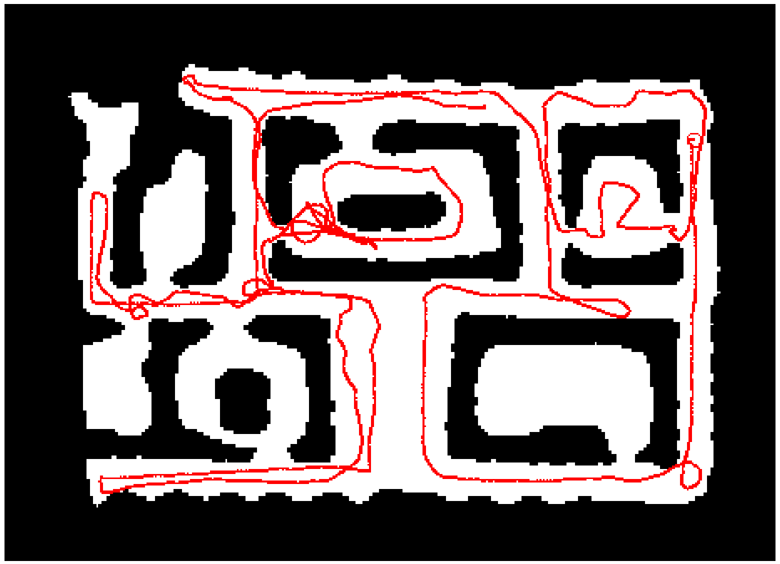
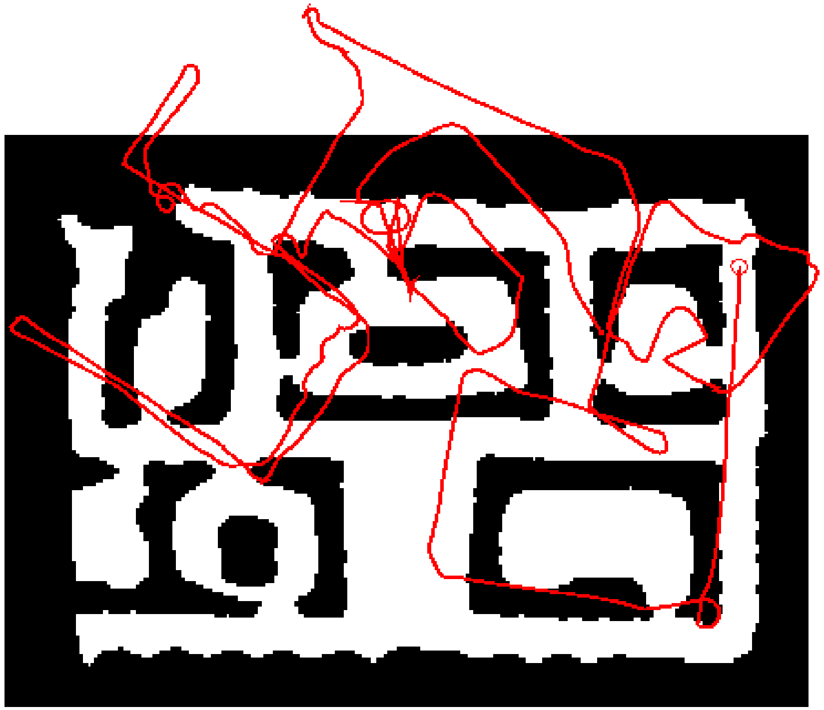
Introduction to Mobile Robotics

Probabilistic Motion Models

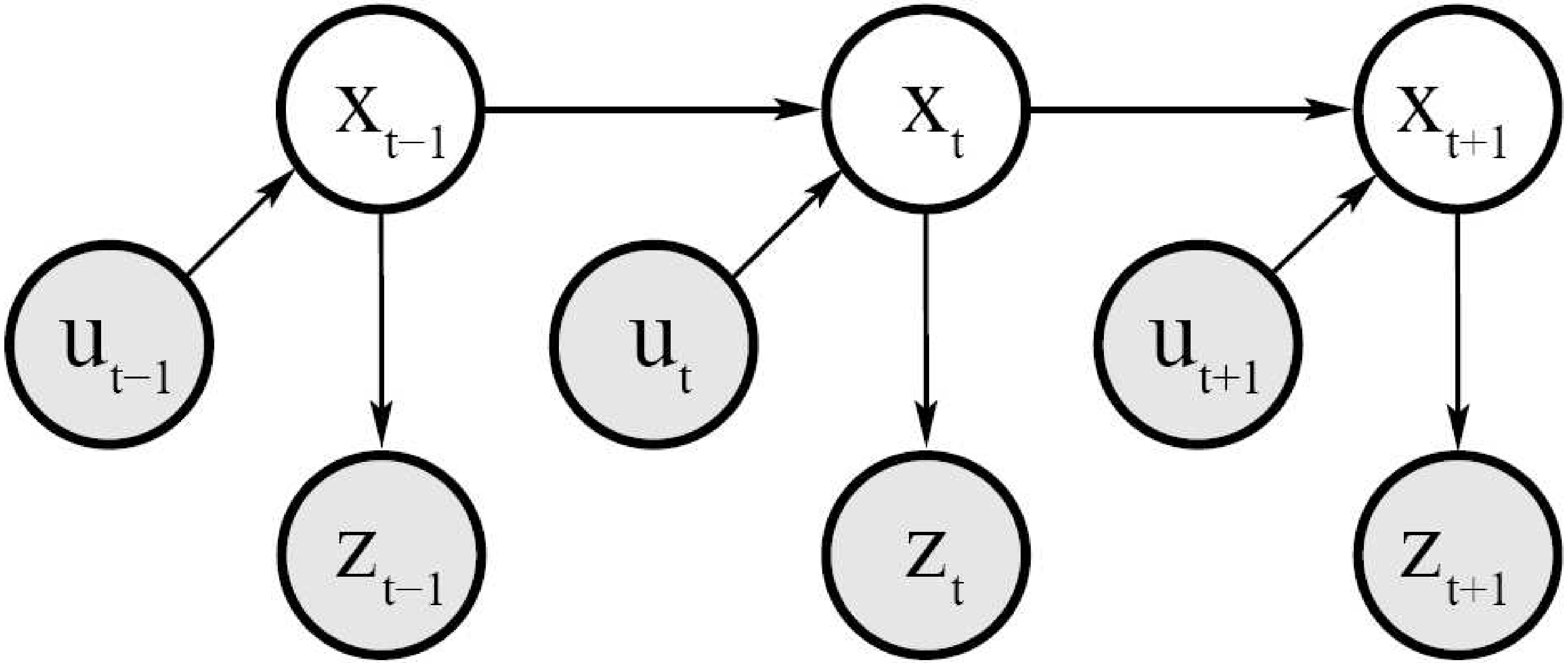
Robot Motion

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





Dynamic Bayesian Network for Controls, States, and Sensations

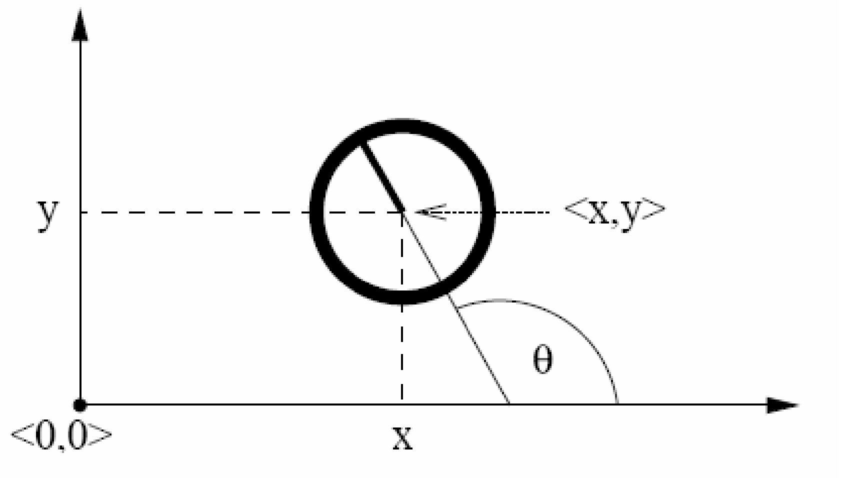


Probabilistic Motion Models

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

Coordinate Systems

- In general the configuration of a robot can be described by six parameters.
- Three-dimensional cartesian coordinates plus three Euler angles pitch, roll, and tilt.
- Throughout this section, we consider robots operating on a planar surface.
- The state space of such systems is three-dimensional (x,y,θ).

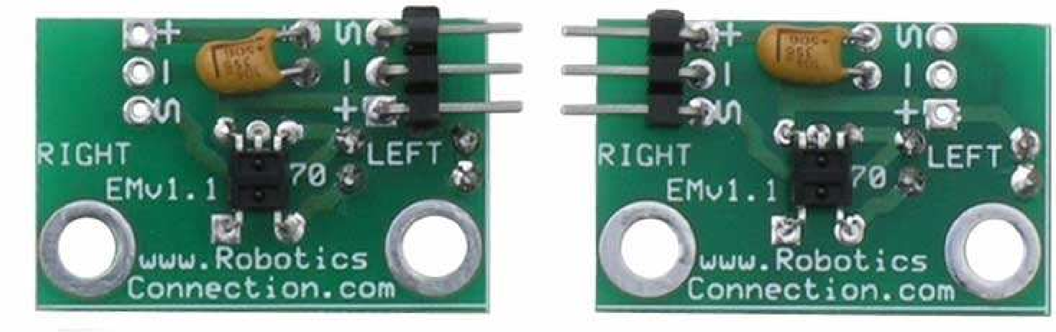


Typical Motion Models

- In practice, one often finds two types of motion models:
  - **Odometry-based**
  - **Velocity-based** (**dead reckoning**)
- Odometry-based models are used when systems are equipped with wheel encoders.
- Velocity-based models have to be applied when no wheel encoders are given.
- They calculate the new pose based on the velocities and the time elapsed.

Example Wheel Encoders

These modules require +5V and GND to power them, and provide a 0 to 5V output. They provide +5V output when they "see" white, and a 0V output when they "see" black.



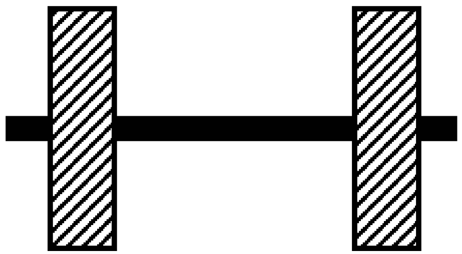
These disks are manufactured out of high quality laminated color plastic to offer a very crisp black to white transition. This enables a wheel encoder sensor to easily see the transitions.

Source: http://www.active-robots.com/

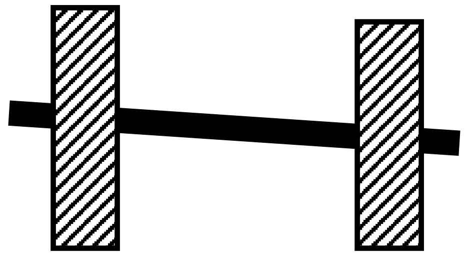
Dead Reckoning

- Derived from “deduced reckoning.”
- Mathematical procedure for determining the present location of a vehicle.
- Achieved by calculating the current pose of the vehicle based on its velocities and the time elapsed.

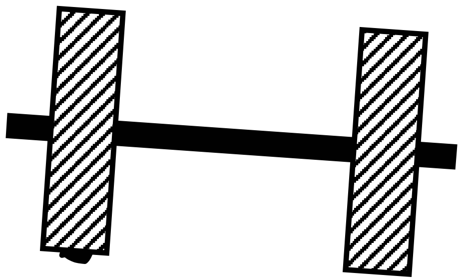
Reasons for Motion Errors



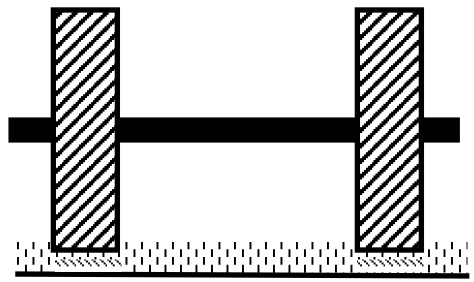
ideal case



different wheel diameters



bump



carpet

and many more …

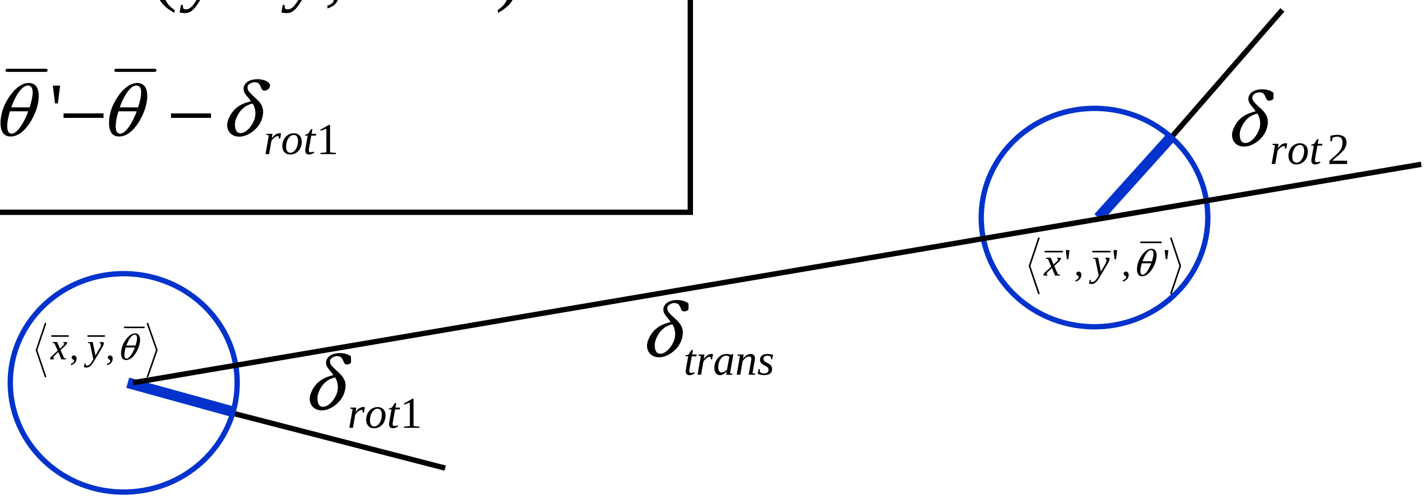
Odometry Model

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

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

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

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



The atan2 Function

- Extends the inverse tangent and correctly copes with the signs of x and y.

$$\text{atan2}(y, x) \;=\; \begin{cases} \text{atan}(y/x) & \text{if } x > 0 \\ \text{sign}(y)\,(\pi - \text{atan}(|y/x|)) & \text{if } x < 0 \\ 0 & \text{if } x = y = 0 \\ \text{sign}(y)\,\pi/2 & \text{if } x = 0, y \neq 0 \end{cases}$$

Noise Model for Odometry

- The measured motion is given by the true motion corrupted with noise.

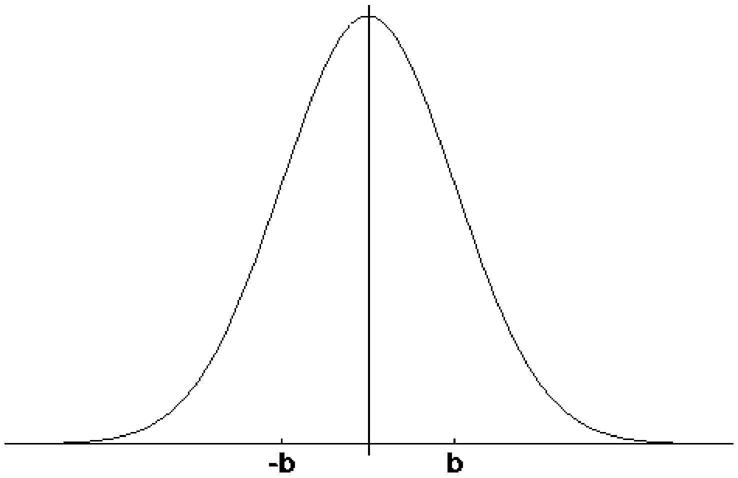
$$\hat{\delta}_{rot1} = \delta_{rot1} + \varepsilon_{\alpha_1 |\delta_{rot1}| + \alpha_2 |\delta_{trans}|}$$

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

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

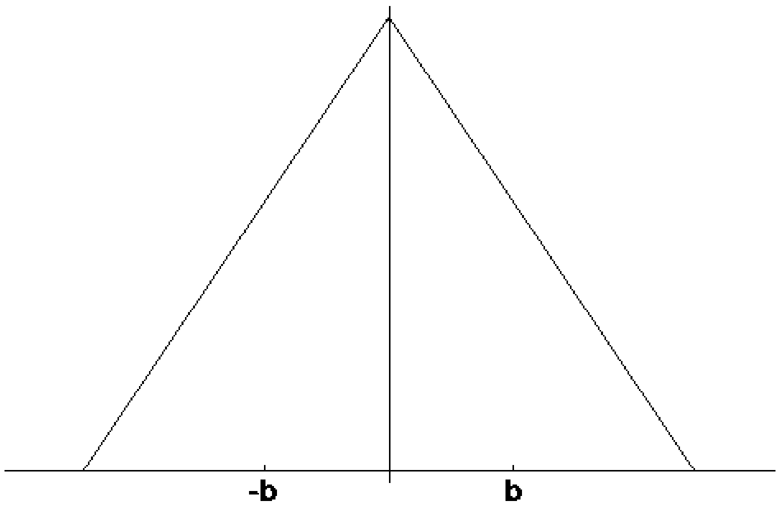
Typical Distributions for Probabilistic Motion Models

Normal distribution



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

Triangular distribution



$$\varepsilon_{\sigma^2}(x) = \begin{cases} 0 \text{ if } |x| > \sqrt{6\sigma^2} \\ \dfrac{\sqrt{6\sigma^2} - |x|}{6\sigma^2} \end{cases}$$

Calculating the Probability (zero-centered)

- For a normal distribution

1. Algorithm **prob_normal_distribution**(a,b):
2. return $\frac{1}{\sqrt{2\pi\, b^2}} \exp\left\{-\frac{1}{2}\frac{a^2}{b^2}\right\}$

- For a triangular distribution

1. Algorithm **prob_triangular_distribution**(a,b):
2. return $\max\left\{0, \frac{1}{\sqrt{6}\, b} - \frac{|a|}{6\, b^2}\right\}$

Calculating the Posterior Given x, x', and u

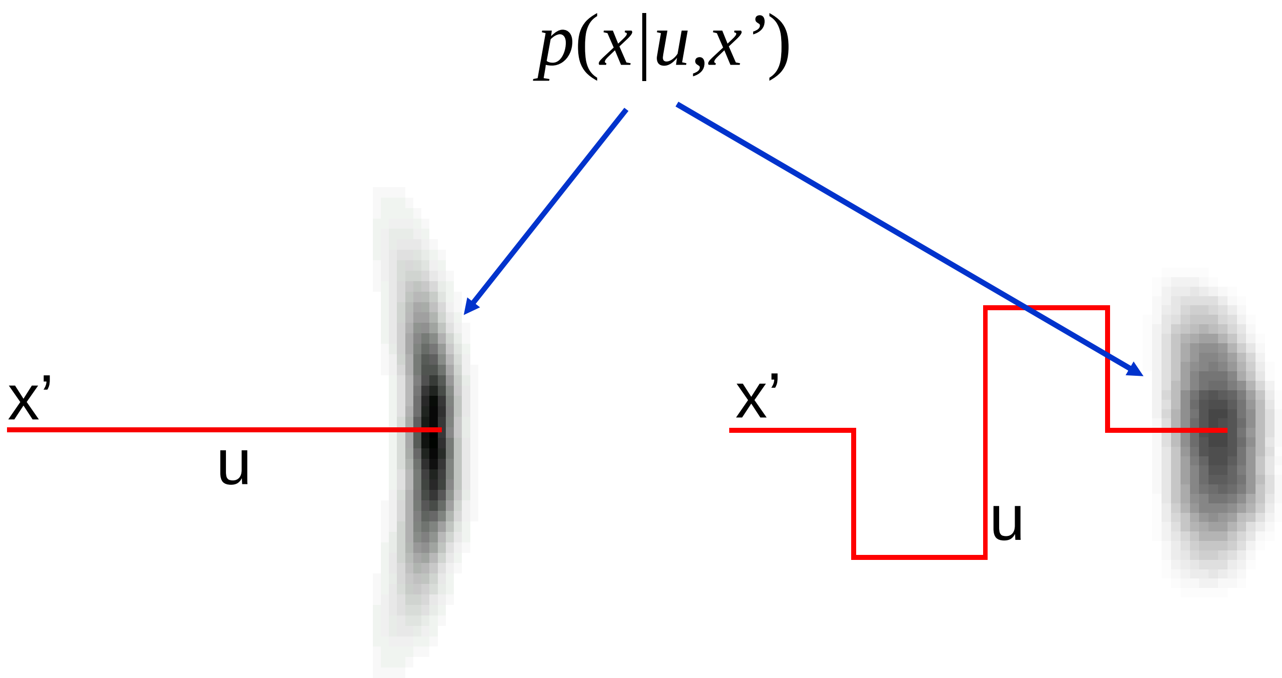
1. Algorithm **motion_model_odometry(x,x',u)**
2. $\delta_{trans} = \sqrt{(\bar{x}' - \bar{x})^2 + (\bar{y}' - \bar{y})^2}$
3. $\delta_{rot1} = \mathrm{atan2}(\bar{y}' - \bar{y}, \bar{x}' - \bar{x}) - \bar{\theta}$
4. $\delta_{rot2} = \bar{\theta}' - \bar{\theta} - \delta_{rot1}$
5. $\hat{\delta}_{trans}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

odometry values (u)

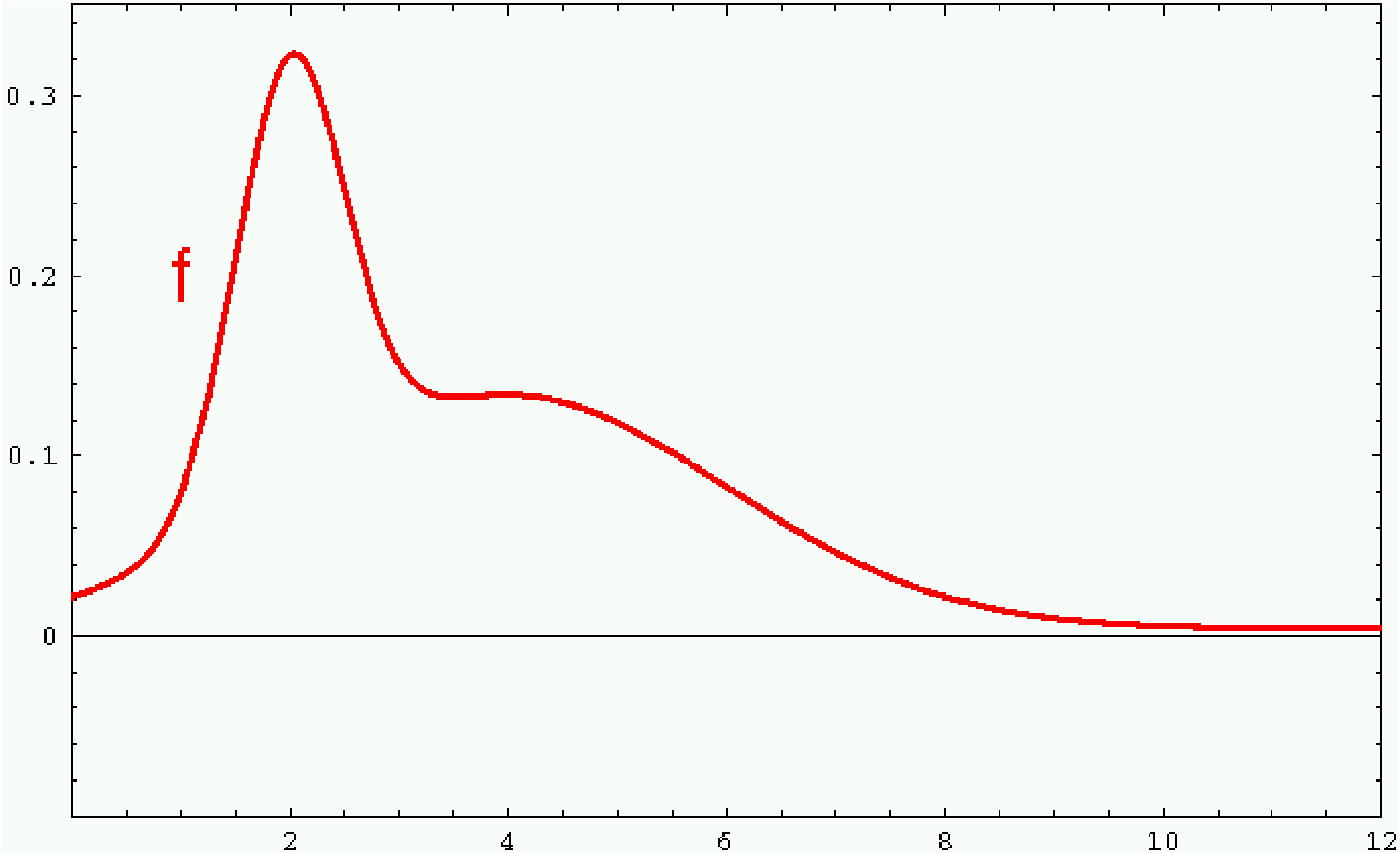
values of interest (x,x')

Application

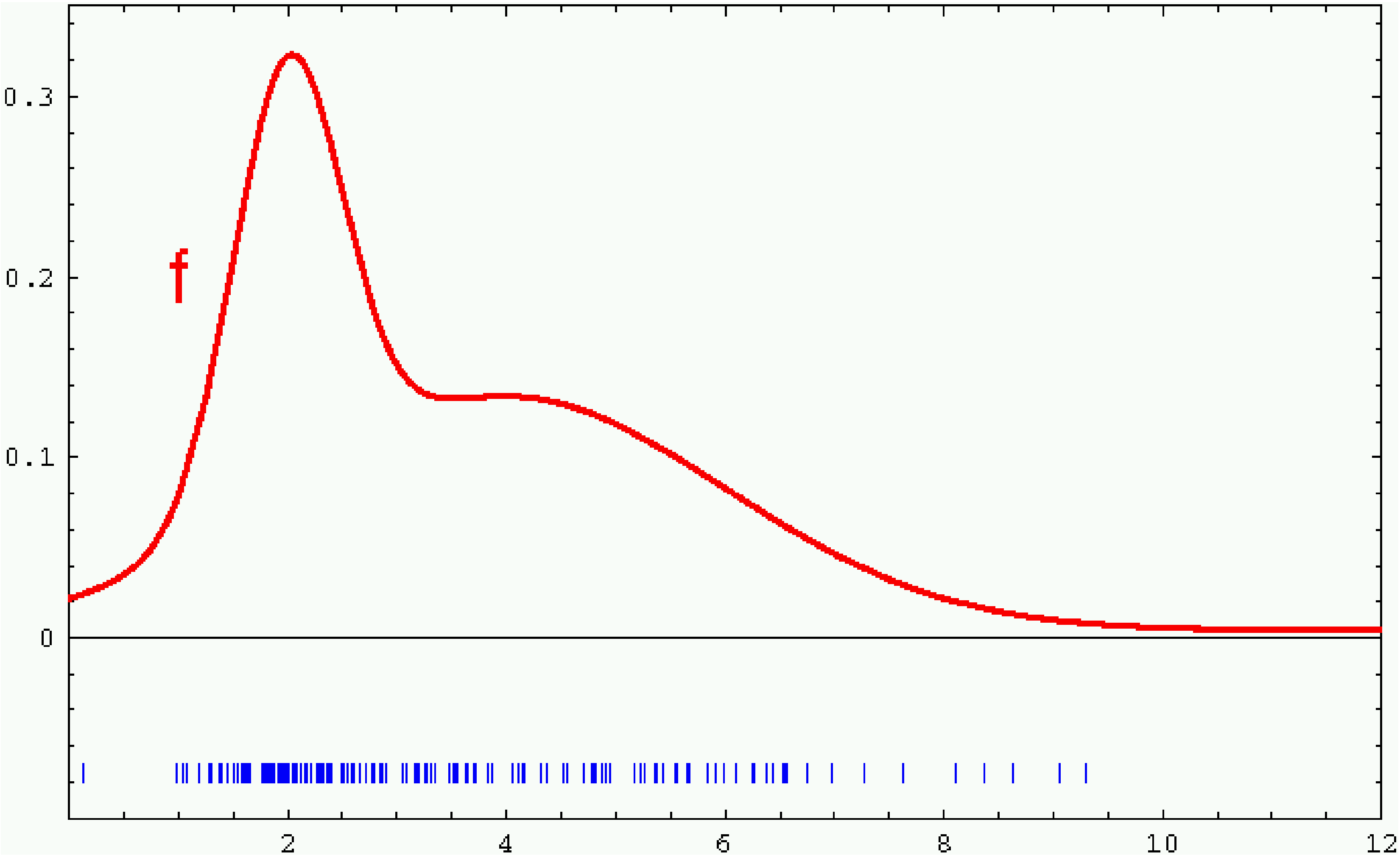
- Repeated application of the sensor model for short movements.
- Typical banana-shaped distributions obtained for 2d-projection of 3d posterior.



Sample-based Density Representation



Sample-based Density Representation



How to Sample from Normal or Triangular Distributions?

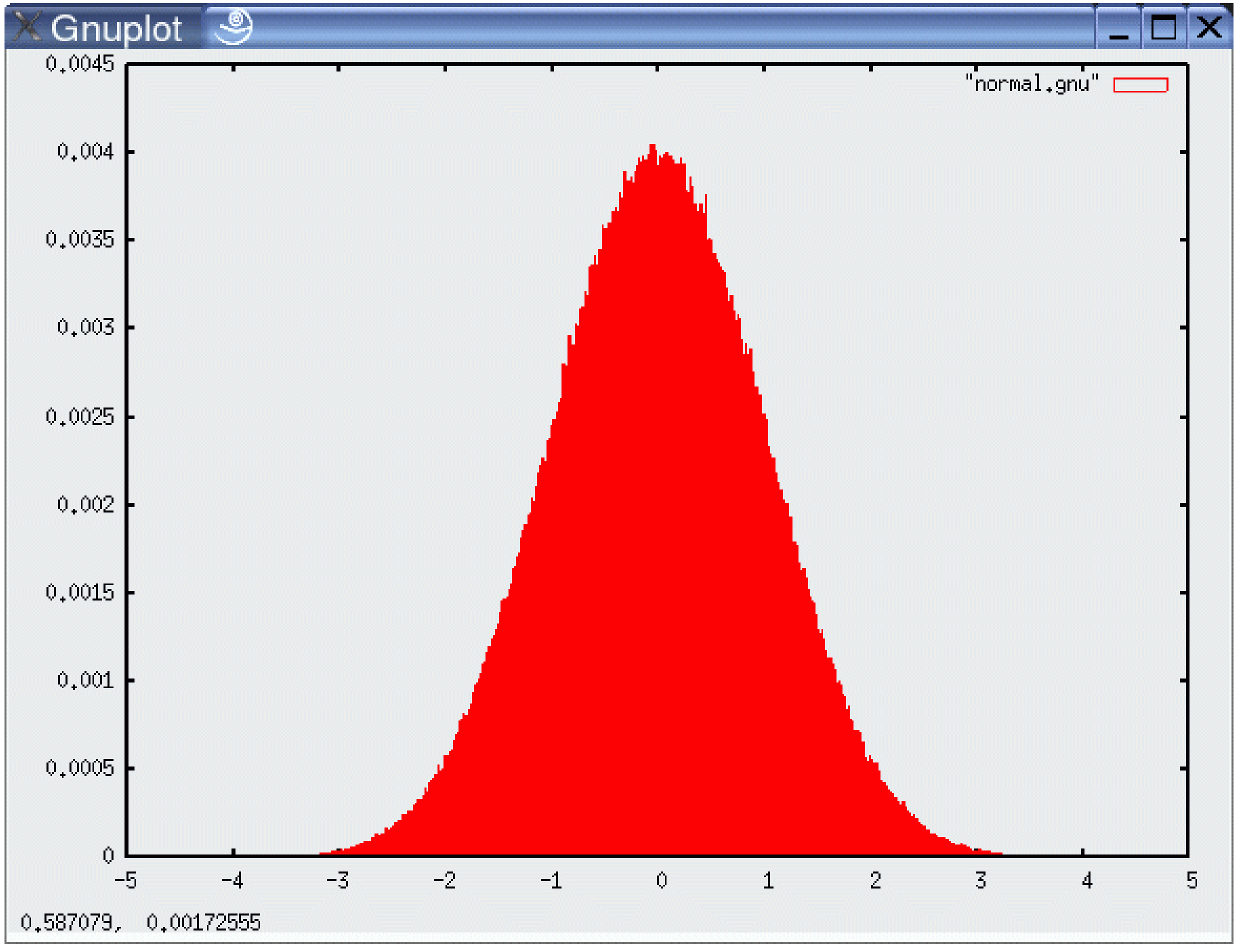
- Sampling from a normal distribution

1. Algorithm **sample_normal_distribution**(b):
2. return $\frac{1}{2}\sum_{i=1}^{12} rand(-b, b)$

- Sampling from a triangular distribution

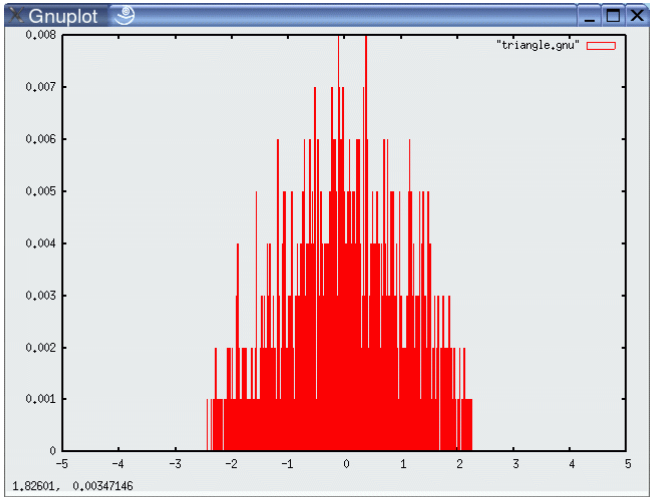
1. Algorithm **sample_triangular_distribution**(b):
2. return $\frac{\sqrt{6}}{2}\left[\mathbf{rand}(-b, b) + \mathbf{rand}(-b, b)\right]$

Normally Distributed Samples

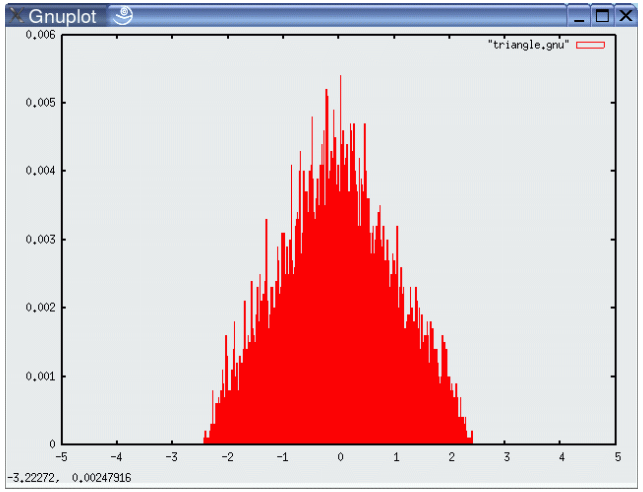


10^6 samples

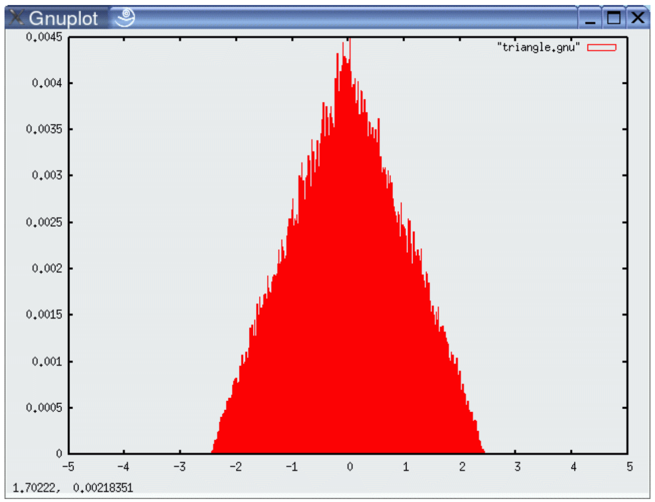
For Triangular Distribution



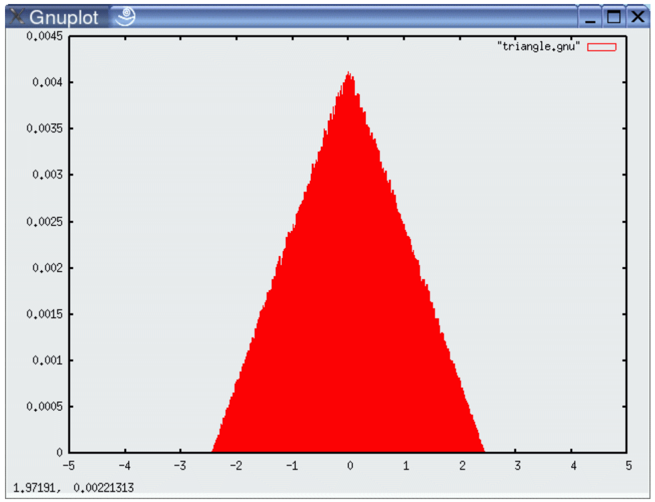
10^3 samples



10^4 samples



10^5 samples



10^6 samples

Rejection Sampling

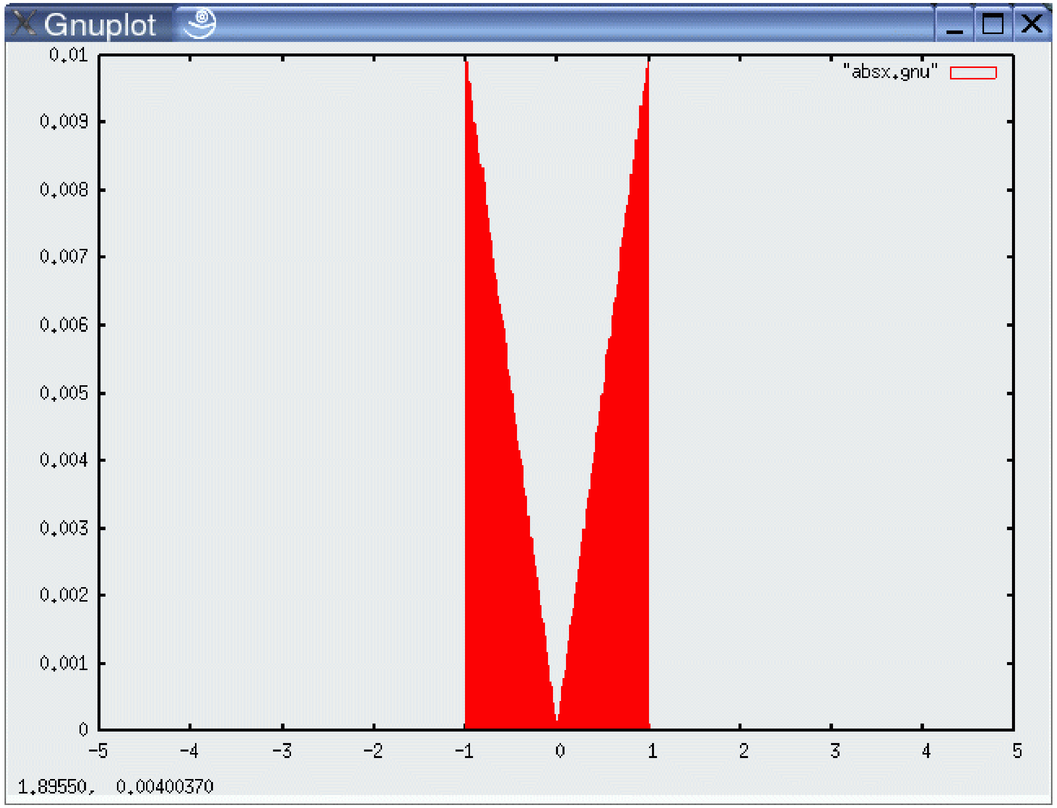
- Sampling from arbitrary distributions

1. Algorithm **sample_distribution**(f,b):
2. repeat
3. $x = \mathbf{rand}(-b, b)$
4. $y = \mathbf{rand}(0, \max\{f(x) \mid x \in (-b, b)\})$
5. until $(y \leq f(x))$
6. return x

Example

- Sampling from

$$f(x) = \begin{cases} \text{abs}(x) & x \in [-1; 1] \\ 0 & \text{otherwise} \end{cases}$$



Sample Odometry Motion Model

1. Algorithm **sample_motion_model**(u, x):

   $u = \langle \delta_{rot1}, \delta_{rot2}, \delta_{trans} \rangle, x = \langle x, y, \theta \rangle$

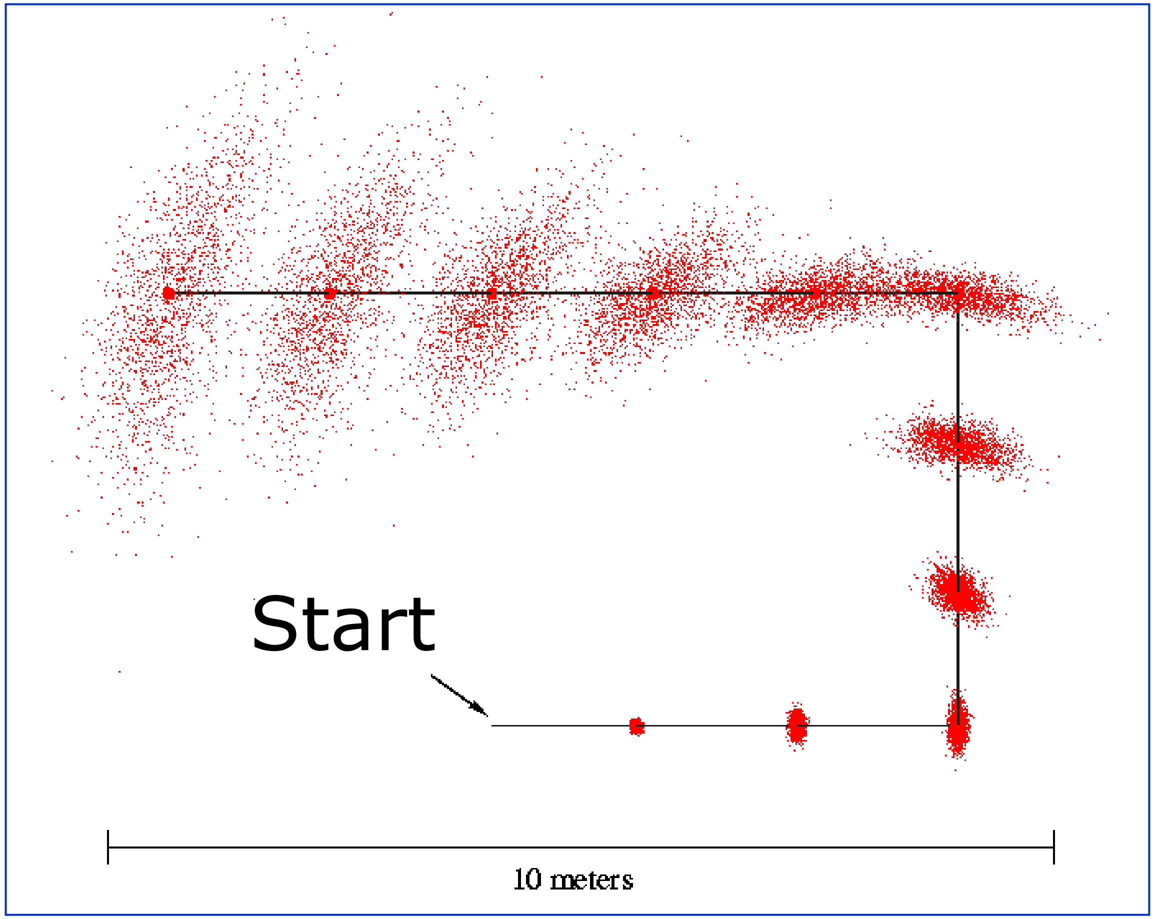
1. $\hat{\delta}_{rot1} = \delta_{rot1} + \text{sample}(\alpha_1 |\delta_{rot1}| + \alpha_2\ \delta_{trans})$
2. $\hat{\delta}_{trans} = \delta_{trans} + \text{sample}(\alpha_3\ \delta_{trans} + \alpha_4\ (|\delta_{rot1}| + |\delta_{rot2}|))$
3. $\hat{\delta}_{rot2} = \delta_{rot2} + \text{sample}(\alpha_1 |\delta_{rot2}| + \alpha_2\ \delta_{trans})$

4. $x' = x + \hat{\delta}_{trans} \cos(\theta + \hat{\delta}_{rot1})$
5. $y' = y + \hat{\delta}_{trans} \sin(\theta + \hat{\delta}_{rot1})$
6. $\theta' = \theta + \hat{\delta}_{rot1} + \hat{\delta}_{rot2}$

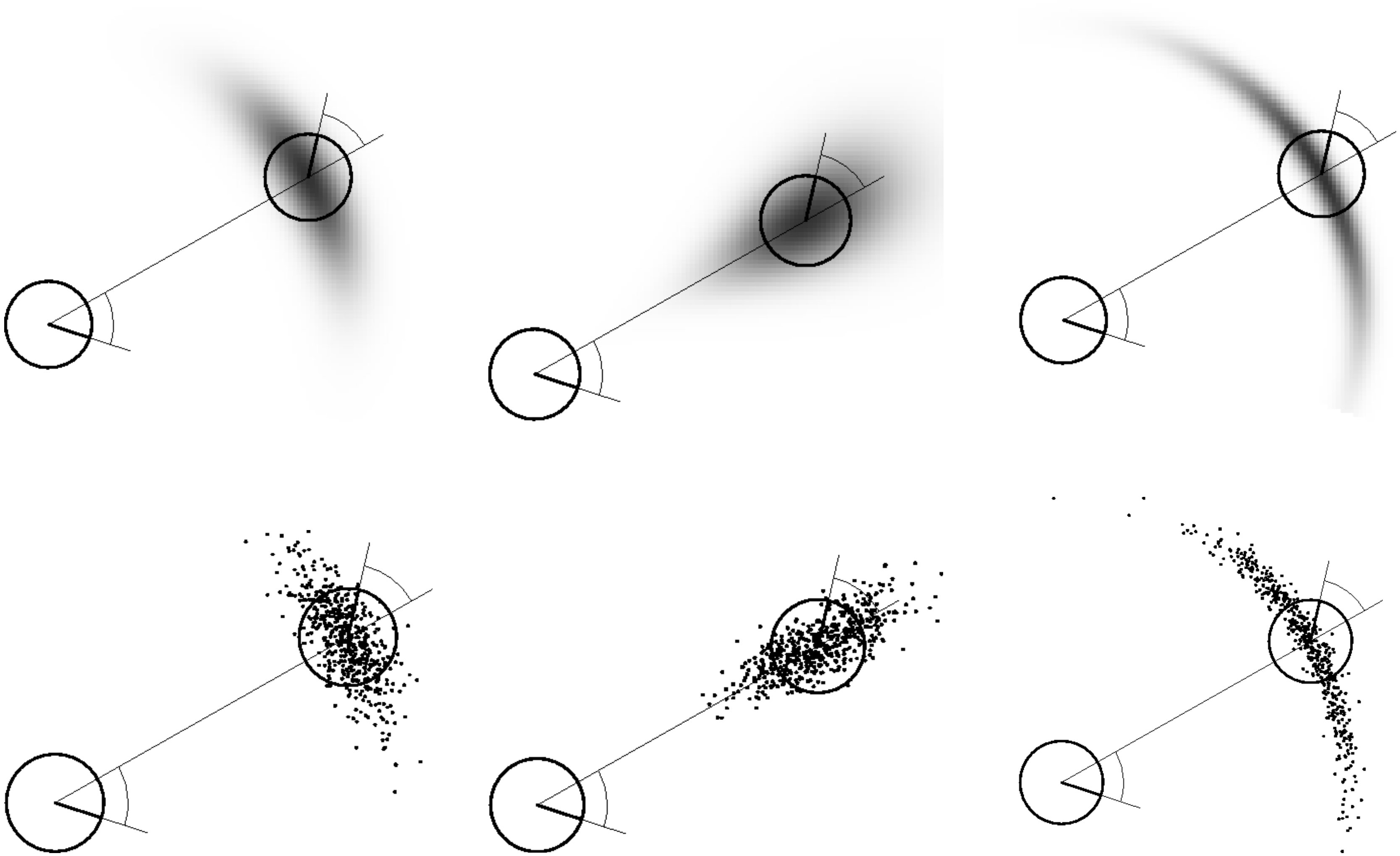
7. Return $\langle x', y', \theta' \rangle$

sample_normal_distribution

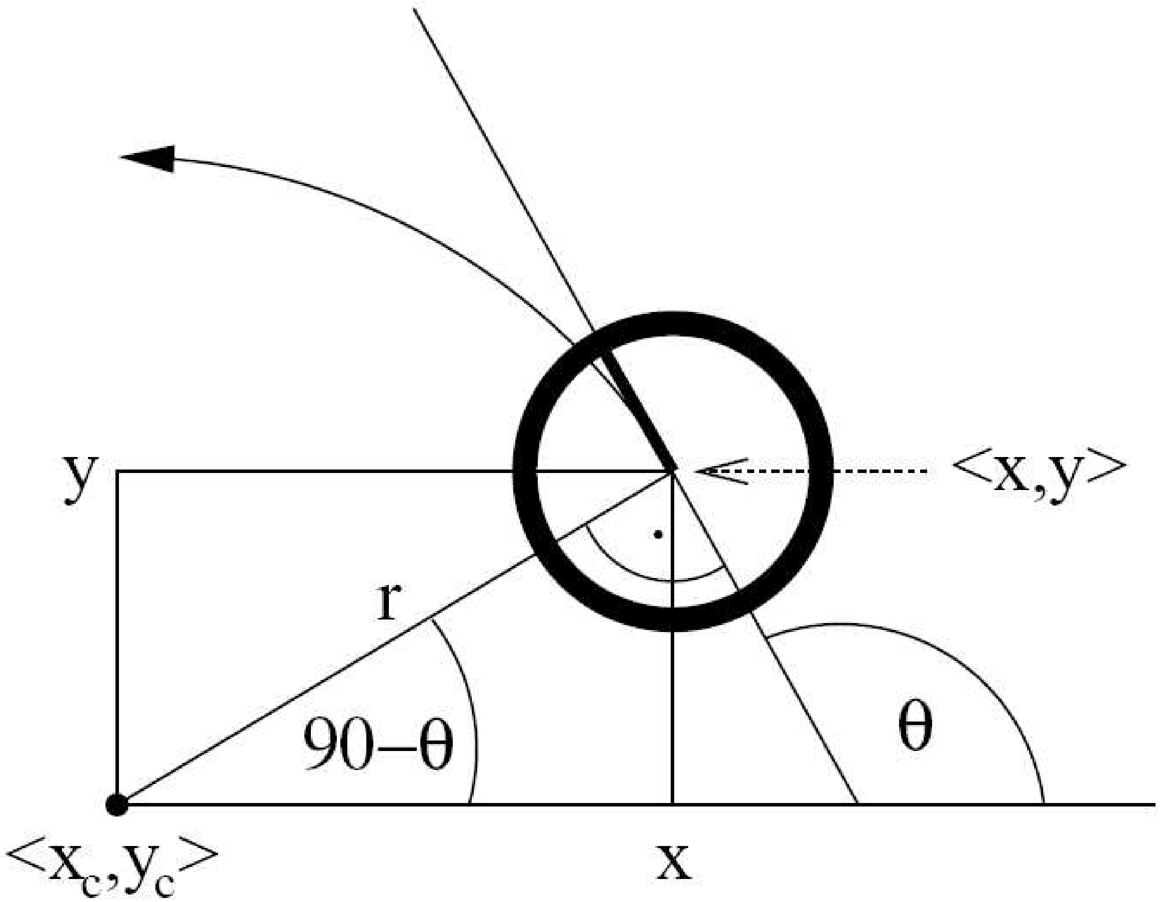
Sampling from Our Motion Model



Examples (Odometry-Based)



Velocity-Based Model



Equation for the Velocity Model

Center of circle:

$$\begin{pmatrix} x^* \\ y^* \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} -\lambda \sin\theta \\ \lambda \cos\theta \end{pmatrix} = \begin{pmatrix} \frac{x+x'}{2} + \mu(y-y') \\ \frac{y+y'}{2} + \mu(x'-x) \end{pmatrix}$$

with

$$\mu = \frac{1}{2} \frac{(x-x')\cos\theta + (y-y')\sin\theta}{(y-y')\cos\theta - (x-x')\sin\theta}$$

Posterior Probability for Velocity Model

Algorithm motion_model_velocity(x_t, u_t, x_{t-1})**:**

$$\mu = \frac{1}{2} \frac{(x - x')\cos\theta + (y - y')\sin\theta}{(y - y')\cos\theta - (x - x')\sin\theta}$$

$$x^* = \frac{x + x'}{2} + \mu(y - y')$$

$$y^* = \frac{y + y'}{2} + \mu(x' - x)$$

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

$$\Delta\theta = \operatorname{atan2}(y' - y^*, x' - x^*) - \operatorname{atan2}(y - y^*, x - x^*)$$

$$\hat{v} = \frac{\Delta\theta}{\Delta t} r^*$$

$$\hat{\omega} = \frac{\Delta\theta}{\Delta t}$$

$$\hat{\gamma} = \frac{\theta' - \theta}{\Delta t} - \hat{\omega}$$

return $\mathbf{prob}(v - \hat{v}, \alpha_1|v| + \alpha_2|\omega|) \cdot \mathbf{prob}(\omega - \hat{\omega}, \alpha_3|v| + \alpha_4|\omega|) \cdot \mathbf{prob}(\hat{\gamma}, \alpha_5|v| + \alpha_6|\omega|)$

Sampling from Velocity Model

Algorithm sample_motion_model_velocity(u_t, x_{t-1}):

$$\hat{v} = v + \textbf{sample}(\alpha_1|v| + \alpha_2|\omega|)$$

$$\hat{\omega} = \omega + \textbf{sample}(\alpha_3|v| + \alpha_4|\omega|)$$

$$\hat{\gamma} = \textbf{sample}(\alpha_5|v| + \alpha_6|\omega|)$$

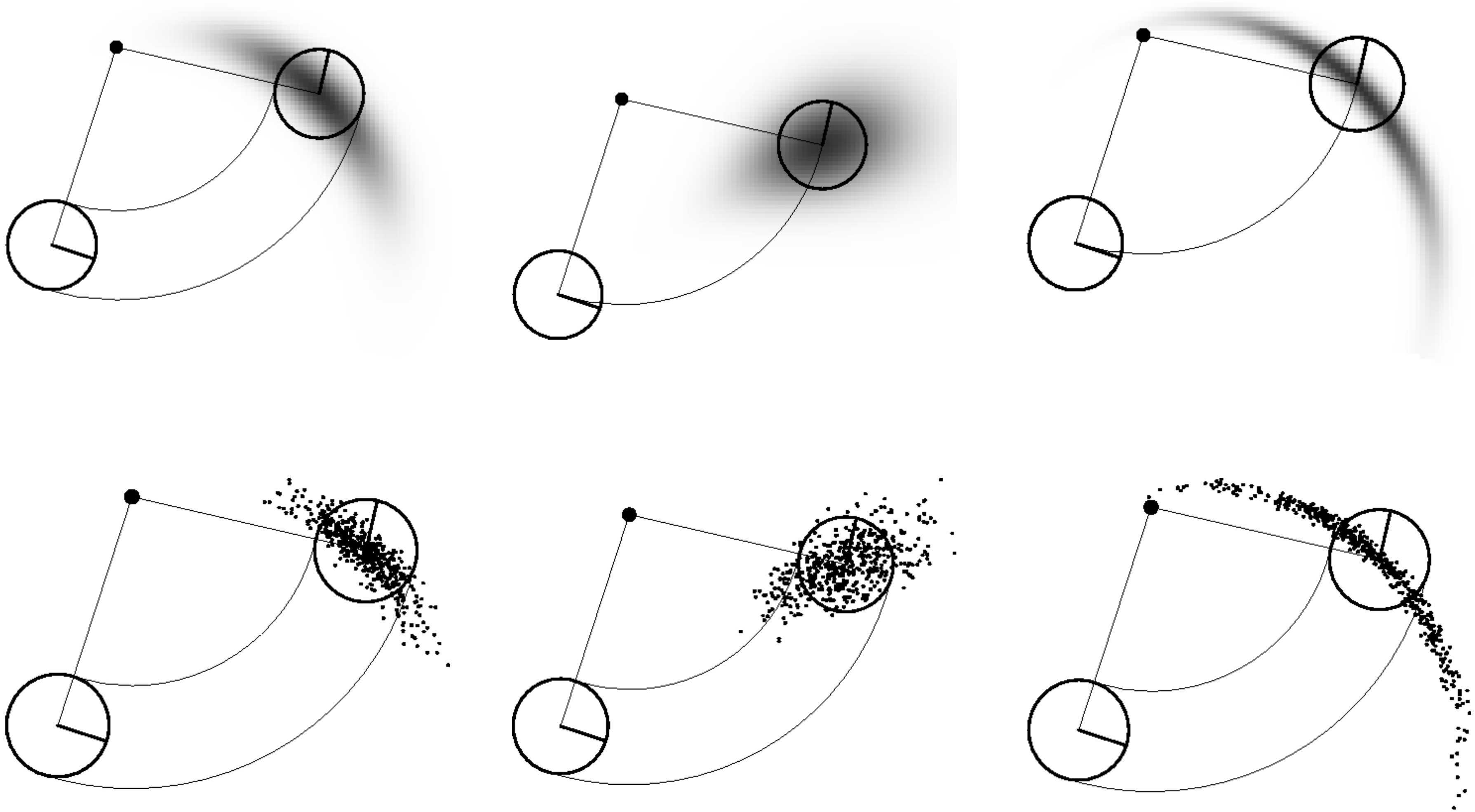
$$x' = x - \frac{\hat{v}}{\hat{\omega}} \sin\theta + \frac{\hat{v}}{\hat{\omega}}\, sin(\theta + \hat{\omega}\Delta t)$$

$$y' = y + \frac{\hat{v}}{\hat{\omega}} \cos\theta - \frac{\hat{v}}{\hat{\omega}}\, cos(\theta + \hat{\omega}\Delta t)$$

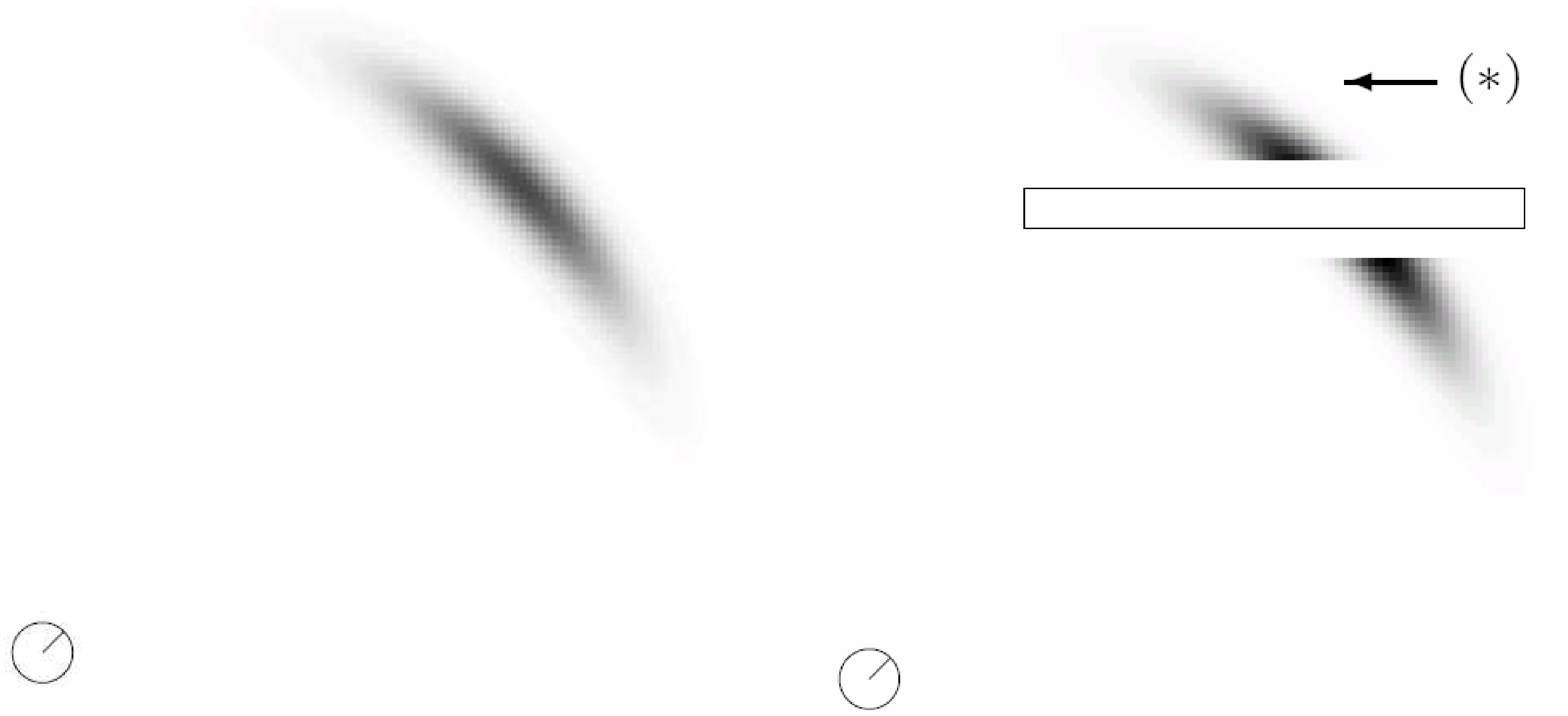
$$\theta' = \theta + \hat{\omega}\Delta t + \hat{\gamma}\Delta t$$

return $x_t = (x', y', \theta')^T$

Examples (velocity based)



Map-Consistent Motion Model



$$p(x \mid u, x') \quad \neq \quad p(x \mid u, x', m)$$

Approximation: $p(x \mid u, x', m) = \eta \; p(x \mid m) \; p(x \mid u, x')$

Summary

- We discussed motion models for odometry-based and velocity-based systems
- We discussed ways to calculate the posterior probability $p(x| x', u)$.
- We also described how to sample from $p(x| x', u)$.
- Typically the calculations are done fixed intervals Δt.
- In practice, the parameters of the models have to be learned.
- We also discussed an extended motion model that takes the map into account.