CSE-490DF
Robotics Capstone

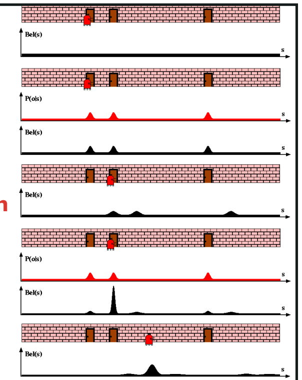
Mobile Robot Localization

Localization

> "Using sensory information to locate the robot in its environment is the most fundamental problem to providing a mobile robot with autonomous capabilities." [Cox '91]

- **Given**
  - Map of the environment.
  - Sequence of sensor measurements.
- **Wanted**
  - Estimate of the robot's position.
- **Problem classes**
  - Position tracking
  - Global localization
  - Kidnapped robot problem (recovery)

Bayes Filters for Robot Localization

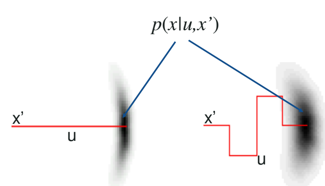


Bayes Filter Algorithm

1. Algorithm **Bayes_filter**(*Bel(x),d*):
2. η=0
3. If *d* is a perceptual data item *z* then
4. For all *x* do
5. $Bel'(x) = P(z \mid x) Bel(x)$
6. $\eta = \eta + Bel'(x)$
7. For all *x* do
8. $Bel'(x) = \eta^{-1} Bel'(x)$
9. Else if *d* is an action data item *u* then
10. For all *x* do
11. $Bel'(x) = \int P(x \mid u, x')\, Bel(x')\, dx'$
12. Return *Bel'(x)*

Probabilistic Kinematics

- Odometry information is inherently noisy.

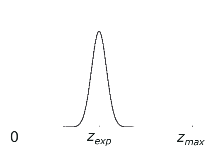


Proximity Measurement

- Measurement can be caused by …
  - a known obstacle.
  - cross-talk.
  - an unexpected obstacle (people, furniture, …).
  - missing all obstacles (total reflection, glass, …).
- Noise is due to uncertainty …
  - in measuring distance to known obstacle.
  - in position of known obstacles.
  - in position of additional obstacles.
  - whether obstacle is missed.

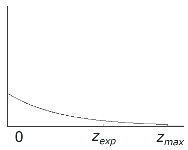
Beam-based Proximity Model

Measurement noise



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

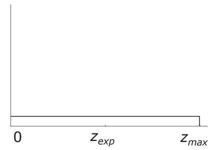
Unexpected obstacles



$$P_{unexp}(z \mid x,m) = \eta\, \lambda\, e^{-\lambda z}$$

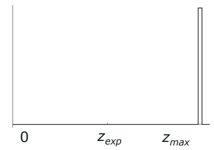
Beam-based Proximity Model

Random measurement



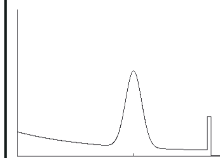
$$P_{rand}(z \mid x,m) = \eta \frac{1}{z_{max}}$$

Max range



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

Mixture Density

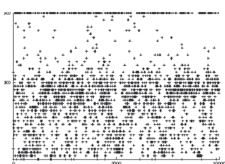


$$P(z \mid x,m) = \begin{pmatrix} \alpha_{hit} \\ \alpha_{unexp} \\ \alpha_{max} \\ \alpha_{rand} \end{pmatrix}^T \cdot \begin{pmatrix} P_{hit}(z \mid x,m) \\ P_{unexp}(z \mid x,m) \\ P_{max}(z \mid x,m) \\ P_{rand}(z \mid x,m) \end{pmatrix}$$

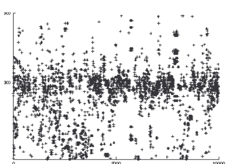
How can we determine the model parameters?

Raw Sensor Data

Measured distances for expected distance of 300 cm.

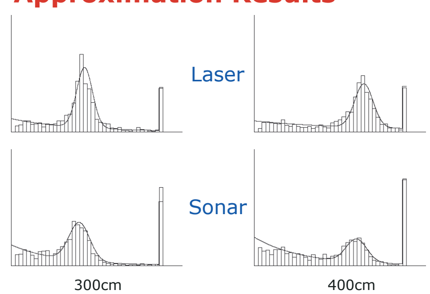


Sonar



Laser

Approximation Results



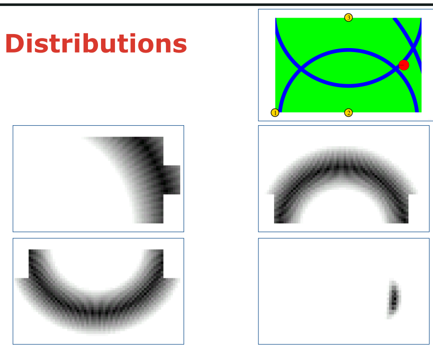
Landmarks

- Active beacons (*e.g.* radio, GPS)
- Passive (*e.g.* visual, retro-reflective)
- Standard approach is triangulation

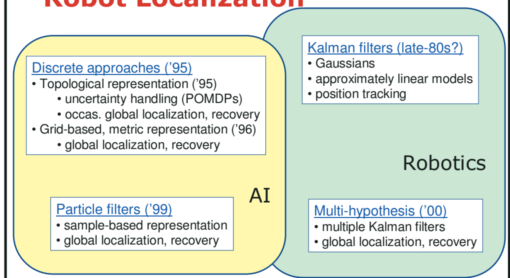
- Sensor provides
  - distance, or
  - bearing, or
  - distance and bearing.

Distance and Bearing

Distributions



Representations for Bayesian Robot Localization



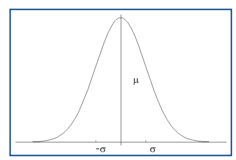
Kalman Filters

Gaussians

$$p(x) \sim N(\mu, \sigma^2):$$

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}}$$

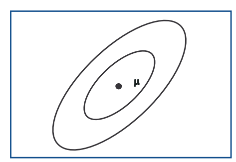
Univariate



$$p(\mathbf{x}) \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma}):$$

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^t \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$$

Multivariate



Properties of Gaussians

$$\left.\begin{array}{l} X \sim N(\mu, \sigma^2) \\ Y = aX + b \end{array}\right\} \Rightarrow Y \sim N(a\mu + b, a^2\sigma^2)$$

$$\left.\begin{array}{l} X_1 \sim N(\mu_1, \sigma_1^2) \\ X_2 \sim N(\mu_2, \sigma_2^2) \end{array}\right\} \Rightarrow \begin{array}{l} X_1 + X_2 \sim N\left(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2\right) \\ X_1 \cdot X_2 \sim N\left(\frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}\mu_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}\mu_2, \ \frac{1}{\sigma_1^{-2} + \sigma_2^{-2}}\right) \end{array}$$

- We stay in the "Gaussian world" as long as we start with Gaussians and perform only linear transformations.

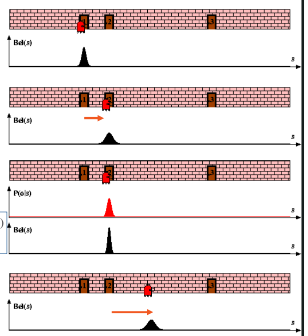
Kalman Filters

$$Bel(x_t) = N(\mu_t, \sigma_t^2)$$

$$Bel(x_{t+1}^-) = \begin{cases} \mu_{t+1}^- = \mu_t + Bu_t \\ \sigma_{t+1}^{-2} = \sigma_t^2 + \sigma_{act}^2 \end{cases}$$

$$Bel(x_{t+1}) = \begin{cases} \mu_{t+1} = \mu_{t+1}^- + K_{t+1}(\mu_{z_{t+1}} - \mu_{t+1}^-) \\ \sigma_{t+1}^2 = (1 - K_{t+1})\sigma_{t+1}^{-2} \end{cases}$$

$$Bel(x_{t+2}^-) = \begin{cases} \mu_{t+2}^- = \mu_{t+1} + Bu_{t+1} \\ \sigma_{t+2}^{-2} = \sigma_{t+1}^2 + \sigma_{act}^2 \end{cases}$$

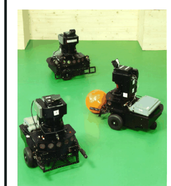


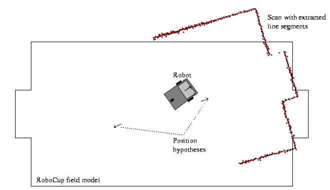
Kalman Filter Algorithm

1. Algorithm **Kalman_filter**(<μ,Σ>, *d*):
2. If *d* is a perceptual data item *z* then
3. $K = \Sigma C^T \left(C \Sigma C^T + \Sigma_{obs} \right)^{-1}$
4. $\mu = \mu + K(z - C\mu)$
5. $\Sigma = (I - KC)\Sigma$
6. Else if *d* is an action data item *u* then
7. $\mu = A\mu + Bu$
8. $\Sigma = A \Sigma A^T + \Sigma_{act}$
9. Return <μ,Σ>

Kalman Filter-based Systems (1)

- [Gutmann et al. 96, 98]:
  - Match LRF scans against map
  - Highly successful in RoboCup mid-size league





Courtesy of S. Gutmann

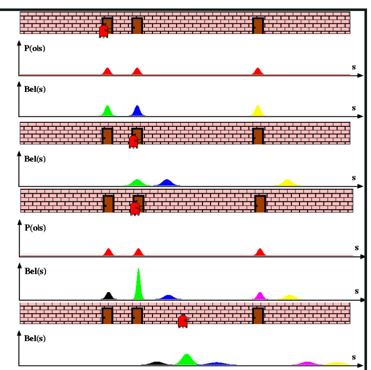
Kalman Filter-based Systems (1)

Courtesy of S. Gutmann

Localization Algorithms - Comparison

	Kalman filter				
Sensors	Gaussian				
Posterior	Gaussian				
Efficiency (memory)	++				
Efficiency (time)	++				
Implementation	+				
Accuracy	++				
Robustness	-				
Global localization	No				

Multi-hypothesis Tracking



Localization With MHT

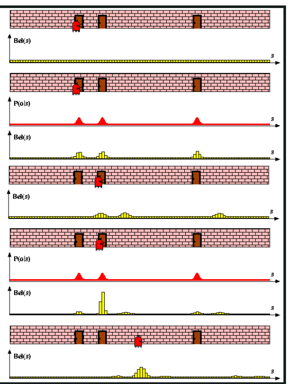
- Belief is represented by multiple hypotheses
- Each hypothesis is tracked by a Kalman filter
- **Additional problems**:
  - Data association: Which observation corresponds to which hypothesis?
  - Hypothesis management: When to add / delete hypotheses?
- Huge body of literature on target tracking, motion correspondence etc.

Localization Algorithms - Comparison

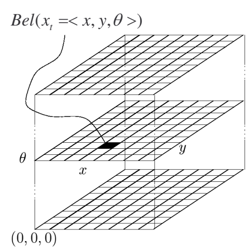
	Kalman filter	Multi-hypothesis tracking			
Sensors	Gaussian	Gaussian			
Posterior	Gaussian	Multi-modal			
Efficiency (memory)	++	++			
Efficiency (time)	++	++			
Implementation	+	o			
Accuracy	++	++			
Robustness	-	+			
Global localization	No	Yes			

Grid-based Approaches

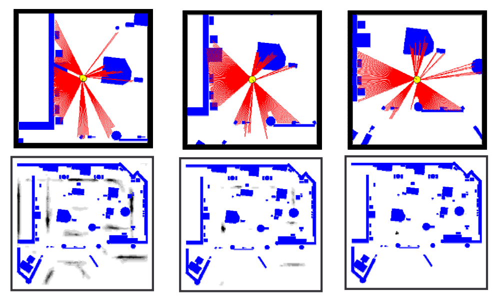
Piecewise Constant



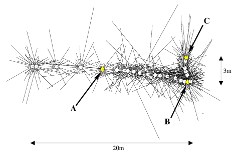
Piecewise Constant Representation

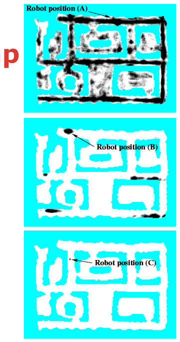


Grid-based Localization



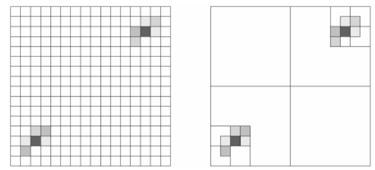
Sonars and Occupancy Grid Map





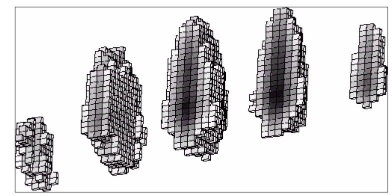
Tree-based Representation

Idea: Represent density using a variant of Octrees



Tree-based Representations

- Efficient in space and time
- Multi-resolution



Localization Algorithms - Comparison

	Kalman filter	Multi-hypothesis tracking	Grid-based (fixed/variable)	Topological maps	
Sensors	Gaussian	Gaussian	Non-Gaussian	Features	
Posterior	Gaussian	Multi-modal	Piecewise constant	Piecewise constant	
Efficiency (memory)	++	++	-/+	++	
Efficiency (time)	++	++	o/+	++	
Implementation	+	o	+/o	+/o	
Accuracy	++	++	+/++	-	
Robustness	-	+	++	+	
Global localization	No	Yes	Yes	Yes	