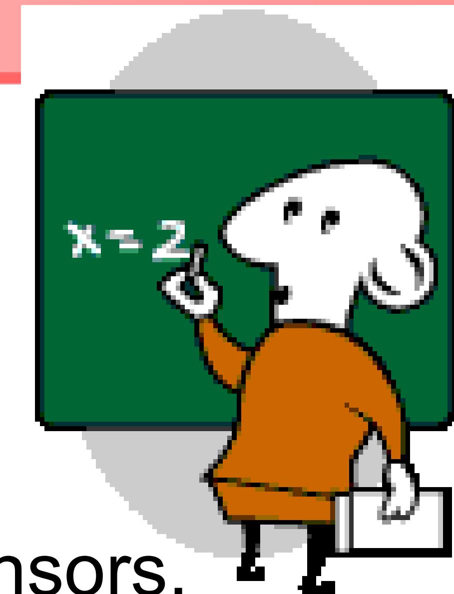
Probabilistic Sensor Models

Department of Electrical and Electronics Engineering
Dr. Afşar Saranlı

Lecture slides heavily use material from the textbook and
Sebastian Thrun, Lecture Slides; http://www.probabilistic-robotics.org/

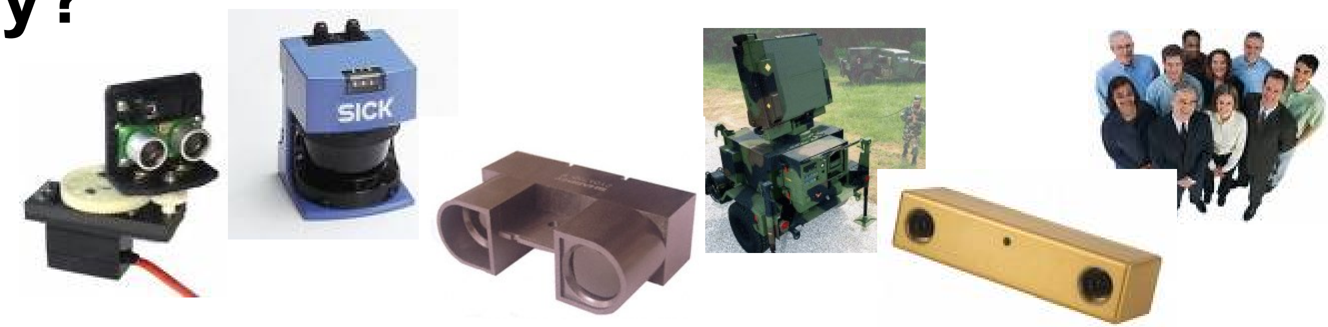
What we will discuss



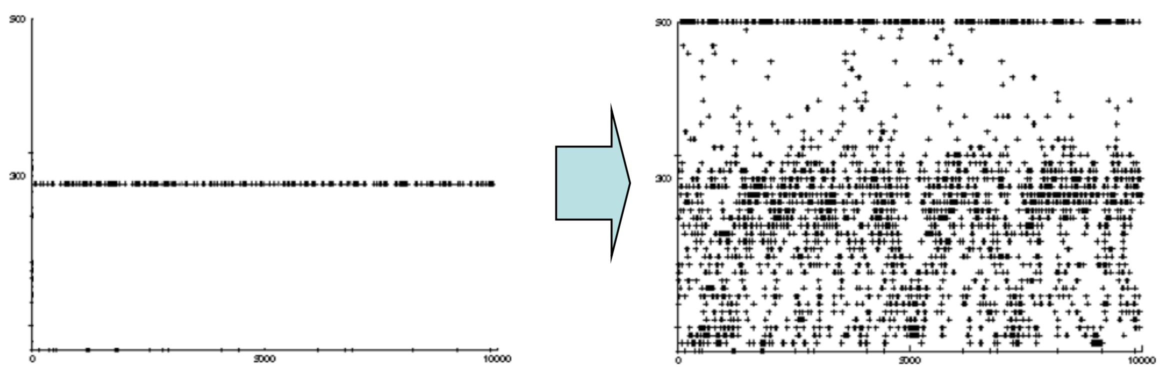
- Remember the inherent uncertainty in robots sensing the environment,
- Overview of Sensors
- Discuss general principles of range sensors,
- *Beam models*,
- *Likelihood Fields*,
- *Correlation Based models*,
- *Feature Based Models*,
- Discussion

Uncertainty in Robotic Sensing

- Robot sensing is inherently uncertain.
- **Why?**



The result?

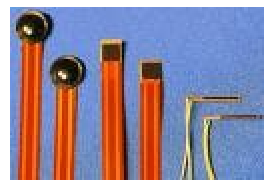


Question: How can we model this uncertainty?

Sensors for Mobile Robots - I

- **Tactile Sensors (Contact, Pressure, Bending)**

Switches or “bumpers”
(binary proximity)



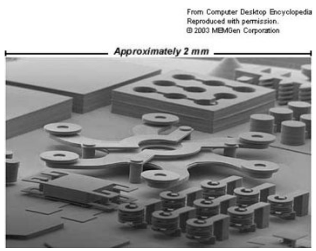

Pressure/Force Sensors

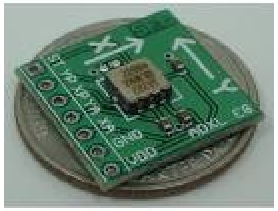
Bending Sensors

Images from: http://www.robotshop.ca

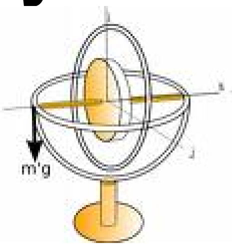
Sensors for Mobile Robots - II

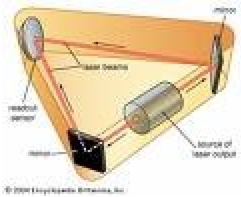
- **Internal Sensors:**
- **Accelerometers (spring mounted masses)**



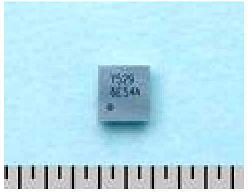



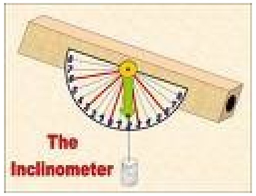
- **Gyroscopes (spinning mass, laser light)**



- **Compass(magn. field), inclinometer (gravity)**



Images from: Various from the web. (try google images)

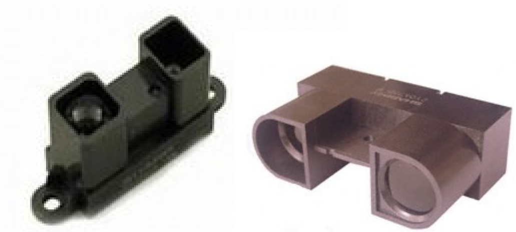
Sensors for Mobile Robots - III

- **Range – Range/Bearing Sensors:**

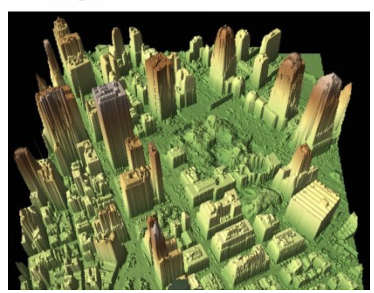
Sonar (sound wave time-of-flight)

Radar (time-of-flight, phase, frequency)

Laser range scanners (laser light, triangulation, time-of-flight, phase)



Infrared Light range finders (triangulation and intensity)



“LIDAR” for larger scale applications

Images from: Various from the web, copyrighted by their respective owners (try google images)

Sensors for Mobile Robots - IV

- **Vision Sensors (Cameras):**

Board cameras

Cased cameras
(separate lens)

Stereo vision cameras (depth from triangulation)

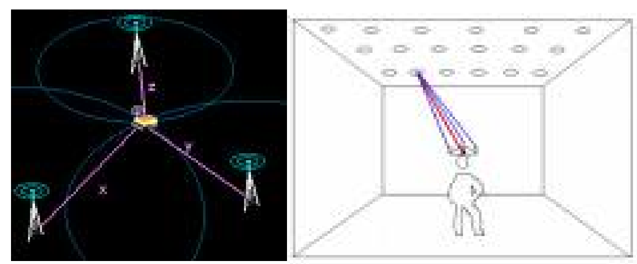
Omni-directional vision cameras and setups (using mirrors or fish-eye lenses

Images from: PointGrey Research (http://www.ptgrey.com) and other web sources

Sensors for Mobile Robots - V

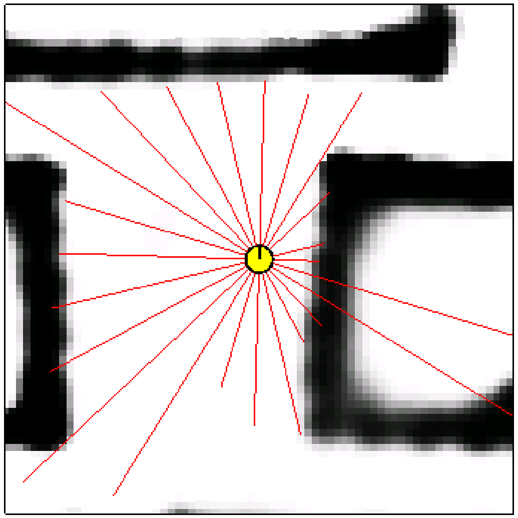
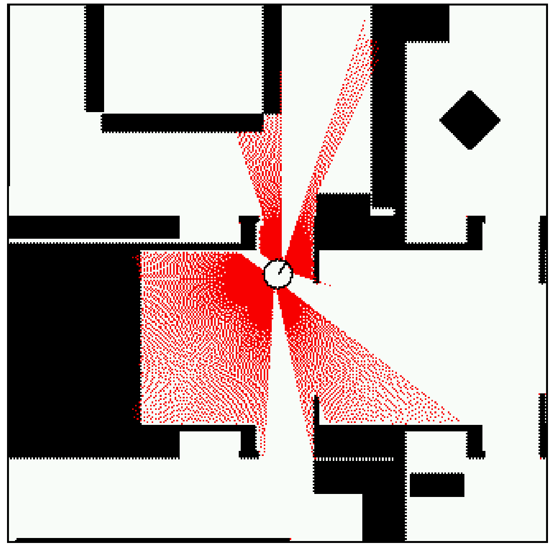
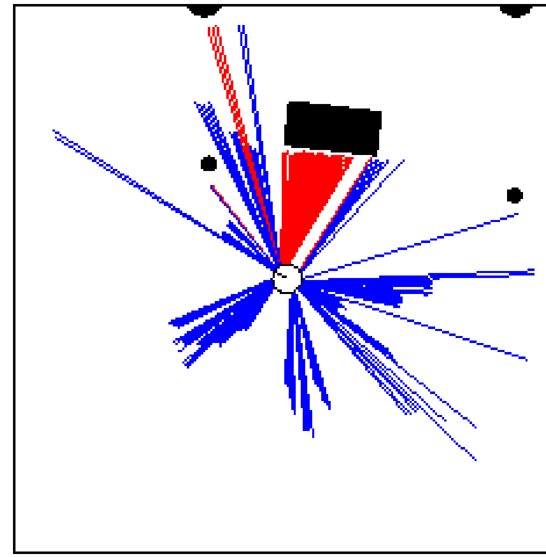
- **External Assisted "global" Sensors**

GPS based "global positioning"



Other beacon "global positioning"
(may use sound, light/camera, RF)

Range/Bearing Sensors

- ***Very popular in robotics applications,***
- The central task of modeling is to determine $p(z_t|x_t,m)$, i.e., the *pdf of a measurement* z_t *given that the robot has state (pose for our case)* x_t *within a map* m*.*
- ***Question***: Where do the “errors” come from?
- ***Approach***: Assume/Build a mixed *pdf* for the measurements.

Some Assumptions First

- Scan z_t consists of K individual measurements.

$$z_t = \{z_t^1, z_t^2, \ldots, z_t^K\}$$

- Individual measurement noises (hence the resulting measurement pdfs) are independent given the robot pose.

$$p(z_t \mid x_t, m) = \prod_{k=1}^{K} p(z_t^k \mid x_t, m)$$

Hence we can first find the pdfs of individual measurements…

We also need to represent this "map" somehow…

The Map

Represent an object and its properties (location, shape, color)

- To model the measurement generation process, we need to have a model of the environment:
- ***The Map*: List of features/objects in the Environment and their locations.**

$$m = \{m_1, m_2, \ldots, m_N\}$$

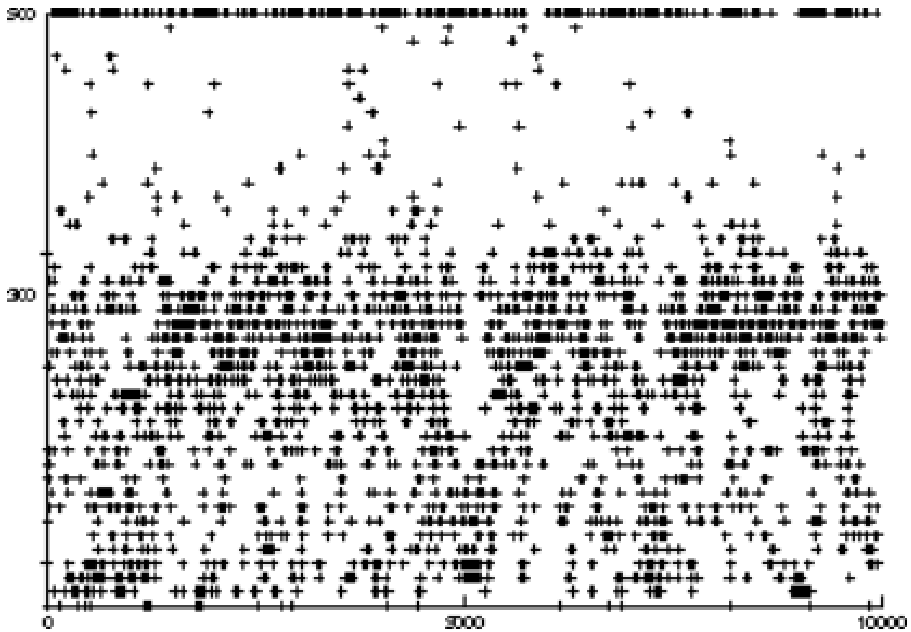
- **Maps: *Feature Based* vs *Location Based***

n is a “feature index”, m_n contains the cartesian location of the feature as well as other properties

Index *n* (which becomes *x*,*y*) already contains the location of the feature, $m_{(x,y)}$ contains the attached properties

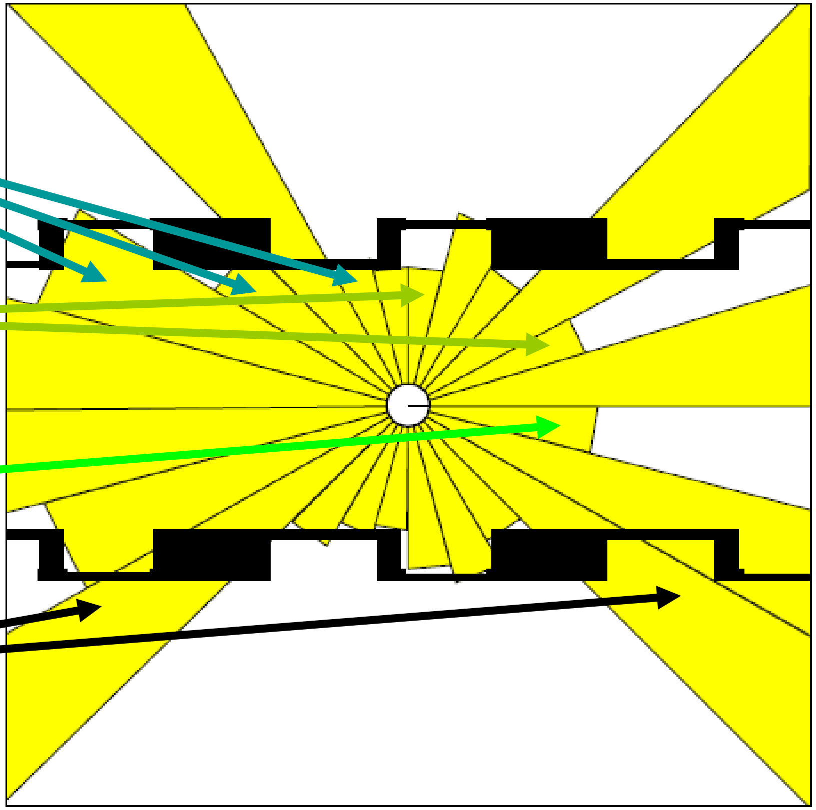
Behavior of Range Data?

- **What do you observe on this Sonar range data?**



Sources of Range Sensor Errors

1. Beams reflected by obstacles
2. Beams reflected by persons / caused by crosstalk
3. Random measurements
4. Maximum range measurements (sensor physics failure or limitations)

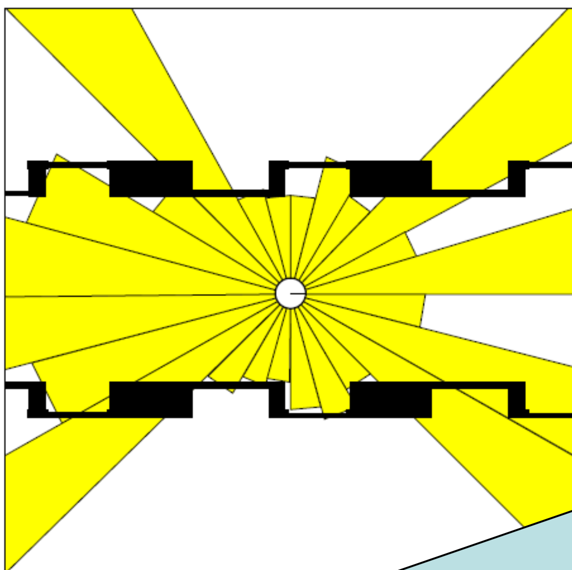


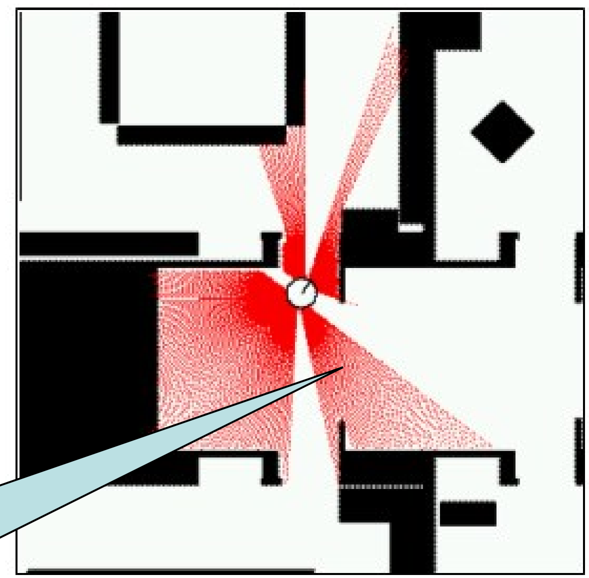
Range Measurement

- **Measurement can be caused by …**
  - a known obstacle.
  - cross-talk between similar sensors,
  - an unexpected obstacle (people, furniture, …),
  - missing all obstacles (total reflection, glass, …),
  - Random sensor behavior,
- **Noise models the uncertainty in…**
  - measuring distance to known obstacle.
  - position of known obstacles.
  - position of additional (unmodeled) obstacles.
  - whether obstacles are missed.

Beam Model of Range Sensors

- Attemps to model the physics of measurement formation by *ray-casting* and the *noise model.*

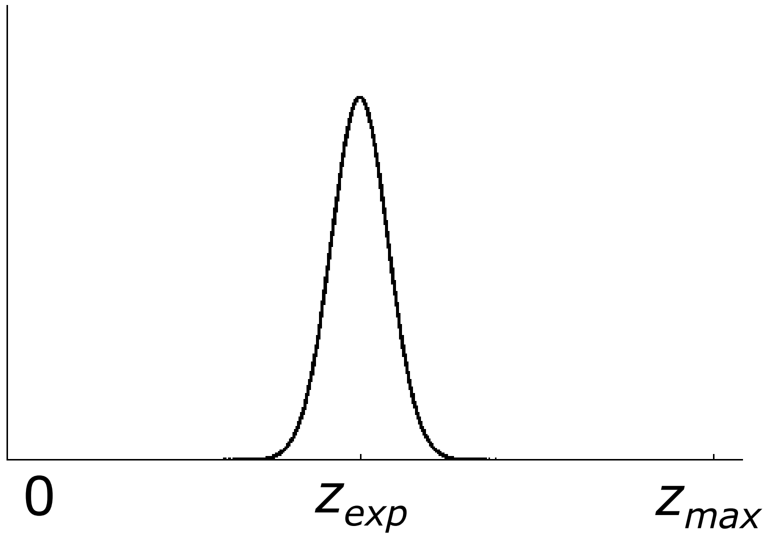




Geometry of rays and map determine the "ideal measurements". Noise model added to account for sources of uncertainty

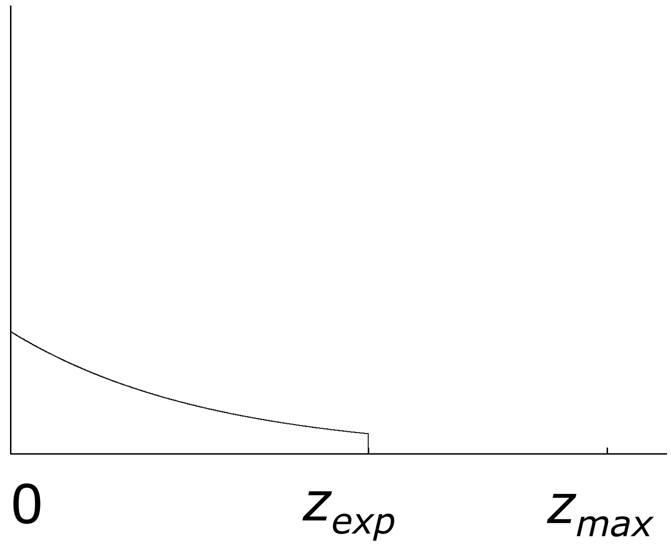
Mixture Model of Sensor Output

Measurement noise



$$P_{hit}(z \mid x, m) = \eta \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\frac{(z - z_{\exp})^2}{\sigma^2}}$$

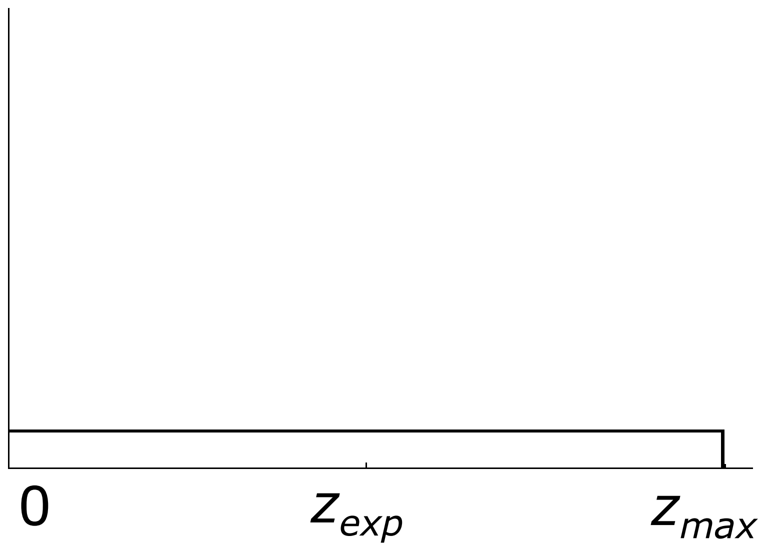
Unexpected obstacles



$$P_{\text{unexp}}(z \mid x, m) = \begin{cases} \eta \, \lambda \, \mathrm{e}^{-\lambda z} & z < z_{\exp} \\ 0 & otherwise \end{cases}$$

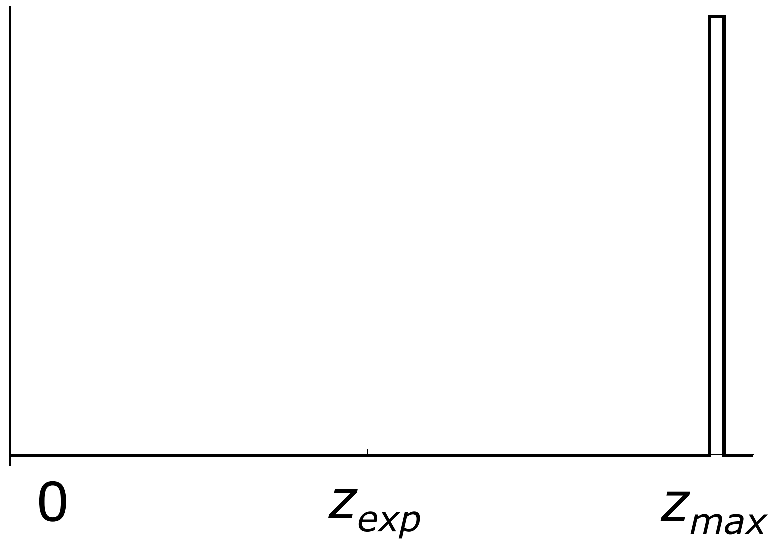
Mixture Model of Sensor Output

Random measurement



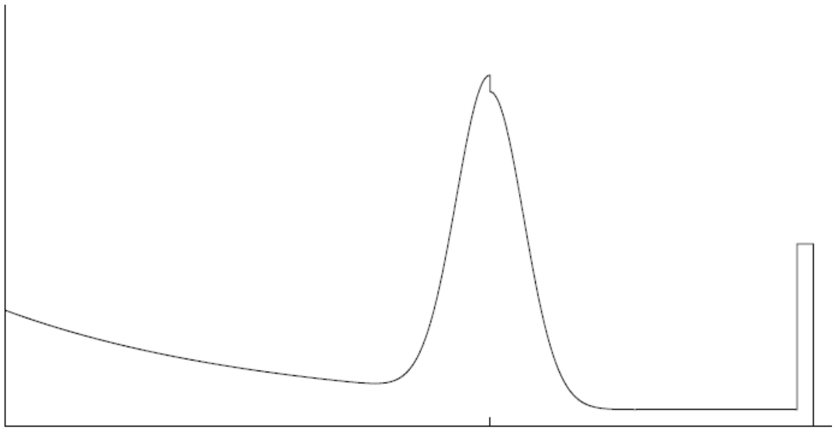
$$P_{rand}(z \mid x, m) = \eta \frac{1}{z_{\max}}$$

Max range



$$P_{\max}(z \mid x, m) = \eta \frac{1}{z_{small}}$$

Resulting Mixture Density



$$P(z \mid x,m) = \begin{pmatrix} \alpha_{\text{hit}} \\ \alpha_{\text{unexp}} \\ \alpha_{\text{max}} \\ \alpha_{\text{rand}} \end{pmatrix}^T \cdot \begin{pmatrix} P_{\text{hit}}(z \mid x,m) \\ P_{\text{unexp}}(z \mid x,m) \\ P_{\text{max}}(z \mid x,m) \\ P_{\text{rand}}(z \mid x,m) \end{pmatrix}$$

Is a weighted combination of the 4 different noise factors

Important Question:

How can we determine the 4 mixing weights, σ^2 and λ parameters?

Beam Model - Algorithm

Algorithm beam_range_finder_model(z_t, x_t, m)**:**

$q = 1$

for $k = 1$ *to* K *do*

compute z_t^{k*} *for the measurement* z_t^k *using ray casting*

$p = z_{\text{hit}} \cdot p_{\text{hit}}(z_t^k \mid x_t, m) + z_{\text{short}} \cdot p_{\text{short}}(z_t^k \mid x_t, m)$

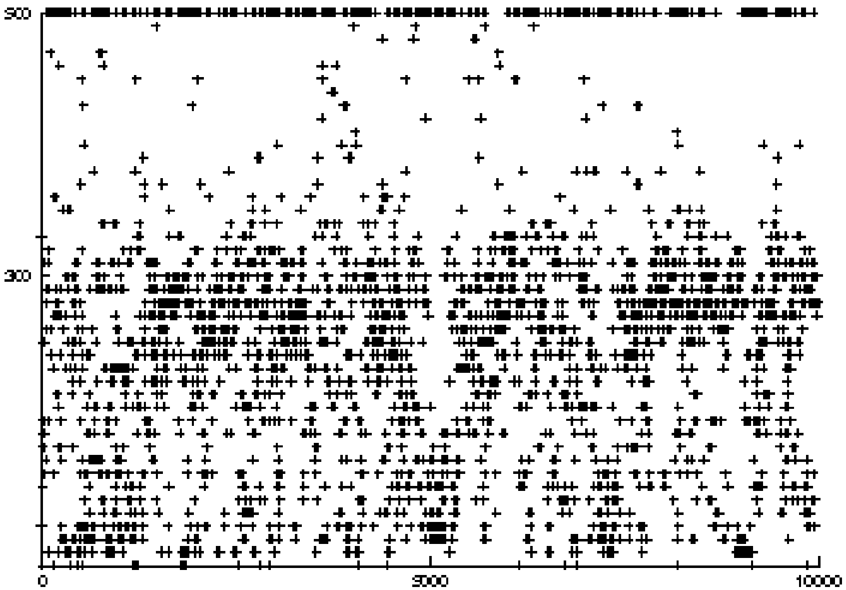
$+ z_{\max} \cdot p_{\max}(z_t^k \mid x_t, m) + z_{\text{rand}} \cdot p_{\text{rand}}(z_t^k \mid x_t, m)$

$q = q \cdot p$

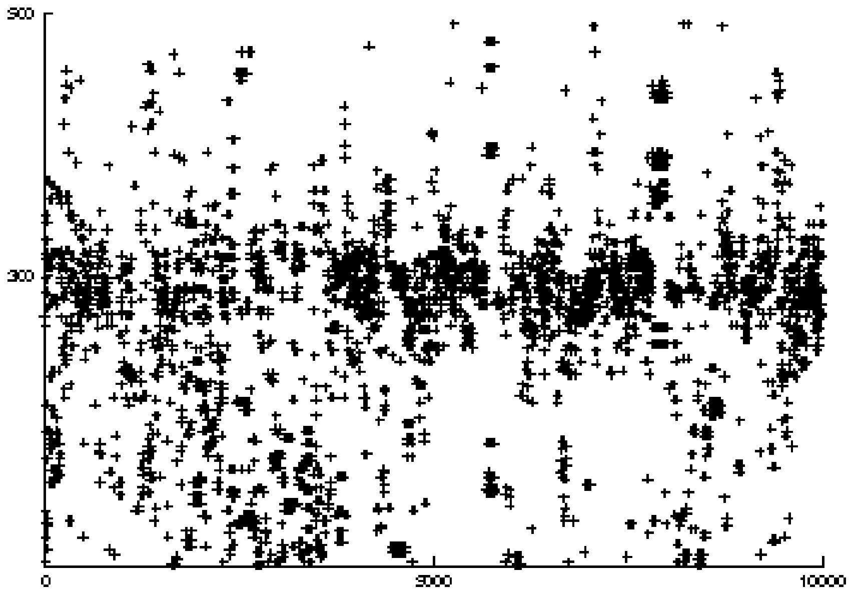
return q

We have called these as *α* to avoid confusing with the measurement *z*

Sonar versus Laser



Sonar



Laser

Are the intrinsic parameters the same?

Learning the Model Parameters

- **The basic idea:**
  Given sensor data, express the "log-likelihood" of the data as a function of model parameters…

$$\log\left(p(Z \mid X, m, \theta)\right)$$

- Maximize this log-likelihood of collected data…
- "*Maximum Likelihood Estimator*"
- ***The "curve fitting" of the probabilistic models!!***
- Log(.) is used to simplify mathematical steps – it is monotonic hence does not change the problem.

Maximum Likelihood Estimation

- We assume that data set can be decomposed into disjoint sets Z_{hit}, Z_{short}, Z_{max}, Z_{rand},
- Hence we assume knowing *which data came from which component of the mixture,*
- (Later this assumption is lifted with the *Expectation Maximization* algorithm)
- Hence we define a "correspondence variable" c_i for each data point z_i. (c_i=*hit* means z_i came from P_{hit} distribution)

Maximum Likelihood Estimation

- Normalized mixing parameters are easy to determine:

|.| is the size of the set

$$\begin{pmatrix} \alpha_{\text{hit}} \\ \alpha_{\text{unexp}} \\ \alpha_{\max} \\ \alpha_{\text{rand}} \end{pmatrix} = |Z|^{-1} \begin{pmatrix} |Z_{\text{hit}}| \\ |Z_{\text{short}}| \\ |Z_{\max}| \\ |Z_{\text{rand}}| \end{pmatrix}$$

- How about the density parameters, e.g., the Variance of the Gaussian for p_{hit}?

Maximum Likelihood Estimation

- Consider the "hit" model variance param. σ^2,

$$p(Z_{\text{hit}} \mid X, m, \Theta) = \prod_{z_i \in Z_{\text{hit}}} p_{\text{hit}}(z_i \mid x_i, m, \Theta)$$

- I will try to find the σ^2 that <u>*maximizes the likelihood of observed data*!!</u>
- A maxima needs to satisfy the *necessary condition:*

$$\frac{\partial P}{\partial \sigma^2} = 0$$

Maximum Likelihood Estimation

- Express P in terms of variance param. σ^2:

$$p(Z_{\text{hit}} \mid X, m, \Theta) = \prod_{z_i \in Z_{\text{hit}}} p_{\text{hit}}(z_i \mid x_i, m, \Theta) = \prod_{z_i \in Z_{\text{hit}}} \frac{1}{\sqrt{2\pi\sigma_{\text{hit}}^2}} e^{-\frac{1}{2}\frac{(z_i - z_i^*)^2}{\sigma_{\text{hit}}^2}}$$

$$\begin{aligned}
\log p(Z_{\text{hit}} \mid X, m, \Theta) &= \sum_{z_i \in Z_{\text{hit}}} \left[-\frac{1}{2} \log 2\pi\sigma_{\text{hit}}^2 - \frac{1}{2} \frac{(z_i - z_i^*)^2}{\sigma_{\text{hit}}^2} \right] \\
&= -\frac{1}{2} \sum_{z_i \in Z_{\text{hit}}} \left[\log 2\pi\sigma_{\text{hit}}^2 + \frac{(z_i - z_i^*)^2}{\sigma_{\text{hit}}^2} \right] \\
&= -\frac{1}{2} \left[|Z_{\text{hit}}| \log 2\pi + 2|Z_{\text{hit}}| \log \sigma_{\text{hit}} + \sum_{z_i \in Z_{\text{hit}}} \frac{(z_i - z_i^*)^2}{\sigma_{\text{hit}}^2} \right] \\
&= \text{const.} - |Z_{\text{hit}}| \log \sigma_{\text{hit}} - \frac{1}{2\sigma_{\text{hit}}^2} \sum_{z_i \in Z_{\text{hit}}} (z_i - z_i^*)^2
\end{aligned}$$

Maximum Likelihood Estimation

- **Now differentiate w.r.t. σ and equate to zero:**

$$\log p(Z_{\text{hit}} \mid X, m, \Theta) = \text{const.} - |Z_{\text{hit}}| \log \sigma_{\text{hit}} - \frac{1}{2\sigma_{\text{hit}}^2} \sum_{z_i \in Z_{\text{hit}}} (z_i - z_i^*)^2$$

$$\frac{\partial \log p(Z_{\text{hit}} \mid X, m, \Theta)}{\partial \sigma_{\text{hit}}} = -\frac{|Z_{\text{hit}}|}{\sigma_{\text{hit}}} + \frac{1}{\sigma_{\text{hit}}^3} \sum_{z_i \in Z_{\text{hit}}} (z_i - z_i^*)^2 = 0$$

- **Hence we have:**

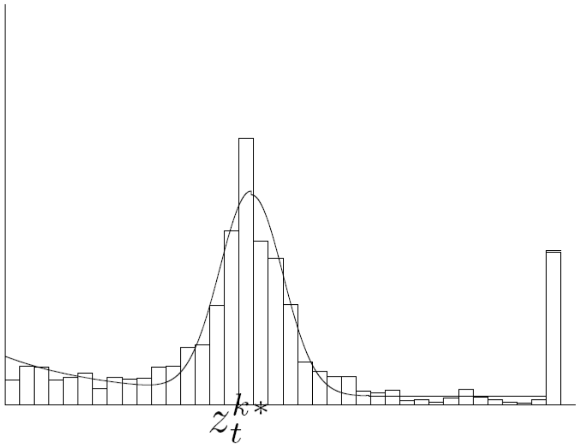
$$\sigma_{\text{hit}} = \sqrt{\frac{1}{|Z_{\text{hit}}|} \sum_{z_i \in Z_{\text{hit}}} (z_i - z_i^*)^2}$$

- **The exponential density parameter proceeds in exactly the same way…**

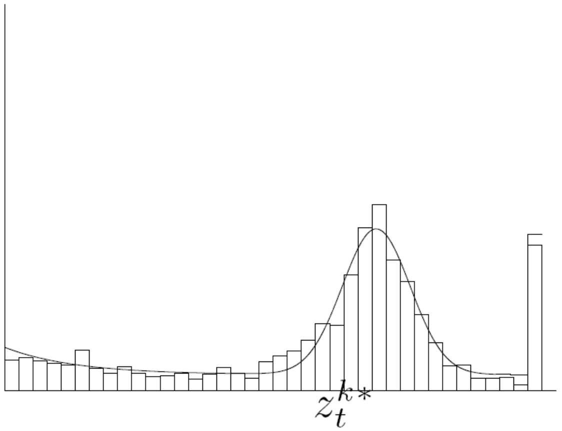
Expectation Maximization - EM

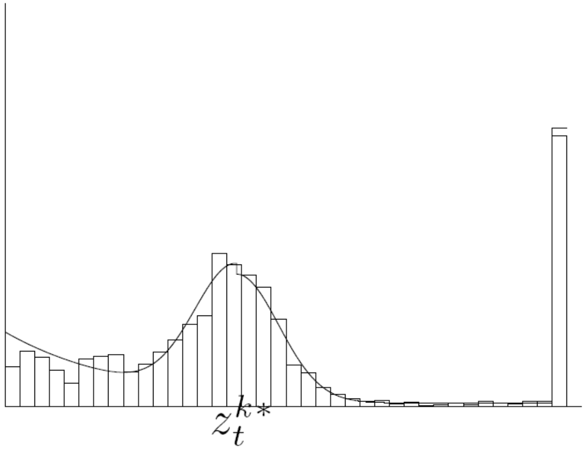
- For unknown "correspondence variables" c_i, there is no such closed form solution,
- But an iterative approach known as "*expectation maximization - EM*" can be applied with success,
- *EM* algorithm is a very useful tool for probabilistic estimation problems
(Reading assignment from Thrun, Burgard & Fox. Also read the Wikipedia Article on the Subject)

Model Estimation Results

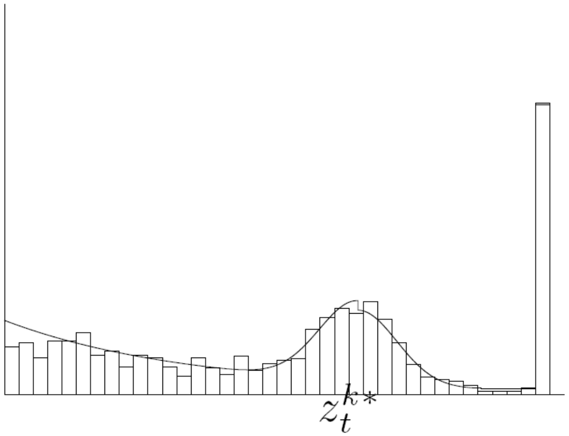


Laser





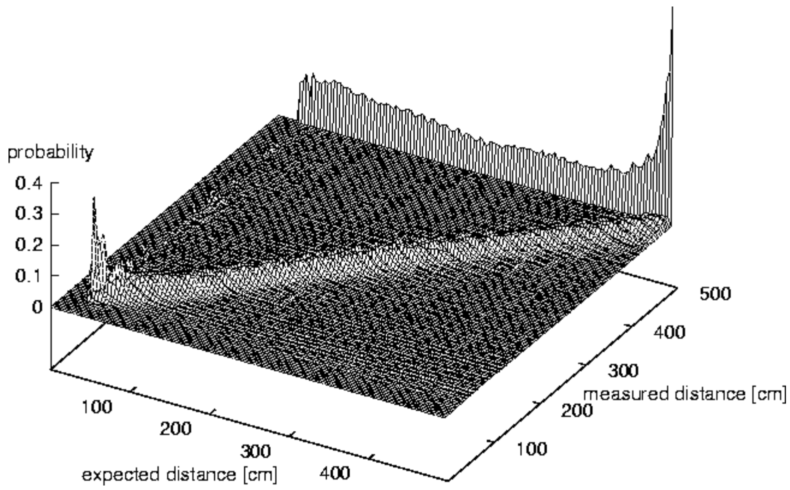
Sonar



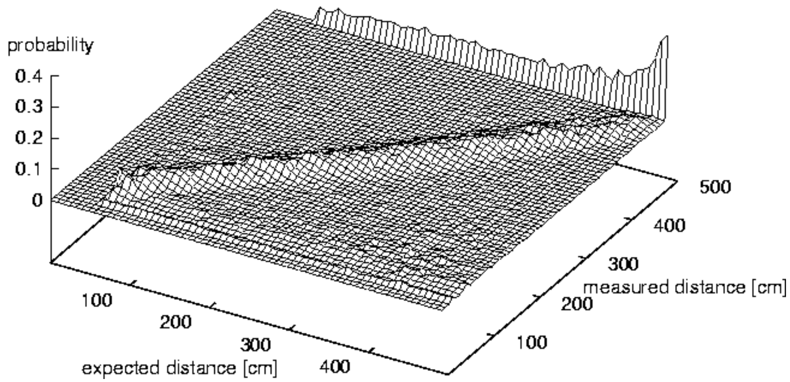
300cm

400cm

Model Estimation Results

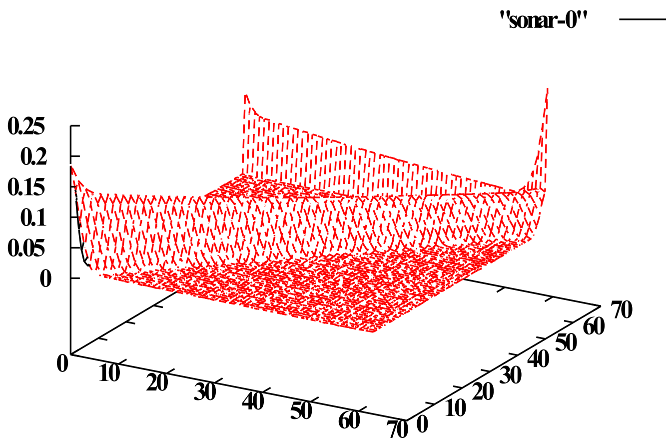


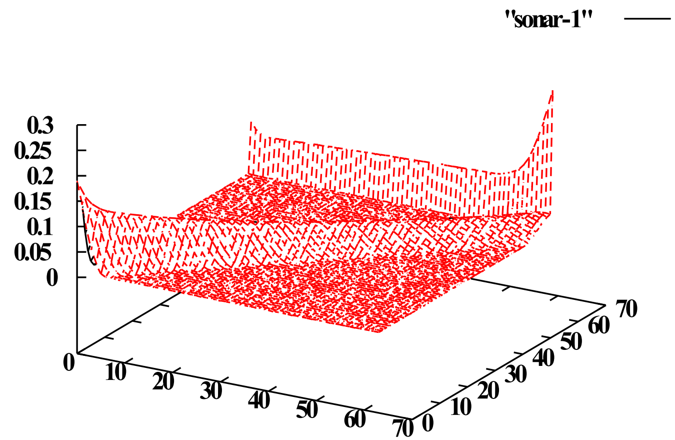
Laser

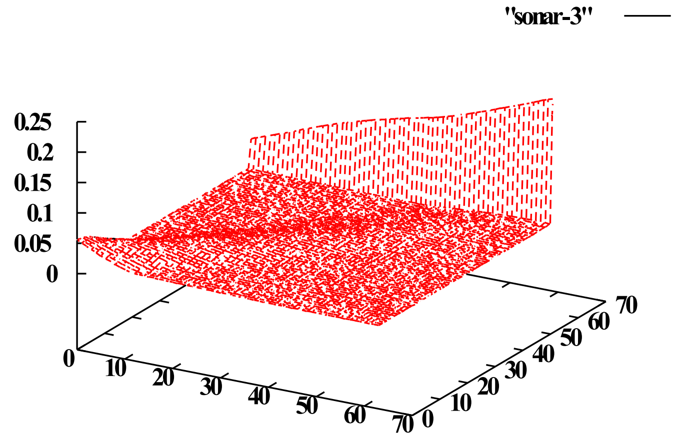


Sonar

Sonar Angle to Obstacle

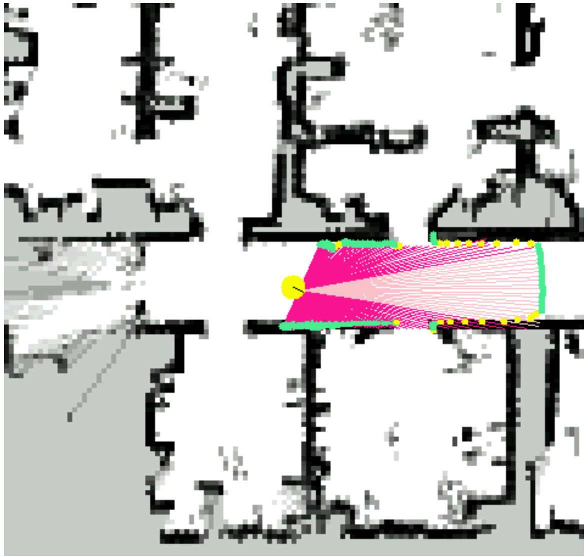




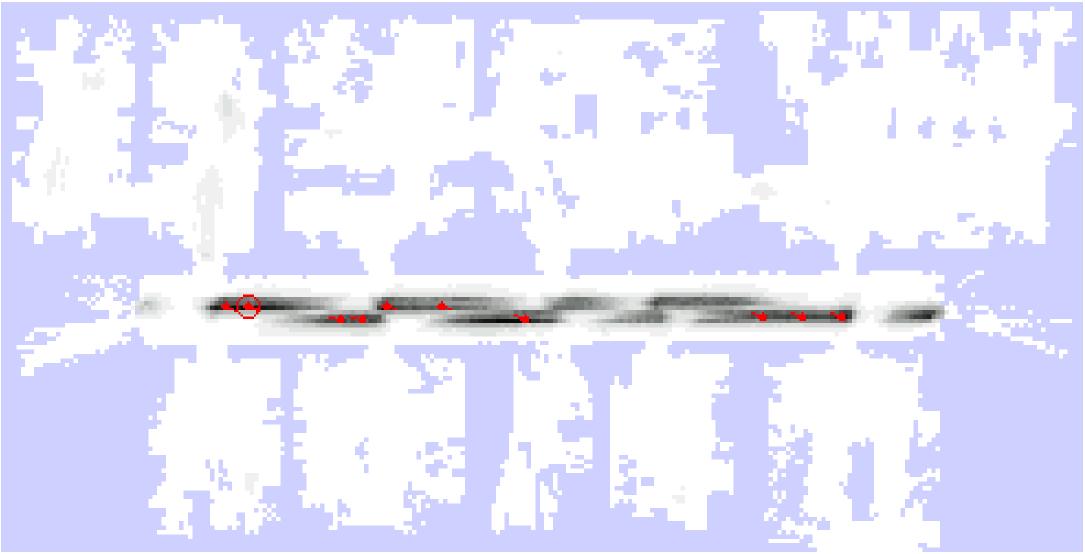


An Example: One Laser Scan

- Measurement likelihood as a function of pose:



z



$P(z|x,m)$

Discussion of Beam Models

- **Assumes independence between beams.**
  - Justification?
  - Overconfident !
- **Models physical causes for measurement noise.**
  - Mixture of densities for these causes.
  - Assumes independence between causes. Any Problems?
- **Implementation**
  - Learn parameters based on real data.
  - Different models should be learned for different angles at which the sensor beam hits the obstacle.
  - Determine expected distances by ray-tracing.
  - Expected distances can be pre-processed to form a look-up table.
- **Primary Problems: Lack of Smoothness, Complexity**

Discussion of Beam Models

- Beam-based model is …
  – not smooth for small obstacles and at edges.
  – not very efficient.

Another Idea: Instead of following along the beam, just check the end point.

Likelihood Field Model

- **It is based on assigning a probability model to the *end point of the measurement geometry***

- Avoids many of the problems of Beam Models,
- In particular: Avoids the lack-of-smoothness problem,
- But… ad-hoc method with no physical interpretation or link with sensor physics,
- Still applied and useful in actual applications.

Measurement Probability

- Conditional probability $p(z_t|x_t,m)$ of a measurement is a mixture of …
  - a Gaussian distribution with zero-mean and function of distance to closest obstacle,
  - a uniform distribution for random measurements, and
  - a small uniform distribution for max range measurements.
- Again, independence between different components is assumed.

Measurement Probability

- First, the end point of the range/bearing measurement is found using geometric transformation:

$$\begin{pmatrix} x_{z_t^k} \\ y_{z_t^k} \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_{k,\text{sens}} \\ y_{k,\text{sens}} \end{pmatrix} + z_t^k \begin{pmatrix} \cos(\theta + \theta_{k,\text{sens}}) \\ \sin(\theta + \theta_{k,\text{sens}}) \end{pmatrix}$$

Robot orientation

Robot location

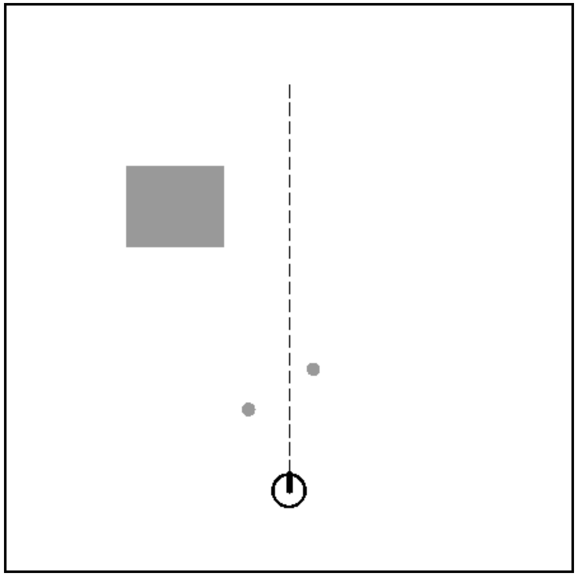
End point of measurement

Range measurement

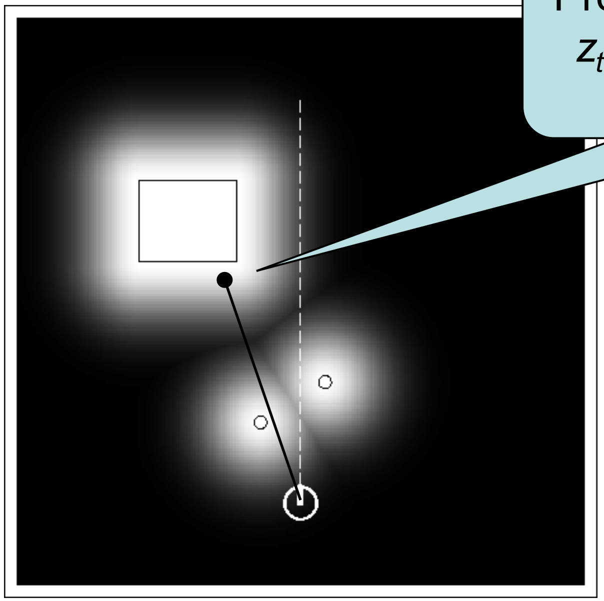
Sensor beam angle with respect to robot orientation

Sensor position in robot body coordinate frame

Example

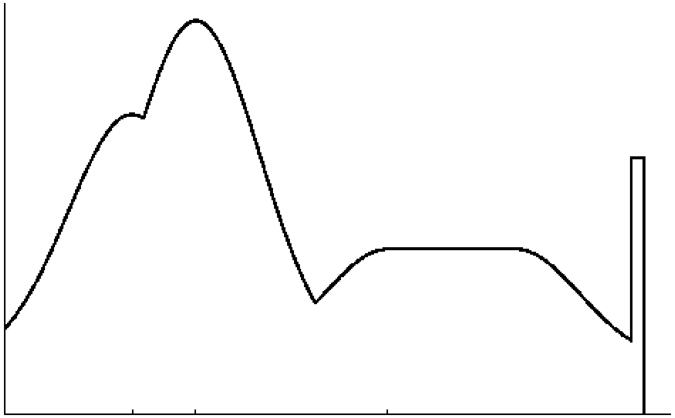


Map m



Likelihood field

$p(z_t|x_t,m)$



Likelihood Field Model - Algorithm

Algorithm likelihood_field_range_finder_model(z_t, x_t, m):

$$q = 1$$

for all k do

if $z_t^k \neq z_{\max}$

$$x_{z_t^k} = x + x_{k,\text{sens}} \cos\theta - y_{k,\text{sens}} \sin\theta + z_t^k \cos(\theta + \theta_{k,\text{sens}})$$

$$y_{z_t^k} = y + y_{k,\text{sens}} \cos\theta + x_{k,\text{sens}} \sin\theta + z_t^k \sin(\theta + \theta_{k,\text{sens}})$$

$$dist = \min_{x',y'} \left\{ \sqrt{(x_{z_t^k} - x')^2 + (y_{z_t^k} - y')^2} \,\middle|\, \langle x', y' \rangle \text{ occupied in } m \right\}$$

$$q = q \cdot \left(z_{\text{hit}} \cdot \mathbf{prob}(dist, \sigma_{\text{hit}}) + \frac{z_{\text{random}}}{z_{\max}} \right)$$

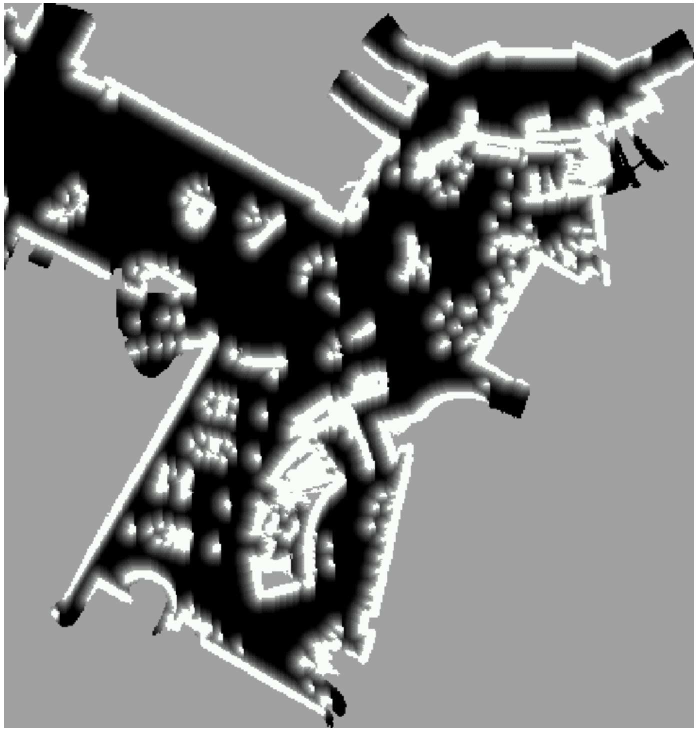
return q

Find the end point of the sensor beam

Find the min distance to the closest obstacle. This is the most costly step (but can be computed off-line)

Example: San Jose Tech Museum

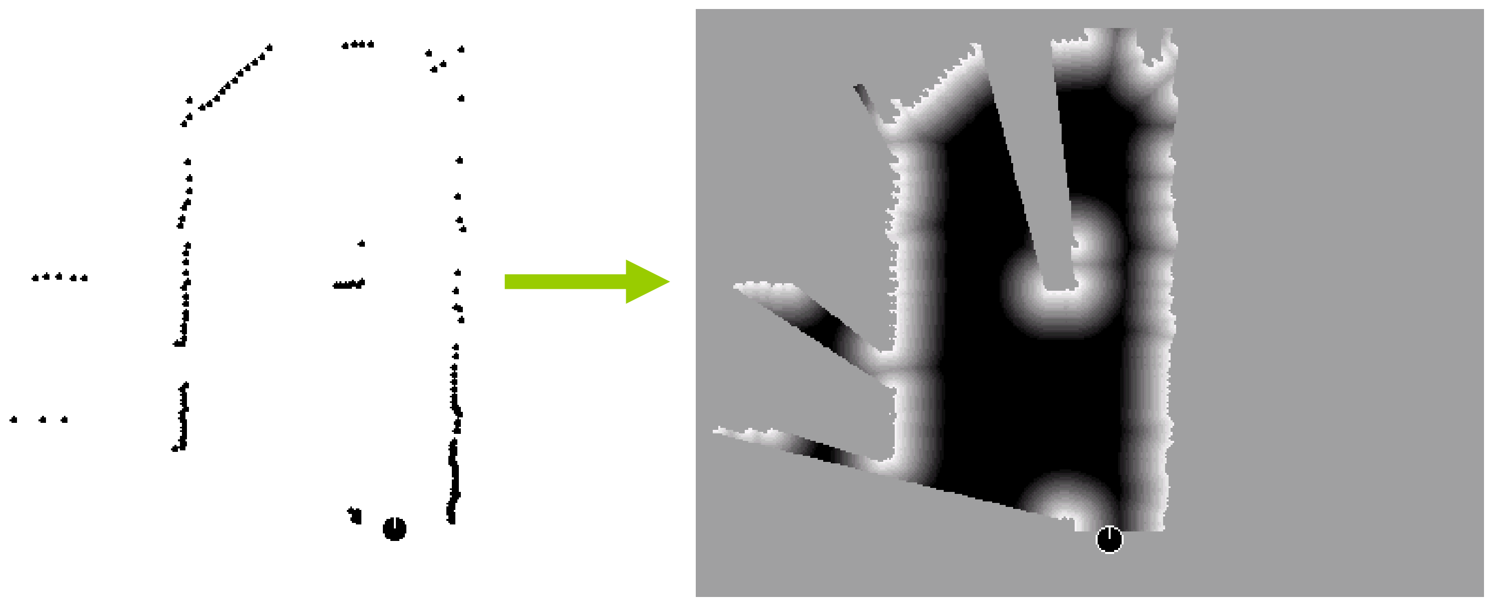
Occupancy grid map



Likelihood field

Likelihood Field over Current Scan

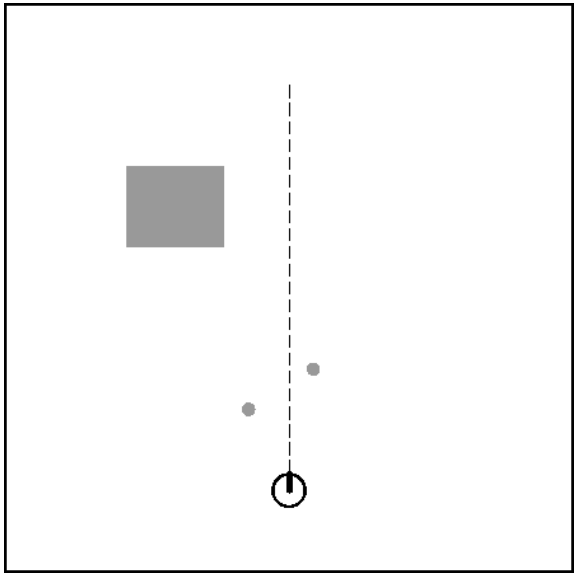
- Likelihood field can also be extracted from a scan,
- Can be used it to “probabilistically” match the scan with a different scan or a map,



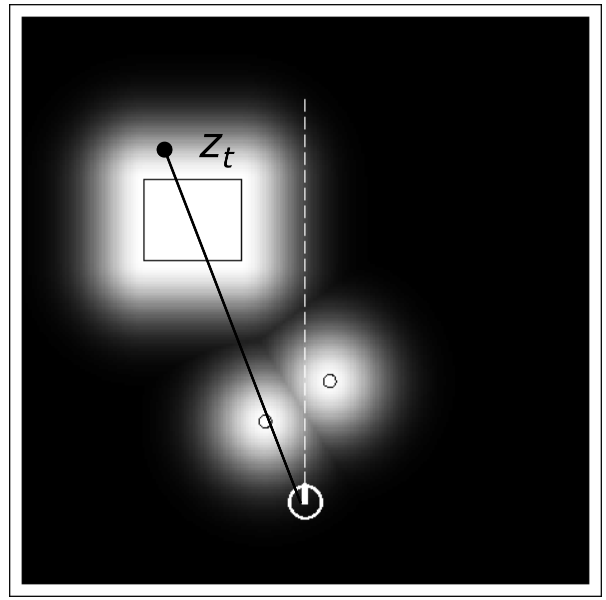
Properties of Likelihood Fields

- Highly efficient, uses 2D tables only.
- Smooth w.r.t. to small changes in robot position.
- Allows gradient descent, scan matching.
- ***However:***
- Ignores physical properties of beams hence not physically meaningful,
- Appears as if "*seeing through the walls*"!!

"Seing through the walls"



Map m



Likelihood field

Likelihood of this particular measurement

$$p(z_t|x_t,m) = \text{large!!}$$

Additional Models: Map Matching

- Map matching (sonar,laser): generate small, local maps from sensor data and match those local maps against a global map.
- Uses a *map correlation function:*

$$\rho_{m,m_{\text{local}},x_t} = \frac{\sum_{x,y}(m_{x,y}-\bar{m})\cdot(m_{x,y,\text{local}}(x_t)-\bar{m})}{\sqrt{\sum_{x,y}(m_{x,y}-\bar{m})^2\sum_{x,y}(m_{x,y,\text{local}}(x_t)-\bar{m})^2}}$$

- Where the "average map" is

$$\bar{m} = \frac{1}{2N}\sum_{x,y}(m_{x,y}+m_{x,y,\text{local}})$$

- Hence the probability of seeing a given local map:

$$p(m_{\text{local}}\mid x_t,m) = \max\{\rho_{m,m_{\text{local}},x_t},0\}$$

Additional Models: Feature Based

- Features (sonar, laser, vision): Extract features such as doors, hallways from sensor data,
- Probabilistic models defined over feature variables instead of raw sensor readings,
- Drastic reduction in computational complexity,
- Given enough time, performance may be inferior to raw measurement approach,
- Very popular in the past, somewhat falling from favor as memory and computation power increases (will always be used and useful though)

Additional Models: Landmark Based

- Finite number of “landmarks” with known IDs on the map,
- Bearing and/or Range to the landmark can be determined (through regular range sensors or specialized cooperation with the landmark – e.g. GPS)
- Each landmark has a signature that can be measured,

Landmark Model - Algorithm

- Assuming "known correspondence" between landmarks observed and landmarks in the map,
- *Algorithm for computing the likelihood of a landmark measurement:*

1: Algorithm landmark_model_known_correspondence(f_t^i, c_t^i, x_t, m):

2: $j = c_t^i$

3: $\hat{r} = \sqrt{(m_{j,x} - x)^2 + (m_{j,y} - y)^2}$

4: $\hat{\phi} = \text{atan2}(m_{j,y} - y, m_{j,x} - x)$

5: $q = \mathbf{prob}(r_t^i - \hat{r}, \sigma_r) \cdot \mathbf{prob}(\phi_t^i - \hat{\phi}, \sigma_\phi) \cdot \mathbf{prob}(s_t^i - s_j, \sigma_s)$

6: *return* q

Find the errors between the known ID landmark and measured landmark. Assume error components are independent.

observed feature $f_t^i = (r_t^i \;\; \phi_t^i \;\; s_t^i)^T$ the robot pose $x_t = (x \;\; y \;\; \theta)^T$

Example Landmarks

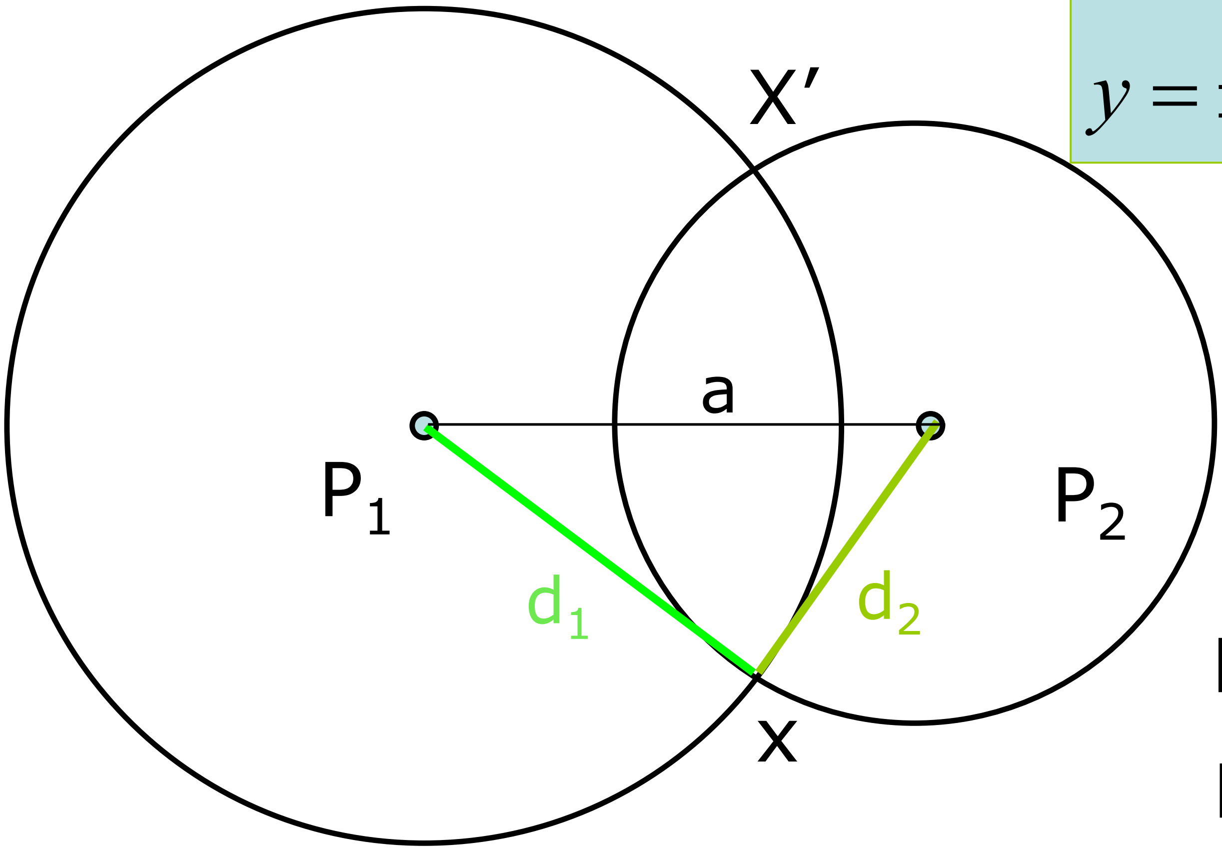
- Active beacons (*e.g.*, radio, GPS)
- Passive (*e.g.*, visual, retro-reflective)
- Standard approach to find range and bearing to landmark is *triangulation*

- Sensor provides
  - distance, or
  - bearing, or
  - distance and bearing.

Example Landmark based Application

Range only – No Uncertainty

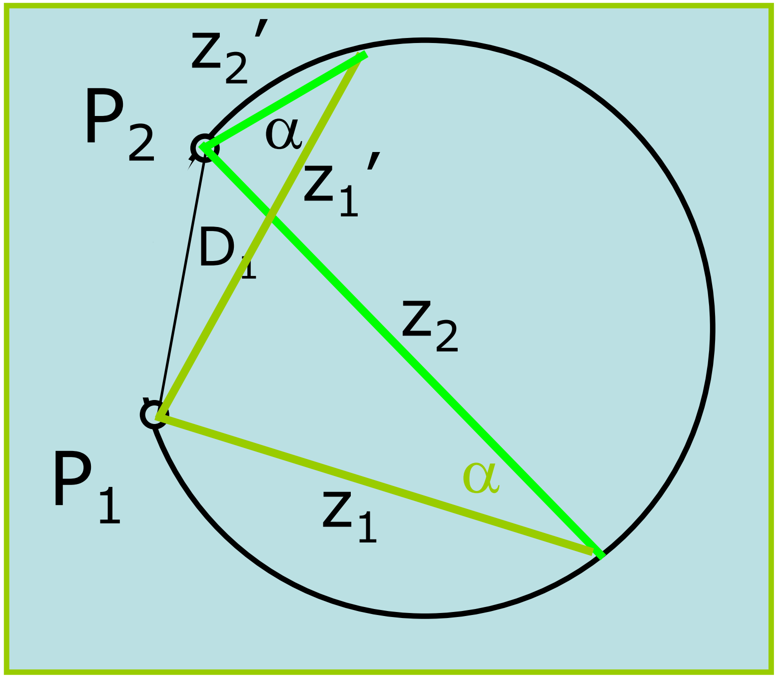
$$x = (a^2 + d_1^2 - d_2^2)/2a$$
$$y = \pm\sqrt{(d_1^2 - x^2)}$$

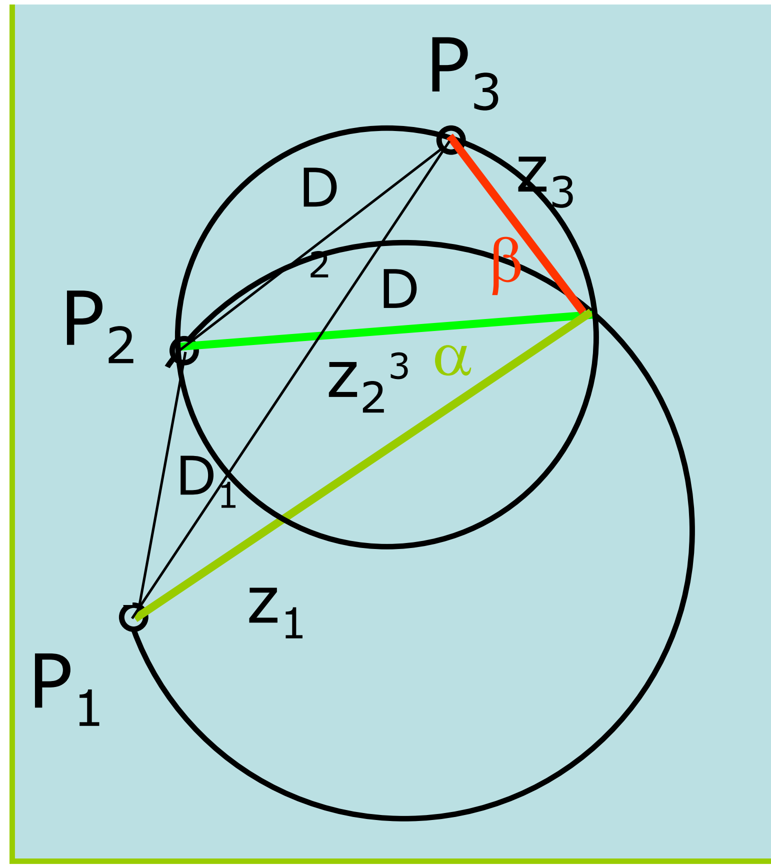


$P_1=(0,0)$

$P_2=(a,0)$

Bearing only – No Uncertainty



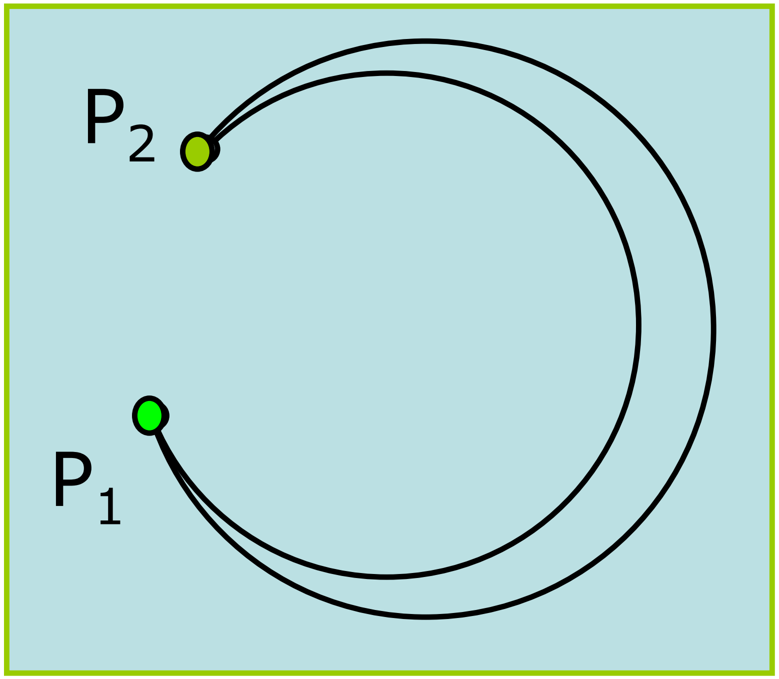


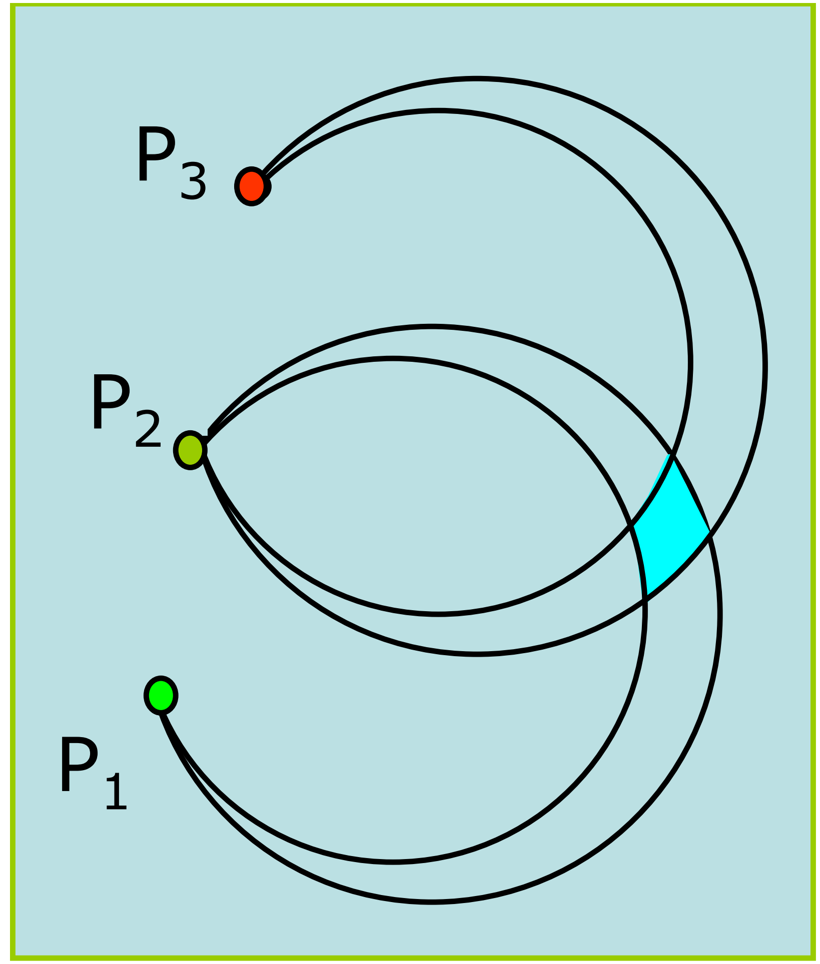
Law of cosine

$$D_1^2 = z_1^2 + z_2^2 - 2\, z_1 z_2 \cos\alpha$$

$$D_1^2 = z_1^2 + z_2^2 - 2\, z_1 z_2 \cos(\alpha)$$
$$D_2^2 = z_2^2 + z_3^2 - 2\, z_1 z_2 \cos(\beta)$$
$$D_3^2 = z_1^2 + z_3^2 - 2\, z_1 z_2 \cos(\alpha + \beta)$$

Bearing only – With Uncertainty





Most approaches attempt to find estimation mean

Sampling from Measurement Models

- Usually more difficult as compared with the motion models,
- However, can be done efficiently for the *landmark based model*,
- Needs some further assumptions
- Reading assignment for you from the book.

Sensor Models - Summary

- Explicitly modeling uncertainty in sensing is key to robustness.
- In many cases, good models can be found by the following approach:
  1. Determine parametric model of noise free measurement.
  2. Analyze sources of noise.
  3. Add adequate noise to parameters (eventually mix in densities for noise).
  4. Learn (and verify) parameters by fitting model to data.
  5. Likelihood of measurement is given by “probabilistically comparing” the actual with the expected measurement.
- This holds for motion models as well.
- It is important to be aware of the underlying assumptions!
- Significant violations of assumptions may cause the model to fail, but only gradually.