Intelligent Robots Practice

Motion and Sensing

Chungbuk National University, Korea
Intelligent Robots Lab. (IRL)

Prof. Gon-Woo Kim

Intelligent Robots Lab.
Chungbuk National University

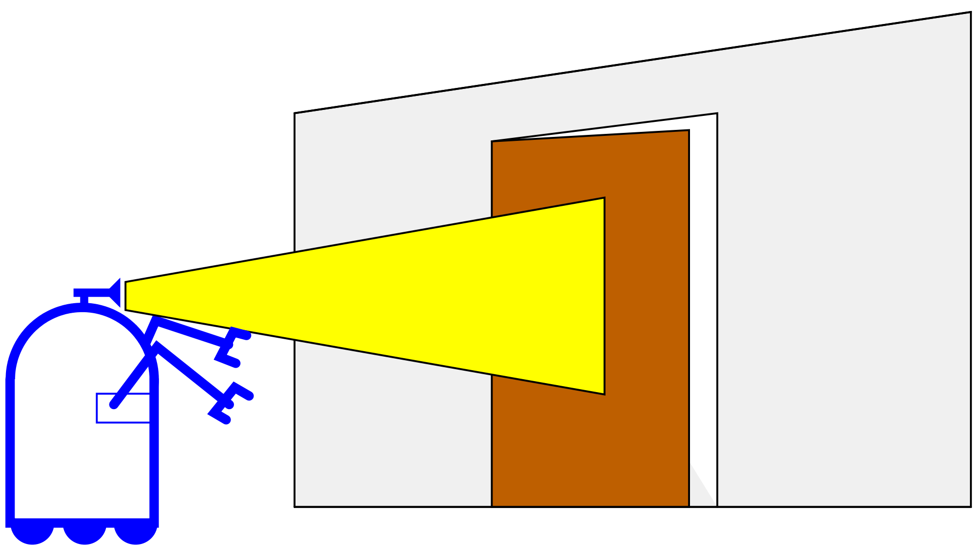
Contents

- Probabilistic Robotics
- Probabilistic Motion Models
- Probabilistic Sensor Models

Probabilistic Robotics

Probabilistic Robotics

- Key idea: Explicit representation of uncertainty
  - Perception = state estimation
  - Action = utility optimization
- Simple Example of State Estimation
  - Suppose a robot obtains measurement z
  - What is P(open|z)?



Causal vs. Diagnostic Reasoning

- P(open|z) is diagnostic.
- P(z|open) is causal.
- Often causal knowledge is easier to obtain.
- Bayes rule allows us to use causal knowledge:

$$P(open \mid z) = \frac{P(z \mid open)P(open)}{P(z)}$$

- Example:
  - P(z|open) = 0.6 P(z|¬open) = 0.3
  - P(open) = P(¬open) = 0.5

$$P(open \mid z) = \frac{P(z \mid open)P(open)}{P(z \mid open)p(open) + P(z \mid \neg open)p(\neg open)}$$

$$P(open \mid z) = \frac{0.6 \cdot 0.5}{0.6 \cdot 0.5 + 0.3 \cdot 0.5} = \frac{2}{3} = 0.67$$

→ *z* raises the probability that the door is open.

Probabilistic Robotics

- Combining Evidence
  - Suppose our robot obtains another observation z_2.
  - How can we integrate this new information?
  - More generally, how can we estimate $P(x| z_1...z_n)$?

  - Recursive Bayesian Updating

$$P(x \mid z_1,\ldots,z_n) = \frac{P(z_n \mid x, z_1,\ldots,z_{n-1})\, P(x \mid z_1,\ldots,z_{n-1})}{P(z_n \mid z_1,\ldots,z_{n-1})}$$

  - **Markov assumption**: z_n is **independent** of $z_1,...,z_{n-1}$ if we know x.

$$P(x \mid z_1,\ldots,z_n) = \frac{P(z_n \mid x)\, P(x \mid z_1,\ldots,z_{n-1})}{P(z_n \mid z_1,\ldots,z_{n-1})}$$

$$= \eta\, P(z_n \mid x)\, P(x \mid z_1,\ldots,z_{n-1})$$

$$= \eta_{1\ldots n} \prod_{i=1\ldots n} P(z_i \mid x)\, P(x)$$

- Second Measurement
  - Example:
    - $P(z_2|open) = 0.5$ $P(z_2|\neg open) = 0.6$
    - $P(open|z_1)=2/3$

$$P(open \mid z_2, z_1) = \frac{P(z_2 \mid open)\, P(open \mid z_1)}{P(z_2 \mid open)\, P(open \mid z_1) + P(z_2 \mid \neg open)\, P(\neg open \mid z_1)}$$

$$= \frac{\frac{1}{2}\cdot\frac{2}{3}}{\frac{1}{2}\cdot\frac{2}{3}+\frac{3}{5}\cdot\frac{1}{3}} = \frac{5}{8} = 0.625$$

→ z_2 lowers the probability that the door is open.

Probabilistic Robotics

- Actions
  - Often the world is **dynamic** since
    - actions carried out by the robot,
    - actions carried out by other agents,
    - or just the time passing by change the world.
  - How can we **incorporate** such **actions**?

- Typical Actions
  - The robot **turns its wheels** to move
  - The robot **uses its manipulator** to grasp an object

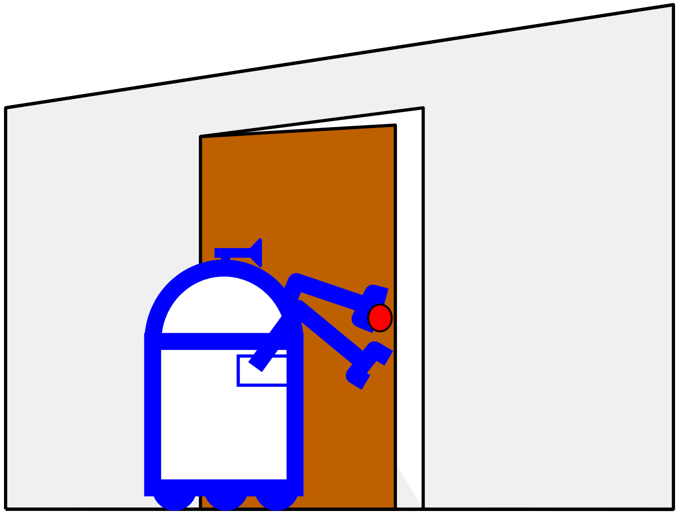
  - Actions are **never carried out with absolute certainty**.
  - In contrast to measurements, **actions generally increase the uncertainty**.

Modeling Actions

- To incorporate the outcome of an action u into the current "belief", we use the conditional PDF

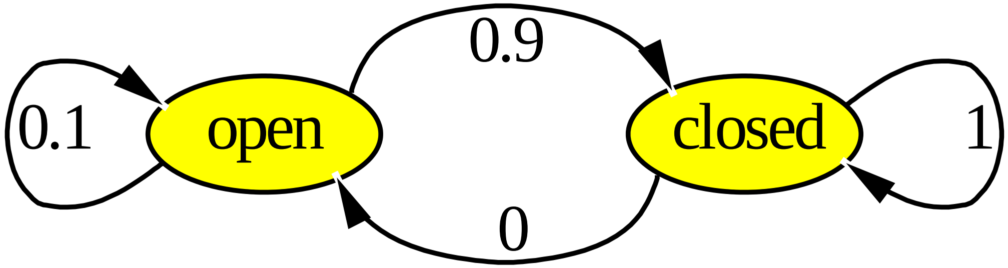
$$P(x|u,x')$$

- This term specifies the pdf that **executing u changes the state from x' to x**.
- Example: Closing the door



Probabilistic Robotics

- State Transitions
  - P(x|u,x') for u = "close door":



  - If the door is open, the action "close door" succeeds in 90% of all cases.
  - Integrating the Outcome of Actions
    - Continuous case:

$$P(x \mid u) = \int P(x \mid u, x')P(x')dx'$$

    - Discrete case:

$$P(x \mid u) = \sum P(x \mid u, x')P(x')$$

- Example: The Resulting Belief

$$
\begin{aligned}
P(closed \mid u) &= \sum P(closed \mid u, x')P(x') \\
&= P(closed \mid u, open)P(open) \\
&\quad + P(closed \mid u, closed)P(closed) \\
&= \frac{9}{10} * \frac{5}{8} + \frac{1}{1} * \frac{3}{8} = \frac{15}{16}
\end{aligned}
$$

$$
\begin{aligned}
P(open \mid u) &= \sum P(open \mid u, x')P(x') \\
&= P(open \mid u, open)P(open) \\
&\quad + P(open \mid u, closed)P(closed) \\
&= \frac{1}{10} * \frac{5}{8} + \frac{0}{1} * \frac{3}{8} = \frac{1}{16} \\
&= 1 - P(closed \mid u)
\end{aligned}
$$

Probabilistic Robotics

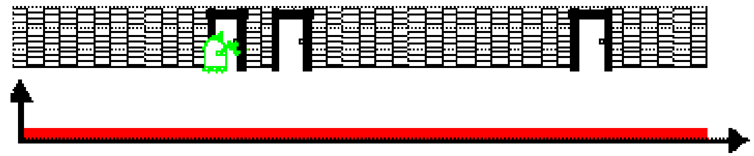
- Bayes Filters: Framework
  - **Given:**
    - Stream of observations z and action data u: $d_t = \{u_1, z_1 \ldots, u_t, z_t\}$
    - Sensor model P(z|x).
    - Action model P(x|u,x').
    - Prior probability of the system state P(x).
  - **Wanted:**
    - Estimate of the state X of a dynamical system.
    - The posterior of the state is also called Belief:

$$Bel(x_t) = P(x_t \mid u_1, z_1 \ldots, u_t, z_t)$$

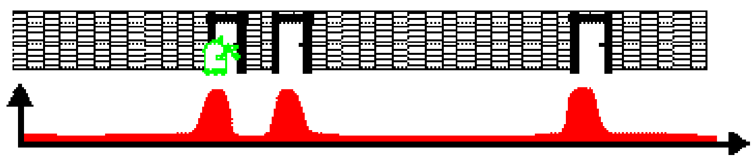
Probabilistic Robotics

- Bayes Filters: Framework

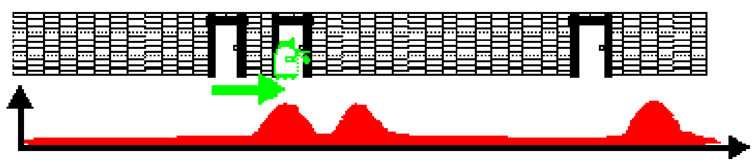
Prior probability of the system state P(x)



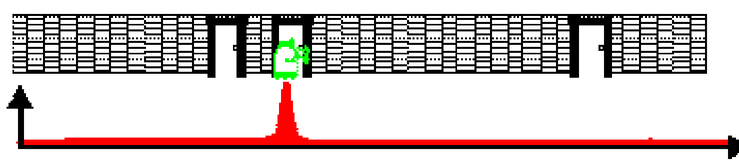
Sensor model P(z|x)



Action model P(x|u,x')



Posterior of the state (belief)



Probabilistic Robotics

z = observation
u = action
x = state

- Bayes Filters

$$\boxed{Bel(x_t)} = P(x_t \mid u_1, z_1 \ldots, u_t, z_t)$$

Bayes $$= \eta\ P(z_t \mid x_t, u_1, z_1, \ldots, u_t)\ P(x_t \mid u_1, z_1, \ldots, u_t)$$

Markov $$= \eta\ P(z_t \mid x_t)\ P(x_t \mid u_1, z_1, \ldots, u_t)$$

Total prob. $$= \eta\ P(z_t \mid x_t) \int P(x_t \mid u_1, z_1, \ldots, u_t, x_{t-1})\ P(x_{t-1} \mid u_1, z_1, \ldots, u_t)\ dx_{t-1}$$

Markov $$= \eta\ P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1})\ P(x_{t-1} \mid u_1, z_1, \ldots, u_t)\ dx_{t-1}$$

Markov $$= \eta P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1}) P(x_{t-1} \mid u_1, z_1, \ldots, z_{t-1})\ dx_{t-1}$$

$$\boxed{= \eta\ P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1})\ Bel(x_{t-1})\ dx_{t-1}}$$

Probabilistic Robotics

- Bayes Filters

$$Bel(x_t) = \eta \ P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1}) \ Bel(x_{t-1}) \ dx_{t-1}$$

1. Algorithm **Bayes_filter**(*Bel(x),d*):
2. $\eta = 0$
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 Robotics

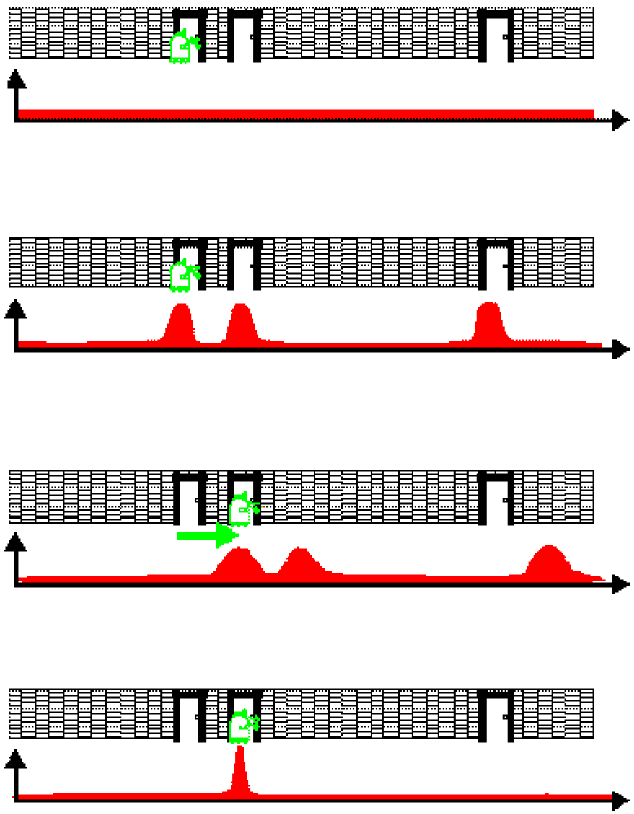
- Bayes Filters
  - **Bayes Filters are Familiar!**

$$Bel(x_t) = \eta \; P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1}) \; Bel(x_{t-1}) \; dx_{t-1}$$

- Kalman filters
- Particle filters
- Hidden Markov models
- Dynamic Bayesian networks
- Partially Observable Markov Decision Processes (POMDPs)

Probabilistic Robotics

- Bayes Filters
  - Bayes Filters in Localization



Action Model

Sensor Model

$$Bel(x_t) = \eta \, P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1}) \, Bel(x_{t-1}) \, dx_{t-1}$$

Probabilistic Motion Models

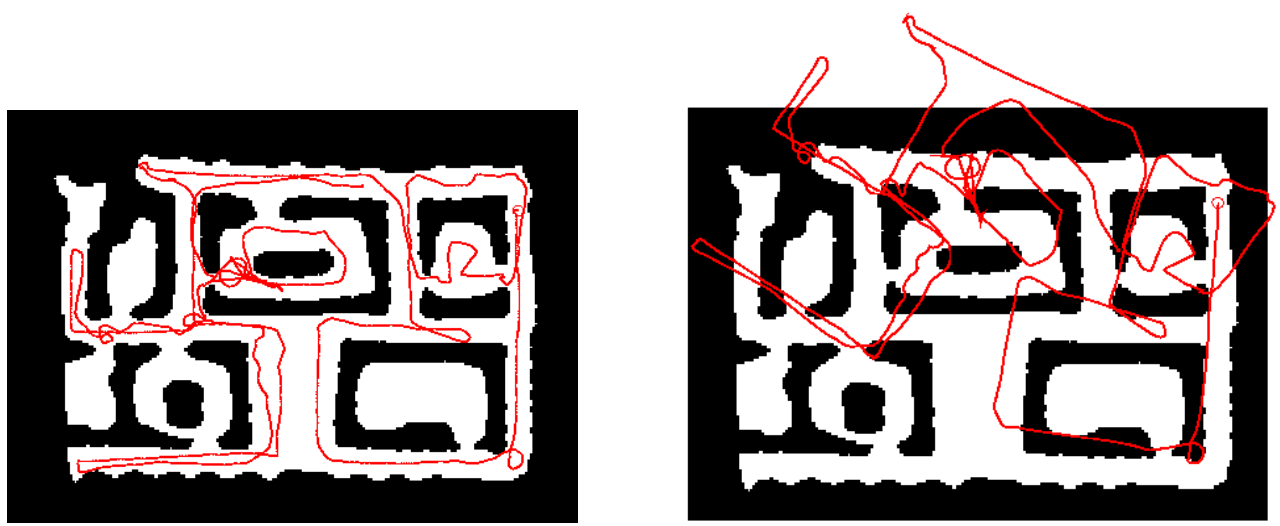
Probabilistic Motion Models

- Bayes Filters
  - Robot Motion

Action Model: p(x’ | u, x)

$$Bel(x_t) = \eta \; P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1}) \; Bel(x_{t-1}) \; dx_{t-1}$$

  - Robot motion is inherently uncertain.
  - How can we model this uncertainty?



Probabilistic Motion Models

- Probabilistic Motion Models
  - To implement the Bayes Filter, we need the transition model p(x | x', u).
  - The term p(x | x', u) specifies a posterior probability, that action u carries the robot from x' to x.
  - In this section we will specify, how p(x | x', u) can be modeled based on the motion equations.

  - Typical Motion Models
    - Odometry-based
      - Odometry-based models are used when systems are equipped with wheel encoders.
    - Velocity-based (dead reckoning)
      - Velocity-based models have to be applied when no wheel encoders are given

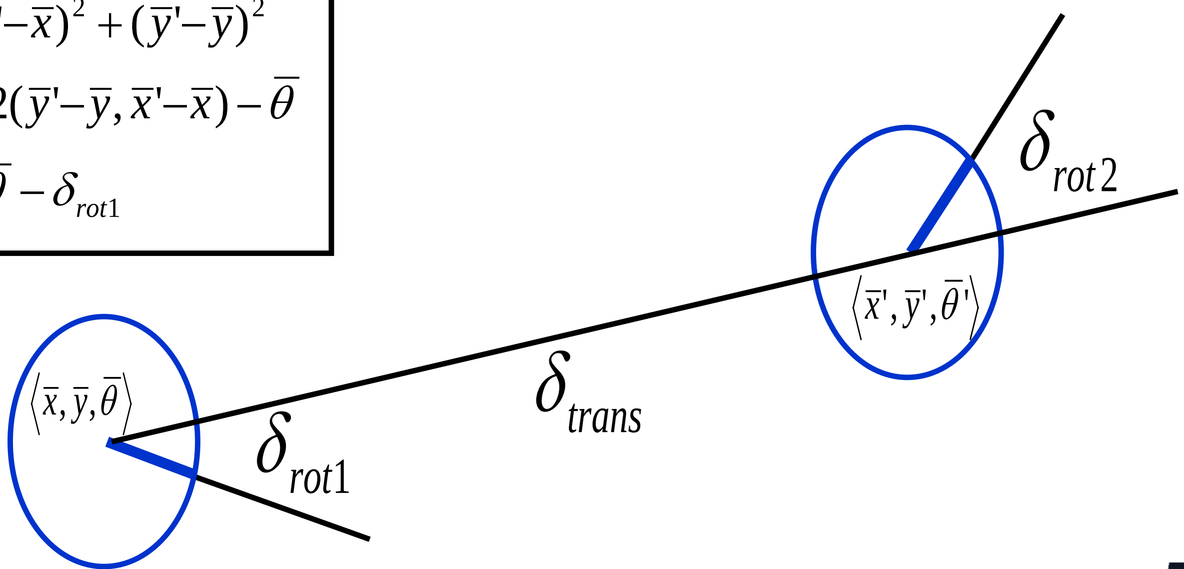
Probabilistic Motion Models

- Probabilistic Motion Models
  - Odometry Model
    - Robot moves from $\langle \bar{x}, \bar{y}, \bar{\theta} \rangle$ to $\langle \bar{x}', \bar{y}', \bar{\theta}' \rangle$
    - Odometry information $u = \langle \delta_{rot1}, \delta_{rot2}, \delta_{trans} \rangle$

$$\delta_{trans} = \sqrt{(\bar{x}' - \bar{x})^2 + (\bar{y}' - \bar{y})^2}$$

$$\delta_{rot1} = \operatorname{atan2}(\bar{y}' - \bar{y}, \bar{x}' - \bar{x}) - \bar{\theta}$$

$$\delta_{rot2} = \bar{\theta}' - \bar{\theta} - \delta_{rot1}$$



Probabilistic Motion Models

- Probabilistic Motion Models
  - Odometry Model
    - Noise Model for Odometry
      - The measured motion is given by the true motion corrupted with noise.

$$\hat{\delta}_{rot1} = \delta_{rot1} + \varepsilon_{\alpha_1 |\delta_{rot1}| + \alpha_2 |\delta_{trans}|}$$

$$\hat{\delta}_{trans} = \delta_{trans} + \varepsilon_{\alpha_3 |\delta_{trans}| + \alpha_4 |\delta_{rot1} + \delta_{rot2}|}$$

$$\hat{\delta}_{rot2} = \delta_{rot2} + \varepsilon_{\alpha_1 |\delta_{rot2}| + \alpha_2 |\delta_{trans}|}$$

Probabilistic Motion Models

- Probabilistic Motion Models
  - Odometry Model
    - Calculating the Posterior Given x, x', and u

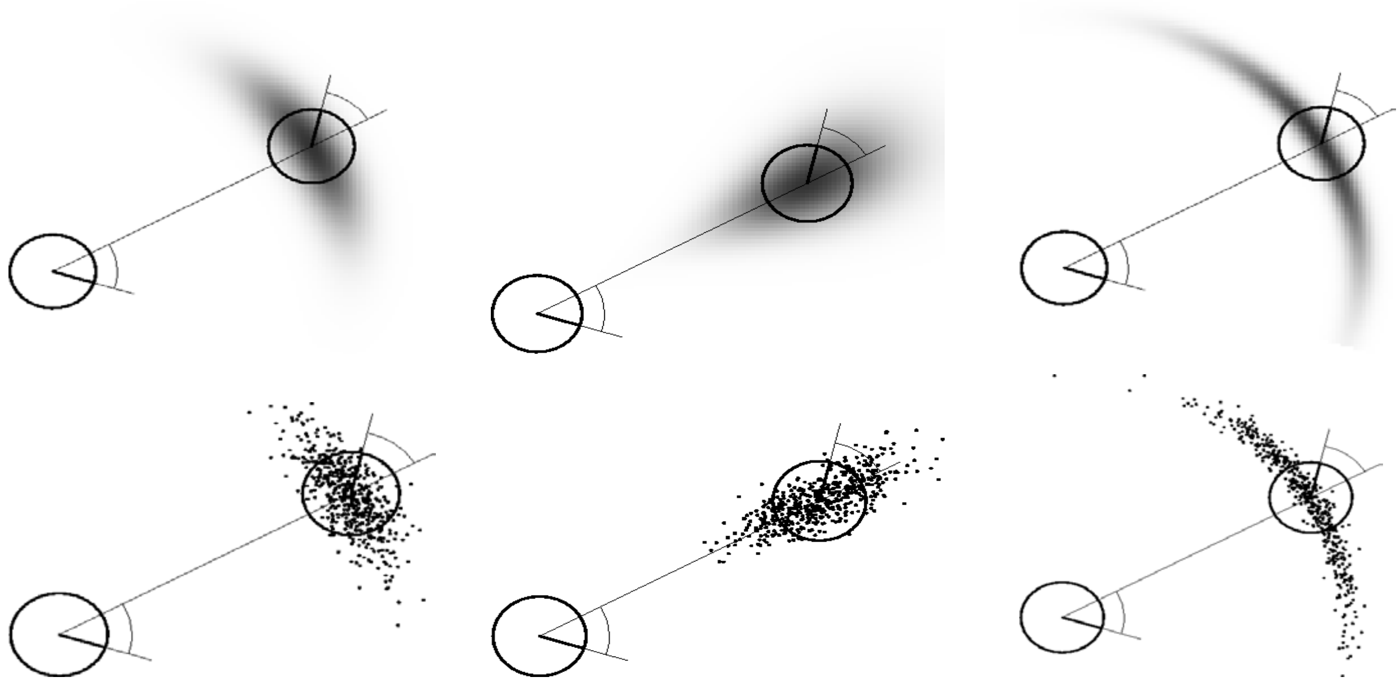
1. Algorithm motion_model_odometry(x,x',u)
2. $\delta_{trans} = \sqrt{(\bar{x}'-\bar{x})^2 + (\bar{y}'-\bar{y})^2}$
3. $\delta_{rot1} = \text{atan2}(\bar{y}'-\bar{y}, \bar{x}'-\bar{x}) - \bar{\theta}$
4. $\delta_{rot2} = \bar{\theta}' - \bar{\theta} - \delta_{rot1}$
5. $\hat{\delta}_{trans} = \sqrt{(x'-x)^2 + (y'-y)^2}$
6. $\hat{\delta}_{rot1} = \text{atan2}(y'-y, x'-x) - \bar{\theta}$
7. $\hat{\delta}_{rot2} = \theta' - \theta - \hat{\delta}_{rot1}$
8. $p_1 = \text{prob}(\delta_{\text{rot1}} - \hat{\delta}_{\text{rot1}}, \alpha_1 |\hat{\delta}_{\text{rot1}}| + \alpha_2 \hat{\delta}_{\text{trans}})$
9. $p_2 = \text{prob}(\delta_{\text{trans}} - \hat{\delta}_{\text{trans}}, \alpha_3 \hat{\delta}_{\text{trans}} + \alpha_4 (|\hat{\delta}_{\text{rot1}}| + |\hat{\delta}_{\text{rot2}}|))$
10. $p_3 = \text{prob}(\delta_{\text{rot2}} - \hat{\delta}_{\text{rot2}}, \alpha_1 |\hat{\delta}_{\text{rot2}}| + \alpha_2 \hat{\delta}_{\text{trans}})$
11. return $p_1 \cdot p_2 \cdot p_3$

odometry values (u)

values of interest (x,x')

Probabilistic Motion Models

- Probabilistic Motion Models
  - Odometry Model
    - Calculating the Posterior Given x, x’, and u
    - Example:



Probabilistic Sensor Models

Probabilistic Sensor Models

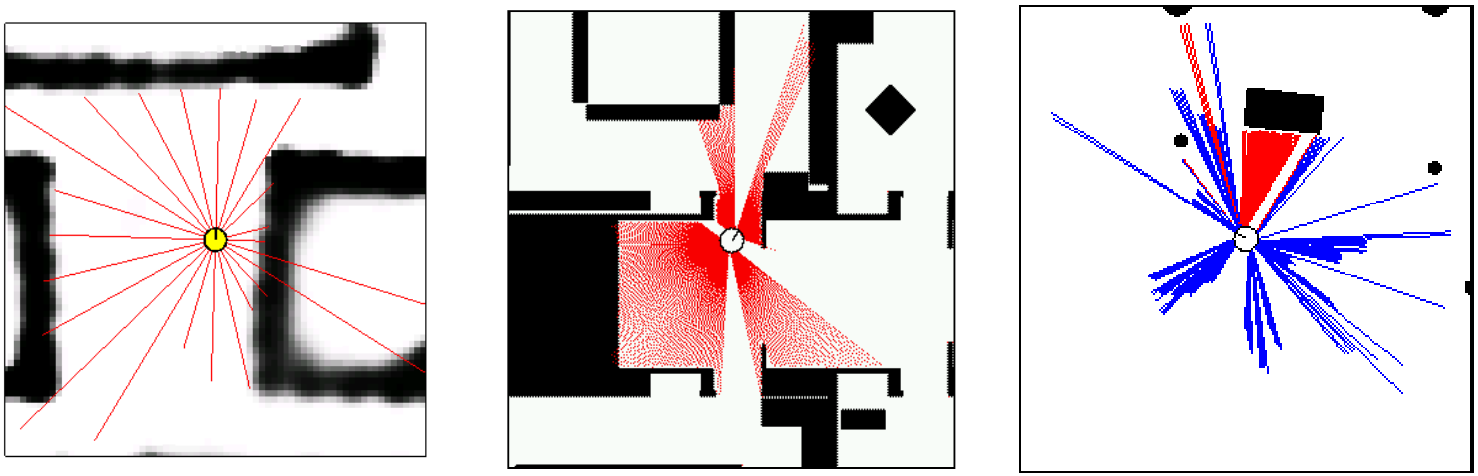
- Bayes Filters
  - Sensor Model

Sensing: p(z | x)

$$Bel(x_t) = \eta \, P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1}) \, Bel(x_{t-1}) \, dx_{t-1}$$

Probabilistic Sensor Models

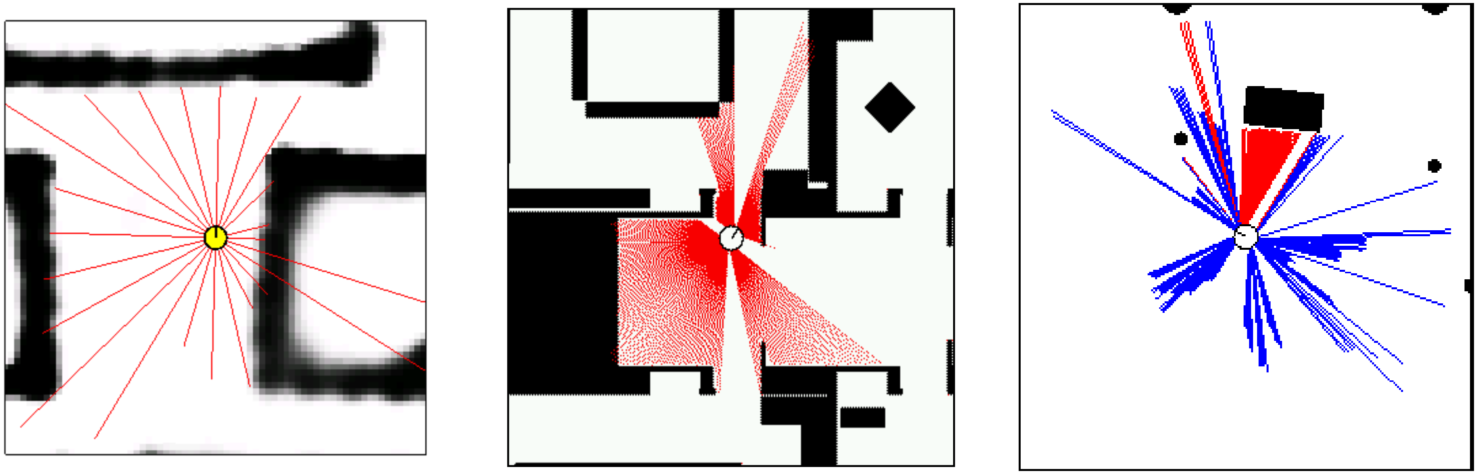
- Probabilistic Sensor Models
  - Sensor Model
    - Proximity Sensors



- The central task is to determine P(z|x), i.e., the probability of a measurement z given that the robot is at position x.
- **Question:** Where do the probabilities come from?
- **Approach:** Let’s try to explain a measurement.

Probabilistic Sensor Models

- Probabilistic Sensor Models
  - Beam-based Sensor Model
    - Scan z consists of K measurements. $z = \{z_1, z_2, \ldots, z_K\}$
    - Individual measurements are independent given the robot position.

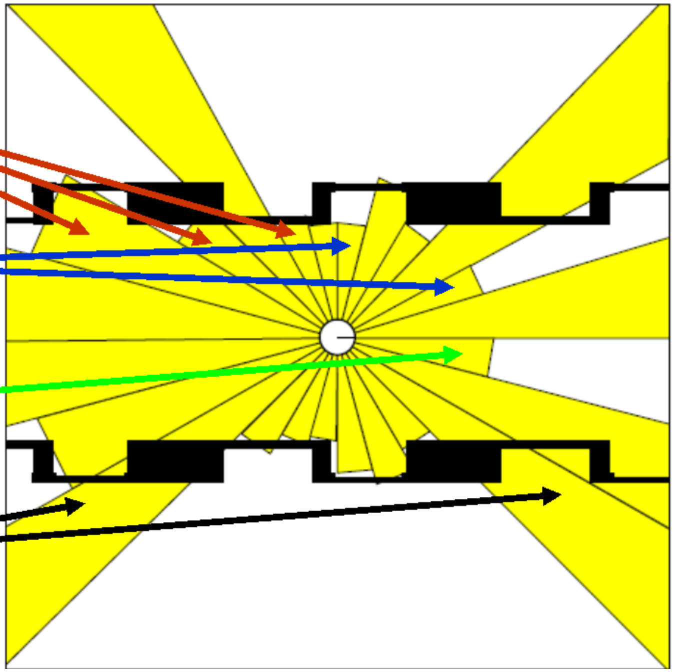


$$P(z \mid x, m) = \prod_{k=1}^{K} P(z_k \mid x, m)$$

Probabilistic Sensor Models

- Probabilistic Sensor Models
  - Beam-based Sensor Model
    - Typical Measurement Errors of Range Measurements

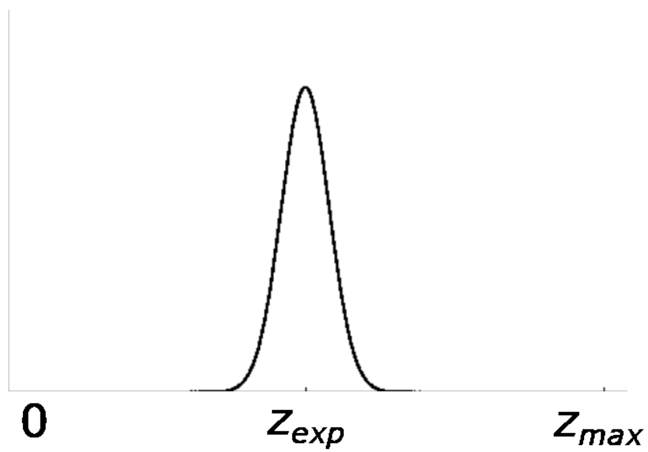
1. Beams reflected by obstacles
2. Beams reflected by persons / caused by crosstalk
3. Random measurements
4. Maximum range measurements



Probabilistic Sensor Models

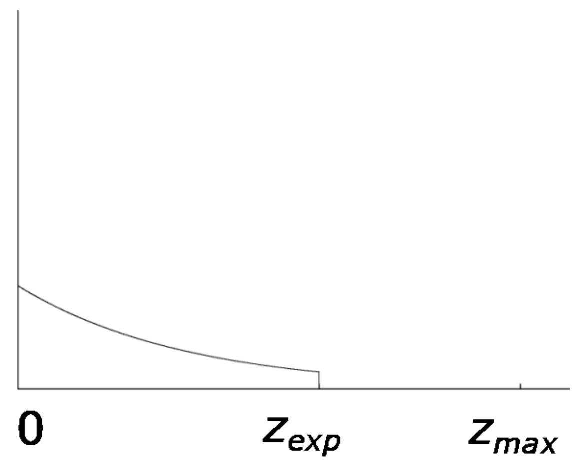
- Probabilistic Sensor Models
  - Beam-based Sensor Model
    - Beam-based Proximity Model

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

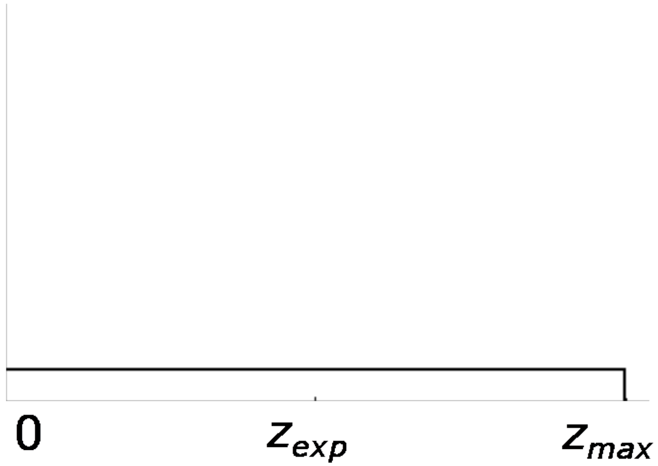


$$P_{unexp}(z \mid x, m) = \begin{cases} \eta \, \lambda \, e^{-\lambda z} & z < z_{exp} \\ 0 & otherwise \end{cases}$$

Probabilistic Sensor Models

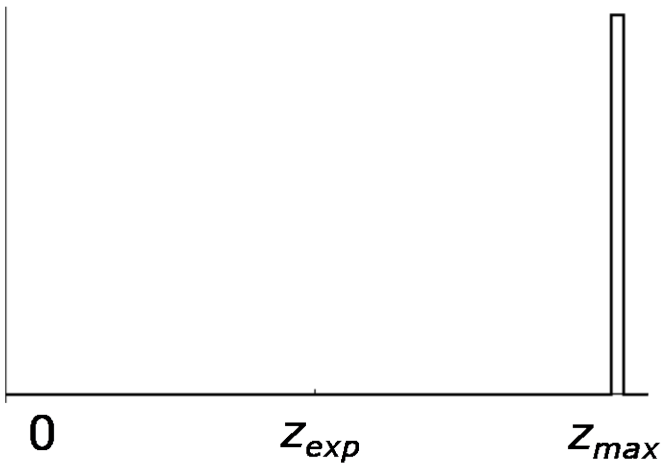
- Probabilistic Sensor Models
  - Beam-based Sensor Model
    - Beam-based Proximity Model

Random measurement



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

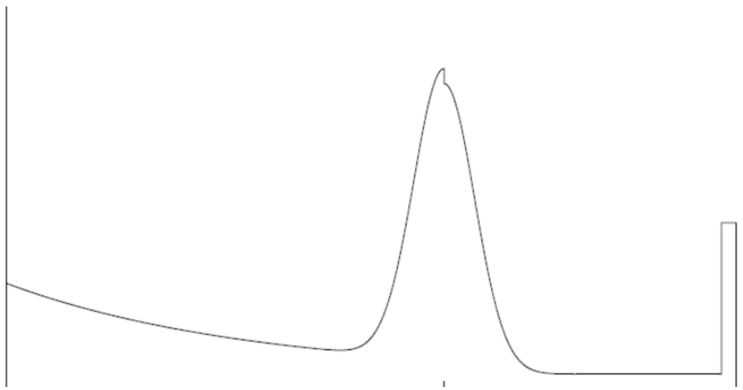
Max range



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

Probabilistic Sensor Models

- Probabilistic Sensor Models
  - Beam-based Sensor Model
    - Beam-based Proximity Model: Resulting Mixture Density



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