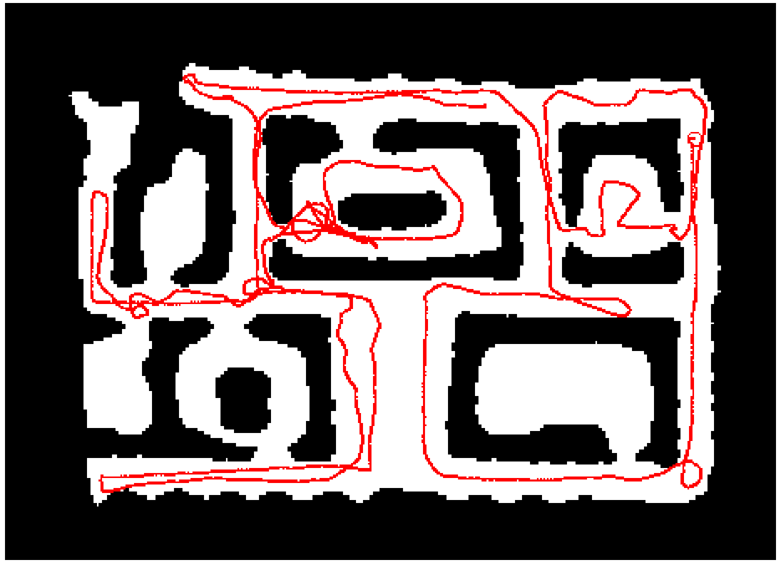
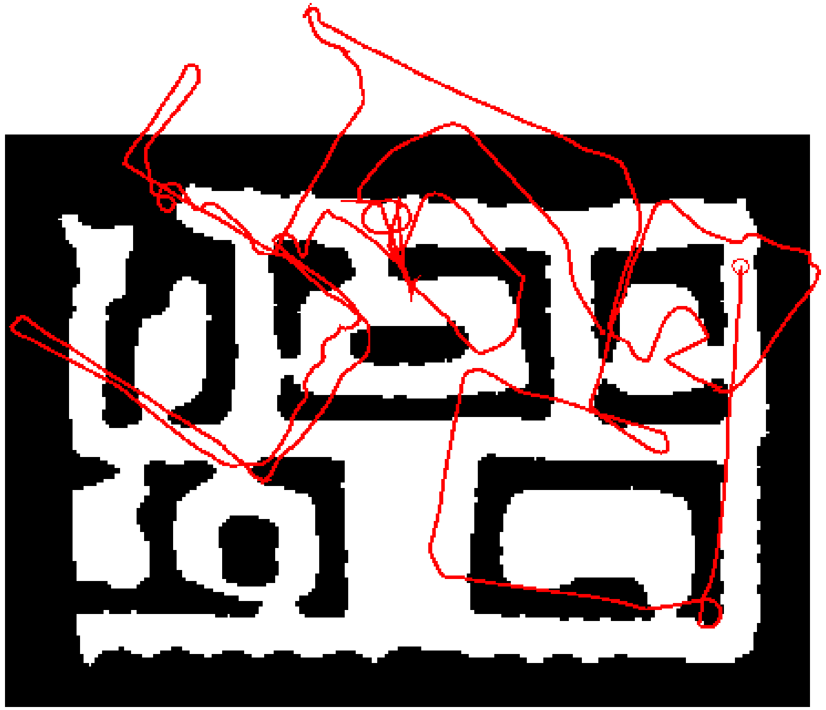
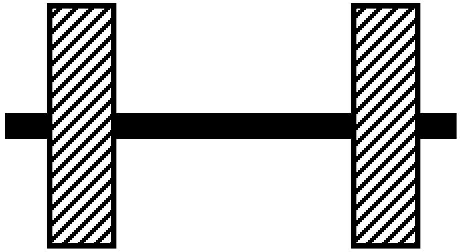
Robot Motion

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

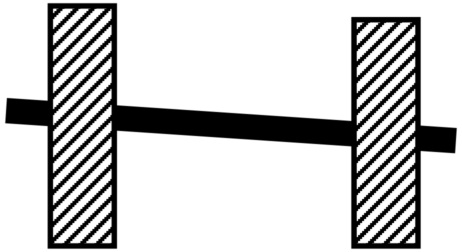




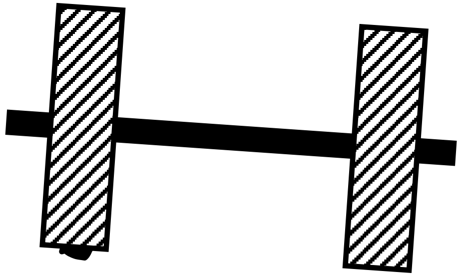
Reasons for Motion Errors of Wheeled Robots



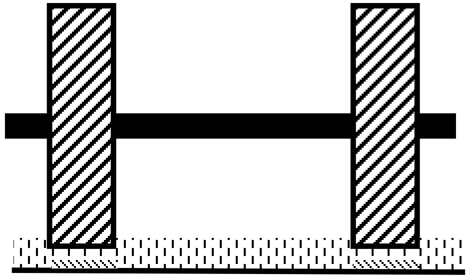
ideal case



different wheel diameters



bump



carpet

and many more …

Probabilistic Motion Models

- To implement the Bayes Filter, we need the transition model $p(x_t \mid x_{t-1}, u_t)$.

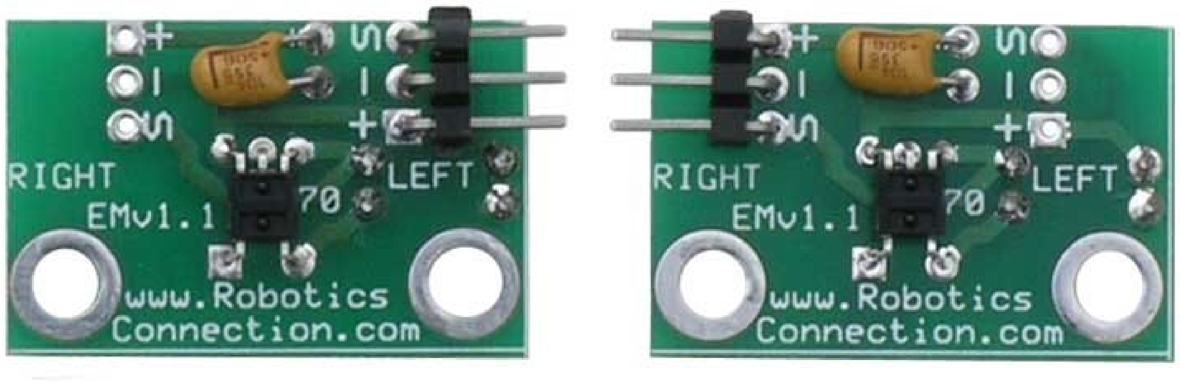
- The term $p(x_t \mid x_{t-1}, u_t)$ specifies a posterior probability, that action u_t carries the robot from x_{t-1} to x_t.

- In this section we will discuss, how $p(x_t \mid x_{t-1}, u_t)$ can be modeled based on the motion equations and the uncertain outcome of the movements.

- We will consider odometry-based models that are used when systems are equipped with wheel encoders.

Example Wheel Encoders

These modules provide +5V output when they "see" white, and a 0V output when they "see" black.

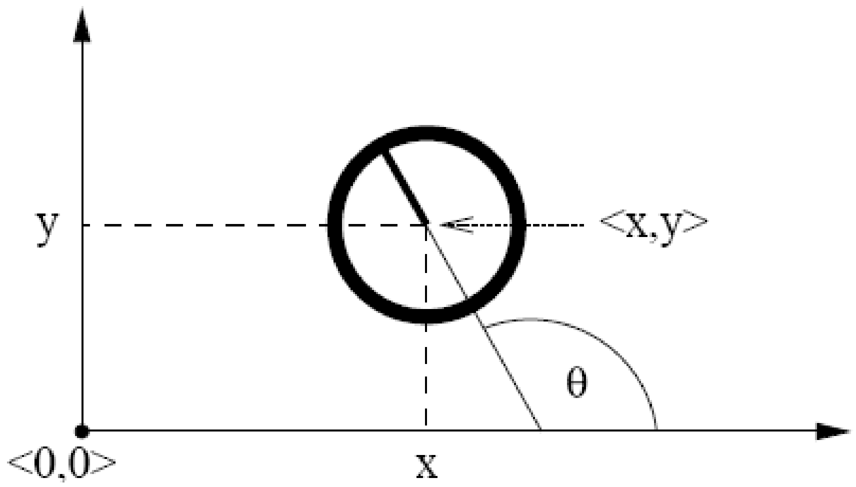


These disks are manufactured out of high quality laminated color plastic to offer a very crisp black to white transition. This enables a wheel encoder sensor to easily see the transitions.

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

Coordinate Systems

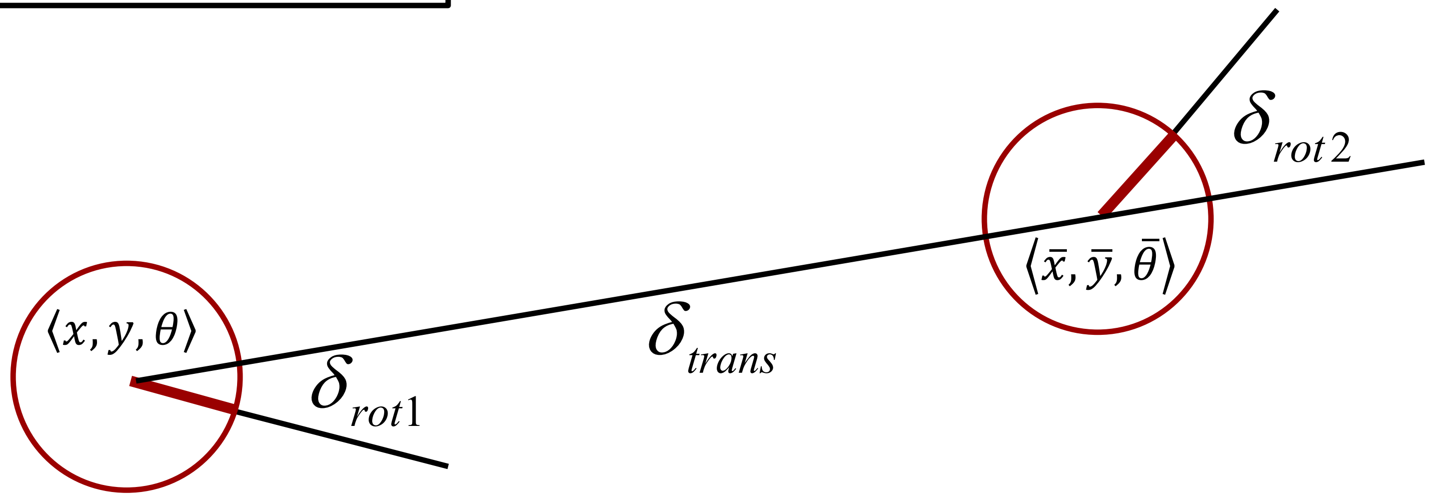
- The configuration of a typical wheeled robot in 3D can be described by six parameters.
- This are the three-dimensional Cartesian coordinates plus the three Euler angles for roll, pitch, and yaw.
- For simplicity, throughout this section we consider robots operating on a planar surface.
- The state space of such systems is three-dimensional (x,y,θ).



Odometry Model

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

$$\boxed{\begin{aligned}\delta_{trans} &= \sqrt{(\bar{x} - x)^2 + (\bar{y} - y)^2} \\ \delta_{rot1} &= \operatorname{atan2}(\bar{y} - y, \bar{x} - x) - \theta \\ \delta_{rot2} &= \bar{\theta} - \theta - \delta_{rot1}\end{aligned}}$$



Example Noise Model for Odometry

- The measured motion is given by the true motion corrupted with noise.
- The example noise functions $\boldsymbol{\varepsilon}$ depend on $\delta_{rot1}, \delta_{rot2}$, and δ_{trans}
- The parameters $\alpha_1, \dots, \alpha_4$ need to be estimated from data for every individual robot

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

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

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

Calculating the Probability Density of a (zero-centered) Normal Distribution

query point

1. Algorithm **prob_normal_distribution**(a,b):

std. deviation

2. return $\dfrac{1}{\sqrt{2\pi\, b^2}} \exp\left\{ -\dfrac{1}{2}\dfrac{a^2}{b^2} \right\}$

Calculating the Posterior $p(x' \mid x, u)$

Algorithm **motion_model_odometry**$(\boldsymbol{x'}, \boldsymbol{x}, \boldsymbol{u})$

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

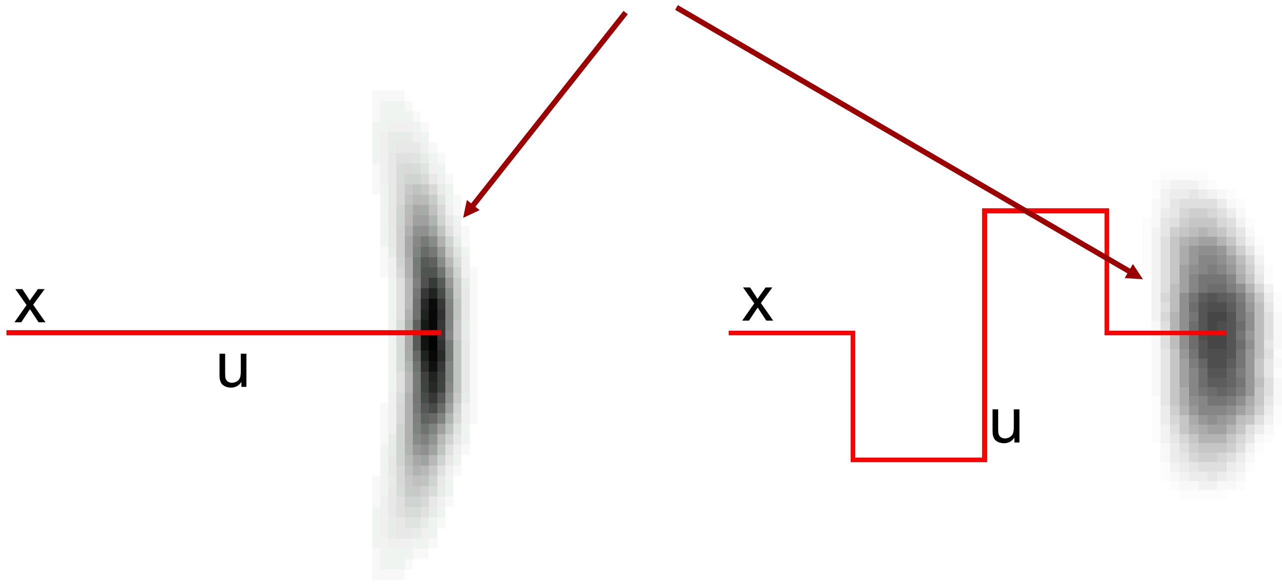
1. $\hat{\delta}_{trans} = \sqrt{(x' - x)^2 + (y' - y)^2}$
2. $\hat{\delta}_{rot1} = \mathrm{atan2}(y' - y, x' - x) - \theta$
3. $\hat{\delta}_{rot2} = \theta' - \theta - \hat{\delta}_{rot1}$
4. $p_1 = \mathrm{prob}(\delta_{rot1} - \hat{\delta}_{rot1}\,,\ \alpha_1 |\delta_{rot1}| + \alpha_2 \delta_{trans})$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

Here, prob() is **prob_normal_distribution()**

Application

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

$$p(\boldsymbol{x}' \mid \boldsymbol{u}, \boldsymbol{x})$$



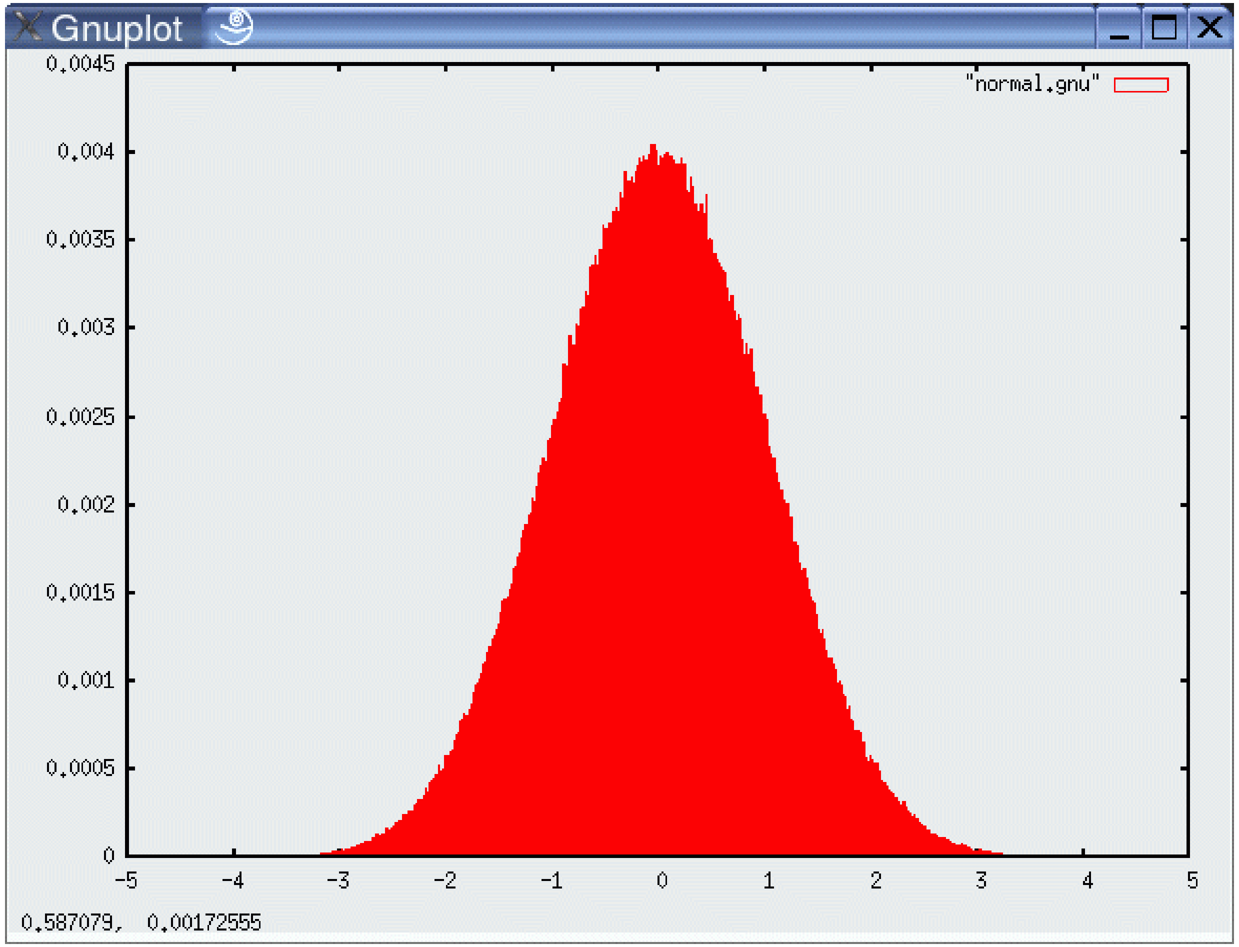
Sampling

- Certain implementations of the Bayes filter require to sample from the motion model
- Sampling means generating random numbers that follow a specific distribution
- We model the noise functions $\boldsymbol{\varepsilon}$ as normal distributions with zero mean
- One algorithm to sample from the normal distribution is:

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

uniformly distributed random number in $[-b, b]$

Normally Distributed Samples



10^6 samples

Sample Odometry Motion Model

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

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

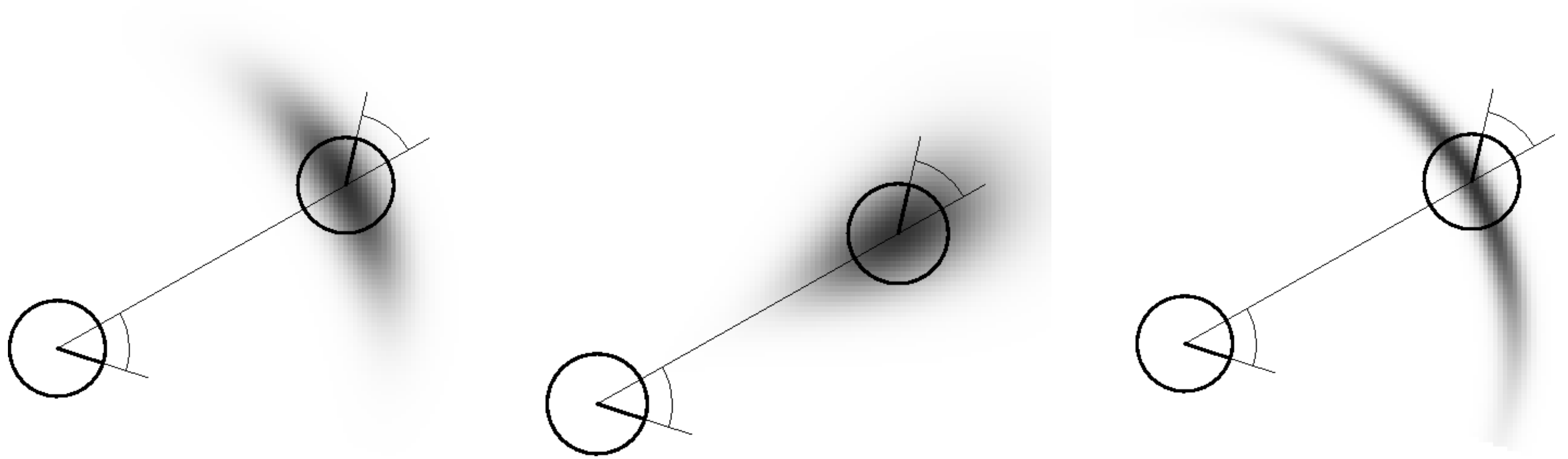
1. $\hat{\delta}_{rot1} = \delta_{rot1} + \text{sample}(\alpha_1 \mid \delta_{rot1} \mid + \alpha_2\ \delta_{trans})$
2. $\hat{\delta}_{trans} = \delta_{trans} + \text{sample}(\alpha_3\ \delta_{trans} + \alpha_4\ (\mid \delta_{rot1} \mid + \mid \delta_{rot2} \mid))$
3. $\hat{\delta}_{rot2} = \delta_{rot2}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

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

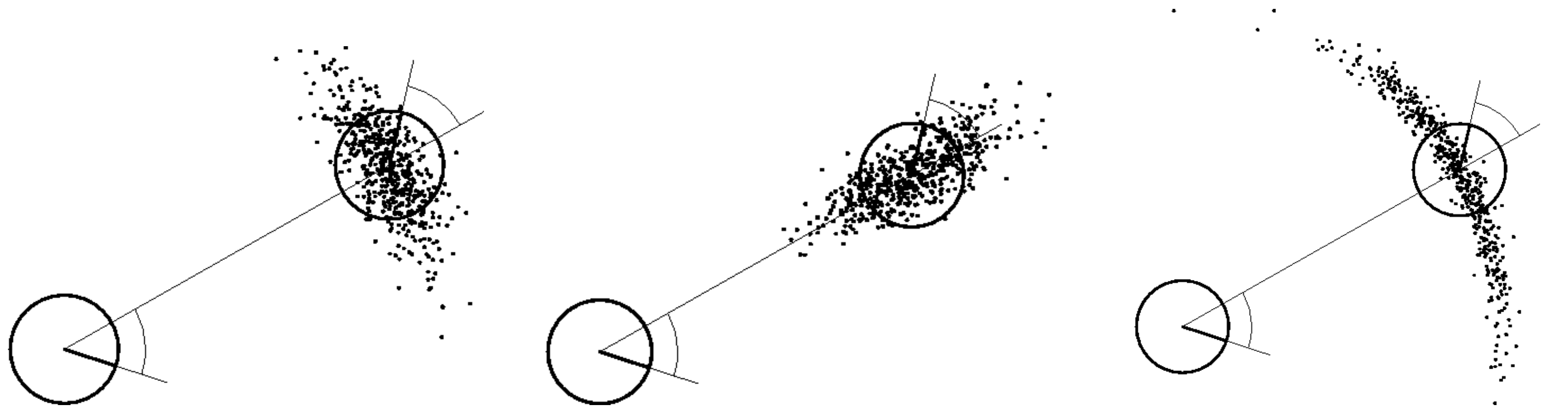
7. return $\langle x', y', \theta' \rangle$

Here, sample() is **sample_normal_distribution()**

Examples (Odometry-Based)



motion model distribution



samples generated from the motion model