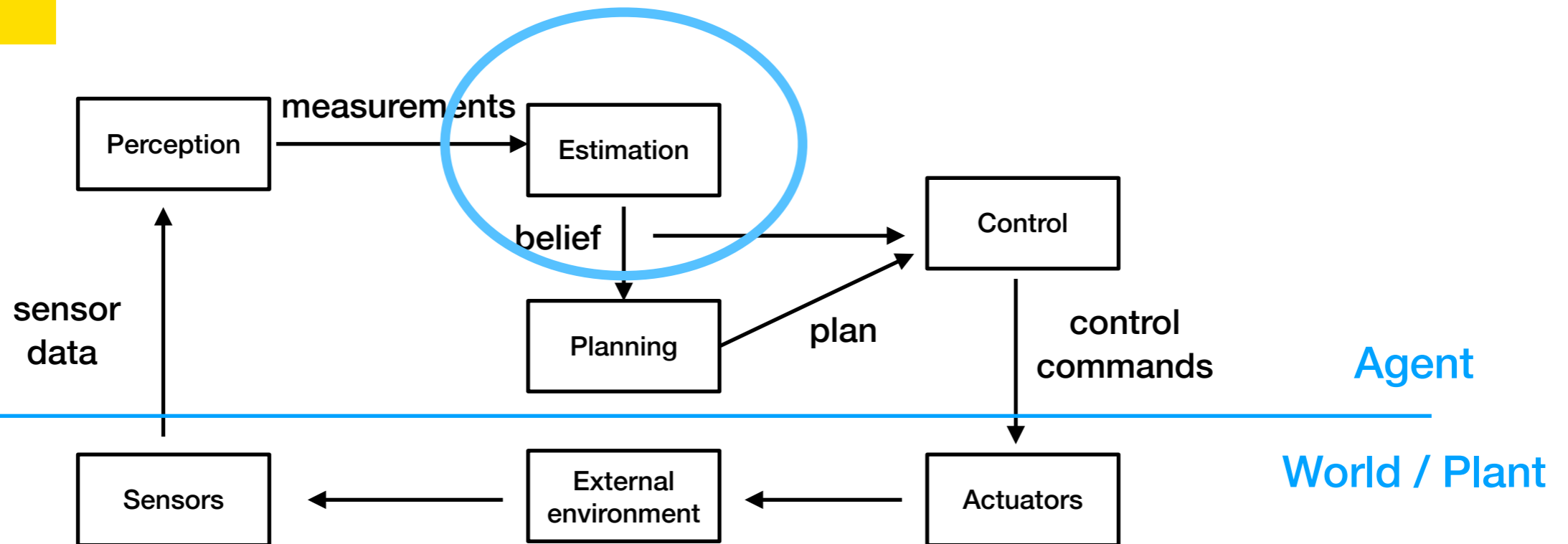


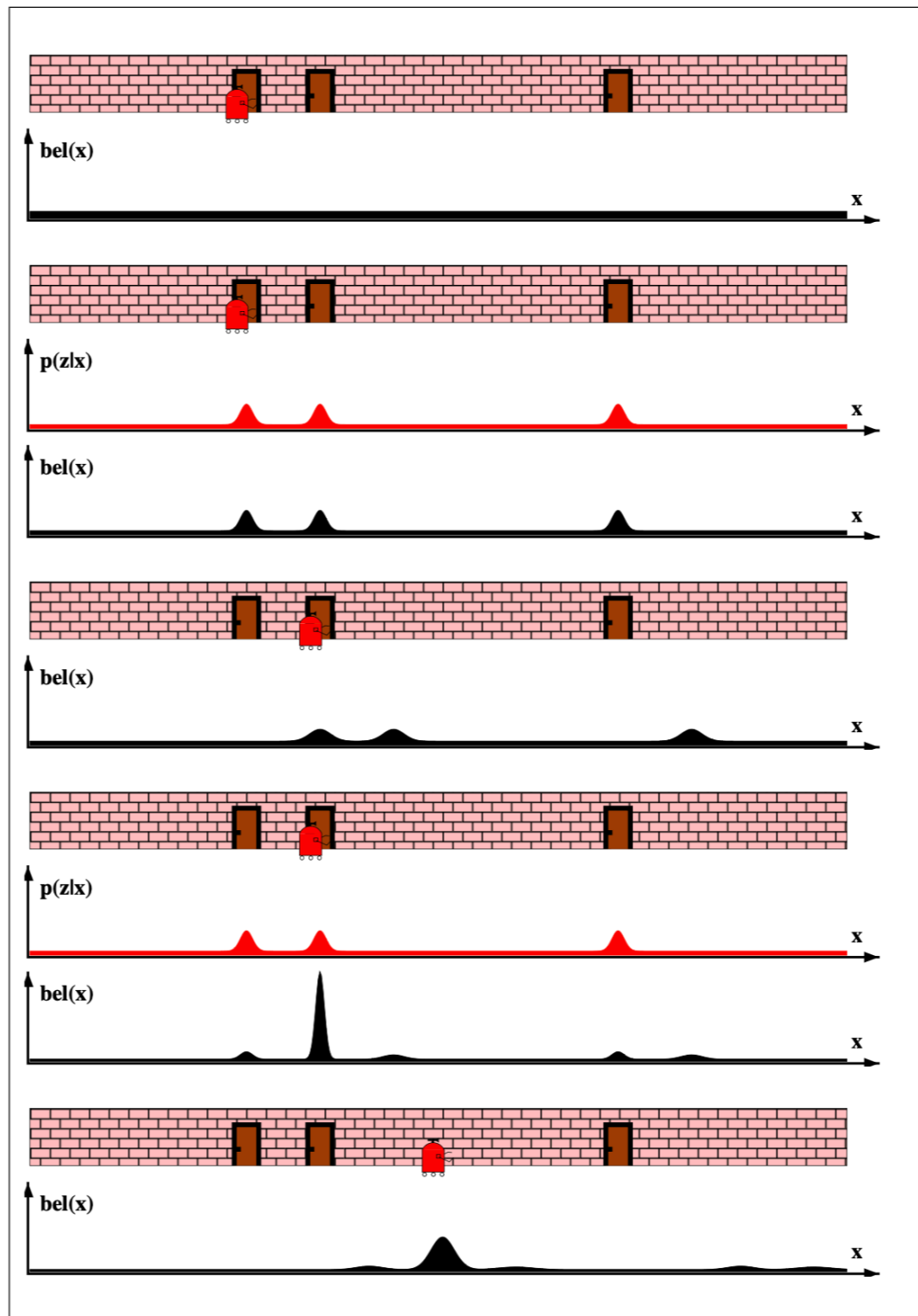


# Filtering

# Big picture

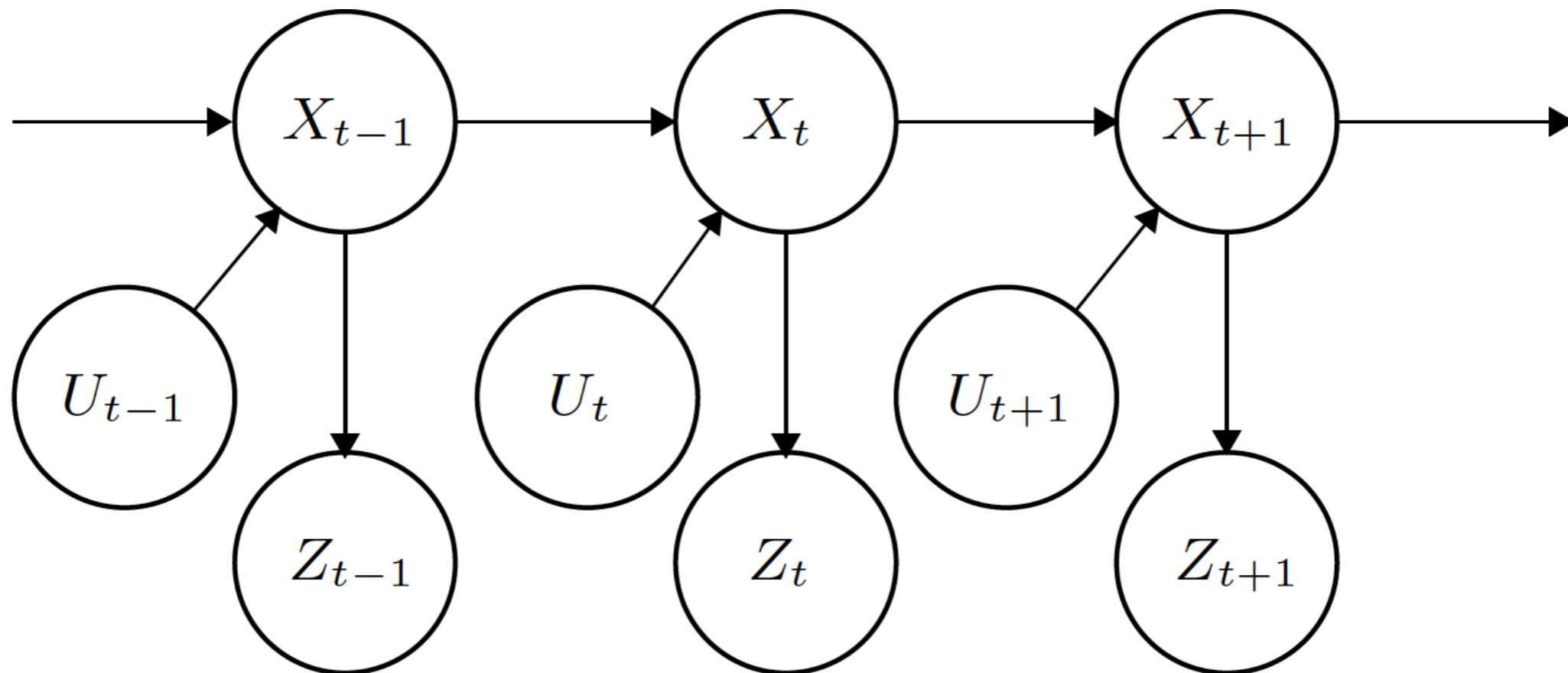


# Exemple



# The Bayes' Net

Process:  $p(x_t | x_{t-1}, u_t)$

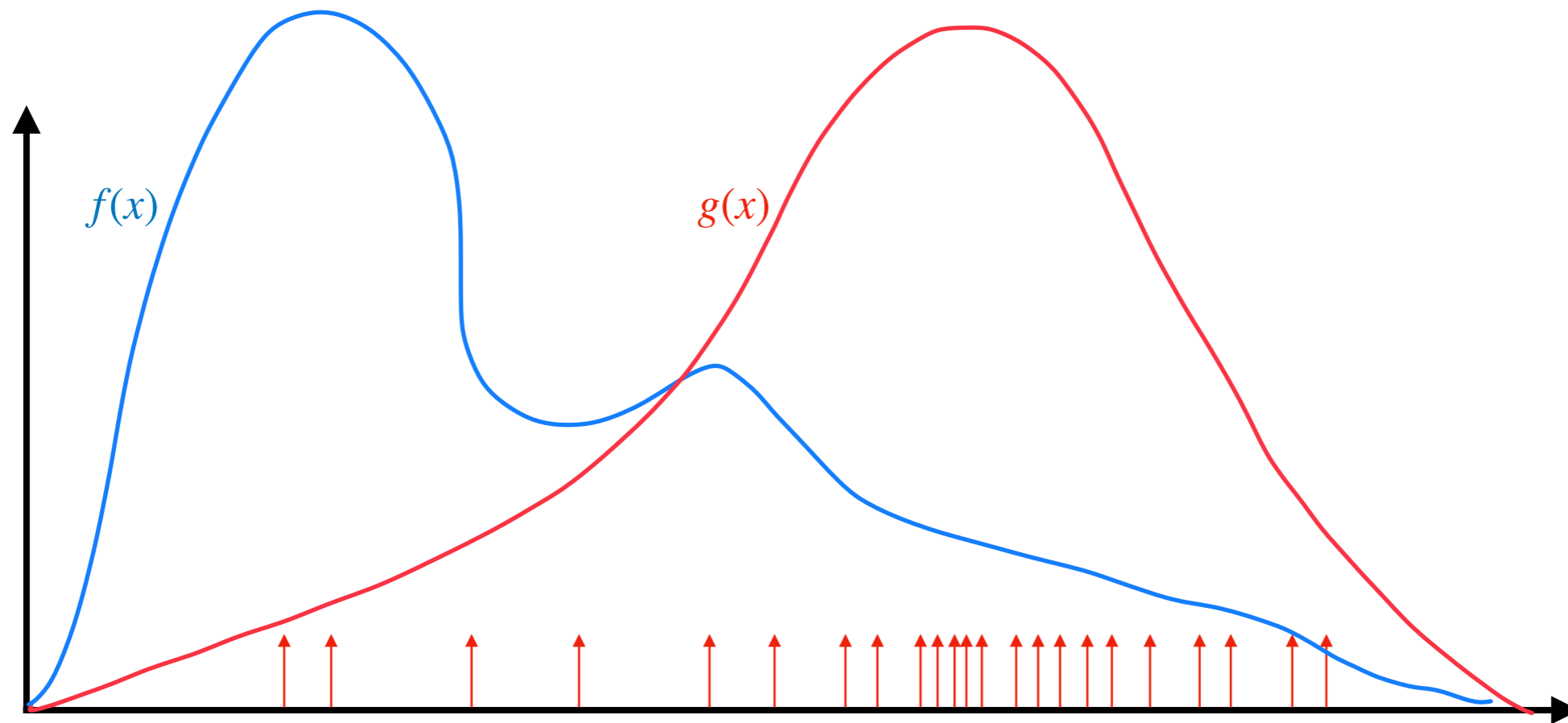


Measurement:  $p(z_t | x_t)$

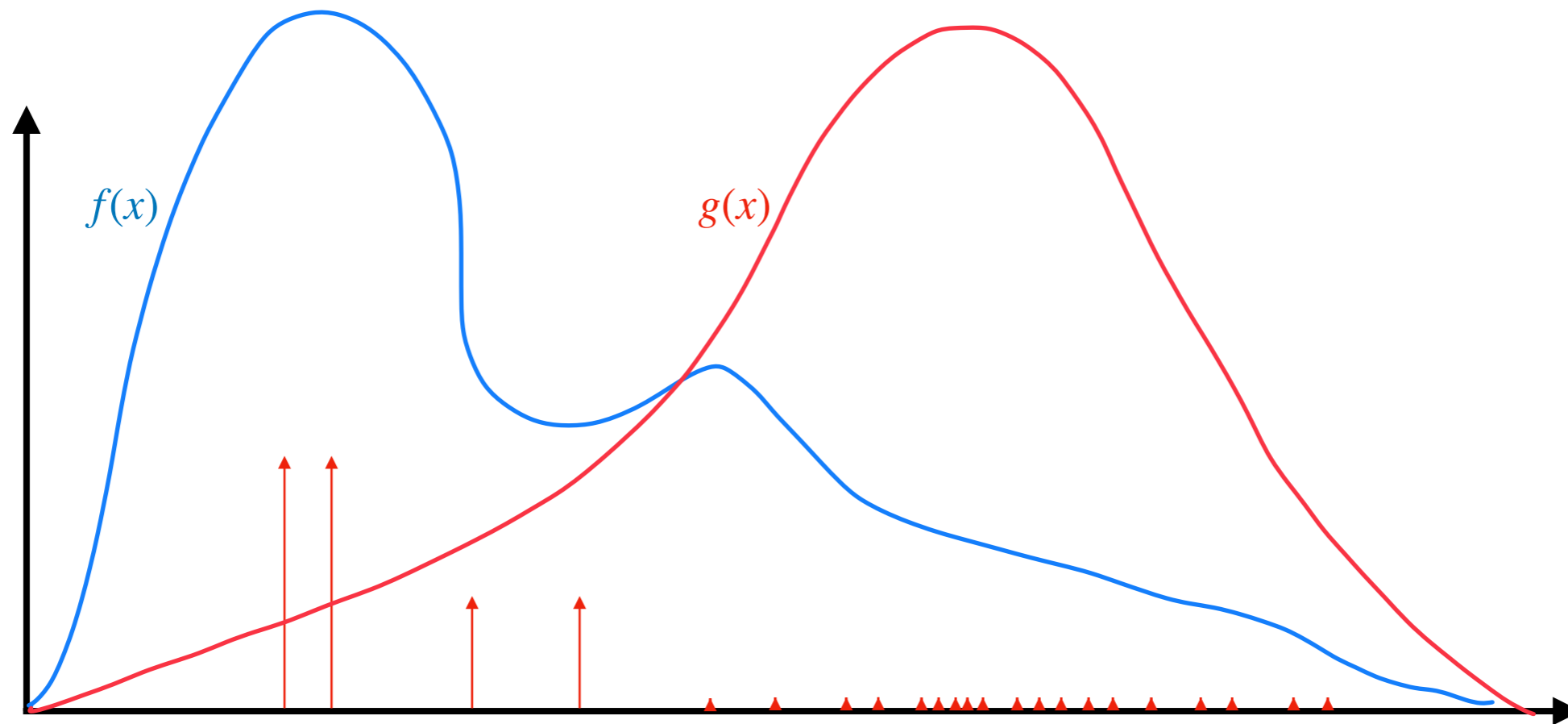
# Extended Kalman filter equations

	Kalman	Extended Kalman
Predict	$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$ $\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t$	$\bar{\mu}_t = f(\mu_{t-1}, u_t, 0)$ $\bar{\Sigma}_t = F_t \Sigma_{t-1} F_t^T + W_t Q_t W_t^T$
Update	$\mu_t = \bar{\mu}_t + K_t (z_t - H_t \bar{\mu}_t)$ $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$ $K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + R_t)^{-1}$	$\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t, 0))$ $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$ $K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + V_t R_t V_t^T)^{-1}$

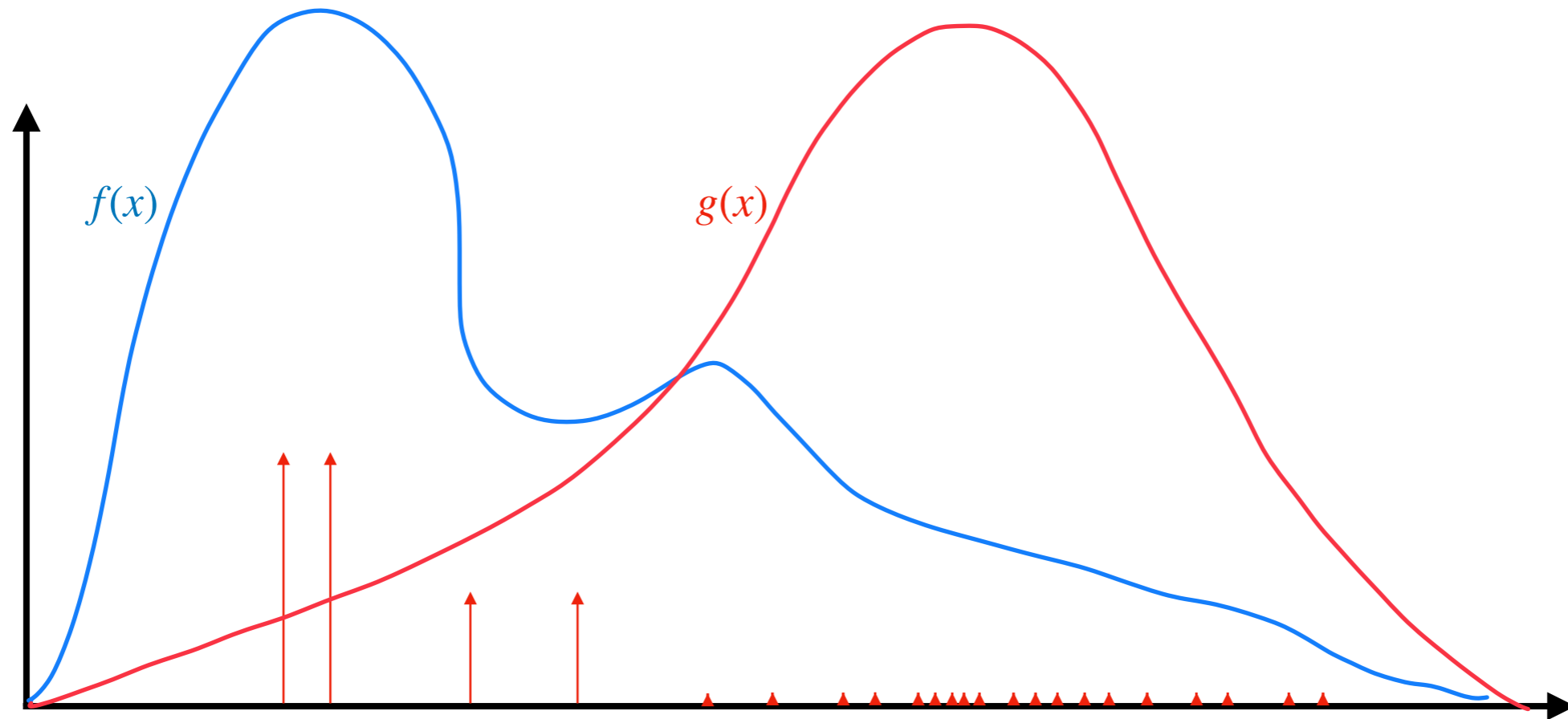
# Importance sampling



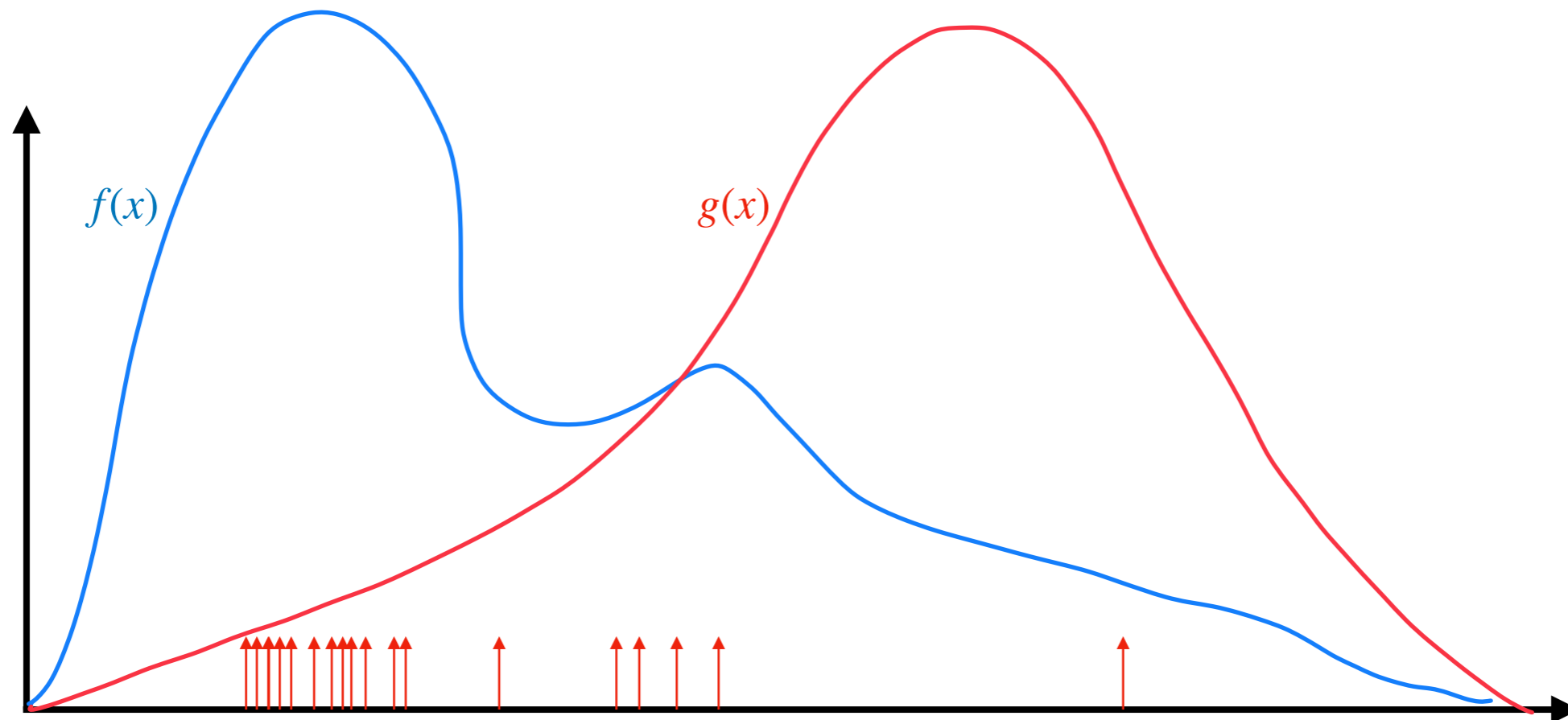
# Importance Sampling



# Sampling importance resampling

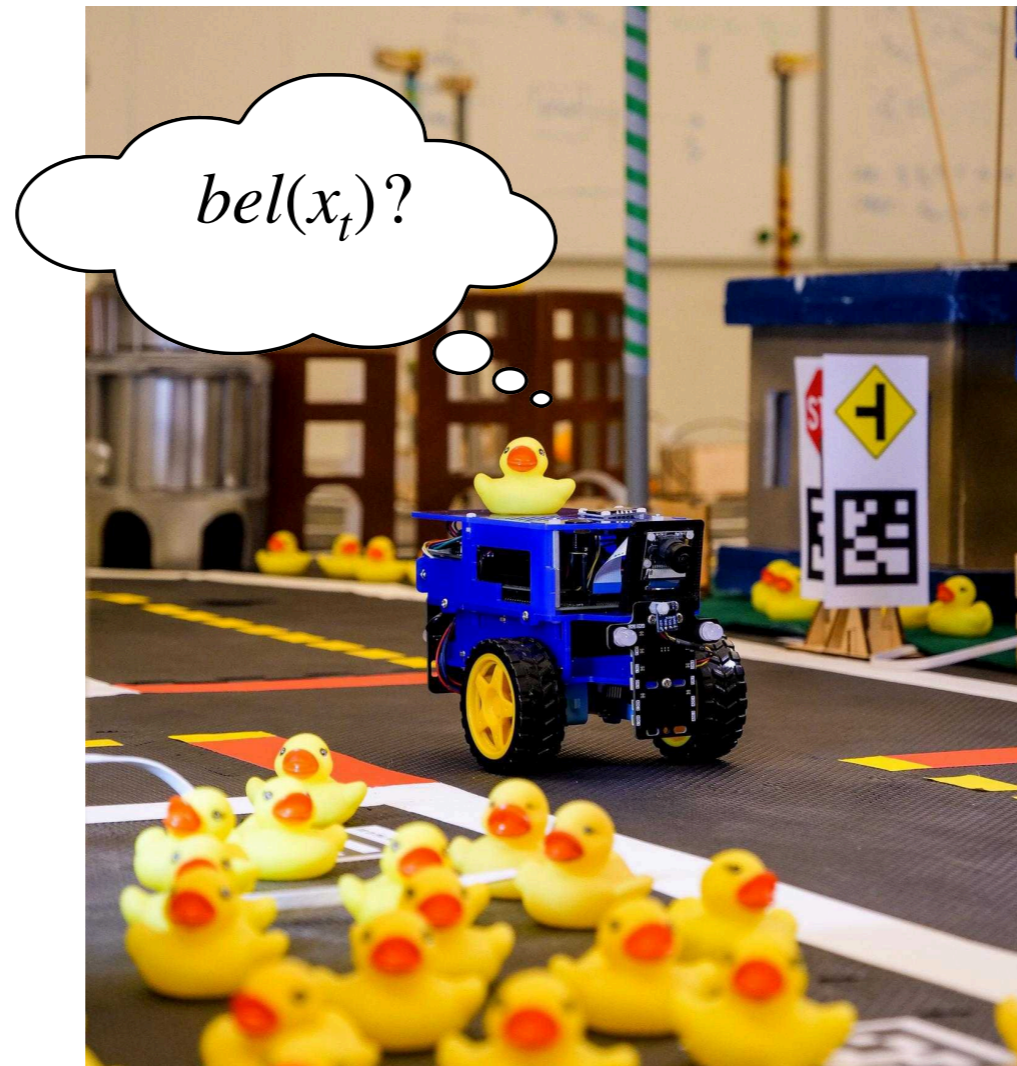


# Sampling importance resampling



Avoids “**particle depletion**”

# Lane Pose Estimation

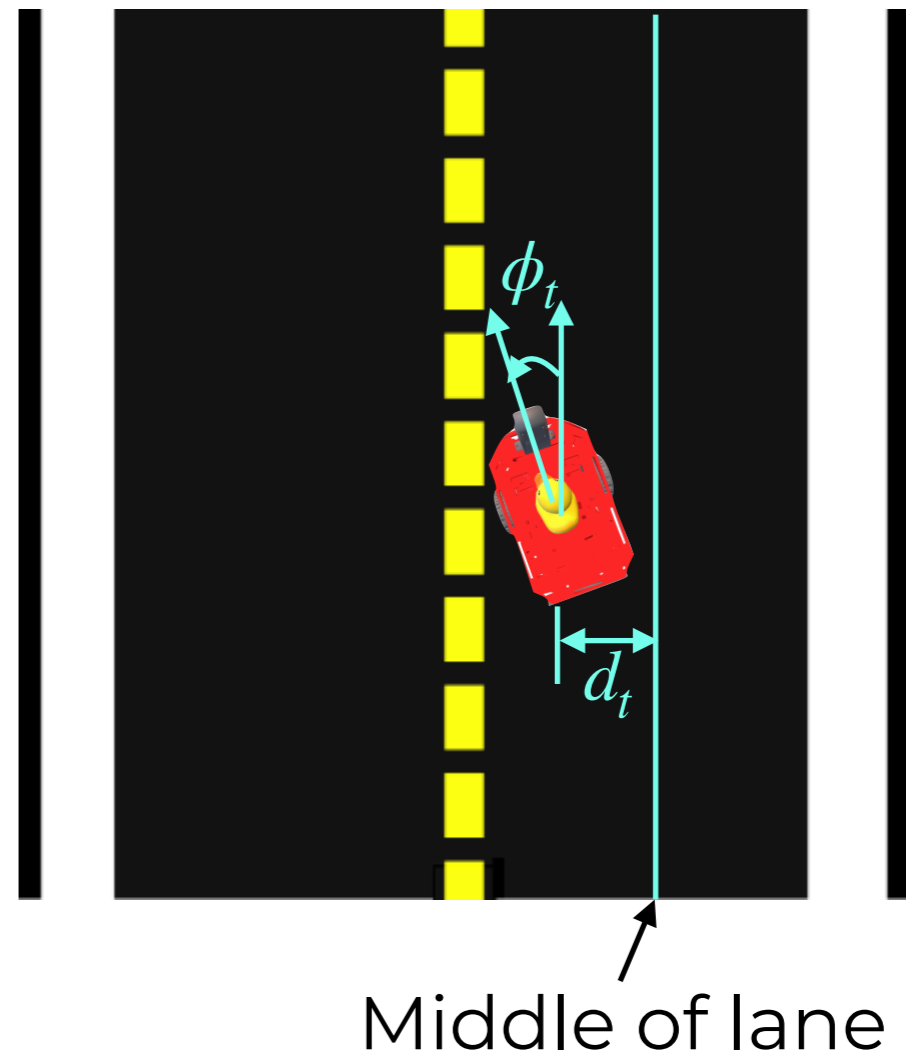


# Lane-relative estimation

State:  $x_t \triangleq [d_t, \phi_t]$

$d_t$ : Duckiebot distance to lane center

$\phi_t$ : Angle of Duckiebot relative to lane



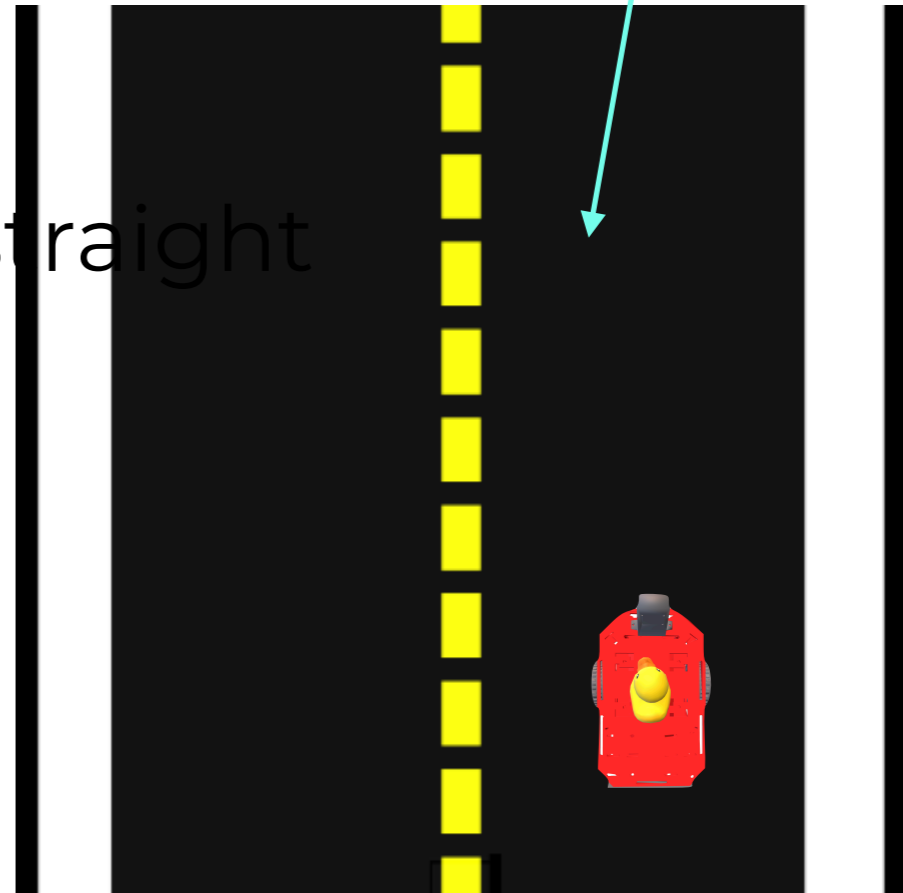
# Predict

$$bel \left( \begin{bmatrix} d_0 \\ \phi_0 \end{bmatrix} \right) = \mathcal{N} \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_{d_0}^2, 0 \\ 0, \sigma_{\phi_0}^2 \end{bmatrix} \right)$$

Robot is commanded to drive straight

$$bel \left( \begin{bmatrix} d_1 \\ \phi_1 \end{bmatrix} \right) = \mathcal{N} \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \bar{\Sigma}_1 \right)$$

But there is some drift

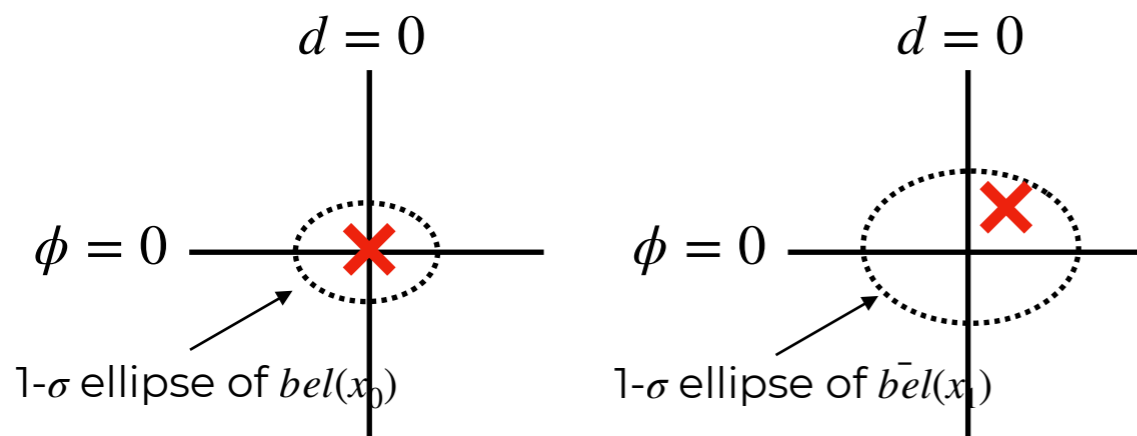


# Predict

$$bel\left(\begin{bmatrix} d_0 \\ \phi_0 \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_{d_0}^2, 0 \\ 0, \sigma_{\phi_0}^2 \end{bmatrix}\right)$$

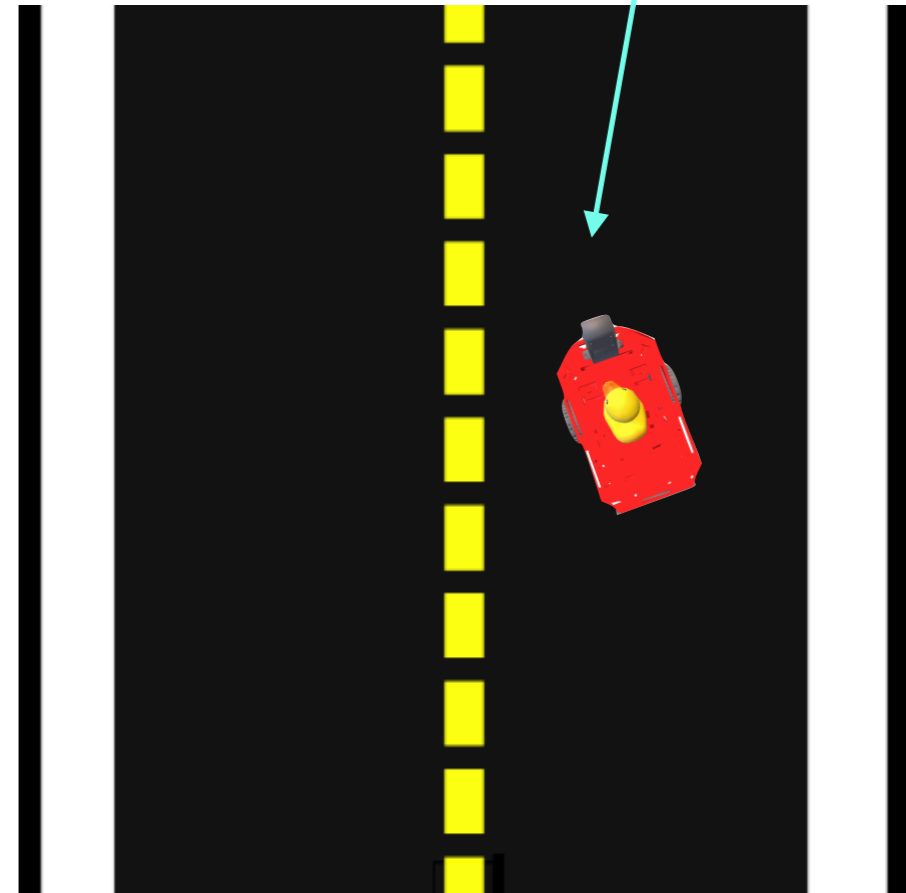
Robot is commanded to drive straight

$$\bar{bel}\left(\begin{bmatrix} d_1 \\ \phi_1 \end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \bar{\Sigma}_1\right)$$



**X** : True state of Duckiebot

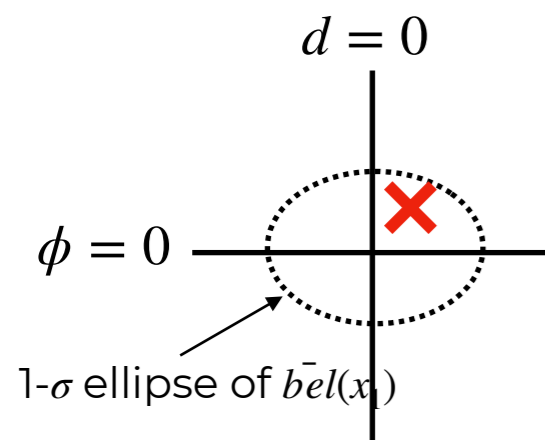
But there is some drift



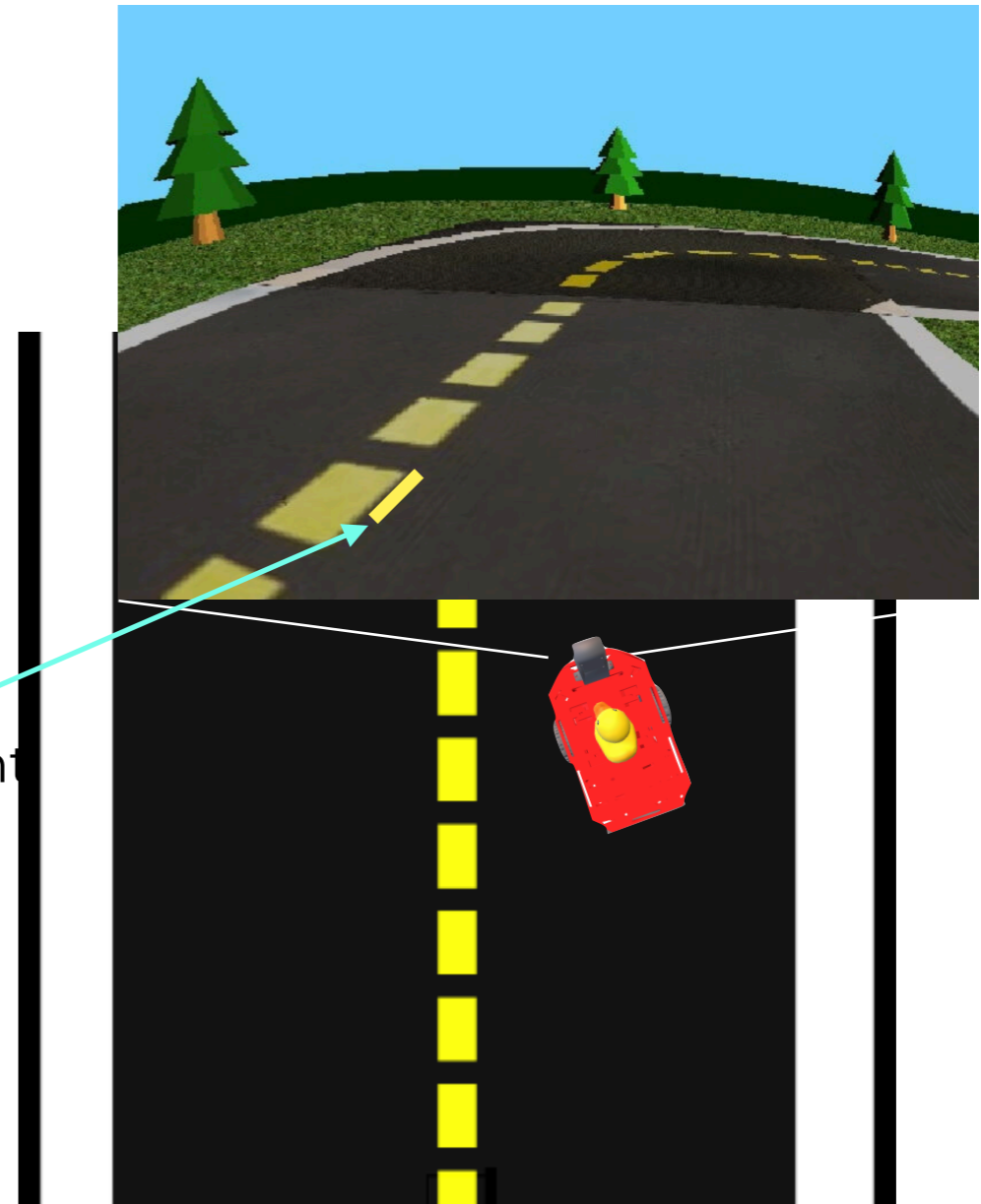
# Update

$$\bar{bel} \left( \begin{bmatrix} d_0 \\ \phi_0 \end{bmatrix} \right) = \mathcal{N} \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \bar{\sigma}_{d_0}^2, 0 \\ 0, \bar{\sigma}_{\phi_0}^2 \end{bmatrix} \right)$$

An image is received



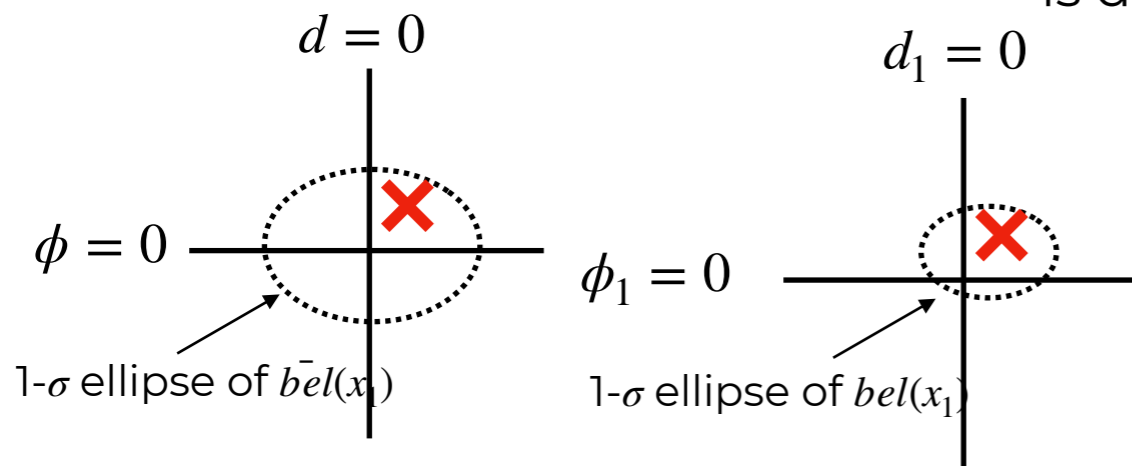
**X** : True state of Duckiebot



# Update

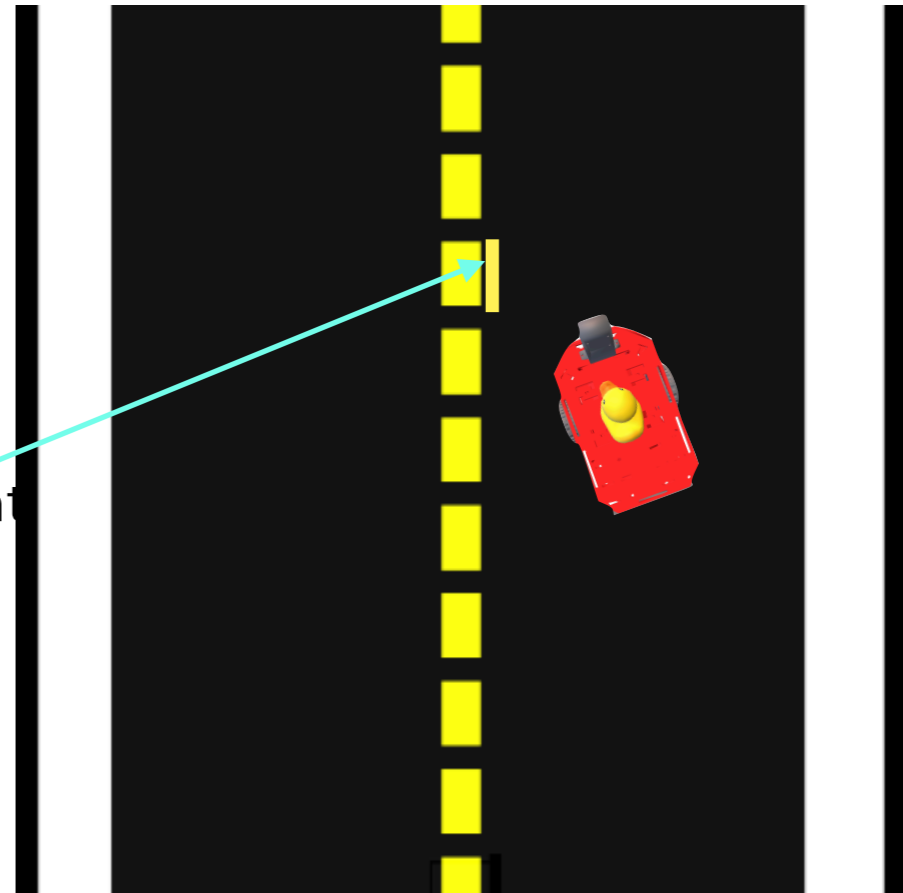
$$\bar{bel} \left( \begin{bmatrix} d_1 \\ \phi_1 \end{bmatrix} \right) = \mathcal{N} \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \bar{\Sigma}_1 \right)$$

$$bel \left( \begin{bmatrix} d_1 \\ \phi_1 \end{bmatrix} \right) = \mathcal{N} (\mu_1, \Sigma_1)$$



**X** : True state of Duckiebot

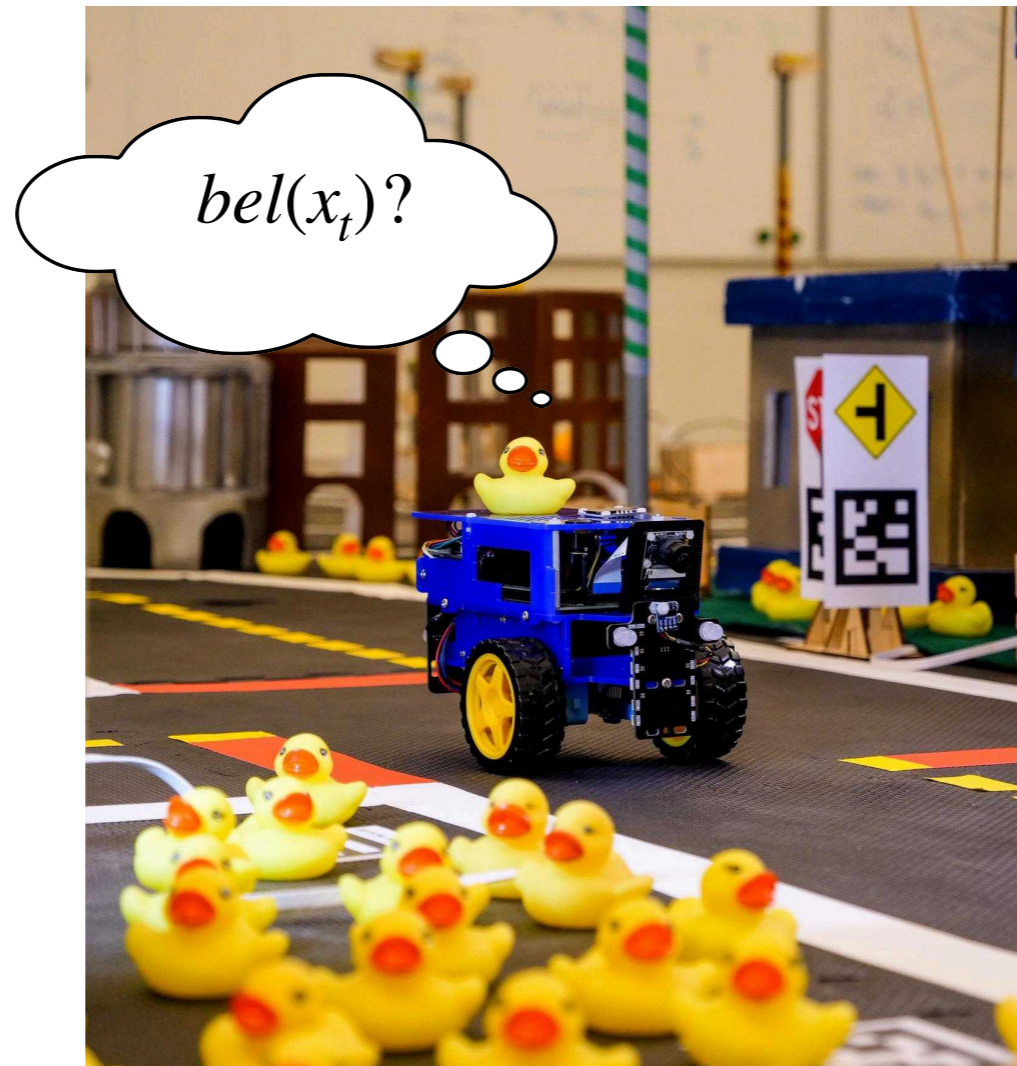
A line segment  
is detected



Outliers?



# Lane Pose Estimation

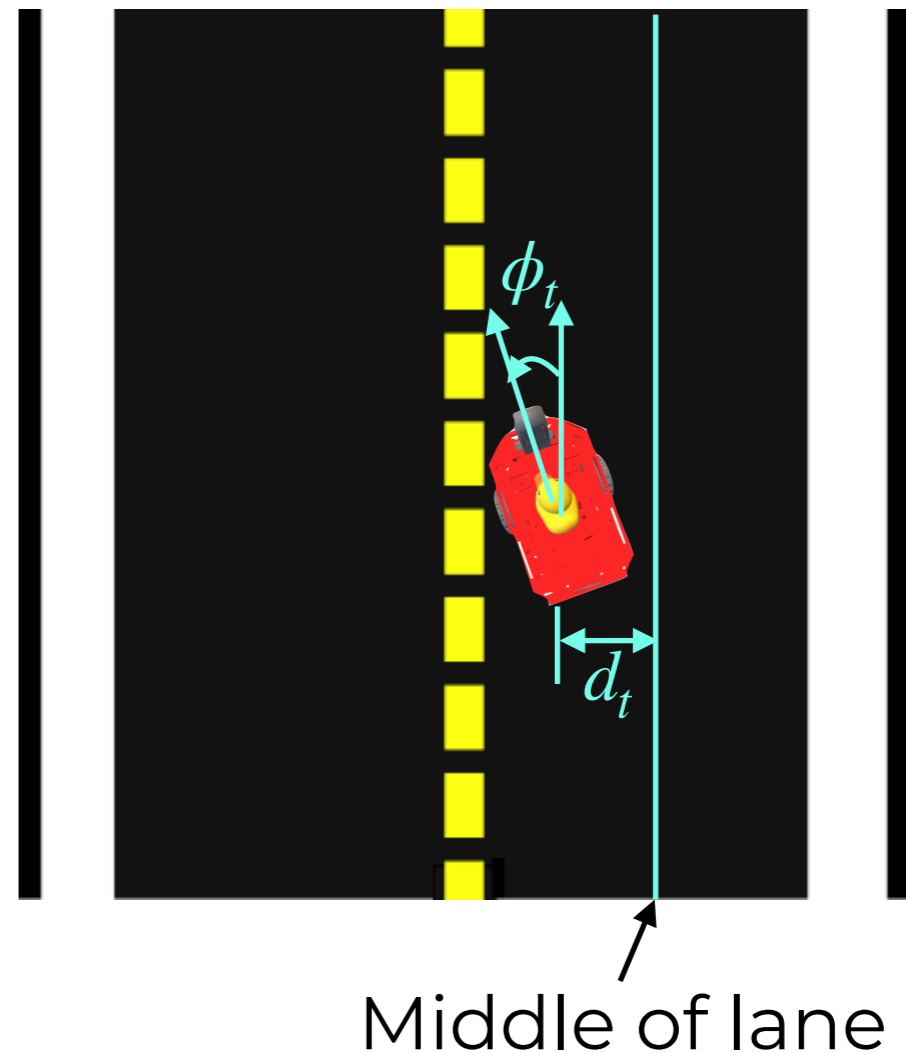


# Lane-relative estimation

State:  $x_t \triangleq [d_t, \phi_t]$

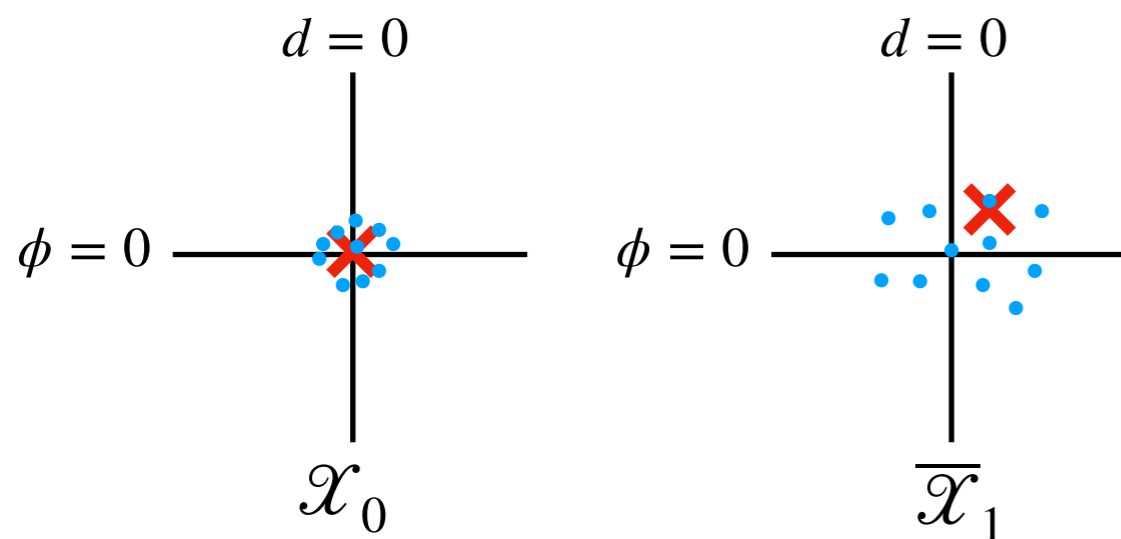
$d_t$ : Duckiebot distance to lane center

$\phi_t$ : Angle of Duckiebot relative to lane



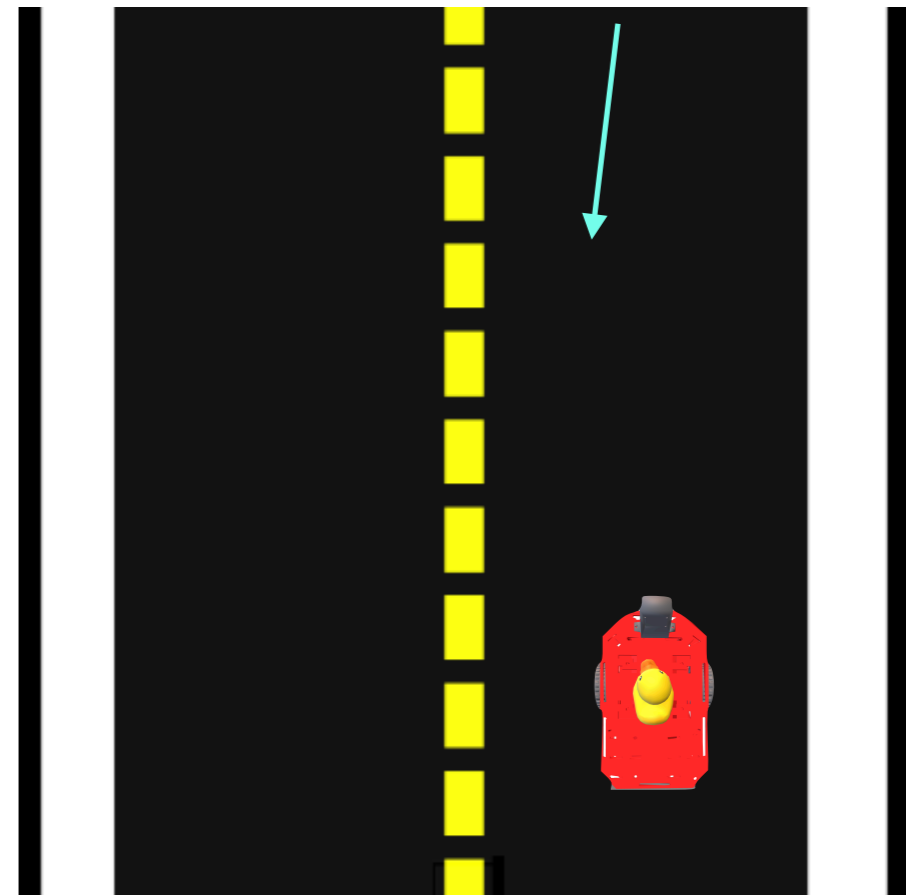
# Sample the process model

Robot is commanded to drive straight



- ✘ : True state of Duckiebot
- : Particles

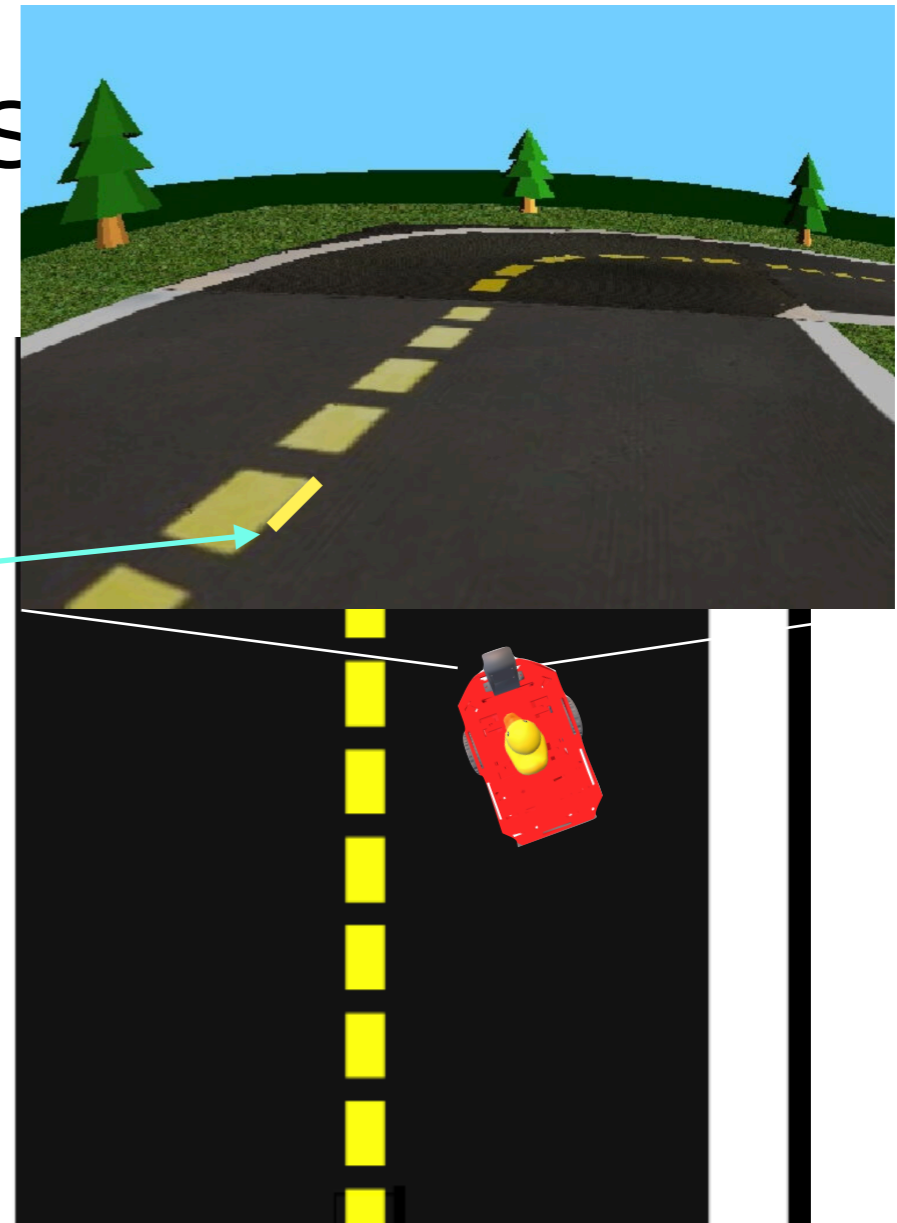
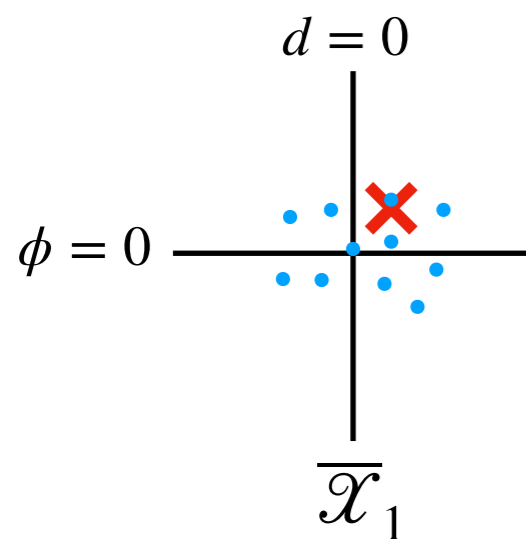
But there is some drift



# Update particle weights

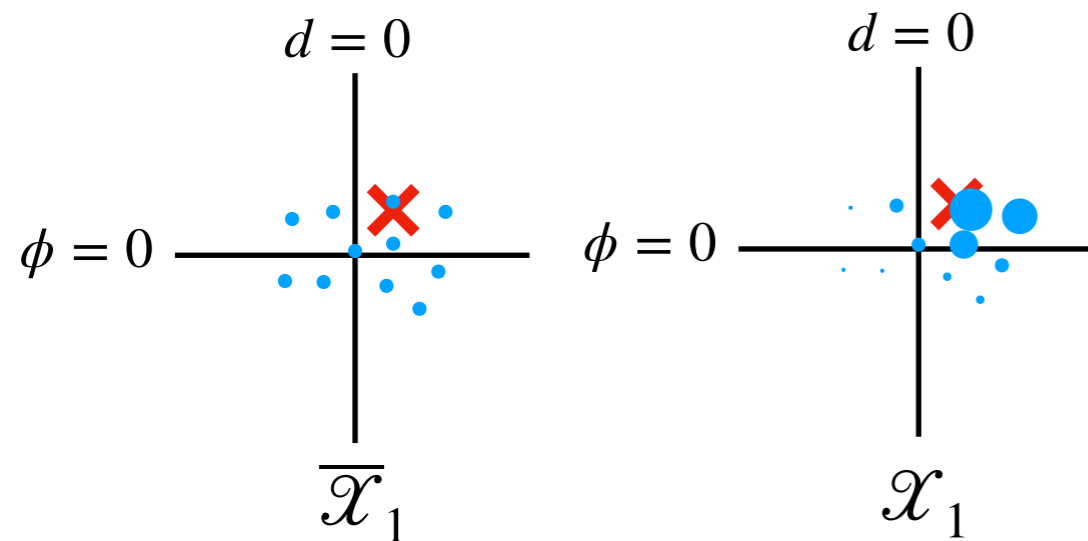
An image is received

A line segment is detected

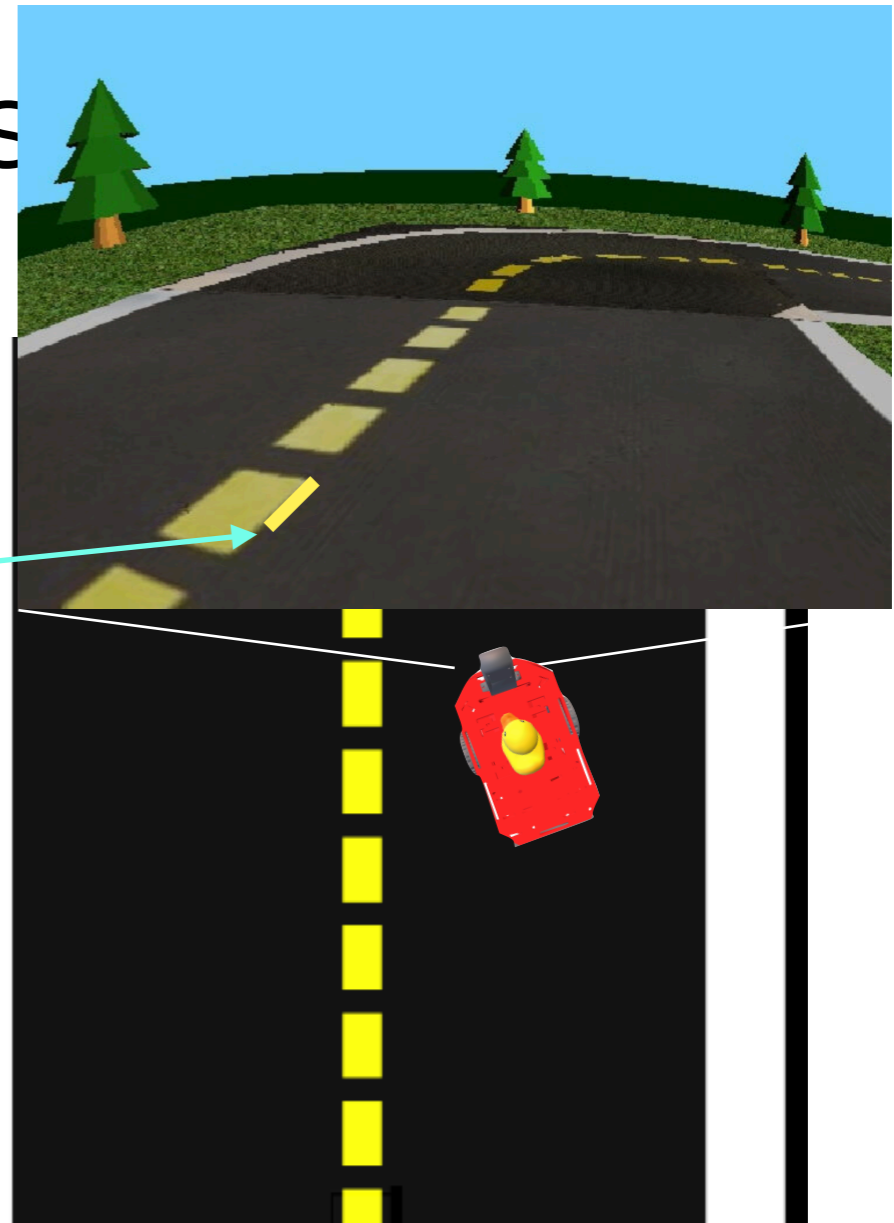


- ✗ : True state of Duckiebot
- : Particles

# Update particle weights

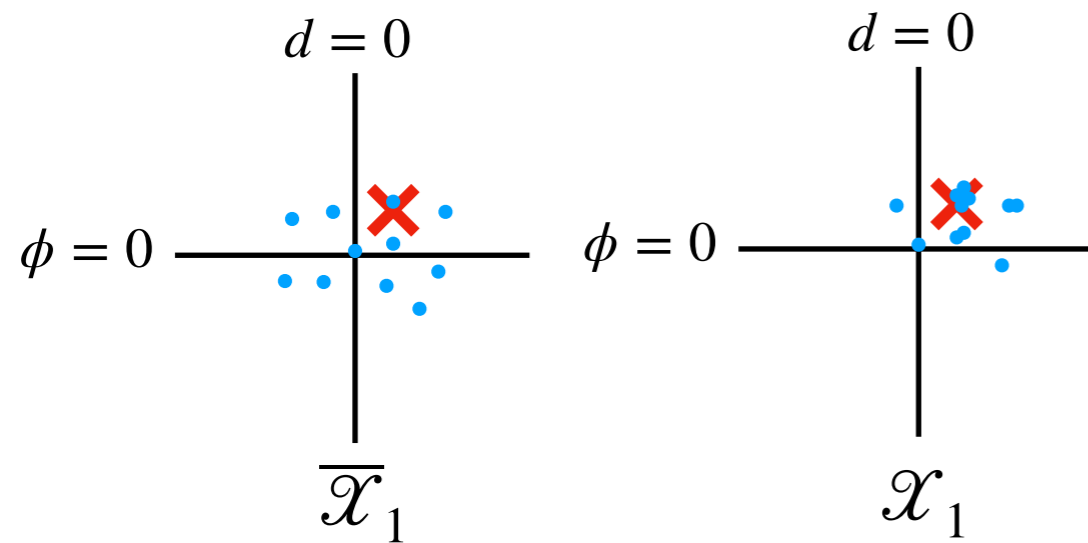


A line segment is detected

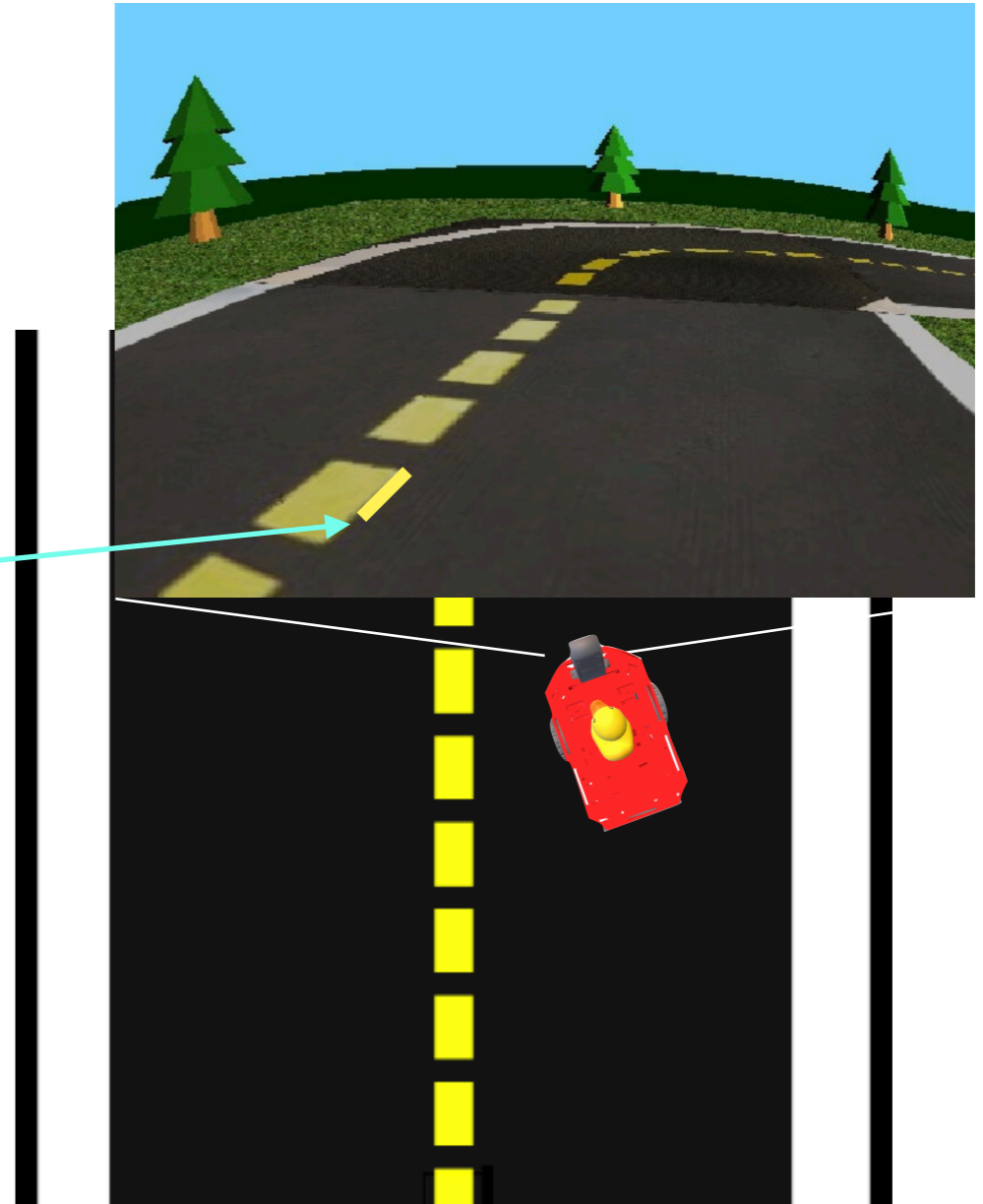


- ✘ : True state of Duckiebot
- : Particles

# Sampling importance resampling

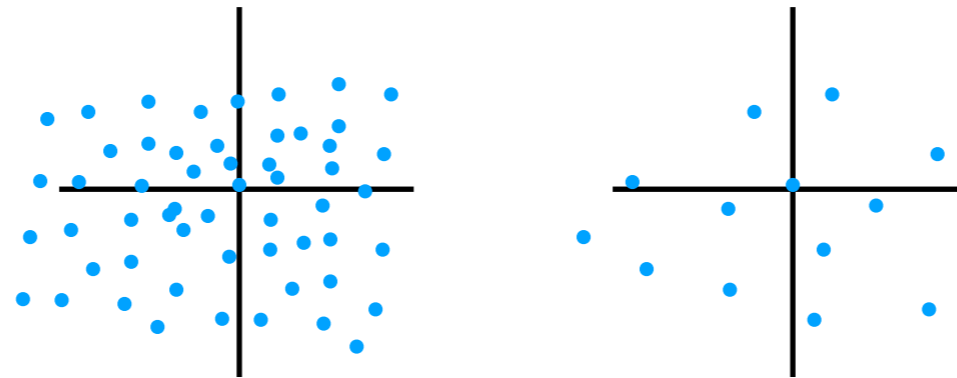


A line segment is detected

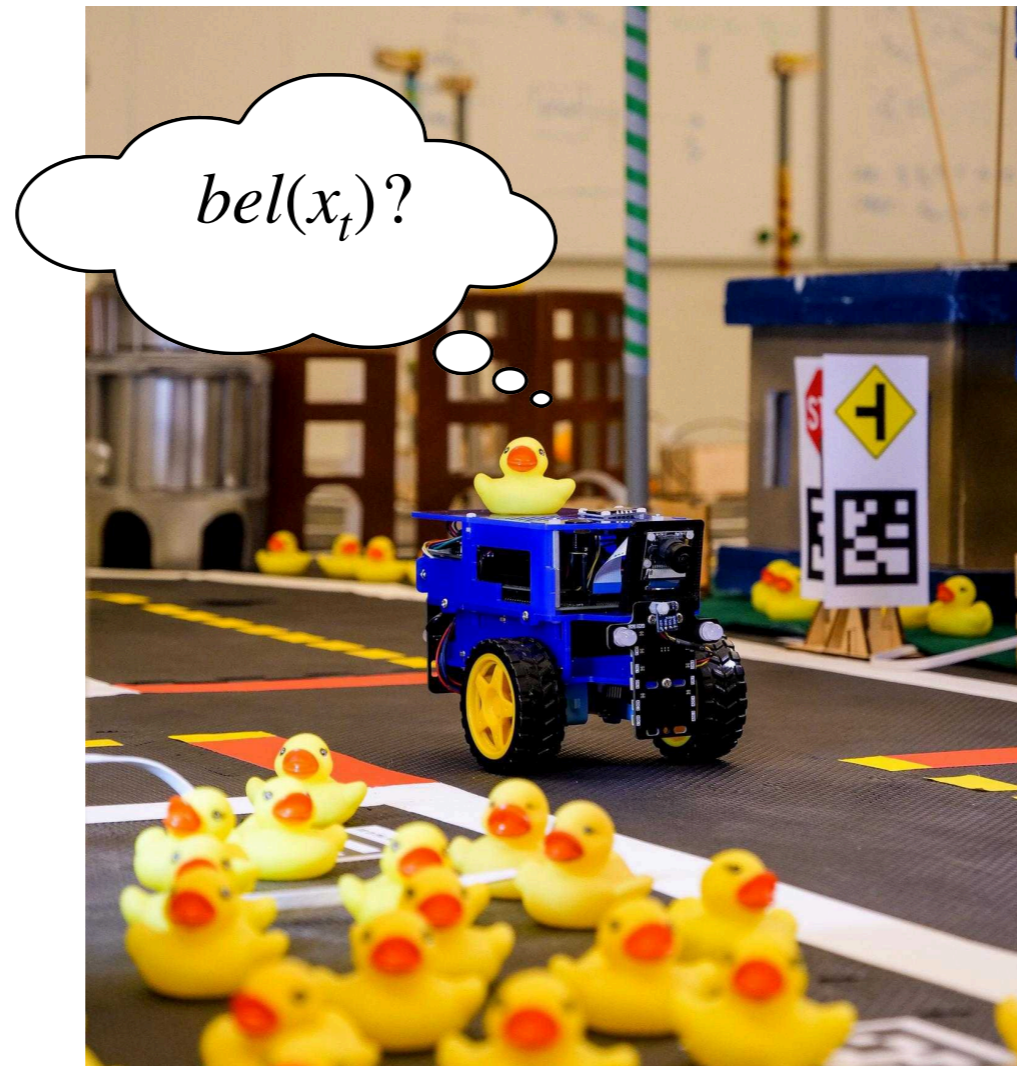


- ✘ : True state of Duckiebot
- : Particles

Size of the  
particle set



# Lane Pose Estimation

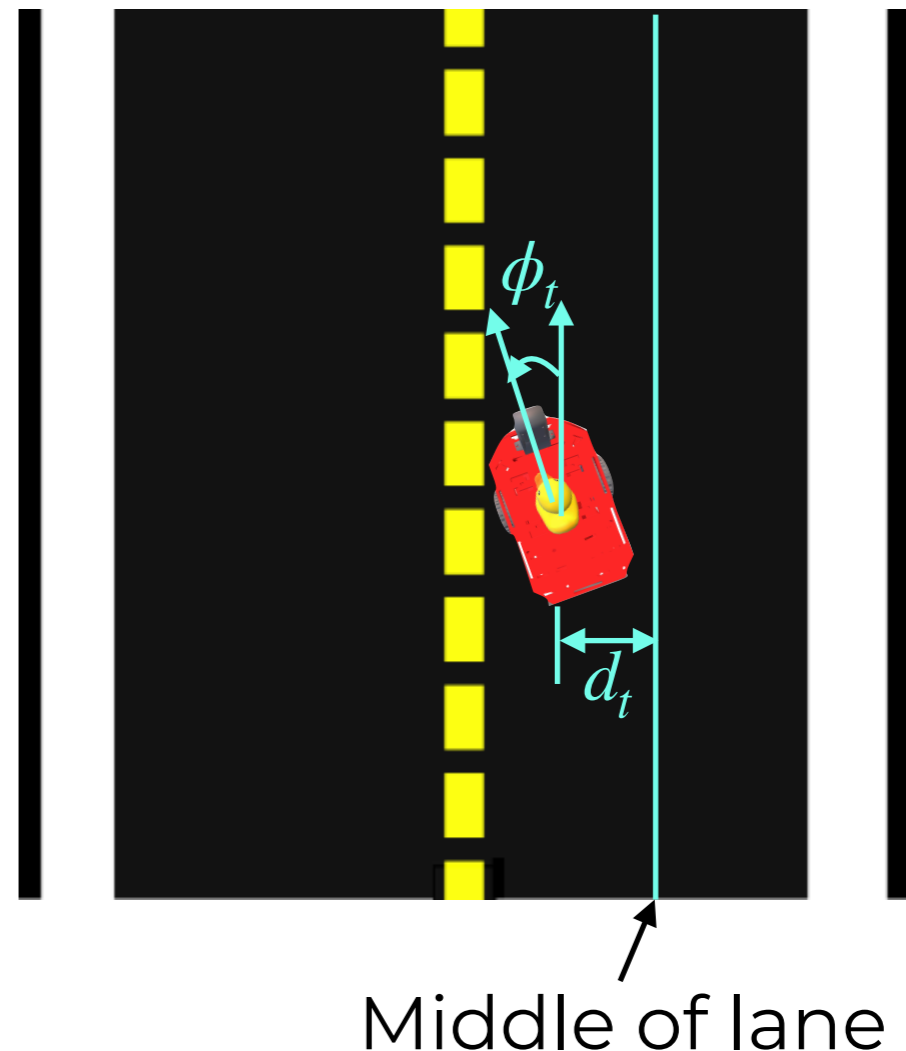


# Lane-relative estimation

State:  $x_t \triangleq [d_t, \phi_t]$

$d_t$ : Duckiebot distance to lane center

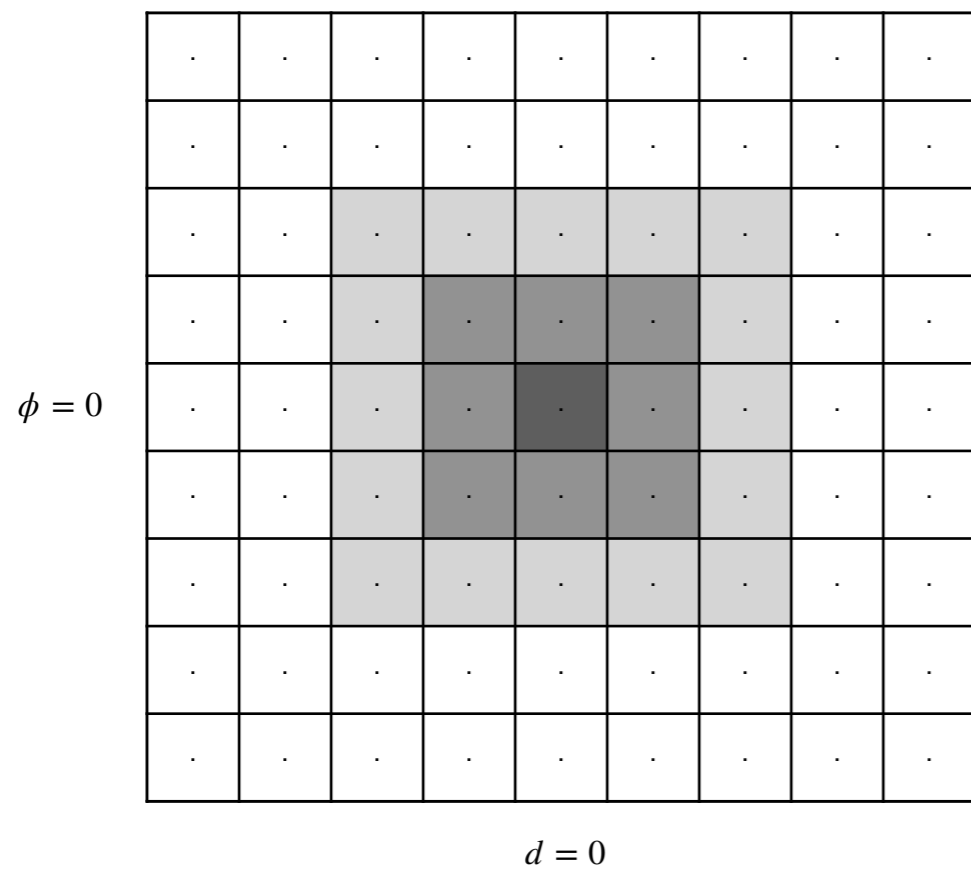
$\phi_t$ : Angle of Duckiebot relative to lane



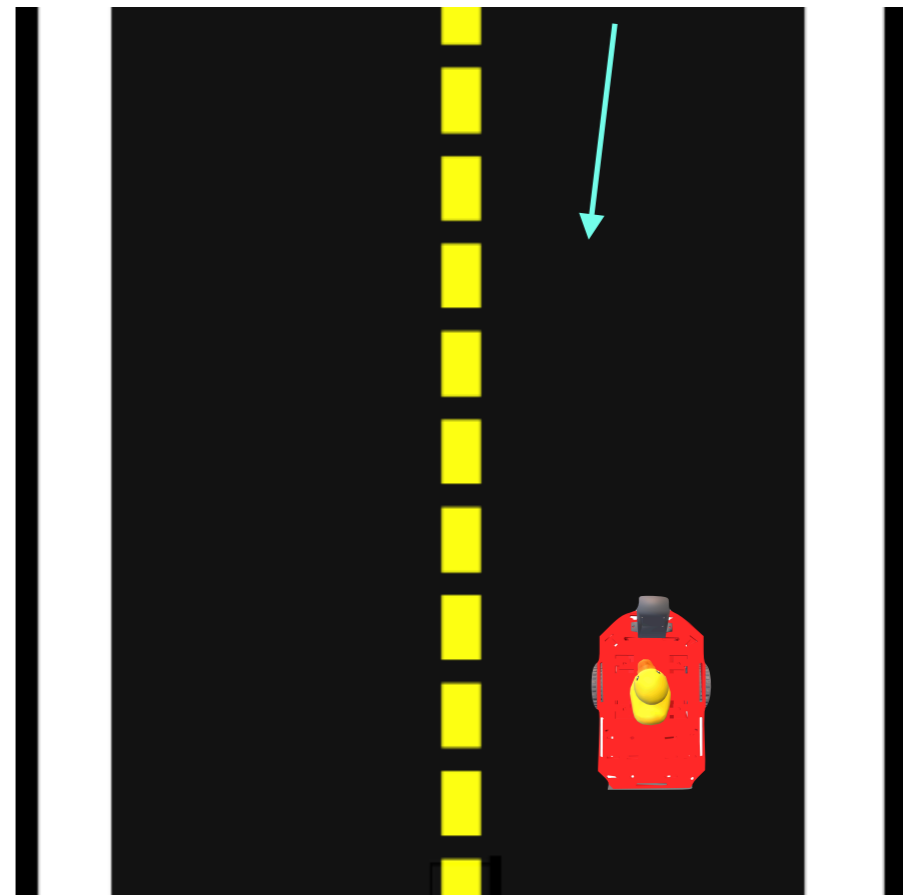
# Propagate the weights

Robot is commanded to drive straight

$$\overline{bel}(x_0)$$



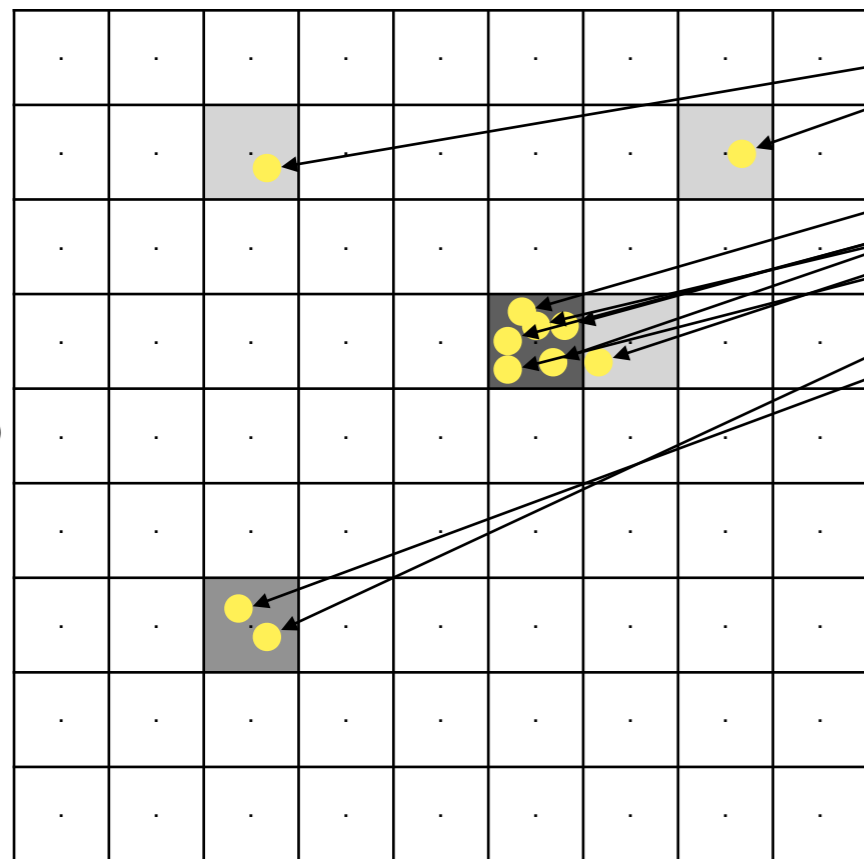
But there is some drift



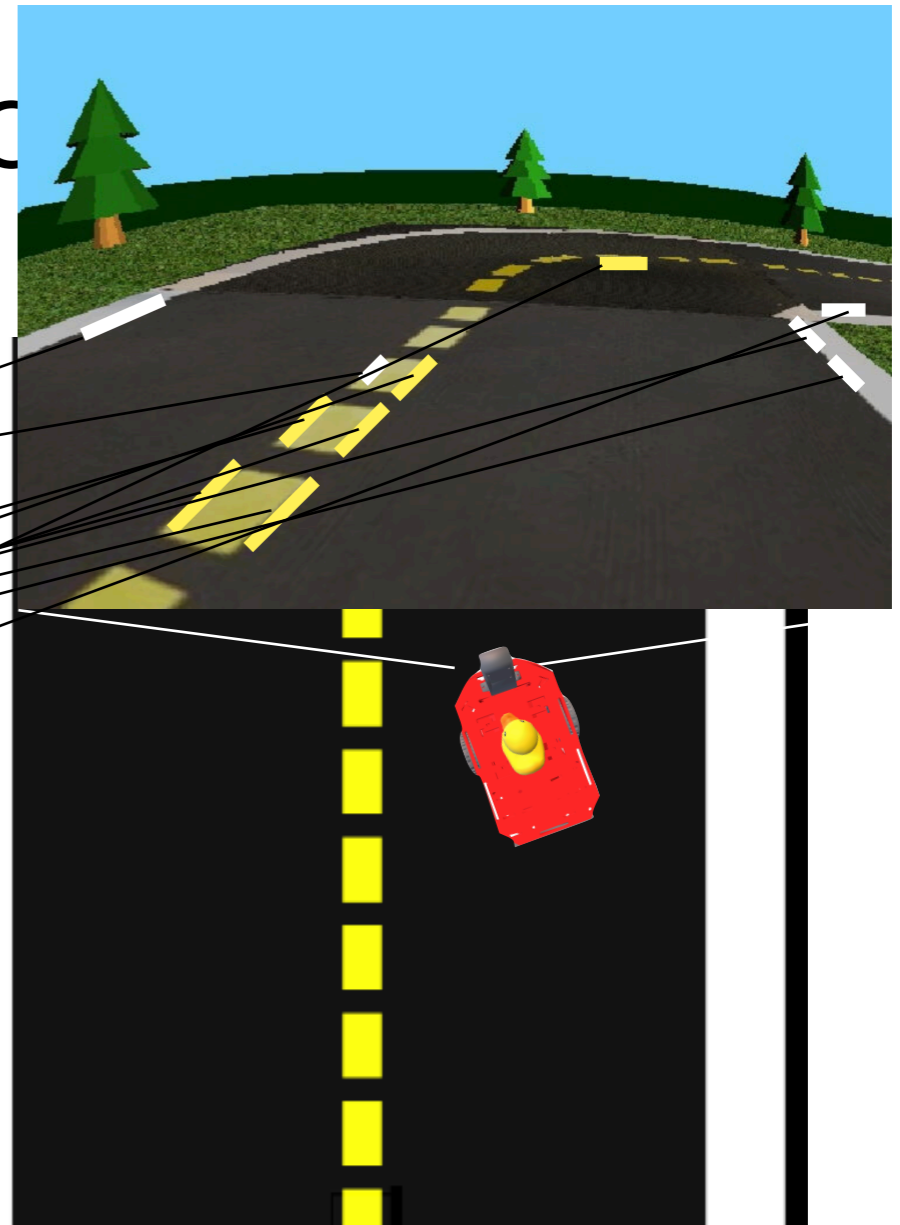
# Measurement likelihood

An image is received

$$p(z_t | x^i)$$

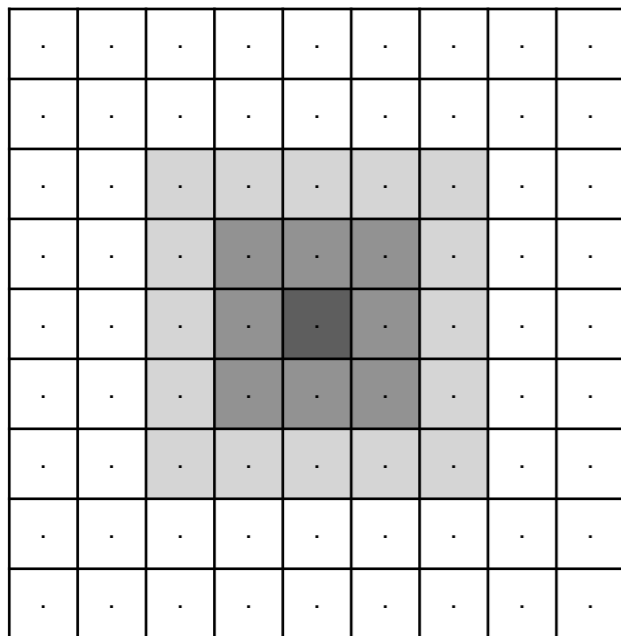


Many line segments  
are detected  
Each one generates  
a vote



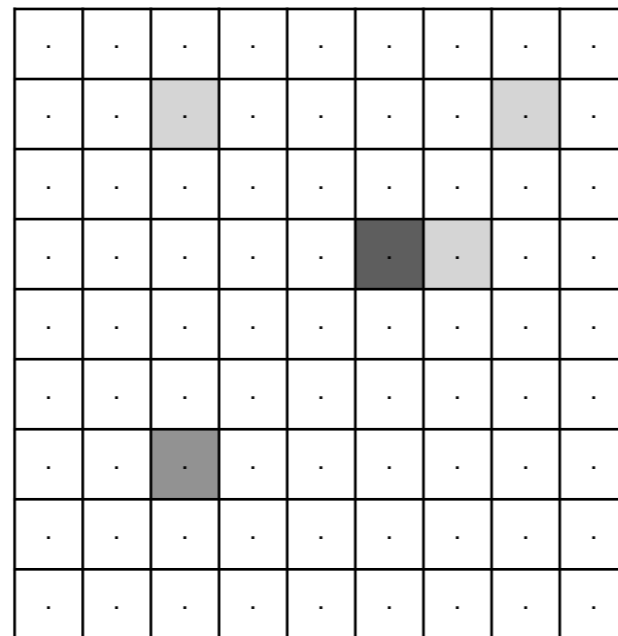
# Update the weights

Prior belief



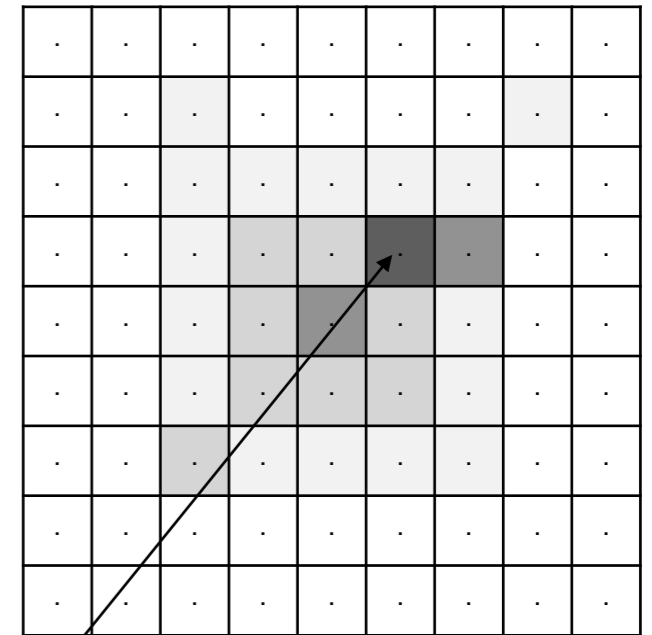
$\times$

Measurement likelihood



$=$

Posterior



Maximum *a posteriori* estimate