EPFL

Disal

Signals, Instruments, and Systems – W10

Dealing with Non-Deterministic Uncertainties in Localization

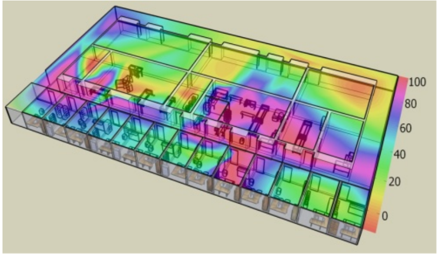
Outline

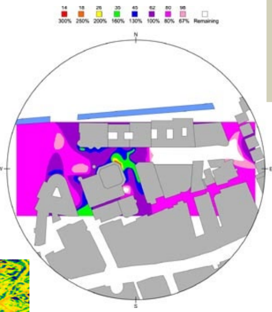
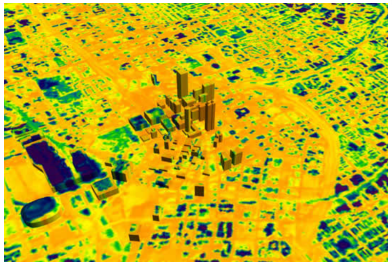
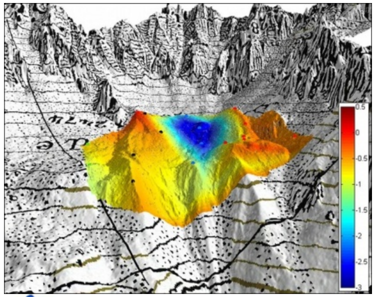
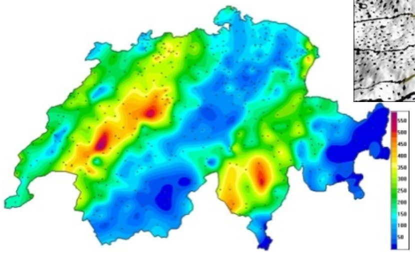
- Motivation for localization and estimation methods
- Fusing information from noisy processes and noisy sensing: the Kalman Filter algorithm in 1D problems
- Localization using proprioceptive sensors with non-deterministic uncertainties
- Feature-based localization
- Kalman filter algorithm for 2D localization
- Particle filters for localization

Motivation for Localization and Sensor Fusion

Motivation for Localization

- Environmental data most often useless without location information
- Localization is required at different scales (1000's km → cm)



Pictures: courtesy of Sensorscope, NASA, NIST

Motivation for Stochastic Models and Estimation Methods

- A mathematical system model is necessarily an approximation of the real system: system structure is often appropriate but parameters are affected by uncertainties
- Non-deterministic sources of noise: we can neither control stochastic disturbances, nor get rid of stochastic sensor noise by calibration, nor model stochastic sources deterministically
- Sensors are noisy and can only partially observe the system state in all its details

Motivation for Stochastic Models and Estimation Methods

- The applicability of estimation methods goes way beyond localization problem
- To this purpose most diverse problems in environmental and civil engineering leverage Kalman filters for handling noisy processes and sensing
- Here are some examples I found in recent literature:
  - Improving accuracy of subsurface contaminant transport model
  - Improving in-situ monitoring of groundwater contamination
  - Identification of civil structural parameters
  - Improving predictive modeling of landslide deformation
  - Improving estimation of environmental performance variables in an acid gas removal process

The Kalman Filter Algorithm in 1D

Two Key Sources of Information

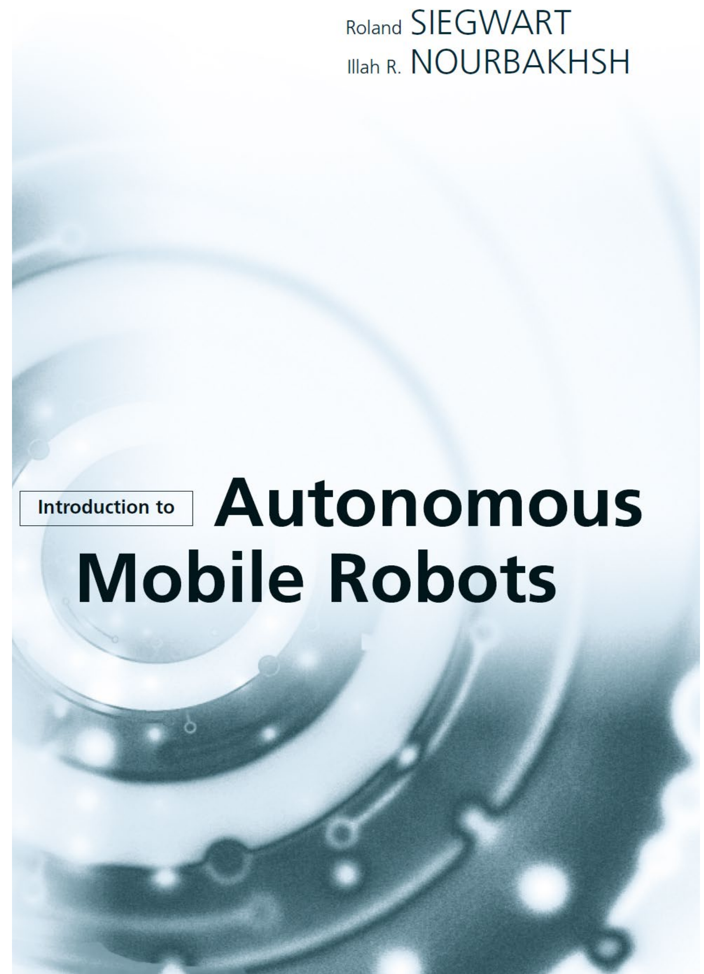
Stochastic models, estimation, and control

VOLUME 1

PETER S. MAYBECK

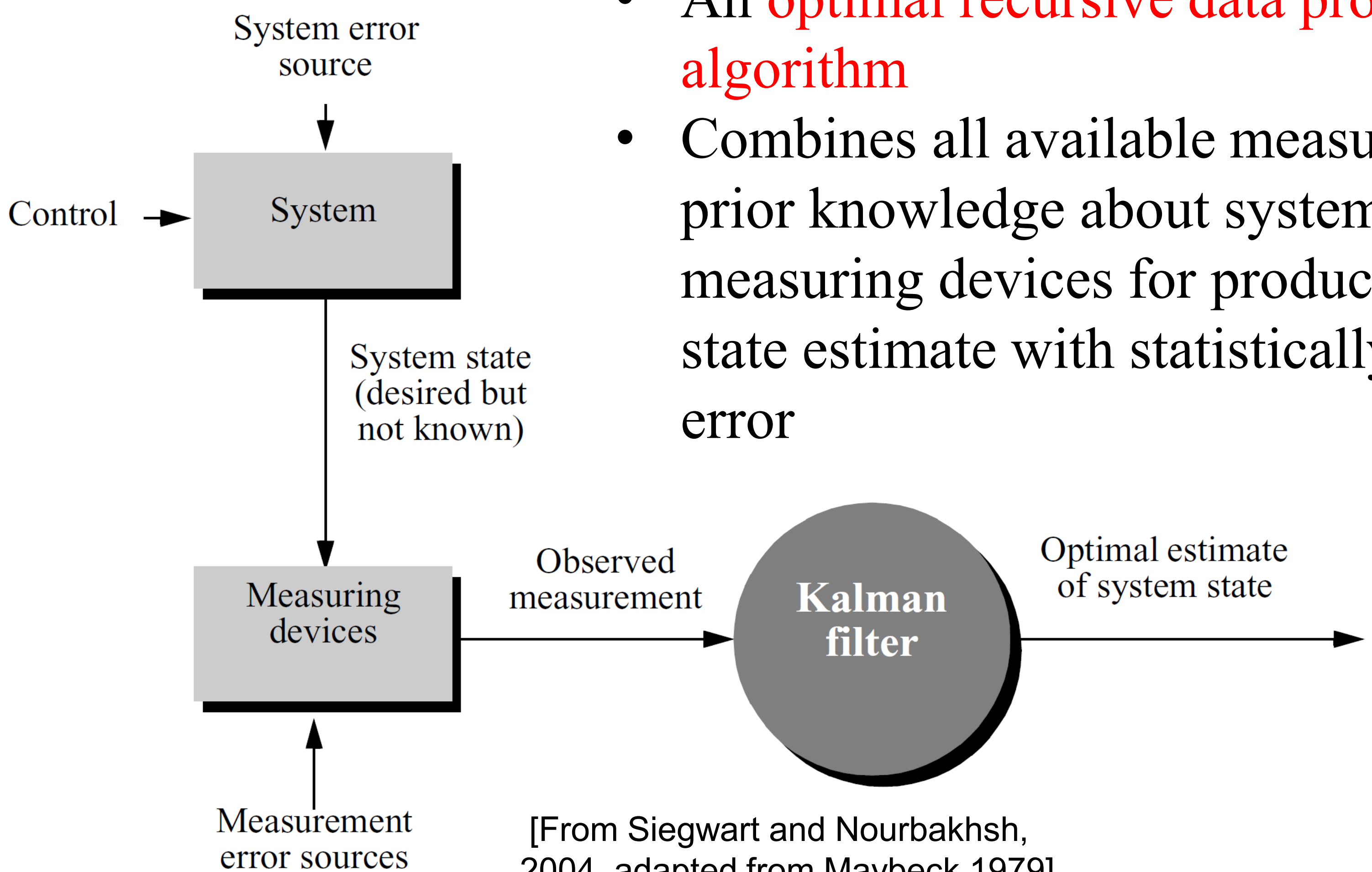
DEPARTMENT OF ELECTRICAL ENGINEERING
AIR FORCE INSTITUTE OF TECHNOLOGY
WRIGHT-PATTERSON AIR FORCE BASE
OHIO

ACADEMIC PRESS New York San Francisco London 1979
A Subsidiary of Harcourt Brace Jovanovich, Publishers



Kalman Filter - Overview

- An optimal recursive data processing algorithm
- Combines all available measurement data, prior knowledge about system and measuring devices for producing a system state estimate with statistically minimized error



[From Siegwart and Nourbakhsh, 2004, adapted from Maybeck 1979]

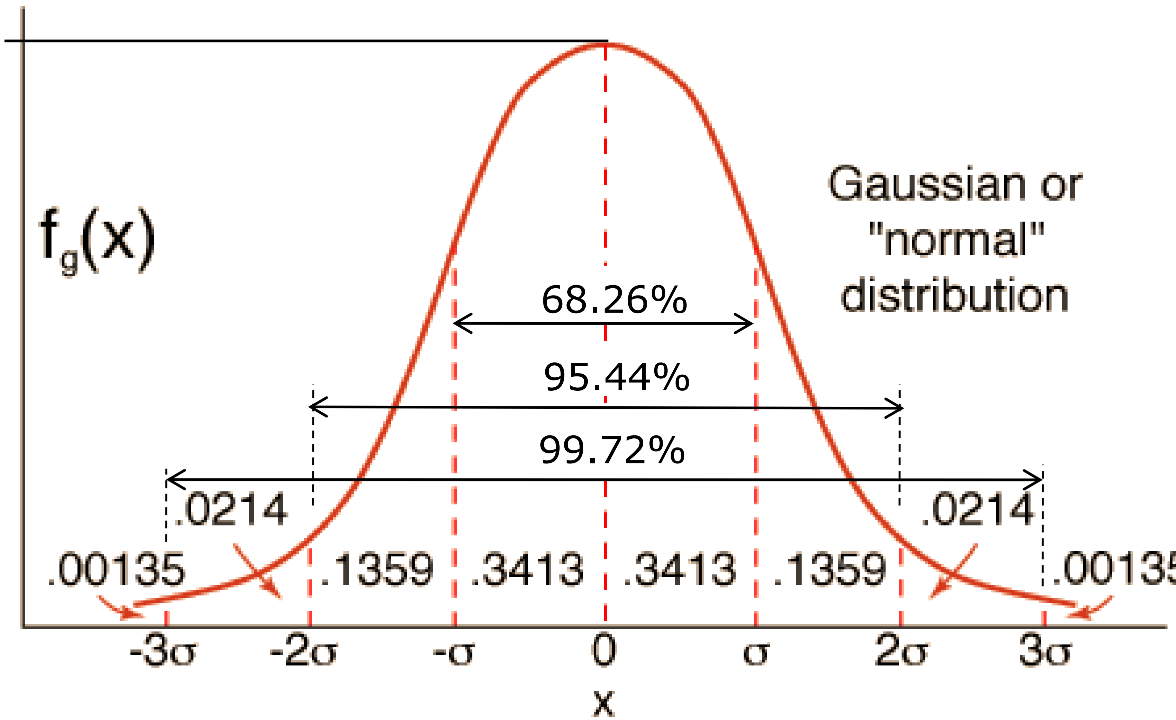
Kalman Filter - Assumptions

- Linear system model
- White Gaussian system and measurement noise

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Notes on noise:

- White: uncorrelated in time
- Gaussian: probability density of amplitude follows bell-shaped curve

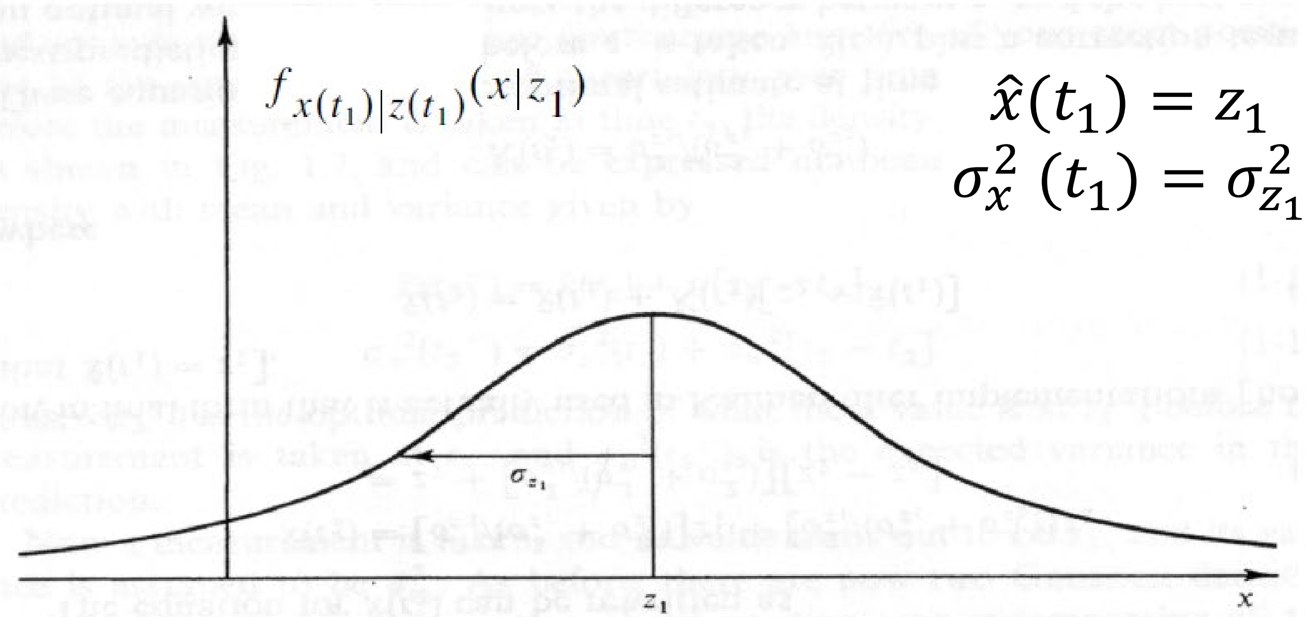


x = μ

A Simple 1D Positioning Example

t = time
x = 1D position
$\hat{x}$ = 1D position estimate
z = measurement or observation

t_i = time instant i
z_i = measurement taken at t_i

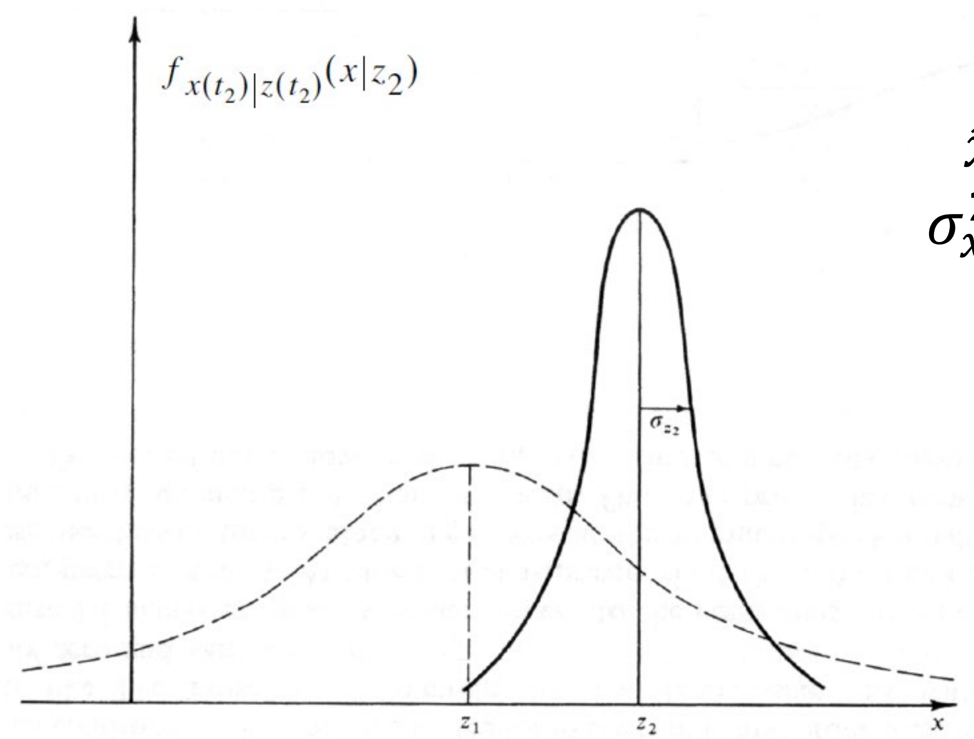


$$\hat{x}(t_1) = z_1$$
$$\sigma_x^2(t_1) = \sigma_{z_1}^2$$

[From Maybeck, 1979]

Estimation Based on Static Measurements

- Second measurement taken at $t_2 \approx t_1$
- The smaller σ, the higher the certitude about the measurement

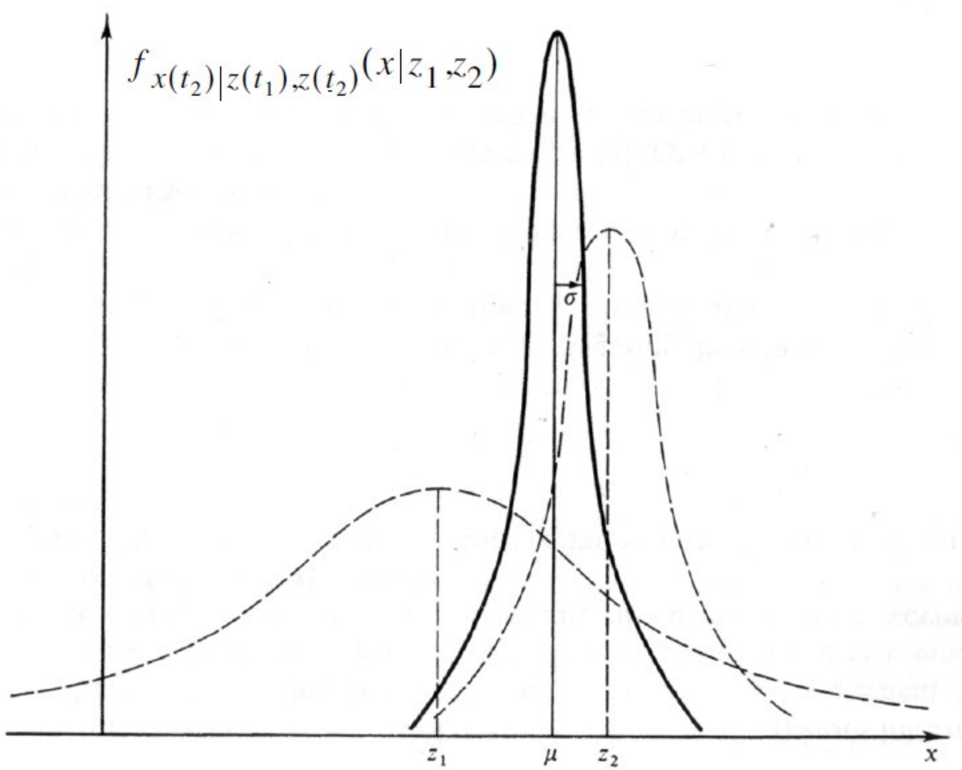


$$\hat{x}(t_2) = z_2$$
$$\sigma_x^2(t_2) = \sigma_{z_2}^2$$

[From Maybeck, 1979]

Improving the Estimate Through Fusion

- Intuition: the smaller σ, and thus σ^2, the higher should be the weight in the fused estimate
- Estimate can be obtained as a weighted average μ of the individual measurement contributions



[From Maybeck, 1979]

$$\mu = \frac{\frac{1}{\sigma_{z_1}^2} z_1 + \frac{1}{\sigma_{z_2}^2} z_2}{\frac{1}{\sigma_{z_1}^2} + \frac{1}{\sigma_{z_2}^2}}$$

$$\frac{1}{\sigma^2} = \frac{1}{\sigma_{z_1}^2} + \frac{1}{\sigma_{z_2}^2}$$

$$\hat{x}(t_2) = \mu$$

$$\sigma_x(t_2) = \sigma$$

$$\sigma < \sigma_{z_1} \text{ and } \sigma < \sigma_{z_2}$$

Mean of the new Estimate

$$\hat{x}(t_2) = \frac{\frac{1}{\sigma_{z_1}^2} z_1 + \frac{1}{\sigma_{z_2}^2} z_2}{\frac{1}{\sigma_{z_1}^2} + \frac{1}{\sigma_{z_2}^2}} = \frac{\sigma_{z_2}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_1 + \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_2$$

$$= \frac{\sigma_{z_2}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_1 \boxed{+ \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_1 - \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_1} + \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_2$$

$$= \frac{\sigma_{z_2}^2 + \sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} z_1 + \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} (z_2 - z_1)$$

$$= z_1 + \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} (z_2 - z_1)$$

Kalman filter formulation

$$\hat{x}(t_2) = \hat{x}(t_1) + K(t_2)[z_2 - \hat{x}(t_1)]$$

with $K(t_2) = \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2}$

$\hat{x}(t_1) = z_1$

Variance of the new Estimate

$$\frac{1}{\sigma_x^2(t_2)} = \frac{1}{\sigma_{z_1}^2} + \frac{1}{\sigma_{z_2}^2}$$

$$\sigma_x^2(t_2) = \frac{\sigma_{z_1}^2 \sigma_{z_2}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2} = \frac{\sigma_{z_2}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2}\,\sigma_{z_1}^2 = \frac{\sigma_{z_1}^2 + \sigma_{z_2}^2 - \sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2}\,\sigma_{z_1}^2 = \left(1 - \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2}\right)\sigma_{z_1}^2$$

$$= \sigma_{z_1}^2 - \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2}\,\sigma_{z_1}^2$$

Kalman filter formulation

$$\sigma_x^2(t_2) = \sigma_x^2\,(t_1) - K(t_2)\sigma_x^2(t_1)$$

with $K(t_2) = \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2}$

$$\sigma_x^2(t_1) = \sigma_{z_1}^2$$

Kalman Filter for Sensor Fusion

$$\hat{x}(t_2) = \hat{x}(t_1) + K(t_2)[z_2 - \hat{x}(t_1)]$$

Mean

Prediction

Correction or Update

$$\sigma_x^2(t_2) = \sigma_x^2(t_1) - K(t_2)\,\sigma_x^2(t_1)$$

Variance

$$K(t_2) = \frac{\sigma_{z_1}^2}{\sigma_{z_1}^2 + \sigma_{z_2}^2}$$

Kalman Gain

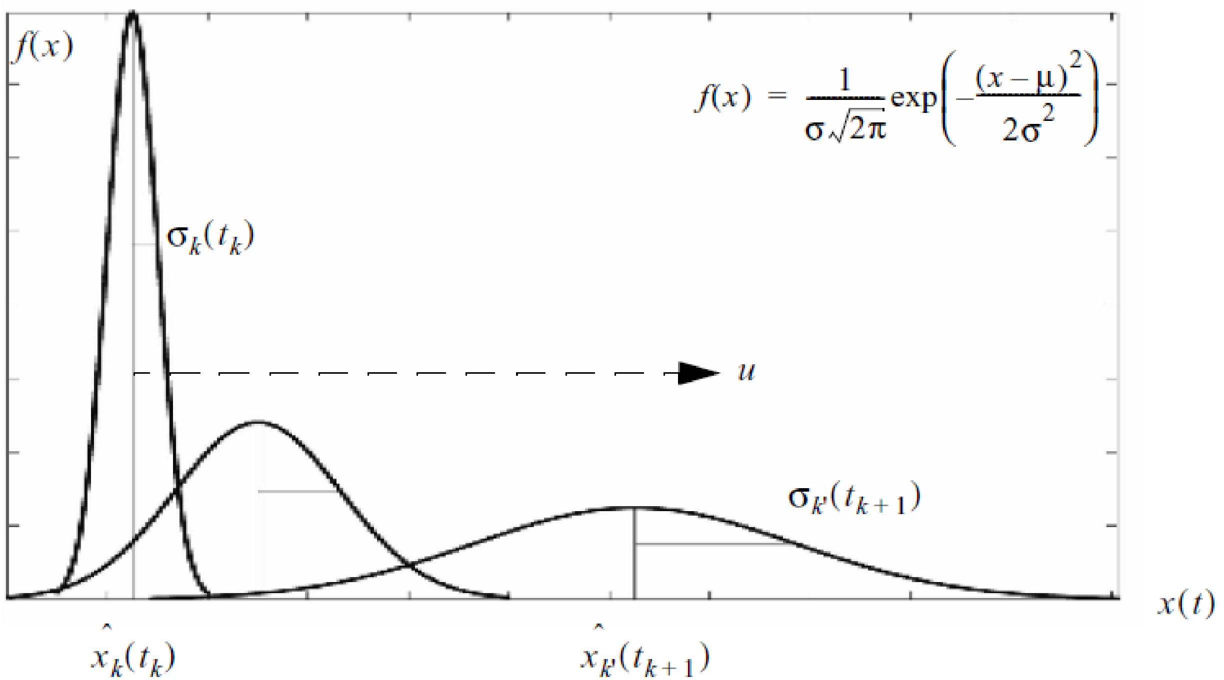
Function of the sensing precision

Estimation Considering Motion Dynamics

Consider a simple but noisy motion model: $\dot{x} = u + w$

u = constant speed (controllable input)
w = Gaussian motion noise

$w = \sigma_w^2$



[From Siegwart and Nourbakhsh, 2004, adapted from Maybeck 1979]

σ_k^2: variance at timestep k (known)

$\sigma_{k'}^2$: variance at timestep k+1

Estimation Based on Motion Model

$$\hat{x}_{k'} = \hat{x}_k + u[t_{k+1} - t_k]$$

- New mean position at timestep t_{k+1}
- Can be estimated with deterministic displacement from motion model

$$\sigma^2_{k'} = \sigma^2_k + \sigma^2_w[t_{k+1} - t_k]$$

- New variance at timestep t_{k+1}
- Variance of noisy motion (constant over time) gets added (cumulated) to previous one

Fusing Motion Model Prediction with New Measurement - Mean

$$\hat{x}_{k+1} = \hat{x}_{k'} + K_{k+1}(z_{k+1} - \hat{x}_{k'}) \qquad \text{with } K_{k+1} = \frac{\sigma_{k'}^2}{\sigma_{k'}^2 + \sigma_z^2}$$

Prediction based on motion model

Correction or update based on observation

z_{k+1} : measurement at timestep k+1

K_{k+1} : Kalman gain at timestep k+1

$\hat{x}_{k+1}$: new estimate at timestep k+ 1 incorporating observation and prediction of motion model

$\hat{x}_{k'}$: estimate just before timestep k+1 based on prediction motion model

EPFL Disal

Fusing Motion Model Prediction with New Measurement - Variance

$$\sigma_{k+1}^2 = \sigma_{k'}^2 - K_{k+1}\,\sigma_{k'}^2 \qquad \text{with } K_{k+1} = \frac{\sigma_{k'}^2}{\sigma_{k'}^2 + \sigma_z^2}$$

Prediction based on motion model

Correction or update based on observation

K_{k+1} : Kalman gain at timestep k+1

σ_{k+1}^2 : variance at timestep k+ 1 incorporating correction from observation and prediction of motion model

$\sigma_{k'}^2$: variance just before timestep k+1 based on prediction motion model

σ_z^2 : variance of the sensor meausurement (constant over time)

Kalman Filter for Sensor and Motion Model Fusion

$$\hat{x}_{k+1} = \hat{x}_{k'} + K_{k+1}(z_{k+1} - \hat{x}_{k'})$$

Mean

Prediction   Correction or Update

$$\sigma^2_{k+1} = \sigma^2_{k'} - K_{k+1}\,\sigma^2_{k'}$$

Variance

$$K_{k+1} = \frac{\sigma^2_{k'}}{\sigma^2_{k'} + \sigma^2_z}$$

Kalman Gain

Function of the motion model and sensing precision

Kalman Filter - Some Extreme Cases

$$\hat{x}_{k+1} = \hat{x}_{k'} + K_{k+1}(z_{k+1} - \hat{x}_{k'})$$

$$\sigma_{k+1}^2 = \sigma_{k'}^2 - K_{k+1}\,\sigma_{k'}^2$$

$$K_{k+1} = \frac{\sigma_{k'}^2}{\sigma_{k'}^2 + \sigma_z^2}$$

$\sigma_z^2 \rightarrow \infty$: new measurement extremely noisy, does not add information, then $K_{k+1} \rightarrow 0$ new estimate based exclusively on motion model (both mean and variance)

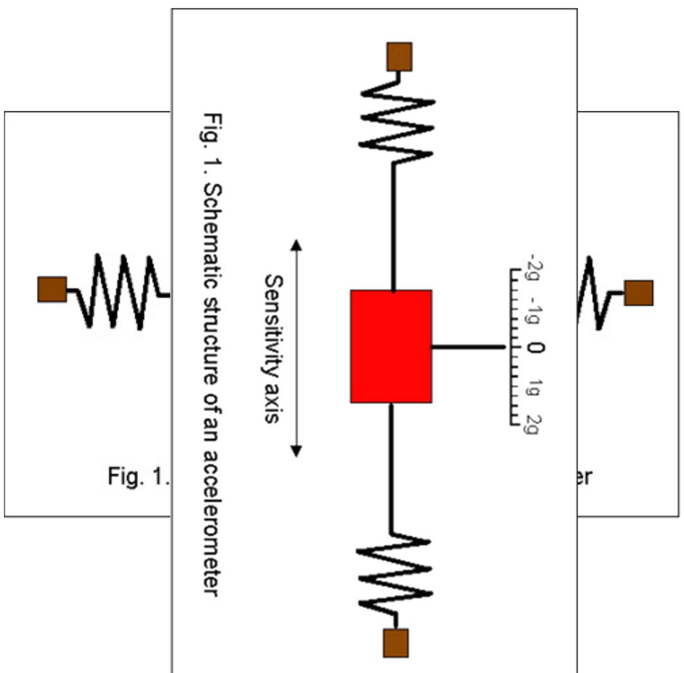
$\sigma_w^2 \rightarrow \infty$, then $\sigma_{k'}^2 \rightarrow \infty$ (see. s. 23): motion model does not add information, then $K_{k+1} \rightarrow 1$ new estimate based exclusively on new observation

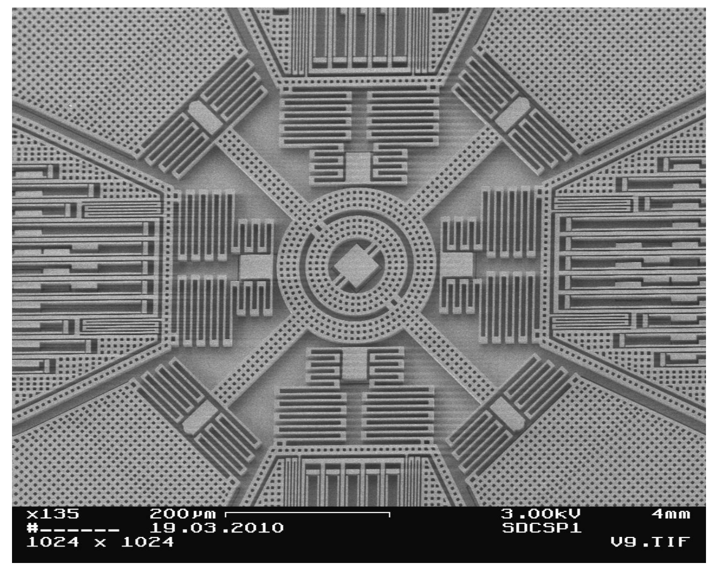
$\sigma_{k'}^2 \rightarrow 0$, motion model is deterministic and perfectly reproducing the reality, then $K_{k+1} \rightarrow 0$, new measurement can be disregarded since model is giving a perfect estimate

Mitigating Localization Uncertainities in Odometry Through Exteroceptive Sensors – The 1D Case

Non-Deterministic Error Sources

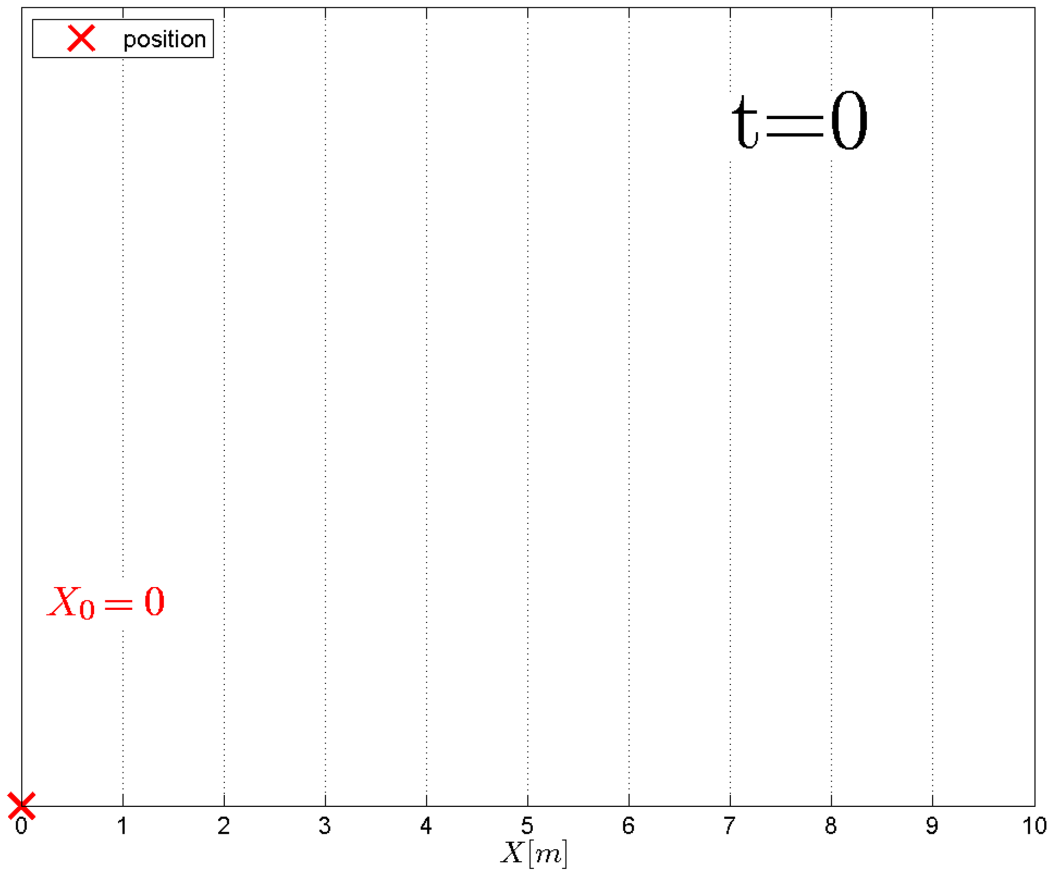
- From Week 9: no deterministic prediction possible
  → we have to describe them probabilistically
- Example: accelerometer-based odometry

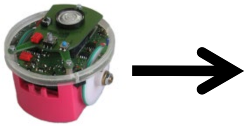




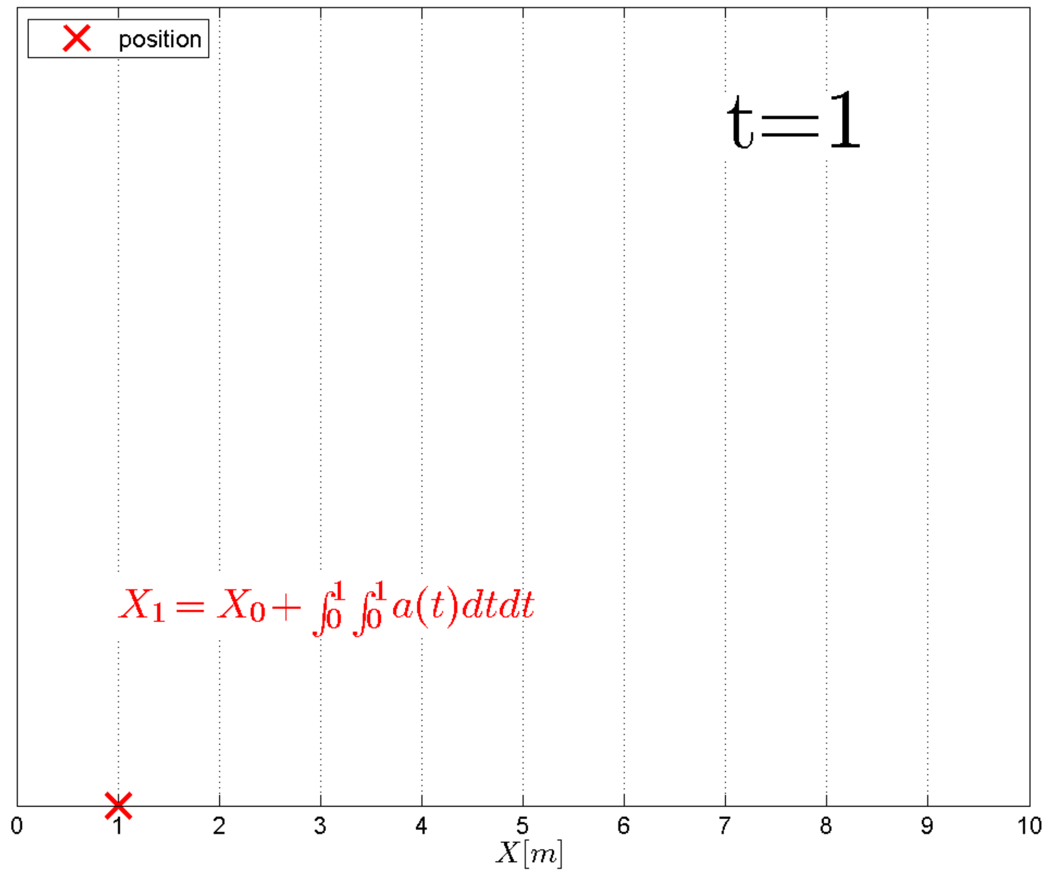
MEMS-Based accelerometer
(e.g., on e-puck)

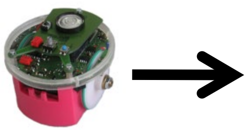
Odometry in 1D



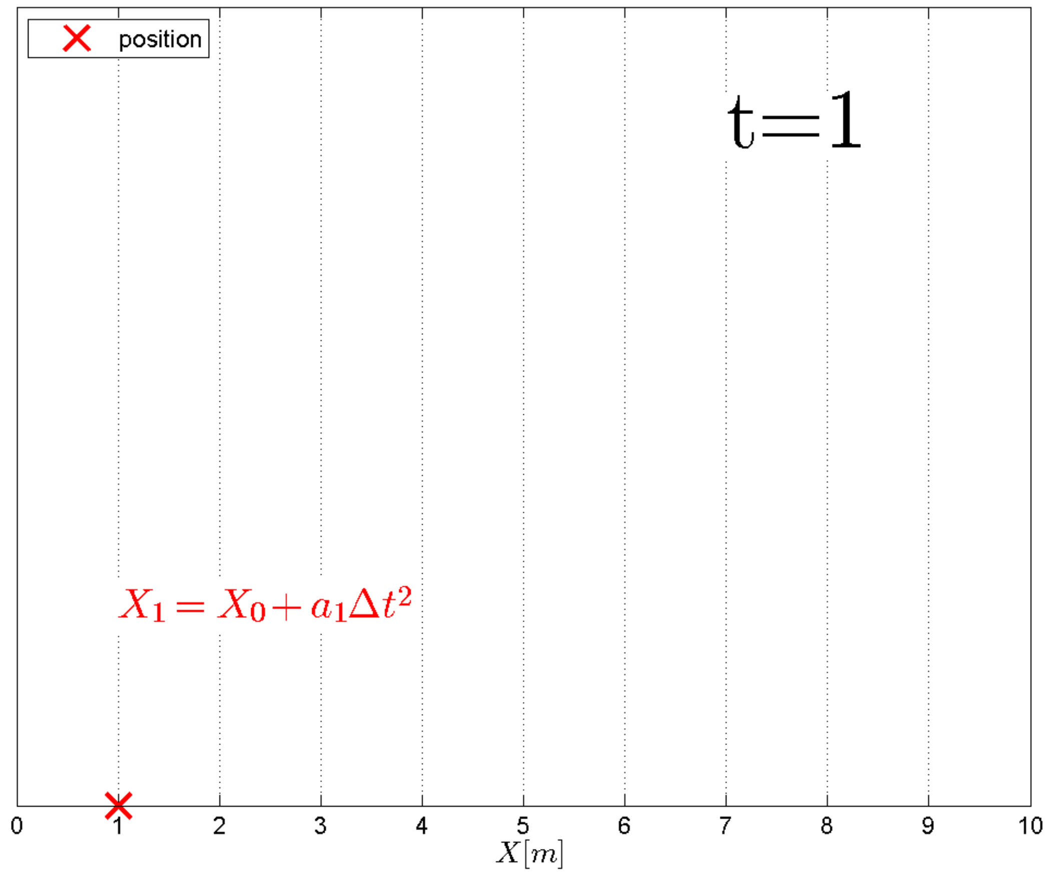


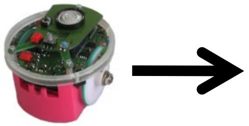
Odometry in 1D



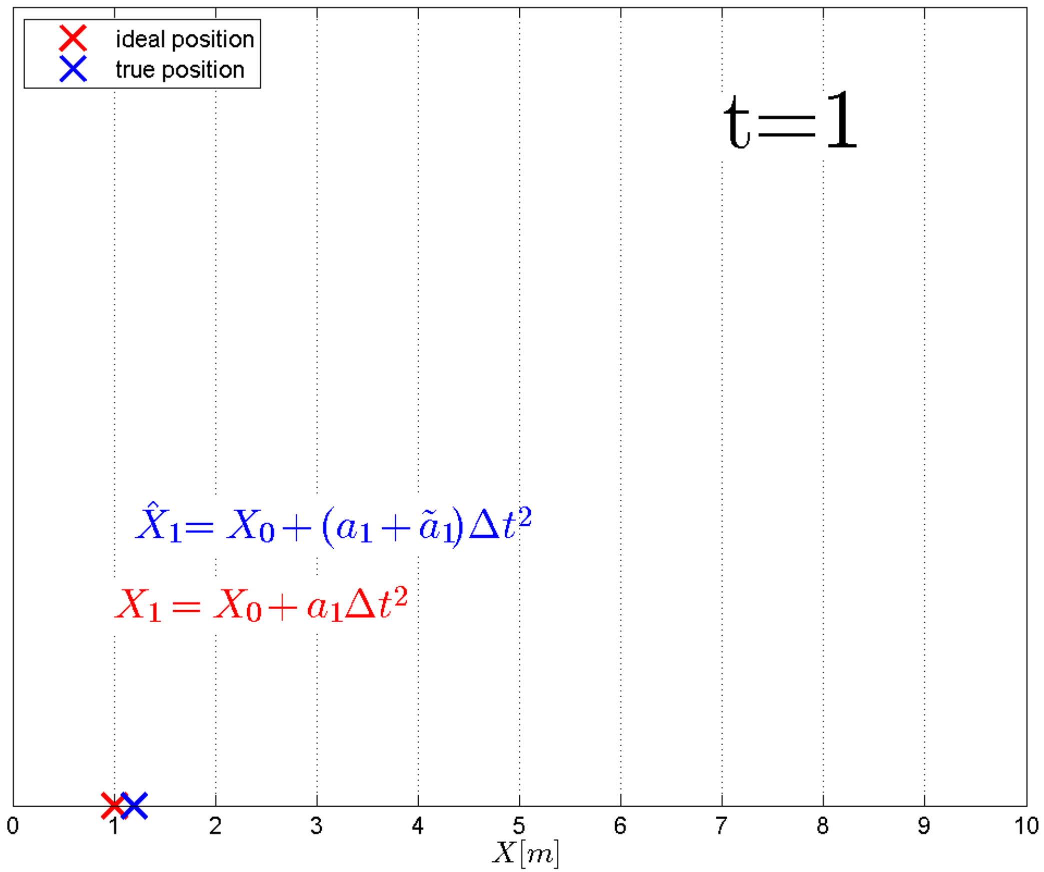


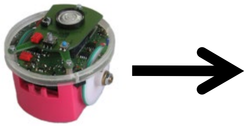
Odometry in 1D



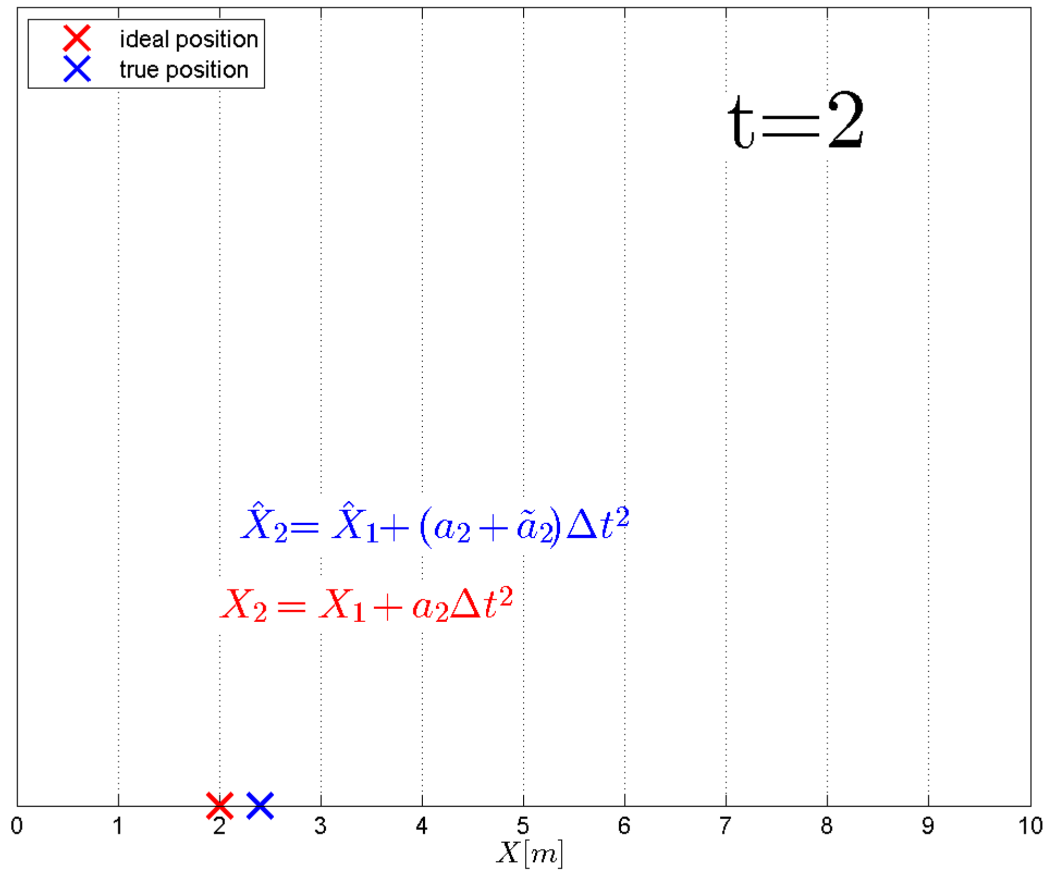


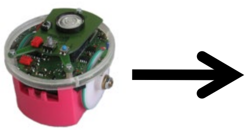
Odometry in 1D



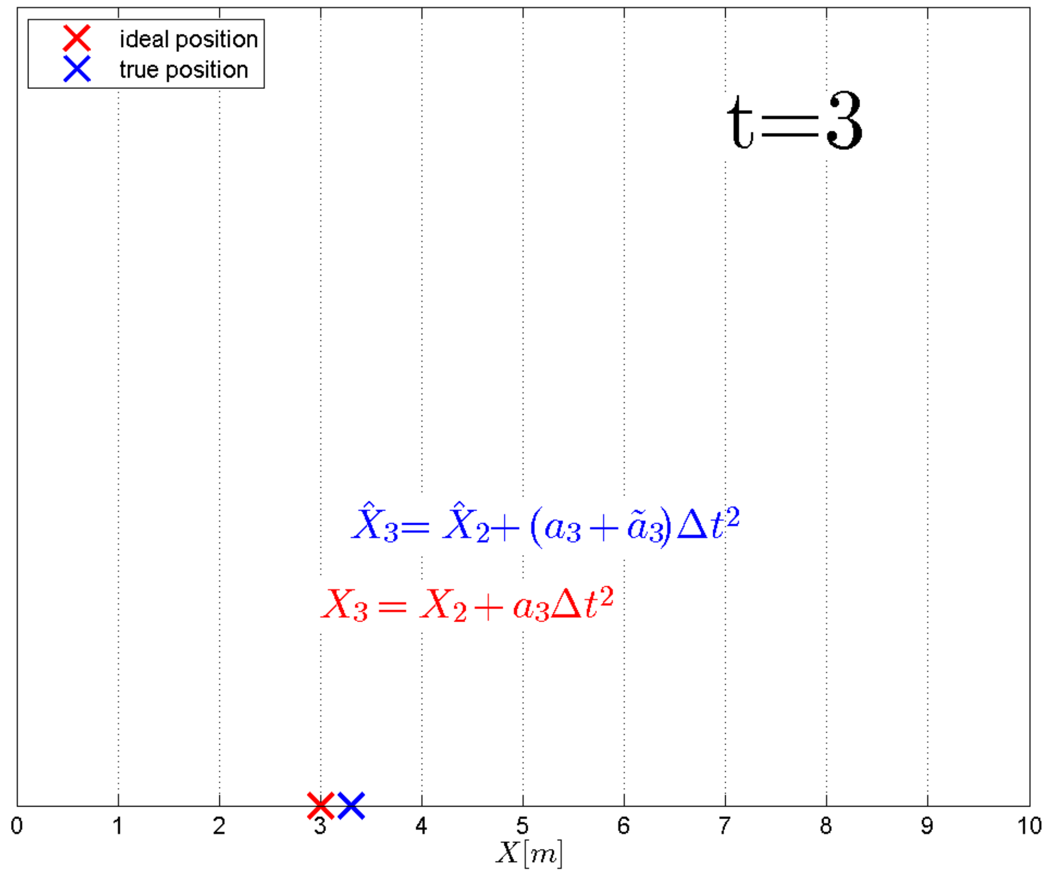


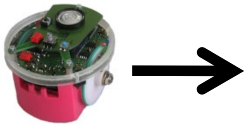
Odometry in 1D



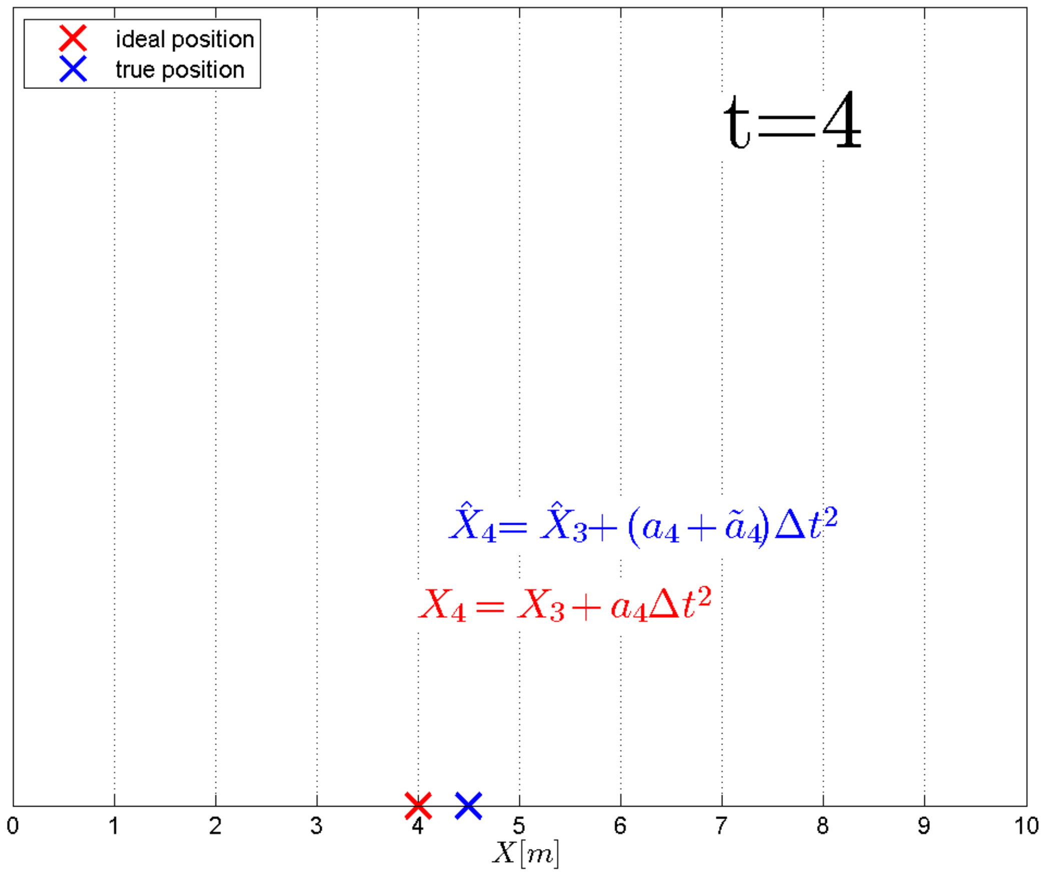


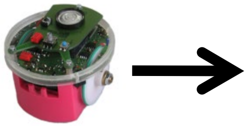
Odometry in 1D



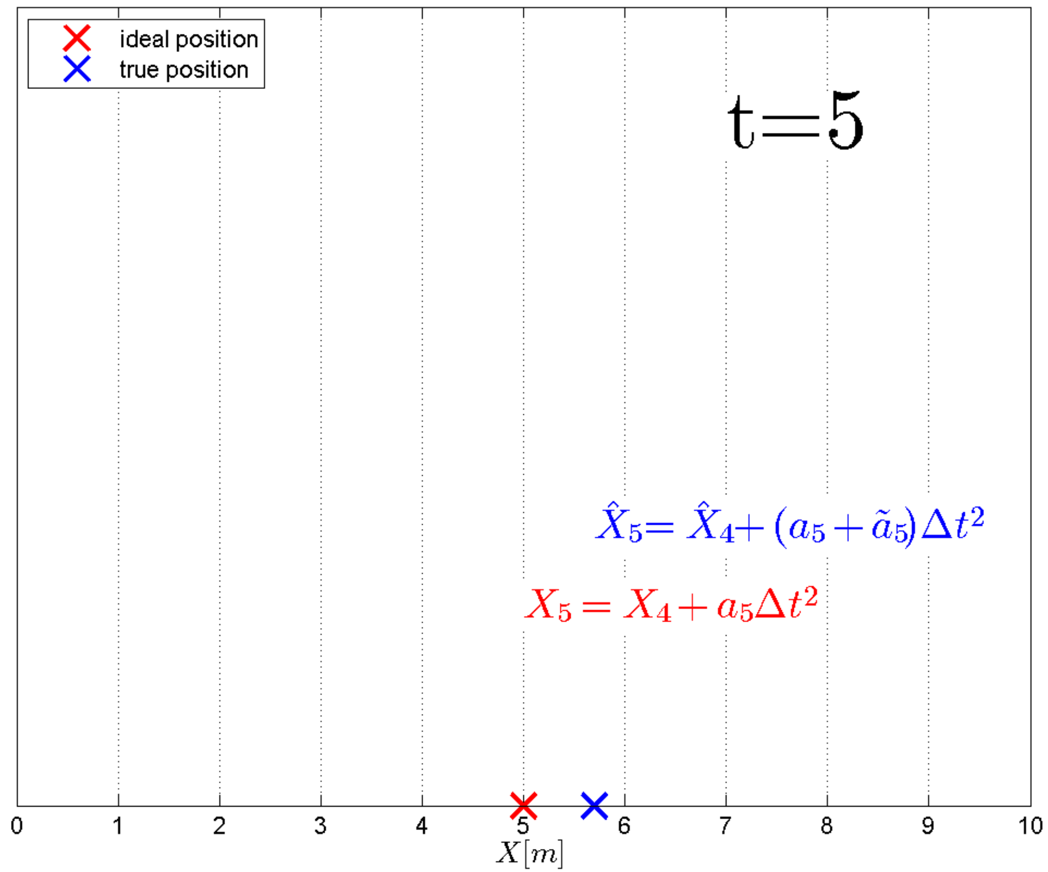


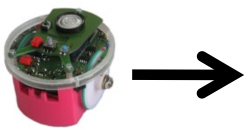
Odometry in 1D





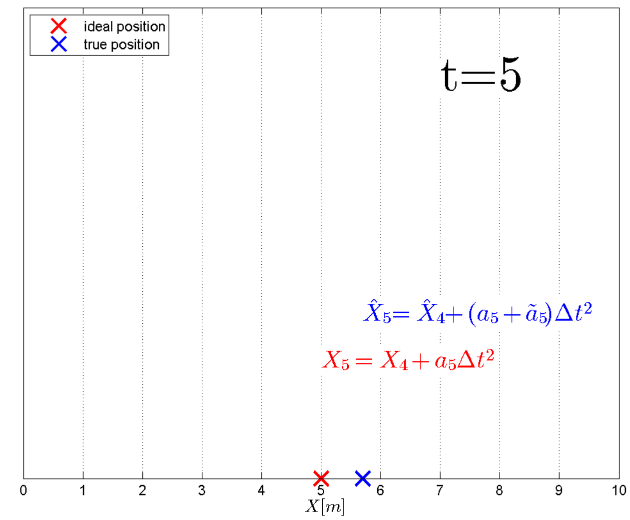
Odometry in 1D

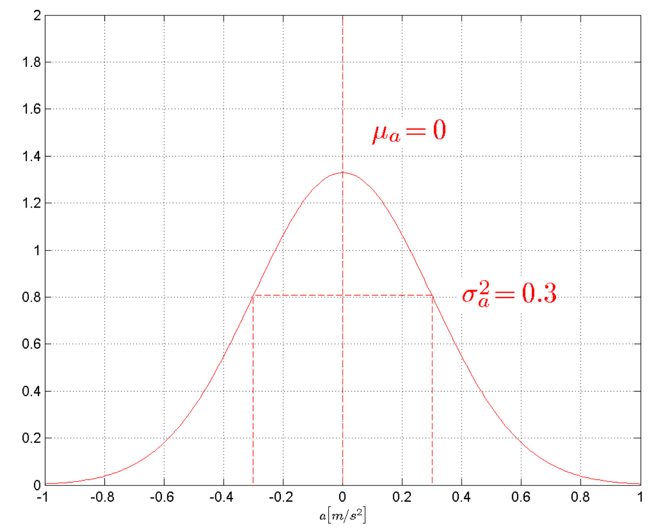




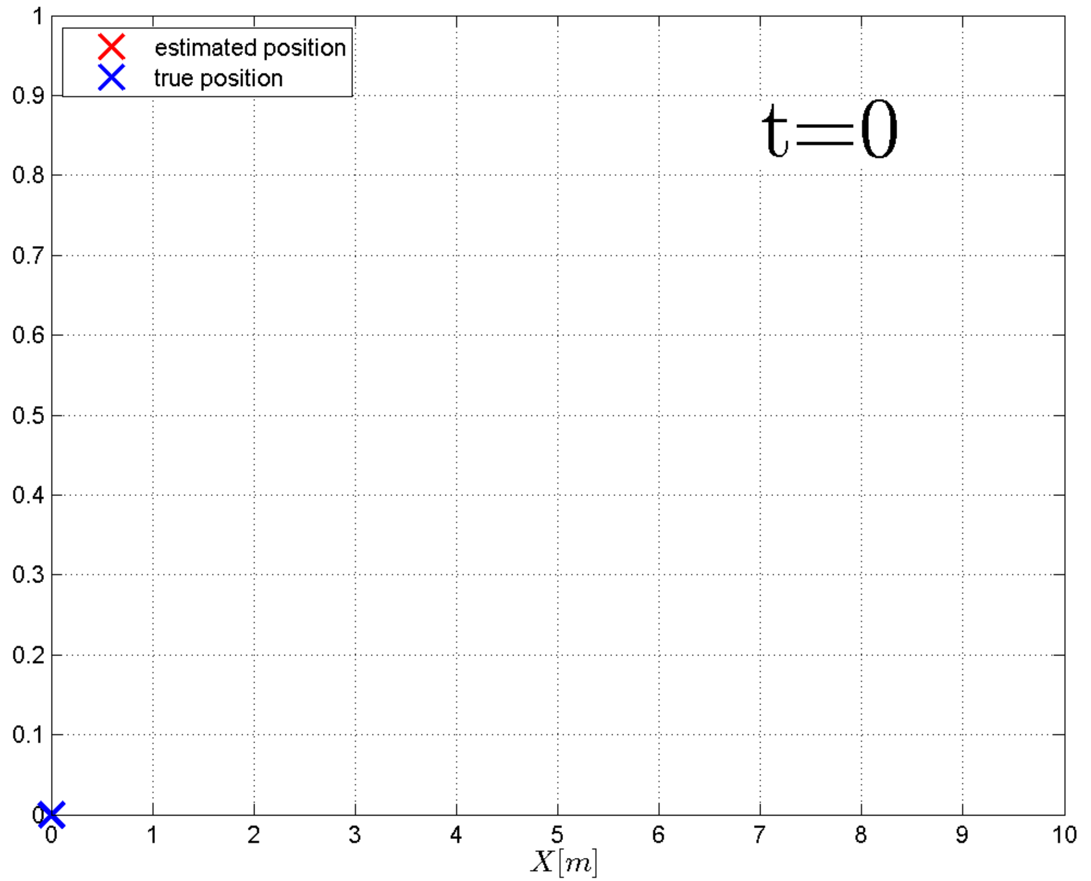
1D Odometry: Error Modeling

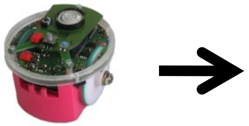
- Error happens!
- Odometry error is cumulative.
  → grows without bound
- We need to be aware of it.
  → We need to model odometry error.
  → We need to model sensor error.
- Multiple independent source of errors with arbitrary distribution combined → Central Limit Theorem → Gaussian assumption reasonable
- Acceleration is random variable A drawn from "mean-free" Gaussian ("Normal") distribution.
  → Position X is random variable with Gaussian distribution.



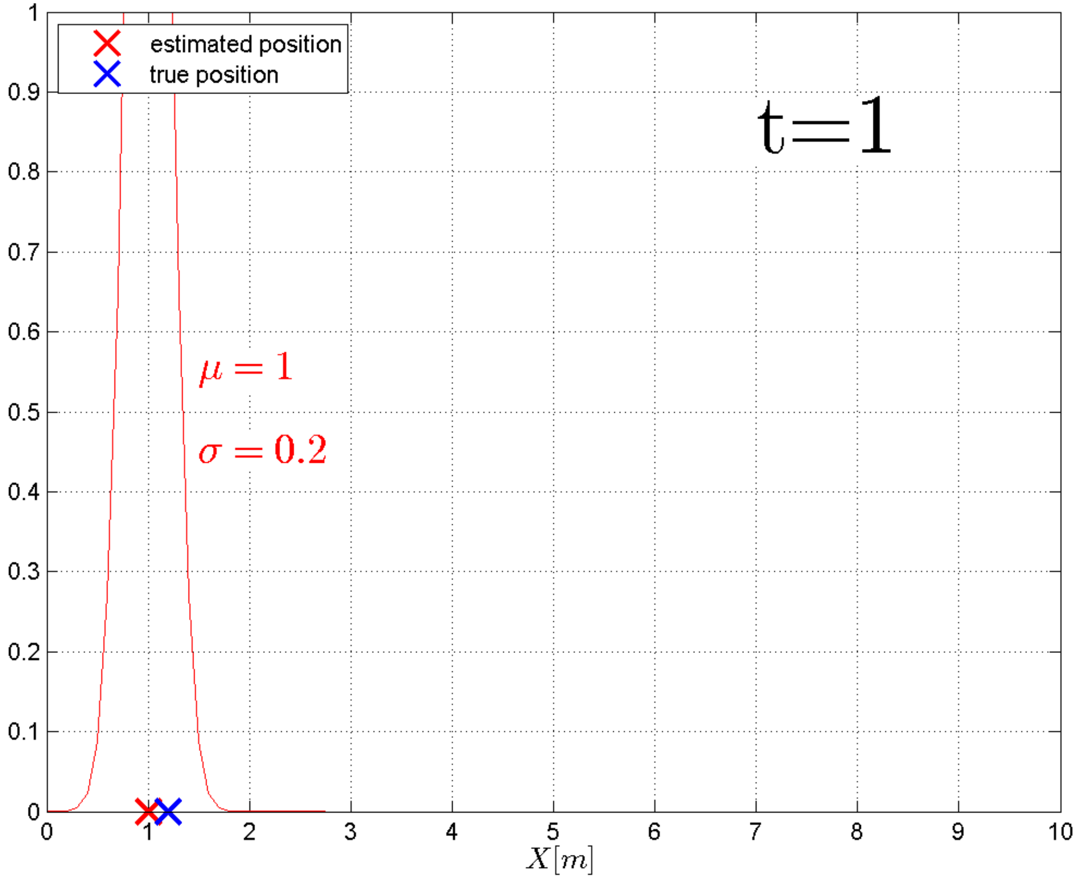


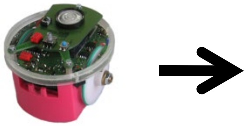
1D Odometry with Gaussian Uncertainty



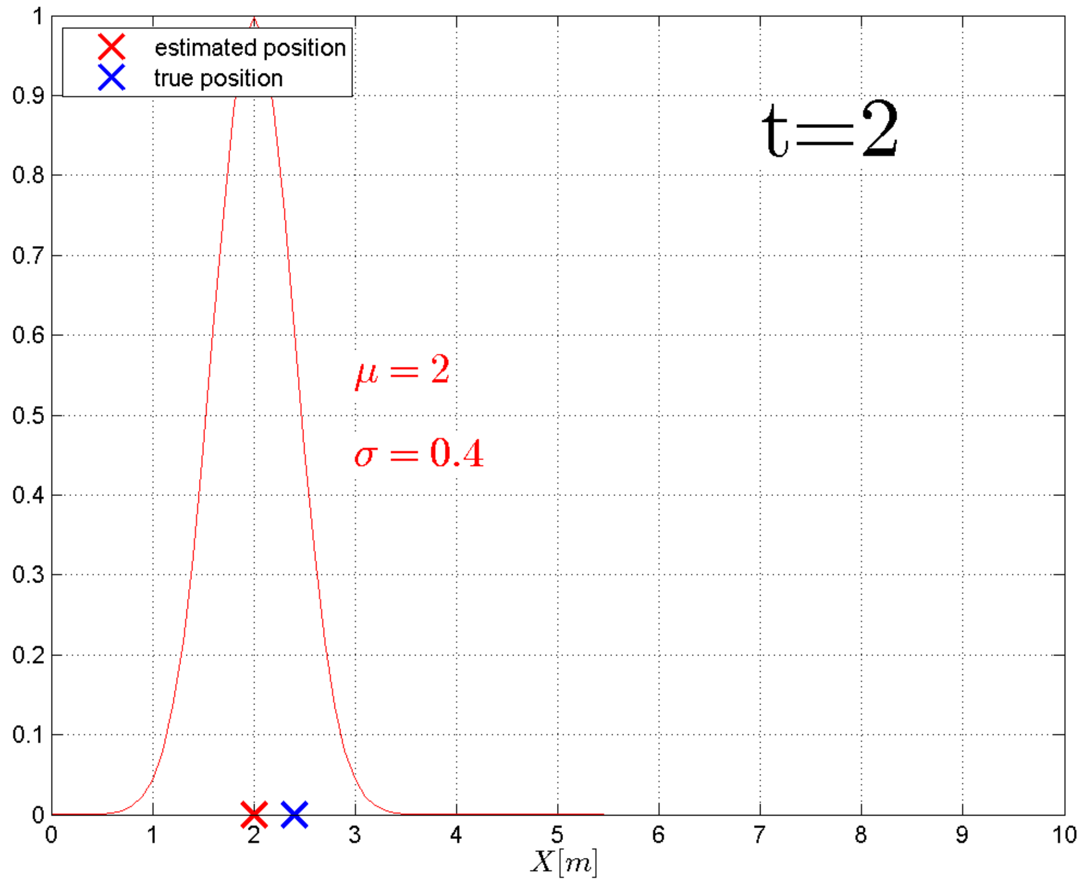


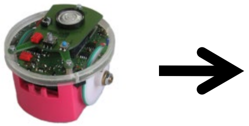
1D Odometry with Gaussian Uncertainty



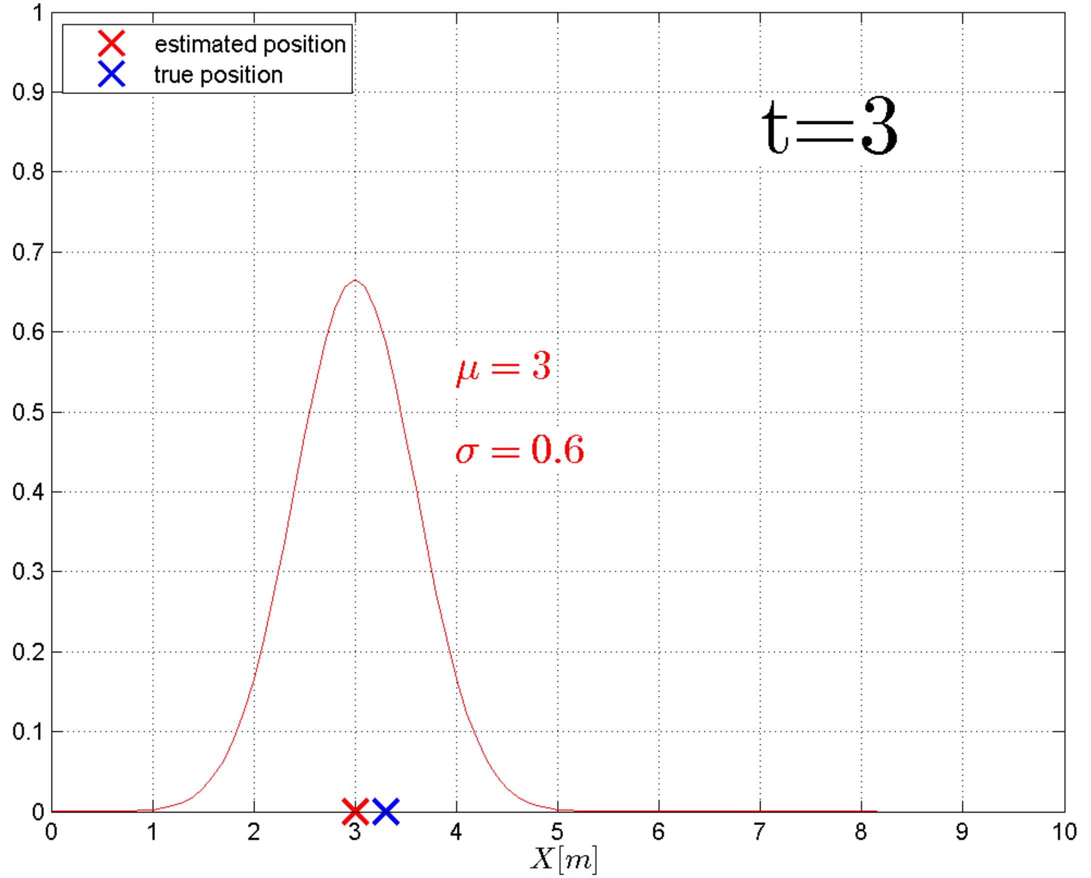


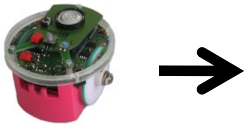
1D Odometry with Gaussian Uncertainty



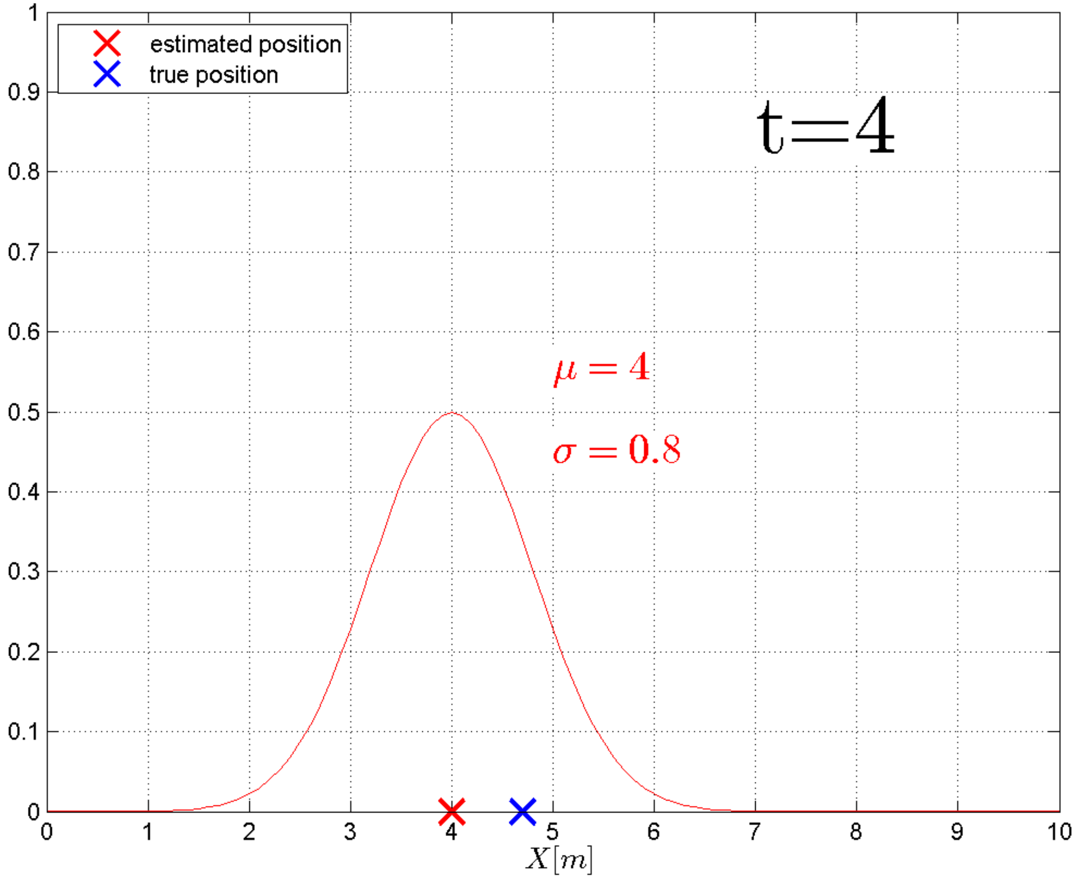


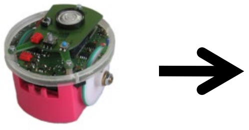
1D Odometry with Gaussian Uncertainty



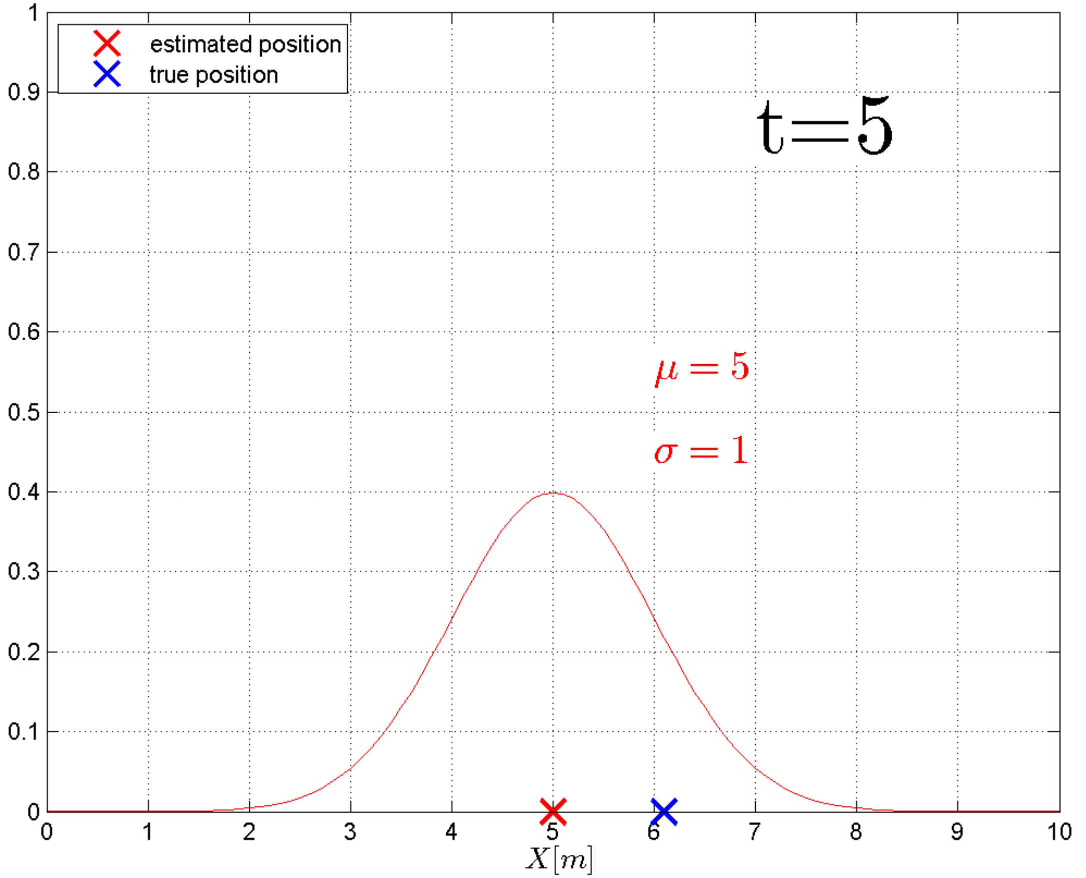


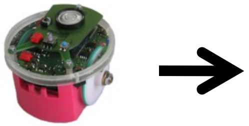
1D Odometry with Gaussian Uncertainty





1D Odometry with Gaussian Uncertainty



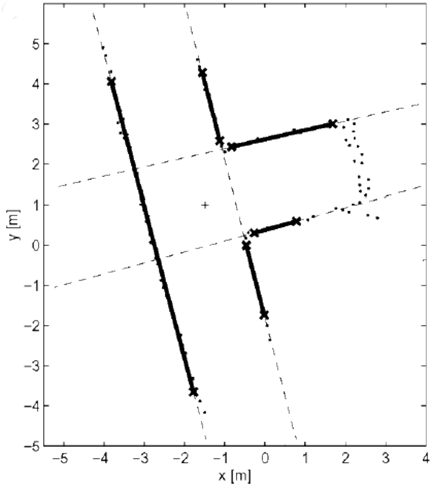


Features

- Odometry based position error grows without bound.
- Use relative measurement to features (“landmarks”) to reduce position uncertainty
- ***Feature***:
  - Uniquely identifiable
  - Position is known
  - We can obtain relative measurements between robot and feature (usually angle or range).
- Examples:
  - Doors, walls, corners, hand rails
  - Buildings, trees, lanes
  - GPS satellites

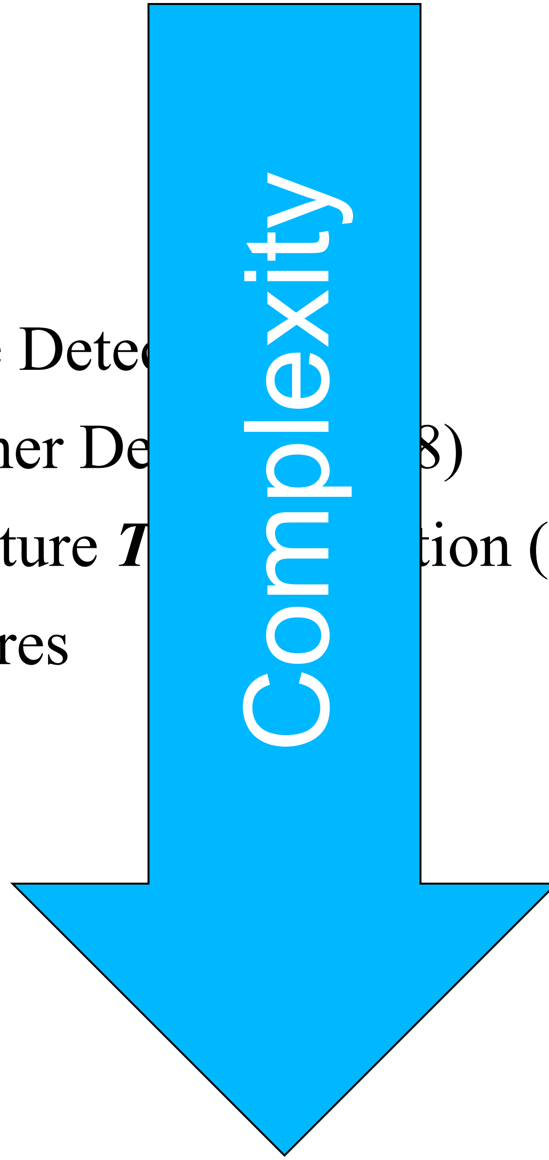
Automatic Feature Extraction

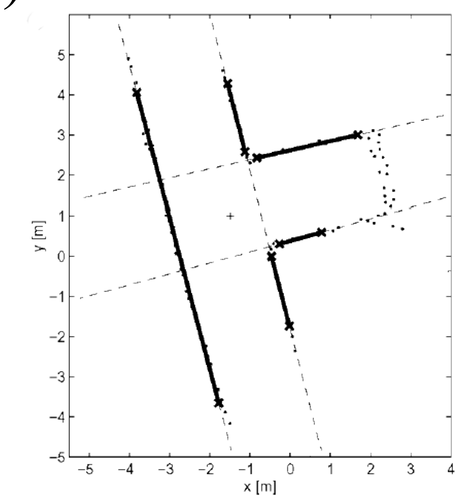
- High level features:
  - Doors, persons
- Simple visual features:
  - Edges (Canny Edge Detector 1983)
  - Corner (Harris Corner Detector 1988)
  - ***S***cale ***I***nvariant ***F***eature ***T***ransformation (2004)
- Simple geometric features
  - Lines
  - Corners
- "Binary" feature



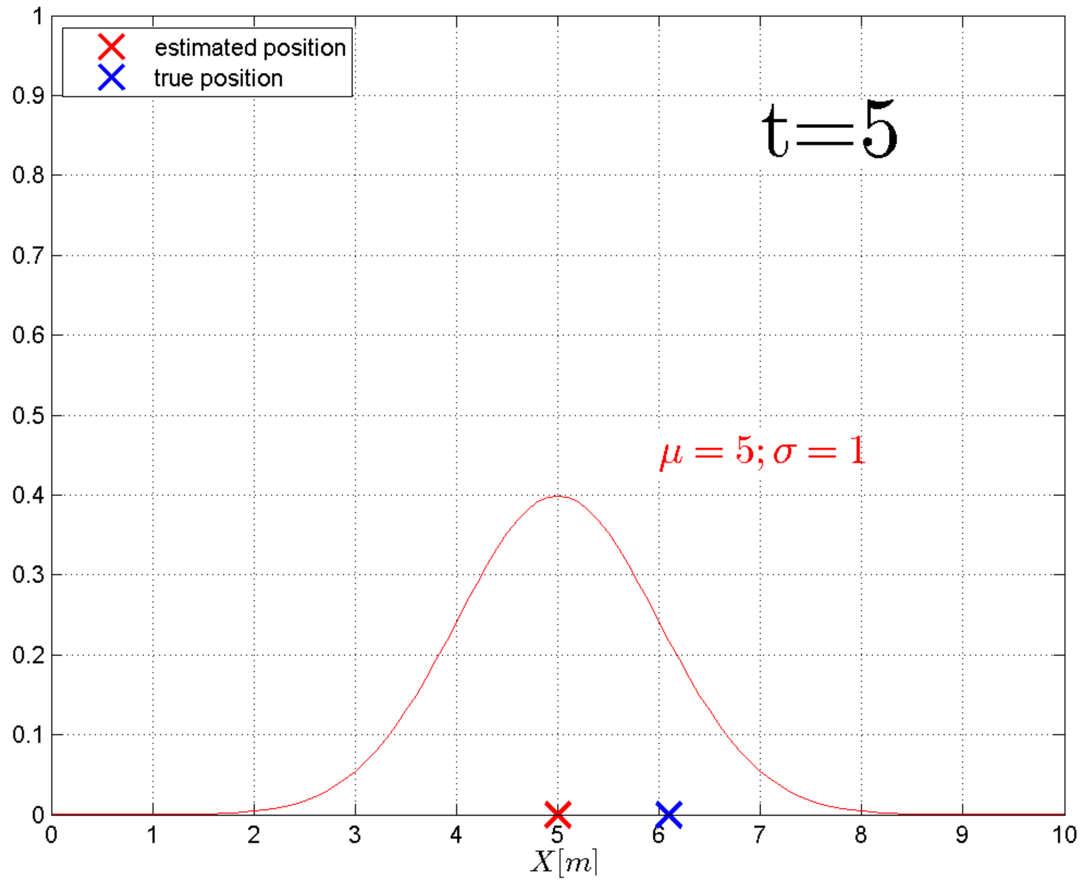
Automatic Feature Extraction

- High level features:
  - Doors, persons
- Simple visual features:
  - Edges (Canny Edge Detec
  - Corner (Harris Corner De 8)
  - ***S***cale ***I***nvariant ***F***eature ***T*** tion (2004)
- Simple geometric features
  - Lines
  - Corners
- “Binary” feature

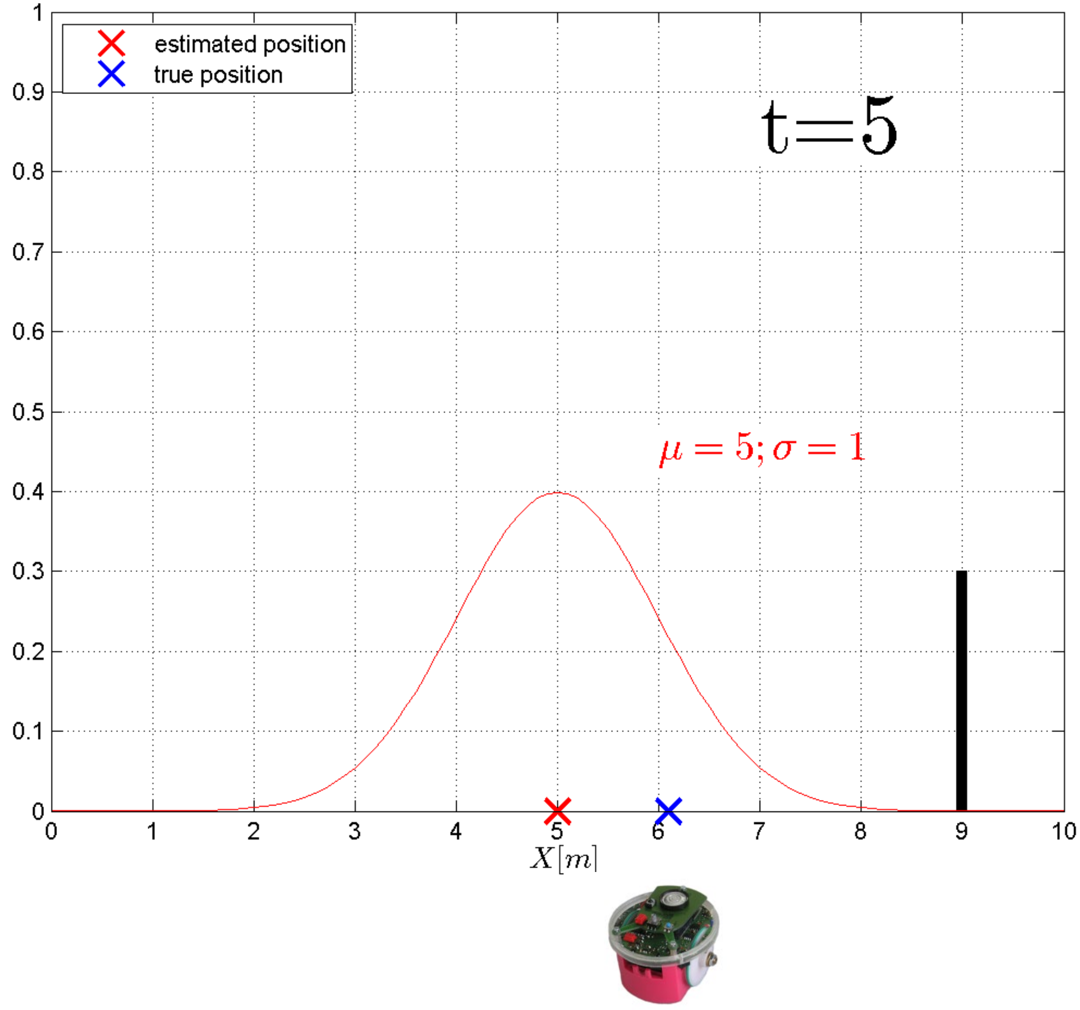




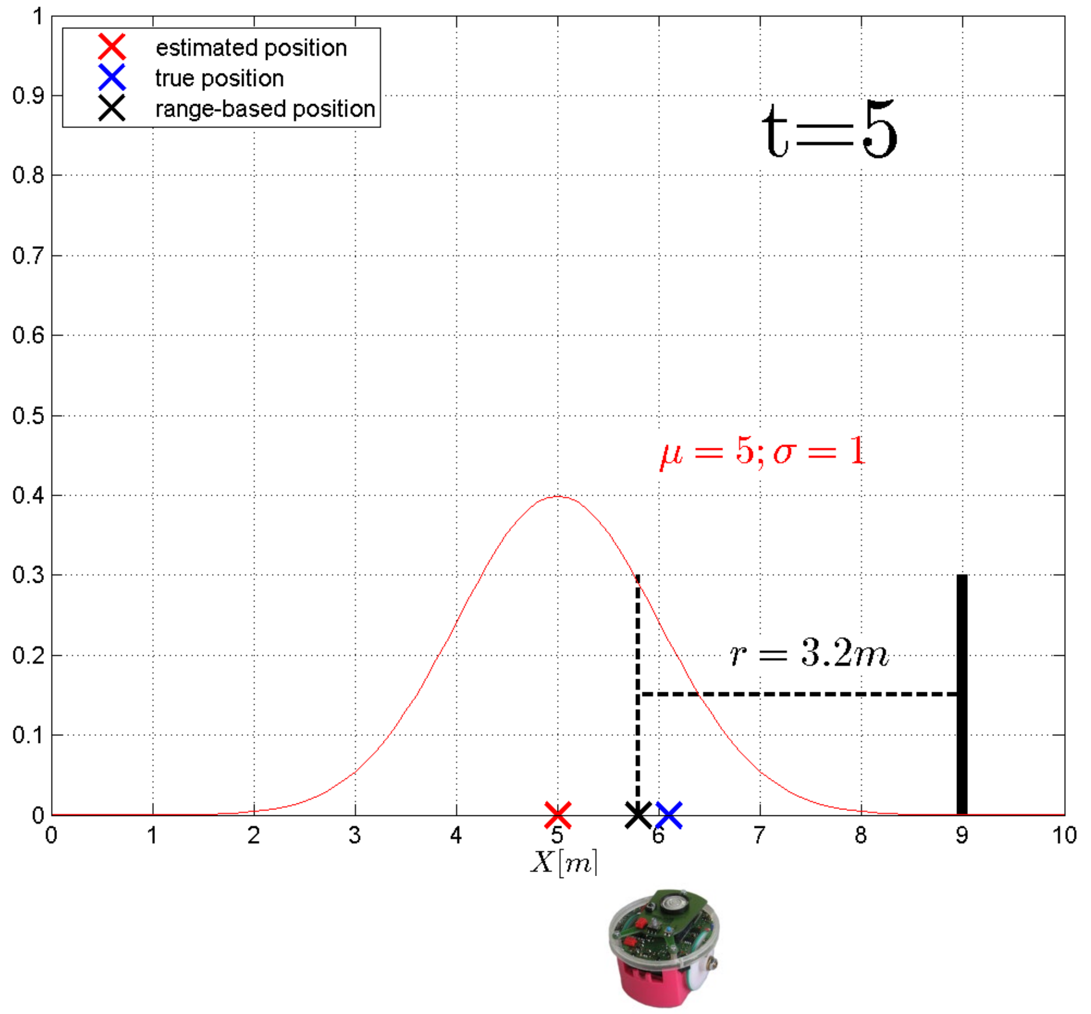
Feature-Based Localization



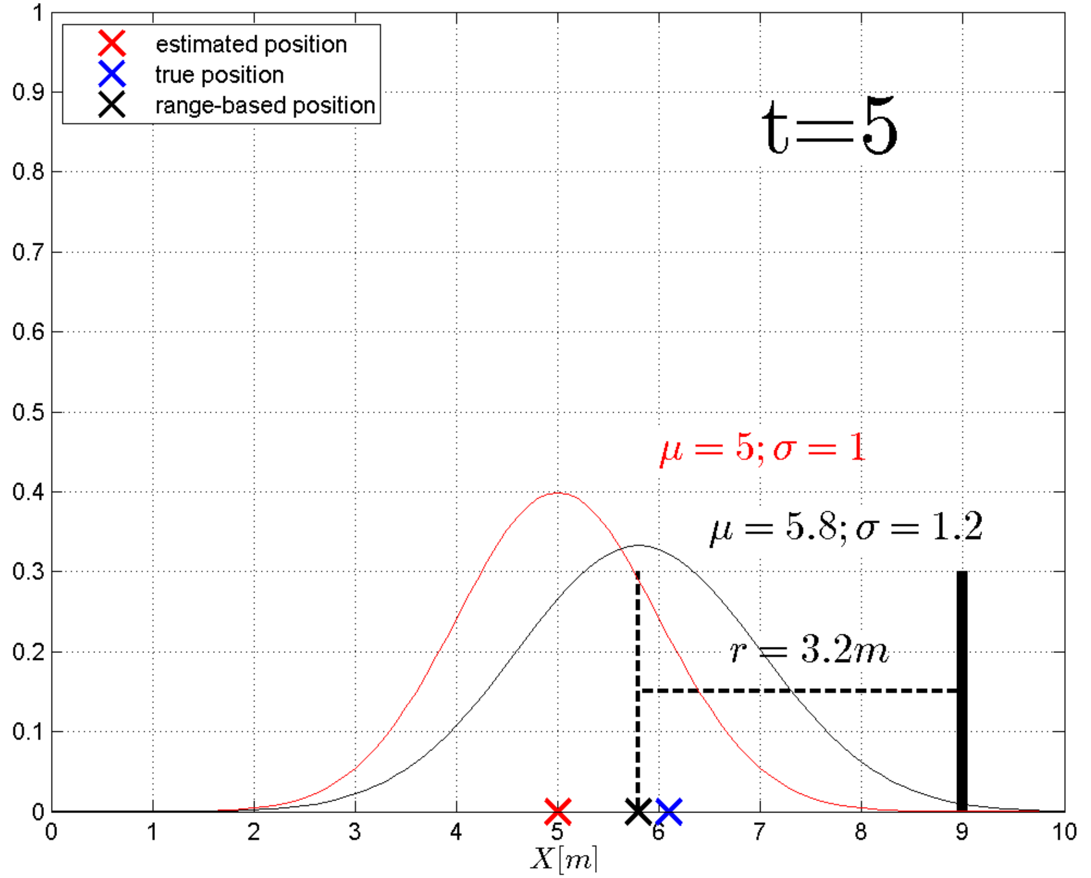
Feature-Based Localization



Feature-Based Localization



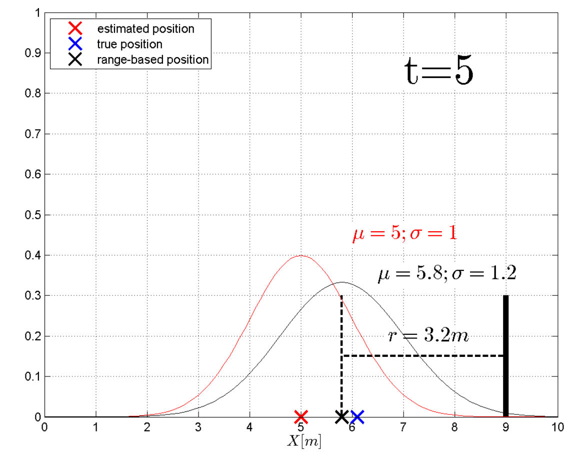
Feature-Based Localization



Sensor Fusion

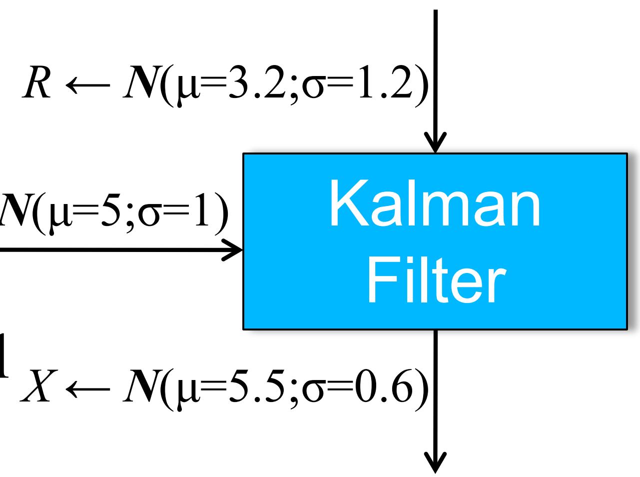
- Given:
  - Position estimate $\underline{X} \leftarrow N(\mu=5;\sigma=1)$
  - Range estimate $R \leftarrow N(\mu=3.2;\sigma=1.2)$

    What is the best estimate AFTER incorporating r ?

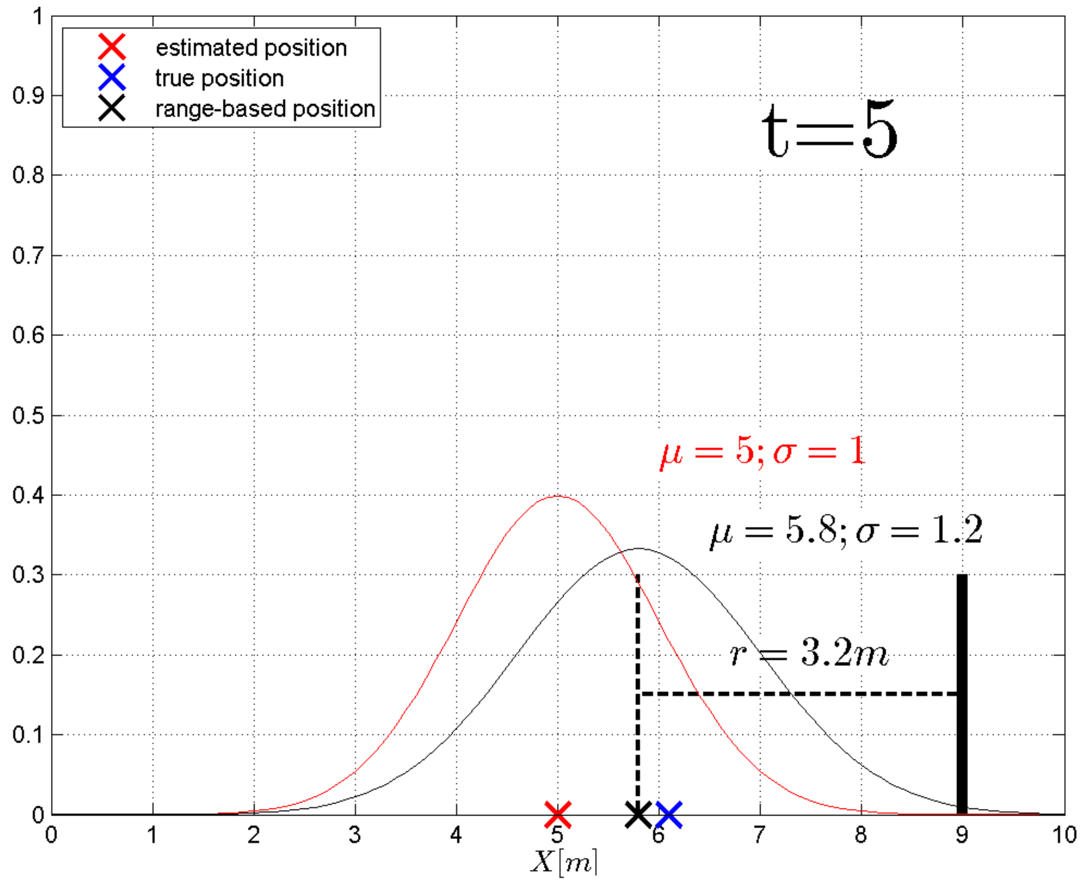


→ ***Kalman Filter***

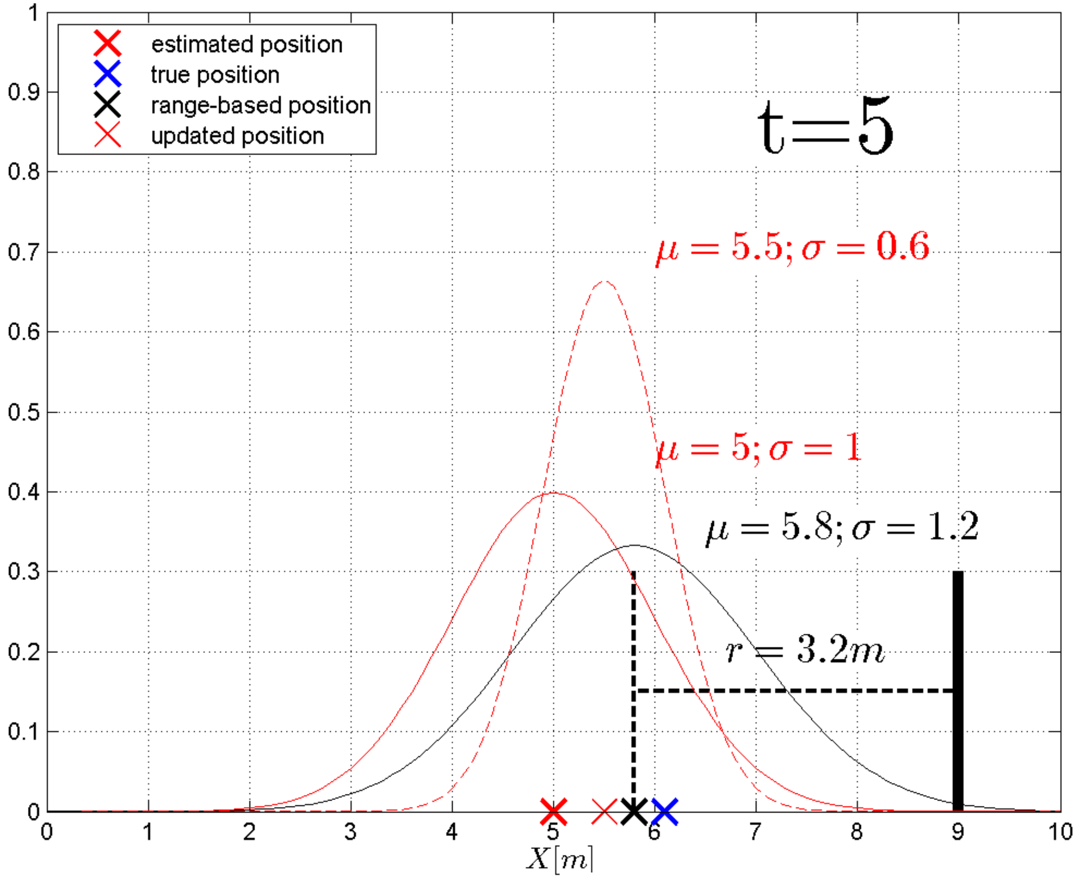
- Requires:
  - White Gaussian noise distribution for all measurements
  - Linear motion and measurement model
  - …



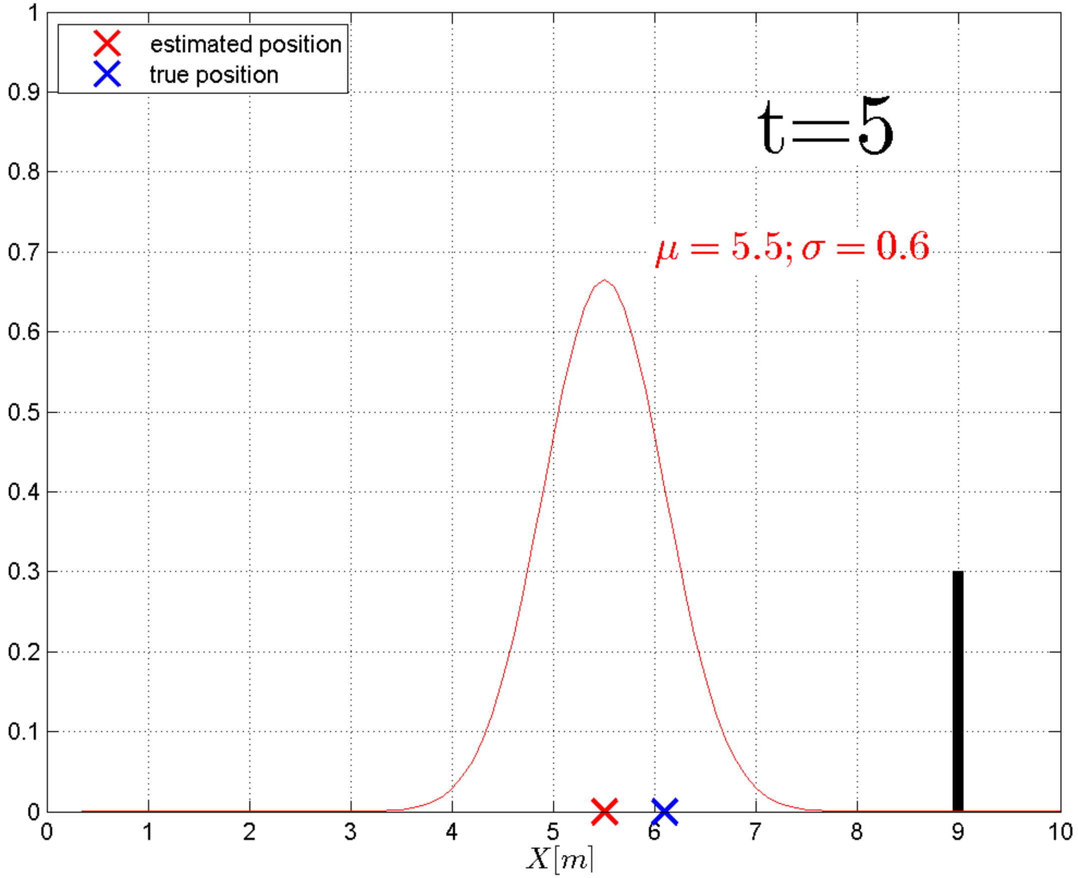
Feature-Based Localization



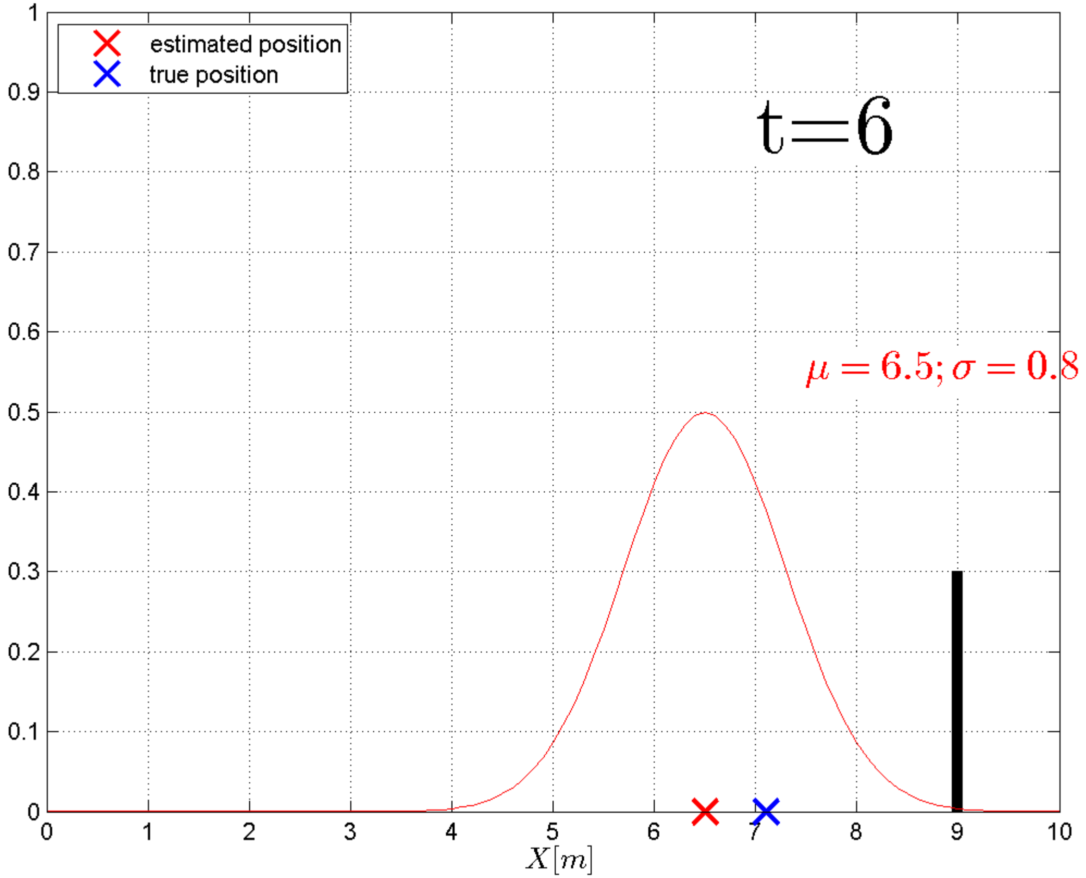
Feature-Based Localization



Feature-Based Localization

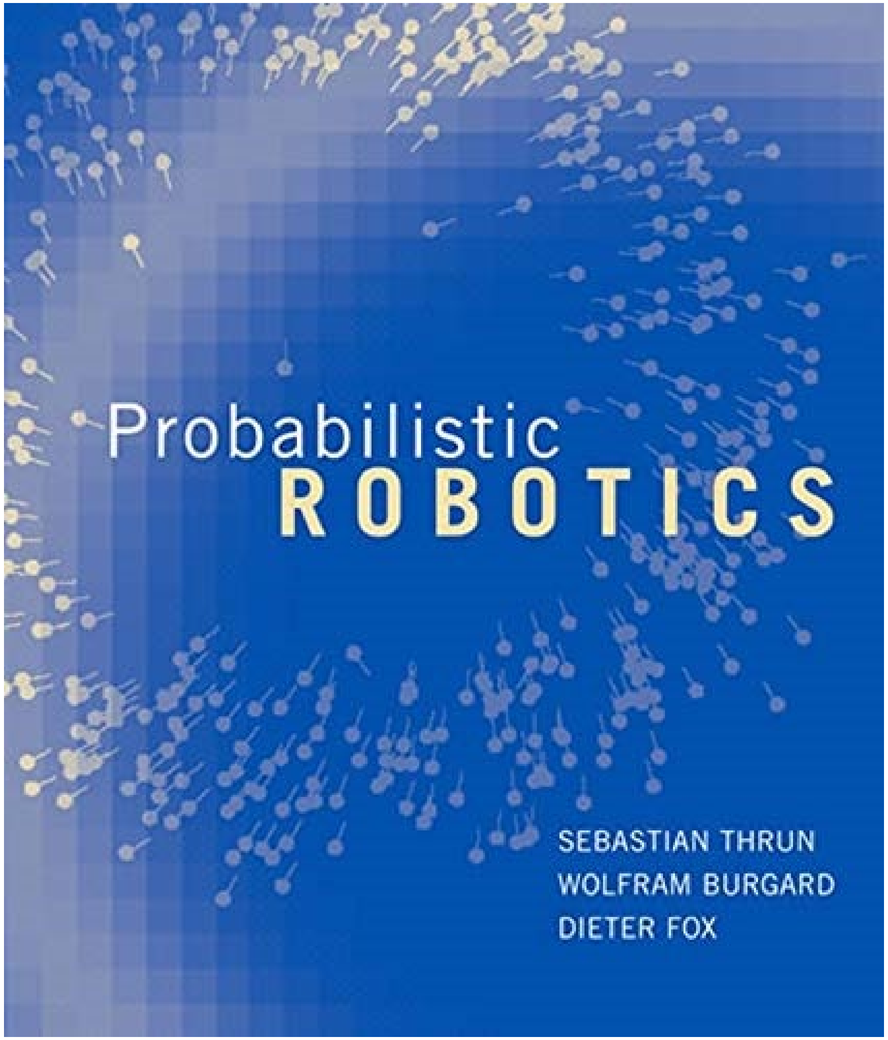


Feature-Based Localization



The Multi-Dimensional Kalman Filter Algorithm and its Application to Localization

Two Key Sources of Information



Multi-Dimensional Kalman Filter

Estimates the state x of a discrete-time controlled process that is governed by the linear stochastic difference equation

$$x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t$$

with a measurement

$$z_t = C_t x_t + \delta_t$$

[Adapted from Thrun et al., 2005]

Components of a Kalman Filter

A_t Matrix (nxn) that describes how the state evolves from *t* to *t-1* without controls or noise.

B_t Matrix (nxl) that describes how the control u_t changes the state from *t* to *t-1*.

C_t Matrix (kxn) that describes how to map the state x_t to an observation z_t.

ε_t

δ_t

Random variables representing the process and measurement noise that are assumed to be independent and normally distributed with covariance R_t and Q_t respectively.

[Adapted from Thrun et al., 2005]

Kalman Filter Algorithm

Algorithm **Kalman_filter**(μ_{t-1}, Σ_{t-1}, u_t, z_t)

1. Prediction:
2. $\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$
3. $\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$
4. Correction or update:
5. $K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1}$
6. $\mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$
7. $\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$
8. Return μ_t, Σ_t

[Adapted from Thrun et al., 2005]

Mitigating Localization Uncertainities in Odometry Through Exteroceptive Sensors – The 2D Case

Nondeterministic Error Sources

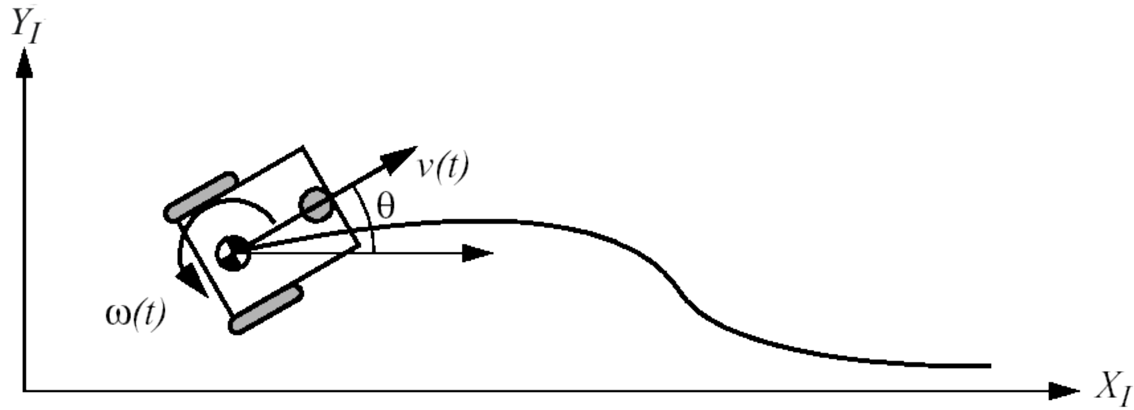
- Variation of the contact point of the wheel
- Unequal floor contact (e.g., wheel slip, nonplanar surface)

- Wheels cannot be assumed to roll perfectly
- Measured encoder values do not perfectly reflect the actual motion
- Pose error is cumulative and incrementally increases
- Probabilistic modeling for assessing quantitatively the error

Pose Variation During Δt

$$\Delta s = \frac{\Delta s_r + \Delta s_l}{2}$$

$$\begin{cases} \Delta x = \Delta s \cos(\theta + \frac{\Delta\theta}{2}) \\ \Delta y = \Delta s \sin(\theta + \frac{\Delta\theta}{2}) \\ \Delta\theta = \frac{\Delta s_r - \Delta s_l}{b} \end{cases}$$



$$p = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \xrightarrow{t'=t+\Delta t} p' = \begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix}$$

b = 2l = inter-wheel distance

Δs_r = traveled distance right wheel

Δs_l = traveled distance left wheel

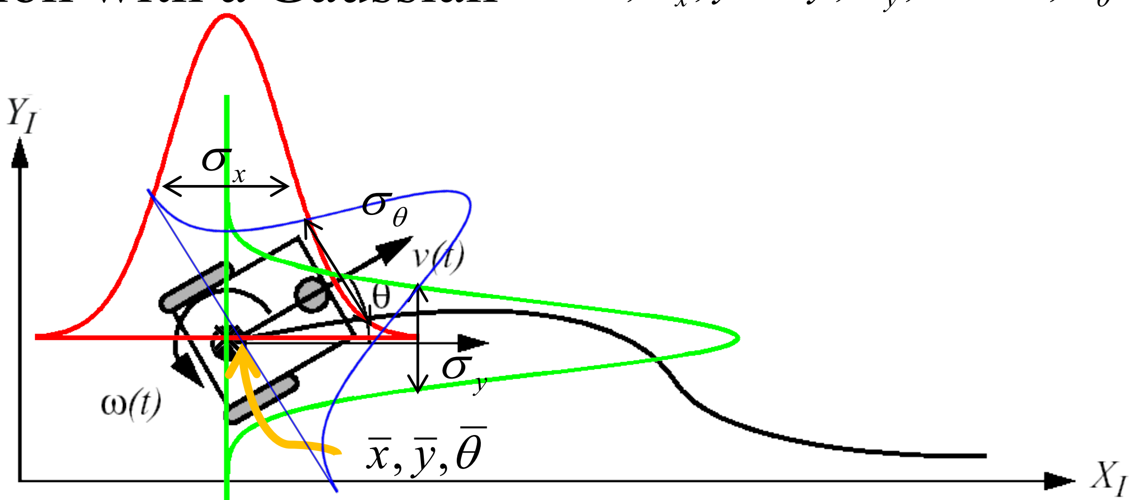
Δθ = orientation change of the vehicle

$$p' = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \Delta s \cos(\theta + \Delta\theta/2) \\ \Delta s \sin(\theta + \Delta\theta/2) \\ \Delta\theta \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \frac{\Delta s_r + \Delta s_l}{2} \cos(\theta + \frac{\Delta s_r - \Delta s_l}{2b}) \\ \frac{\Delta s_r + \Delta s_l}{2} \sin(\theta + \frac{\Delta s_r - \Delta s_l}{2b}) \\ \frac{\Delta s_r - \Delta s_l}{b} \end{bmatrix}$$

Noise modeling

Model error in each dimension with a Gaussian $x \rightarrow \bar{x}, \sigma_x; y \rightarrow \bar{y}, \sigma_y; \theta \rightarrow \bar{\theta}, \sigma_\theta$

$$\Sigma_p = \begin{bmatrix} \sigma_{xx}^2 & \sigma_{xy}^2 & \sigma_{x\theta}^2 \\ \sigma_{yx}^2 & \sigma_{yy}^2 & \sigma_{y\theta}^2 \\ \sigma_{\theta x}^2 & \sigma_{\theta y}^2 & \sigma_{\theta\theta}^2 \end{bmatrix}$$



Assumptions:

- Covariance matrix Σ_p at the beginning is known
- Errors of the two individual wheels are independent
- Errors are independent of direction of motion
- Errors are proportional to the distance traveled (k_r, k_l model parameters)

$$\sum_\Delta = \mathrm{cov}(\Delta s_r, \Delta s_l) = \begin{bmatrix} k_r|\Delta s_r| & 0 \\ 0 & k_l|\Delta s_l| \end{bmatrix} = \begin{bmatrix} \sigma_{s_r}^2 & 0 \\ 0 & \sigma_{s_l}^2 \end{bmatrix}$$

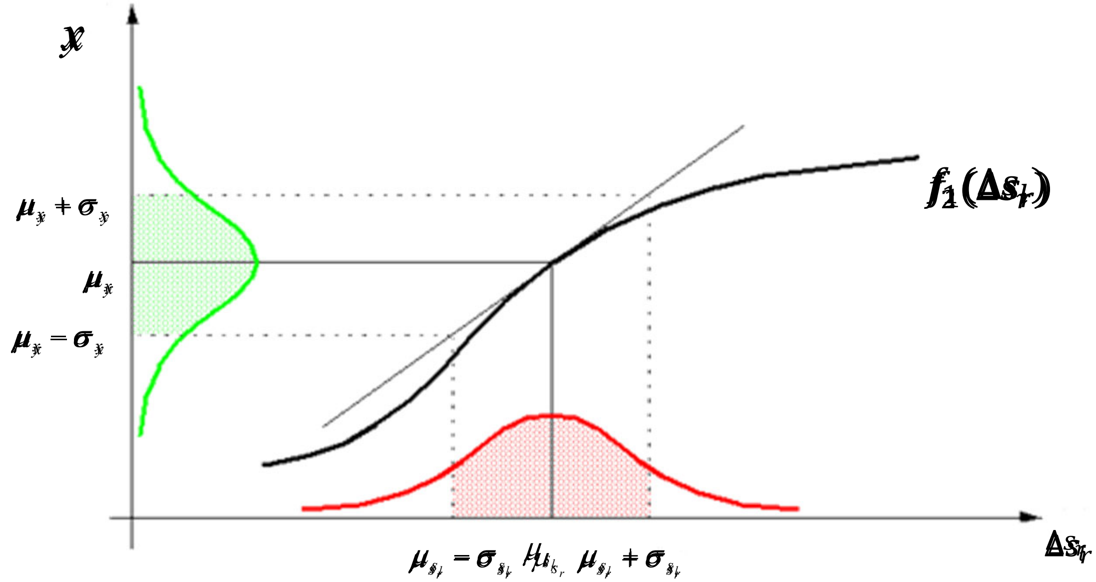
Actuator Noise → Pose Noise

$$x \approx f_1(\Delta s_r)\Big|_{\Delta s_r=\mu_{s_r}} \approx f_1(\Delta s_r) + \frac{\partial f_1}{\partial \Delta s_r}(\Delta s_r - \mu_{s_r})$$

$$x \approx f_1(\Delta s_l)\Big|_{\Delta s_l=\mu_{s_l}} \approx f_1(\Delta s_l) + \frac{\partial f_1}{\partial \Delta s_l}(\Delta s_l - \mu_{s_l})$$

$$y \approx f_2(\Delta s_r)\Big|_{\Delta s_r=\mu_{s_r}} \approx f_2(\Delta s_r) + \frac{\partial f_2}{\partial \Delta s_r}(\Delta s_r - \mu_{s_r})$$

...



$$F_{\Delta rl} = \begin{bmatrix} \frac{\partial f_1}{\partial \Delta s_r} & \frac{\partial f_1}{\partial \Delta s_l} \\ \frac{\partial f_2}{\partial \Delta s_r} & \frac{\partial f_2}{\partial \Delta s_l} \\ \frac{\partial f_3}{\partial \Delta s_r} & \frac{\partial f_3}{\partial \Delta s_l} \end{bmatrix}$$ Jacobian

$$\Sigma_\Delta = \begin{bmatrix} k_r|\Delta s_r| & 0 \\ 0 & k_l|\Delta s_l| \end{bmatrix}$$

- General error propagation law

$$\Sigma_{\Delta rl} = F_{\Delta rl} \Sigma_\Delta F_{\Delta rl}^T$$

Actuator Noise → Pose Noise

Algorithm

Precompute:

- Determine actuator noise Σ_Δ
- Compute mapping actuator-to-pose noise incremental $F_{\Delta rl}$
- Compute mapping pose propagation noise over step F_p

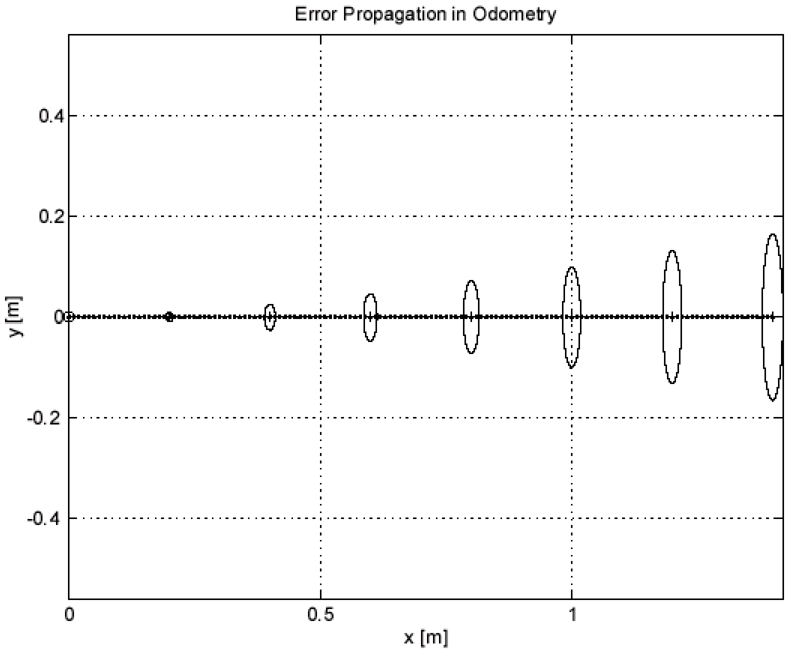
Initialize:

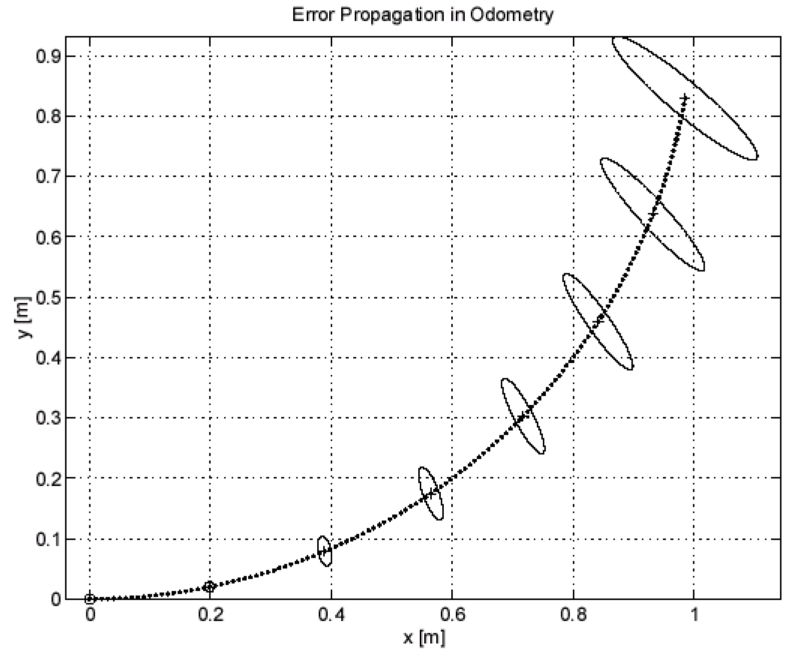
- Initialize $\Sigma_p^{(t=0)} = [0]$

Iterate:

$$\Sigma_p^{(t=(k+1)\Delta t)} = F_p \Sigma_p^{(t=k\Delta t)} F_p^T + F_{\Delta rl} \Sigma_\Delta F_{\Delta rl}^T$$

Classical 2D Representation

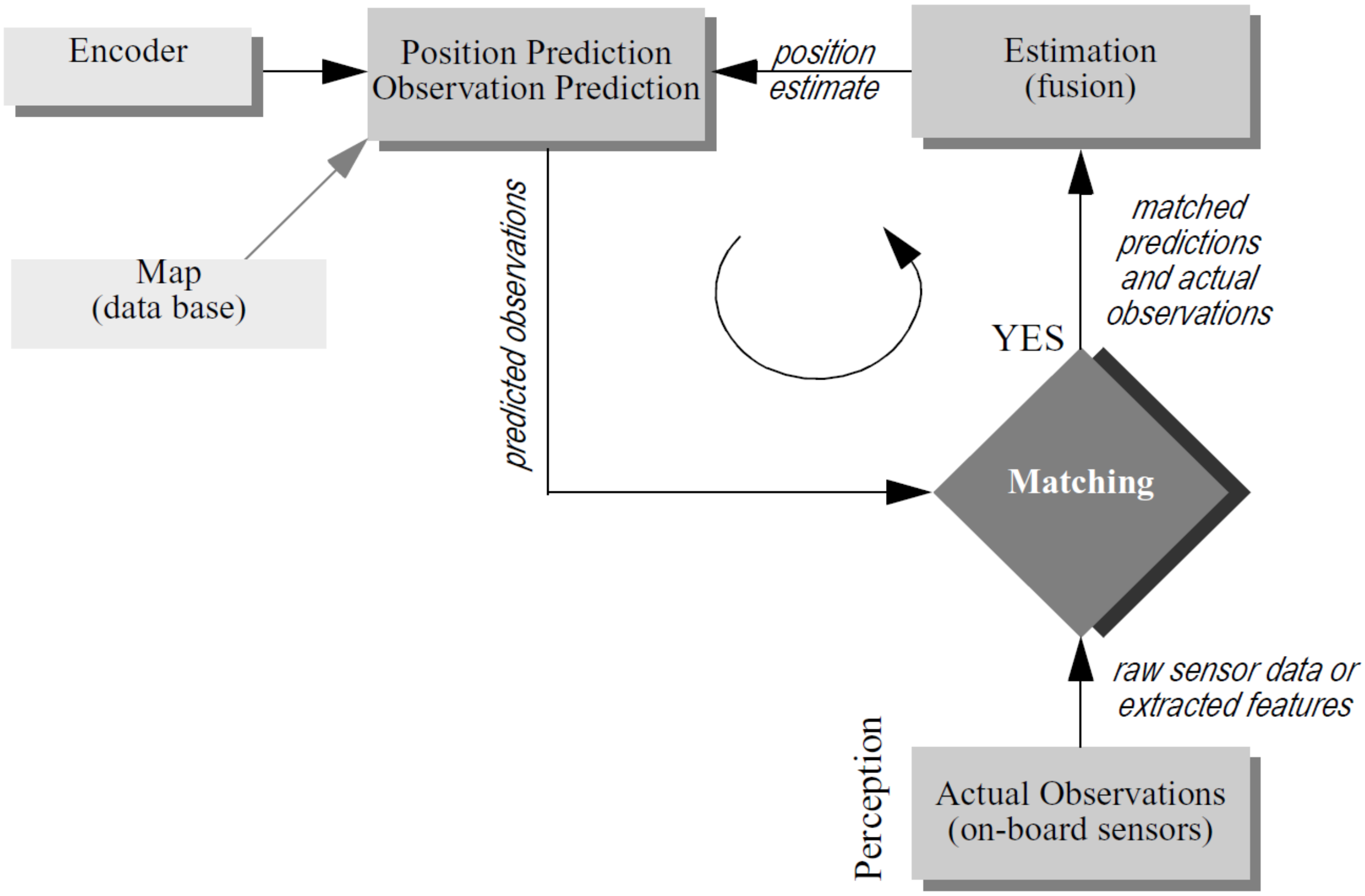




Courtesy of R. Siegwart and R. Nourbakhsh

Ellipses: typical 3σ bounds

2D Localization with Kalman Filter



[From Siegwart and Nourbakhsh, 2004]

A Five-Step Iterative Recipe

1. Robot pose prediction
2. Actual observations
3. Predicted observations
4. Matching
5. Estimation with Kalman filter

Robot Pose Prediction

- Input: motion model of the vehicle, odometry, control actions
- Output: estimation mean and co-variance of the pose and the next timestep (s. 40 and 43)

Actual Observations

- Input: sensor model
- Coordinate transformation needed for getting to the local frame (in this case sensor frame)
- Output: depending on the sensing technology/feature (e.g., distance, line, door)

Predicted Observations

- Input: predicted robot pose and map
- Coordinate transformation needed for getting to the local frame (in this case sensor frame)
- Output: predicted feature observations

Matching

- Input: predicted feature observations and actual observations, all in the sensor frame
- Objective: matching actual observations to predicted features
- Innovation metric: difference predicted and observed measurements (also random variable, with mean and co-variance)
- Validation criterion based on innovation metric
- Output: selection of features worth using in the estimation process

Estimation with Kalman Filter

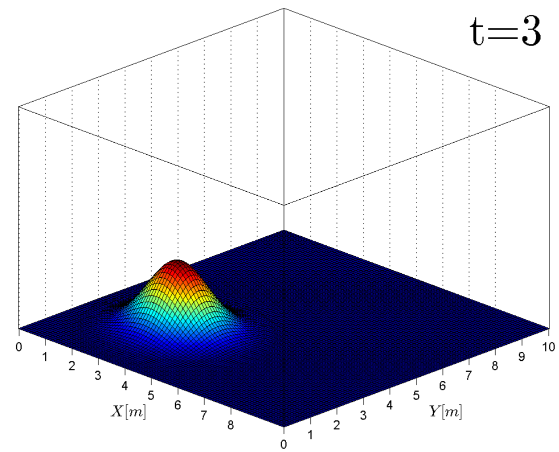
- Input: pose prediction from the motion model and validated observations
- Objective: fuse pose prediction from motion model and all validated observation to create next pose
- Output: next mean and variance of the pose

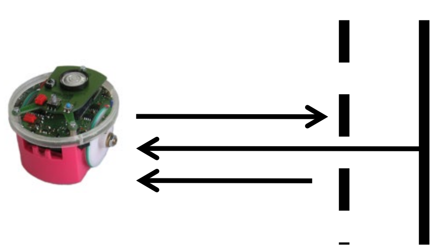
Working Around Kalman Filter Limitations: Particle Filters for Localization

Feature-Based Localization

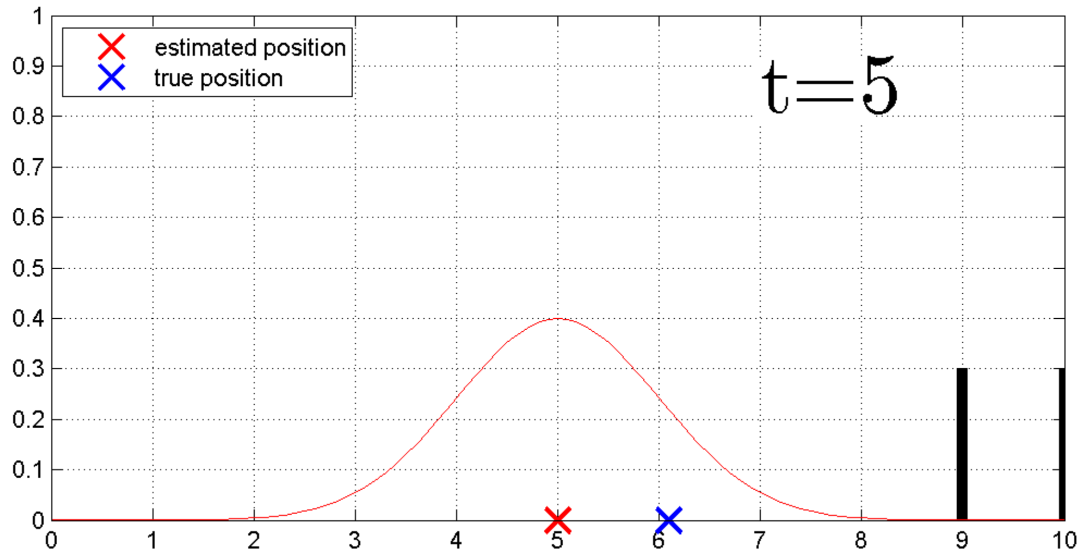
Belief representation through Gaussian distribution

- Advantages:
  - Compact (only mean and variance required)
  - Continuous
  - Powerful tools (Kalman Filter)
- Disadvantages:
  - Requires Gaussian noise assumption
  - Cannot represent ignorance (“kidnapped robot problem”)
  - Uni-modal
- Problematic example:

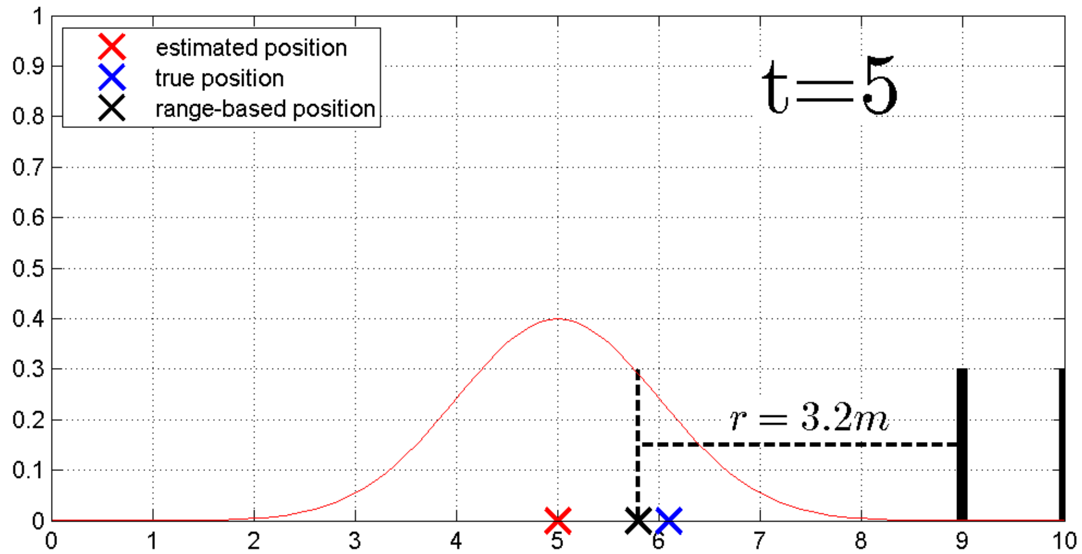




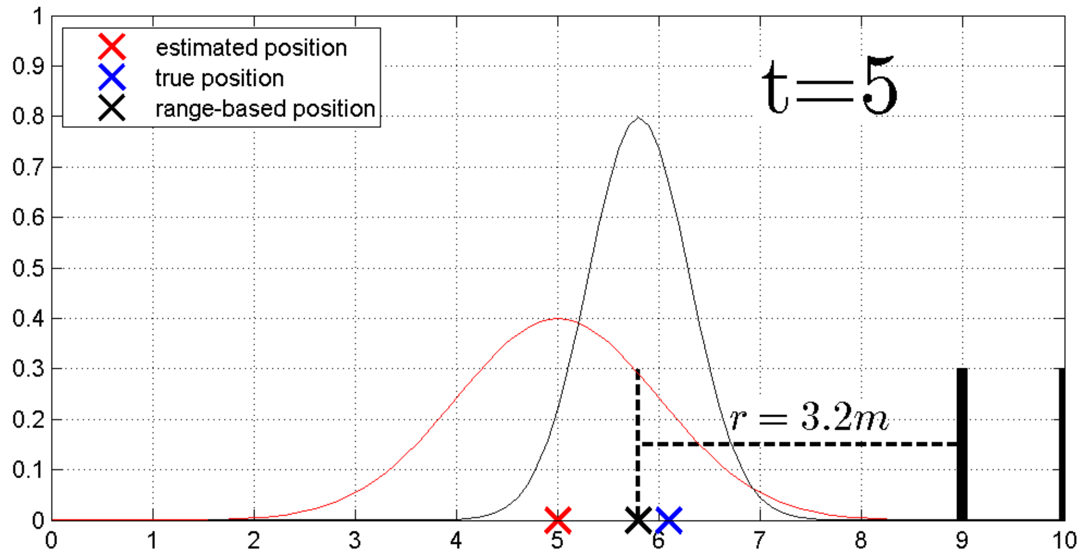
Particle Filter Localization



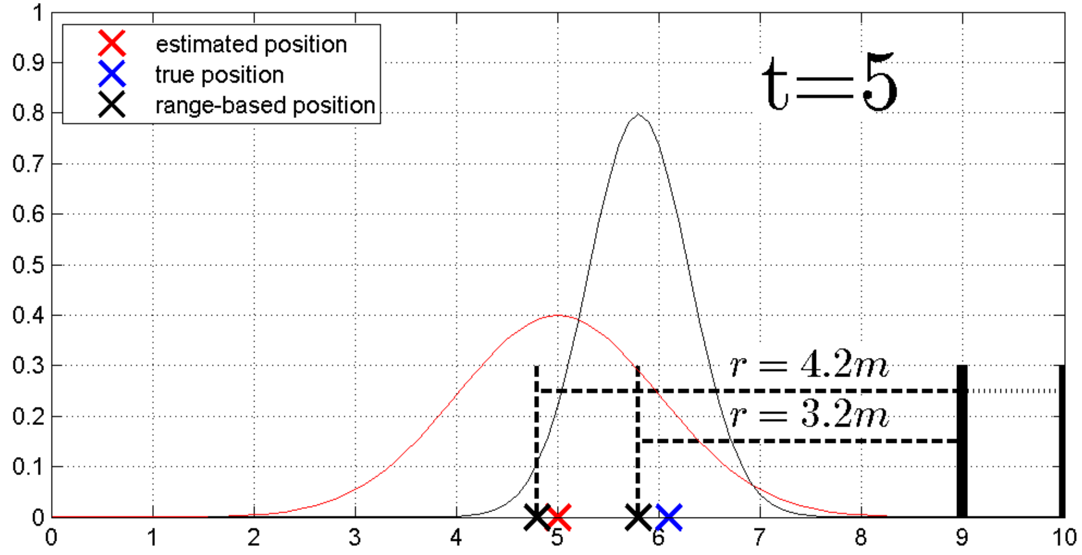
Particle Filter Localization



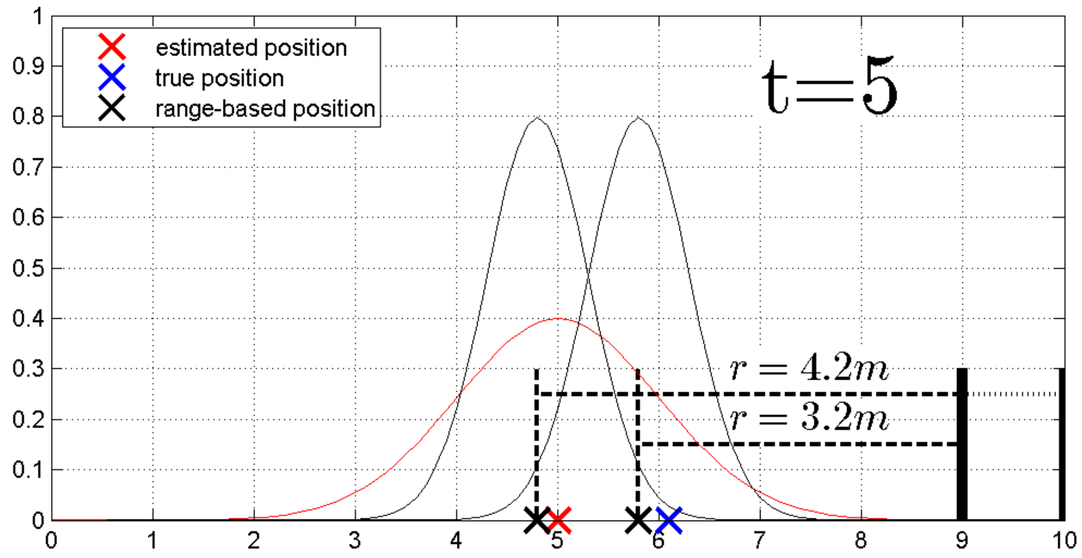
Particle Filter Localization



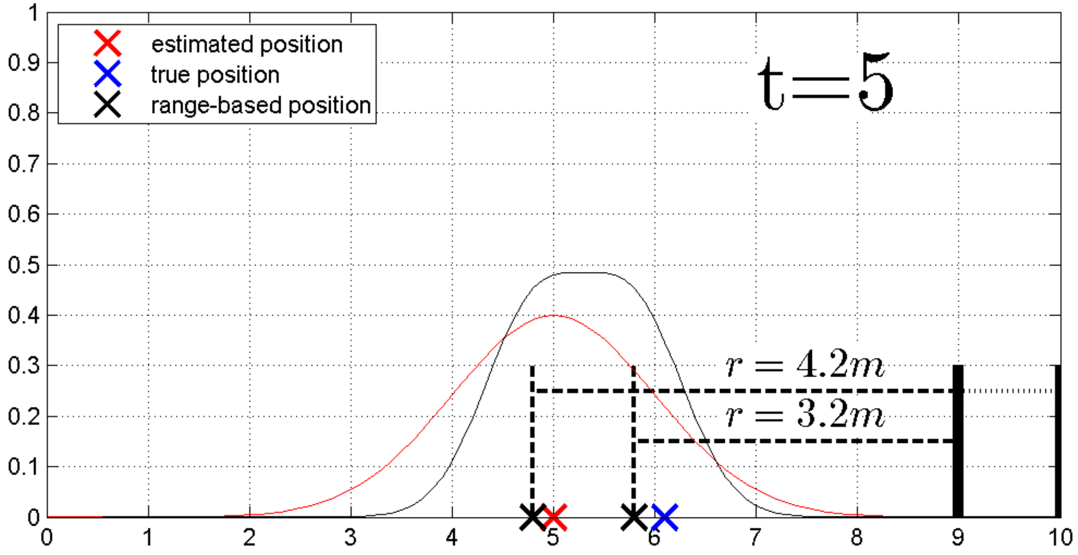
Particle Filter Localization



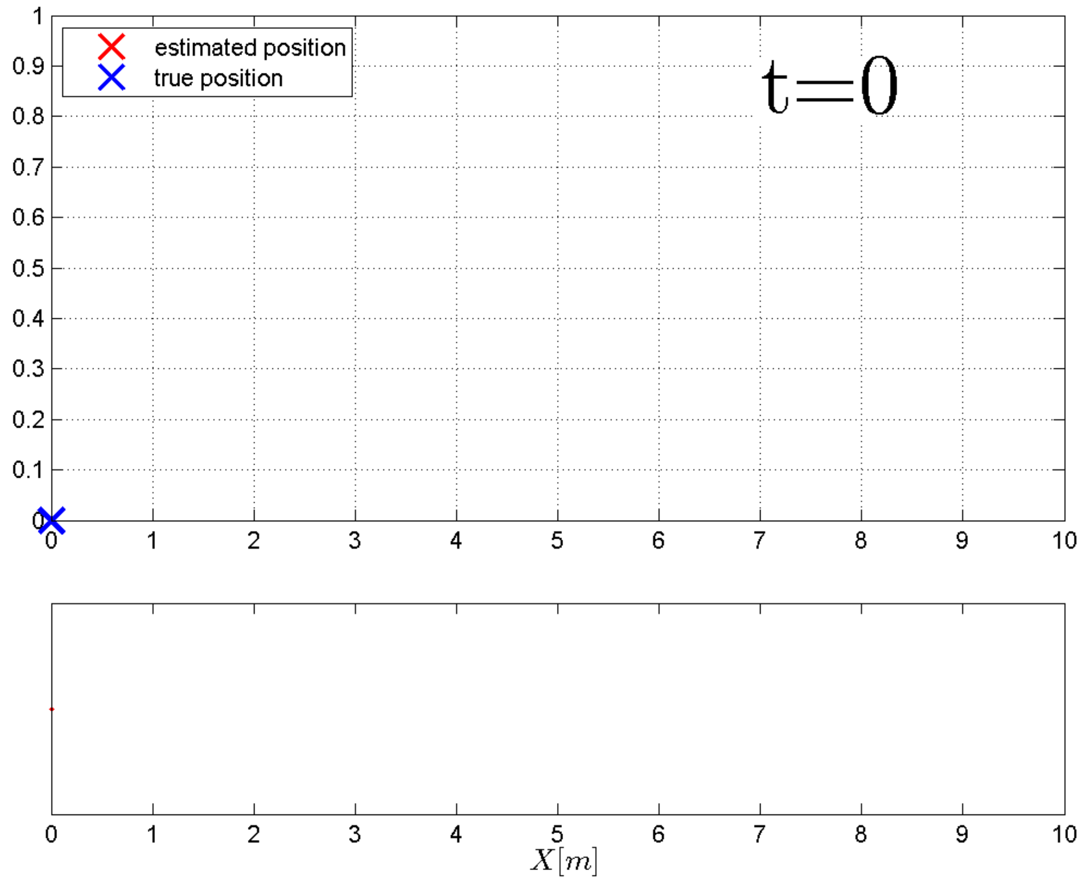
Particle Filter Localization



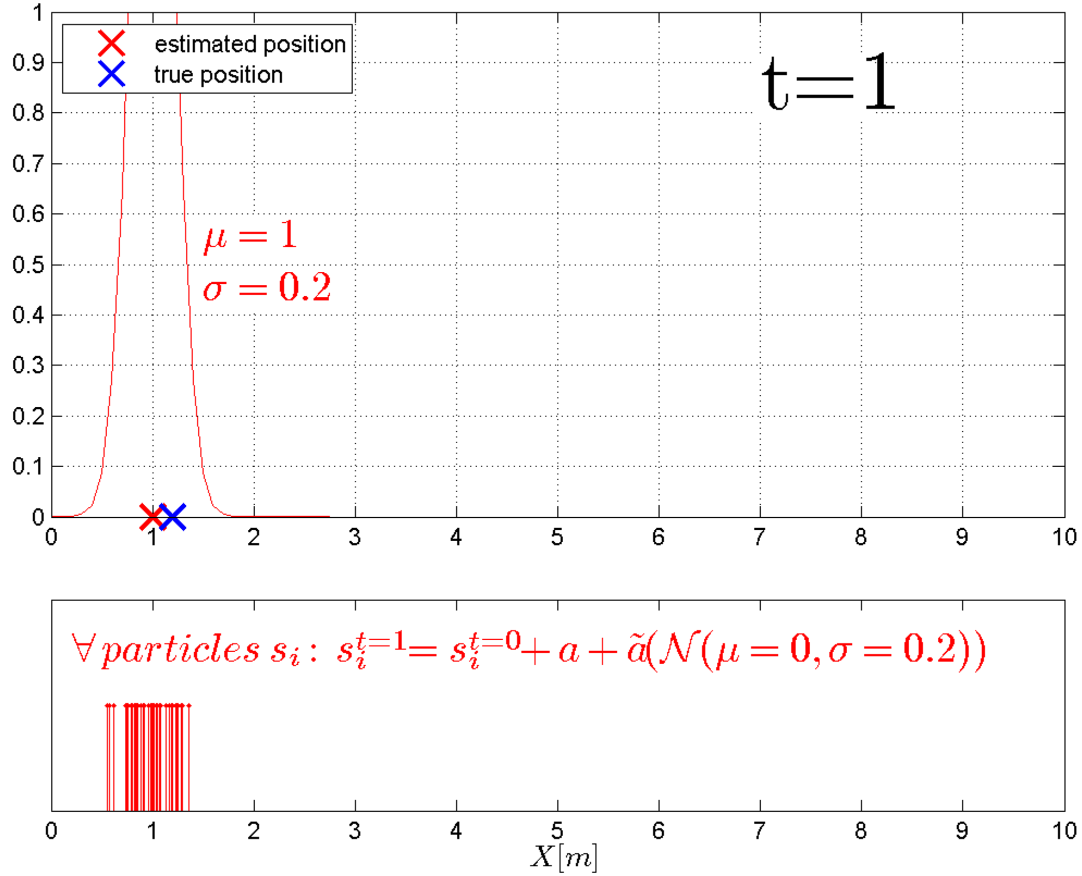
Particle Filter Localization



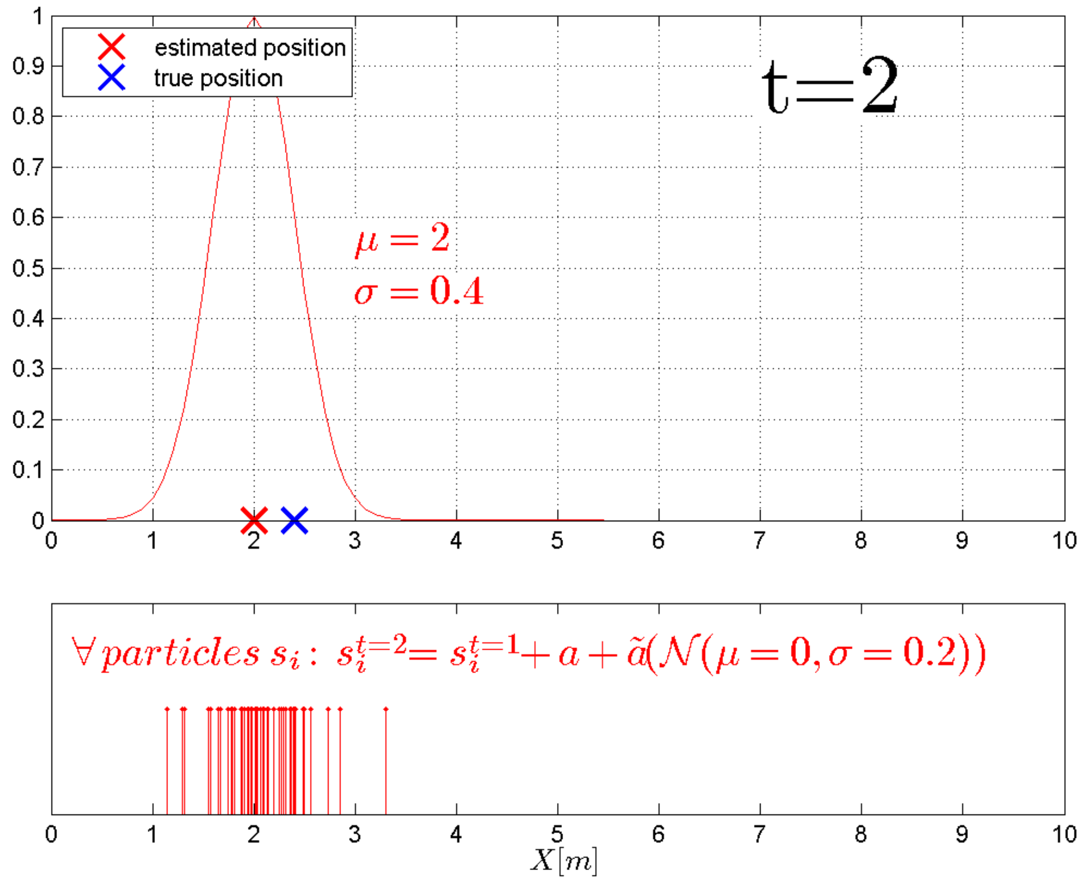
Particle Filter Localization



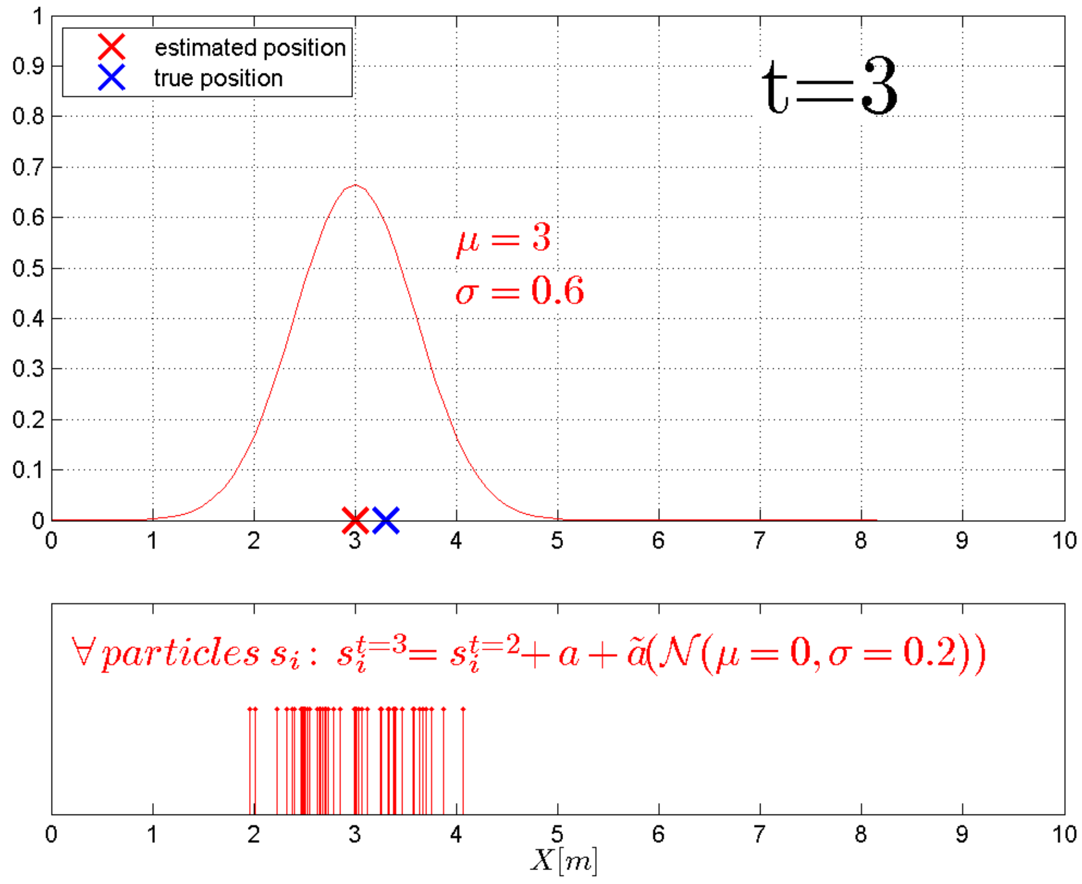
Particle Filter Localization



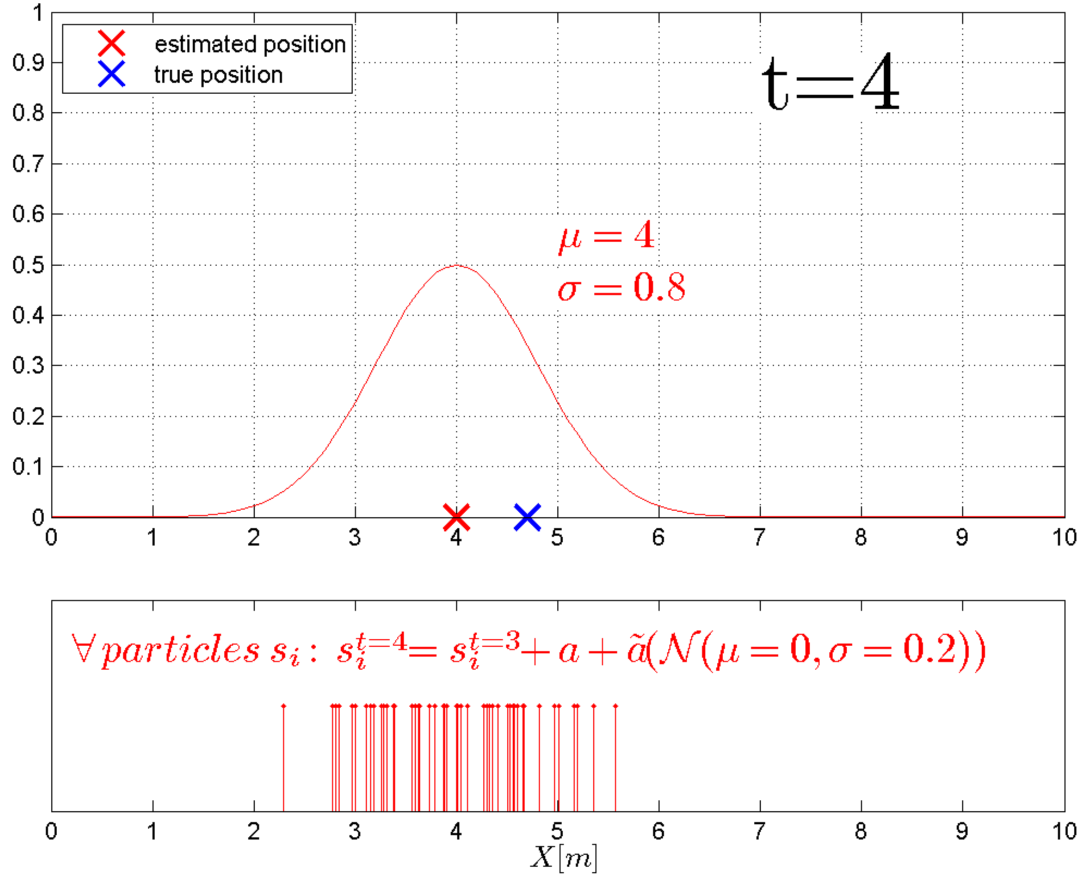
Particle Filter Localization



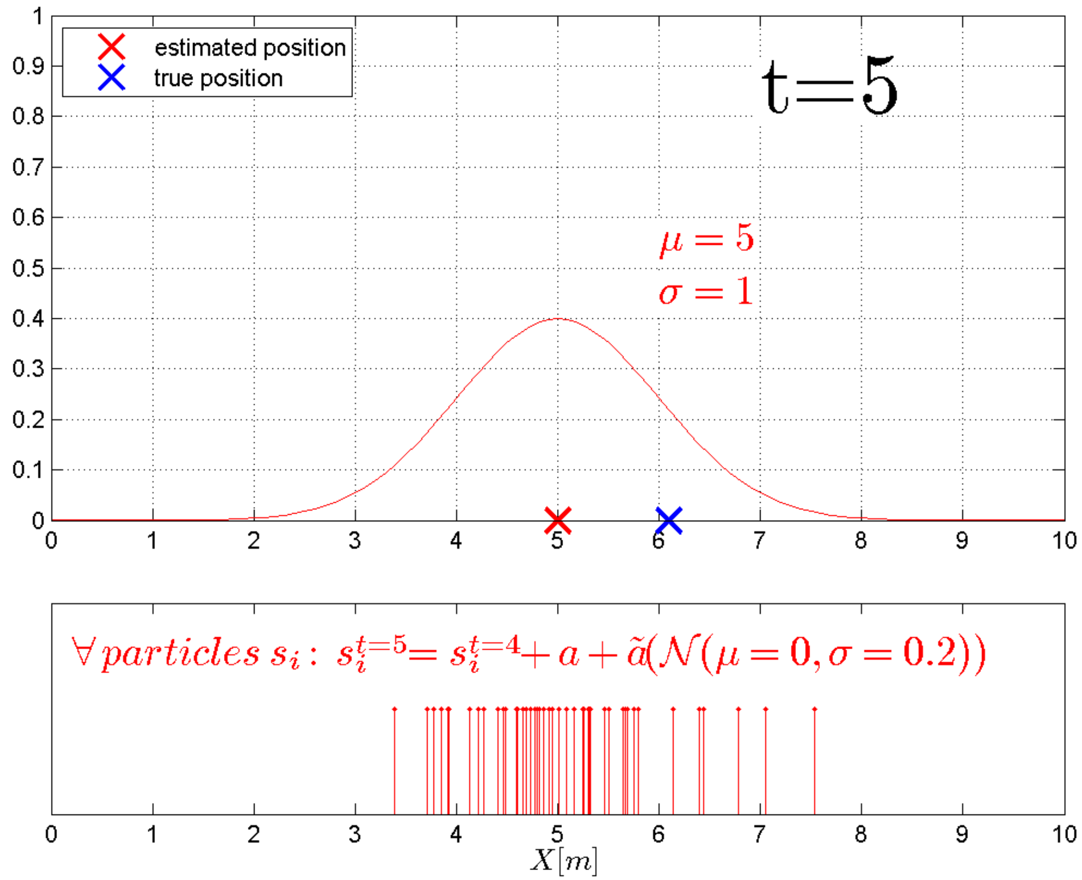
Particle Filter Localization



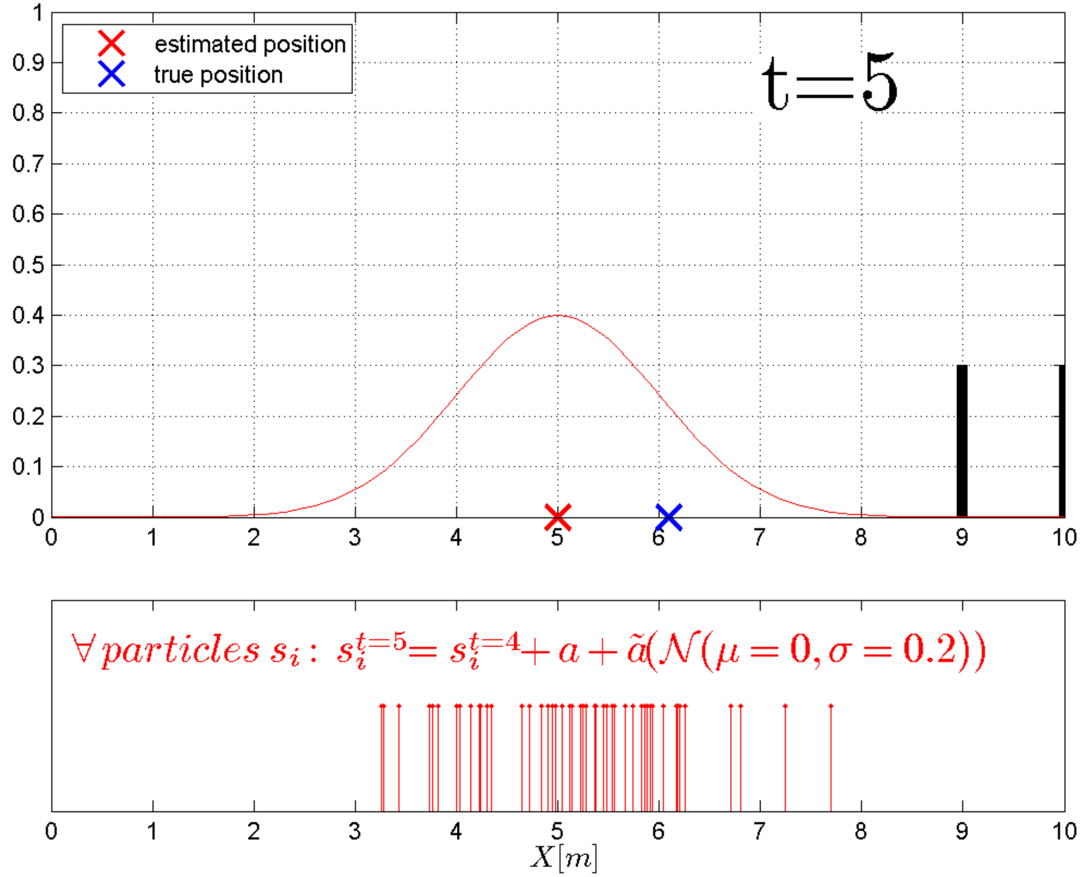
Particle Filter Localization



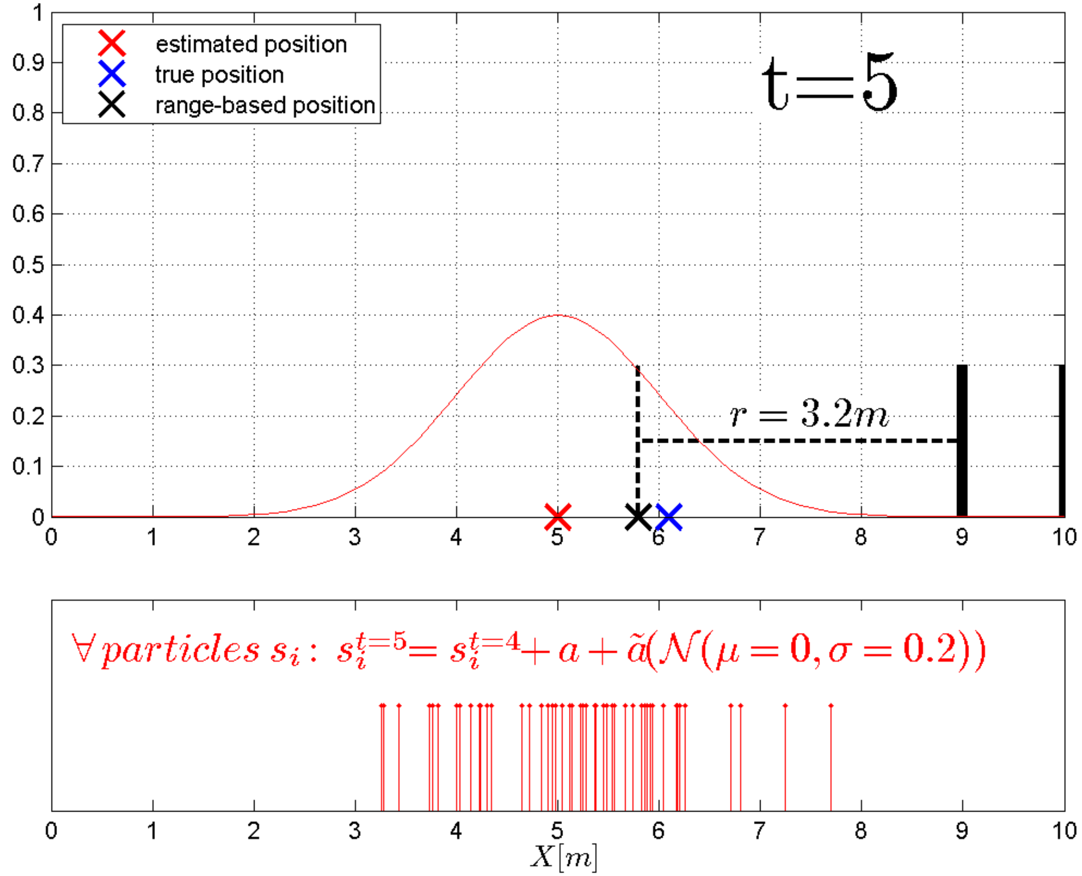
Particle Filter Localization



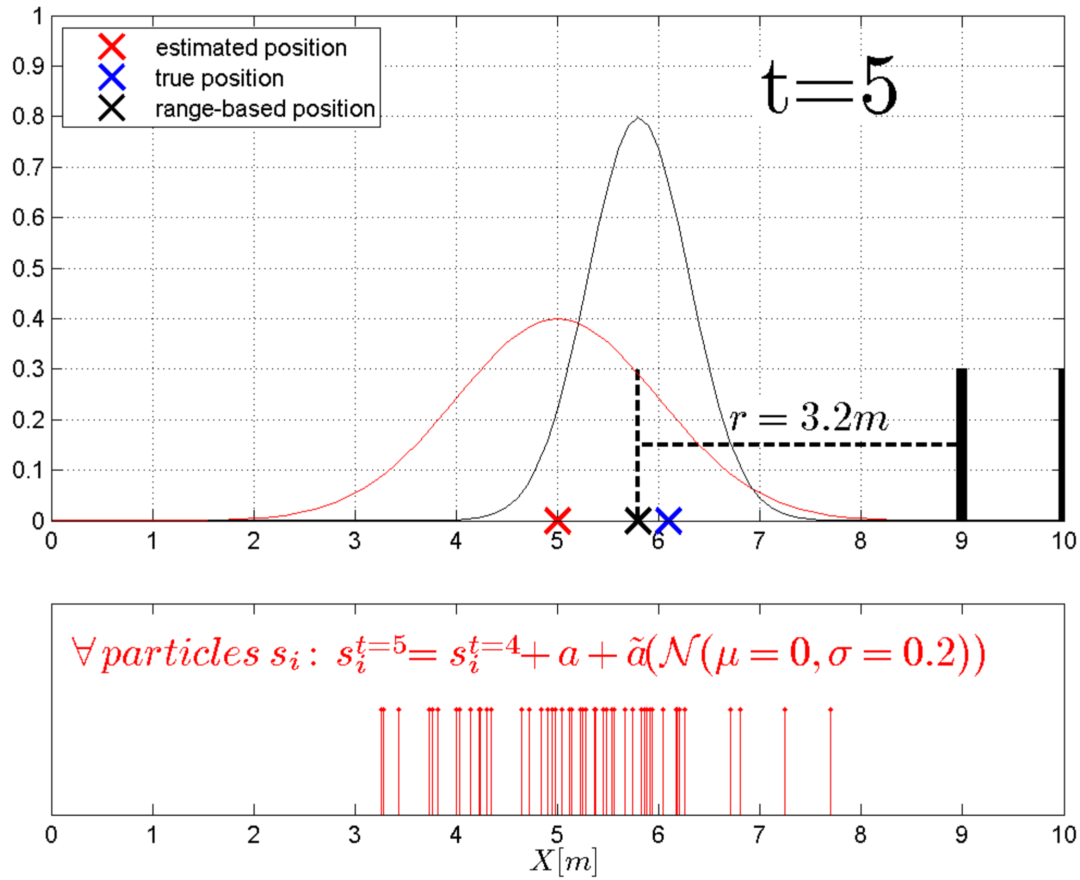
Particle Filter Localization



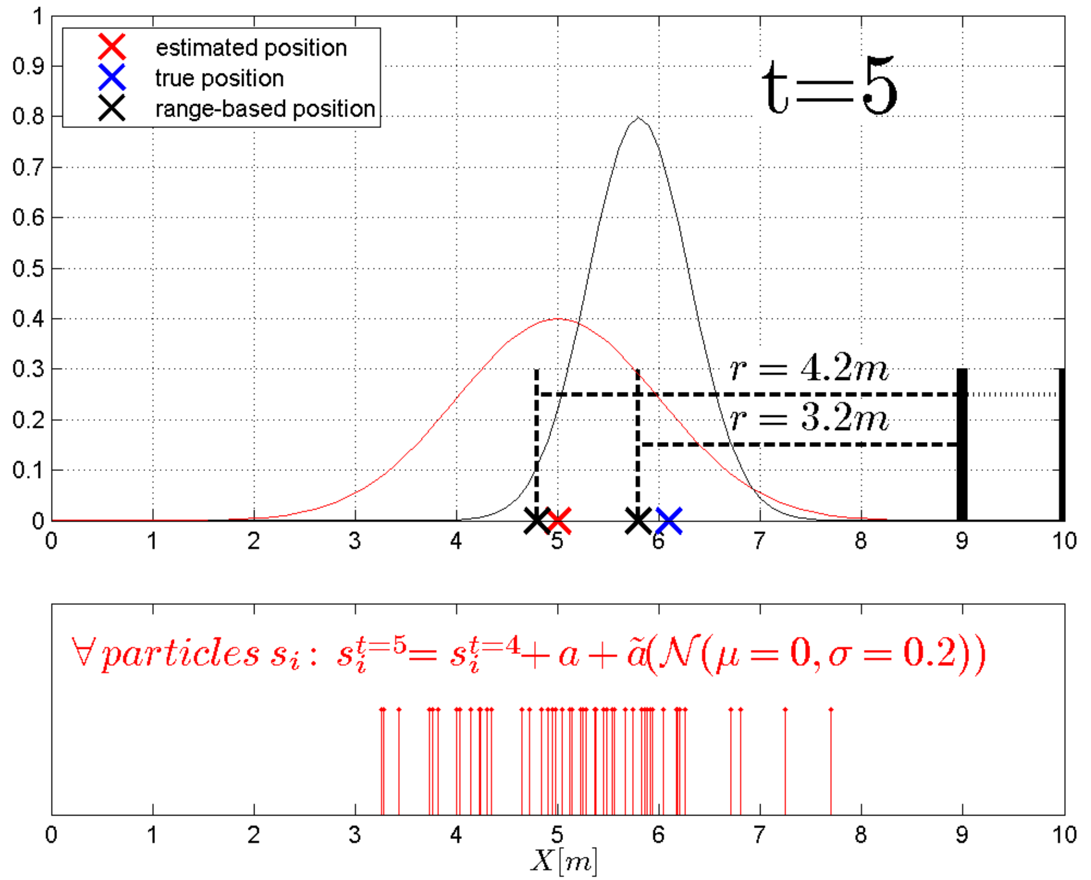
Particle Filter Localization



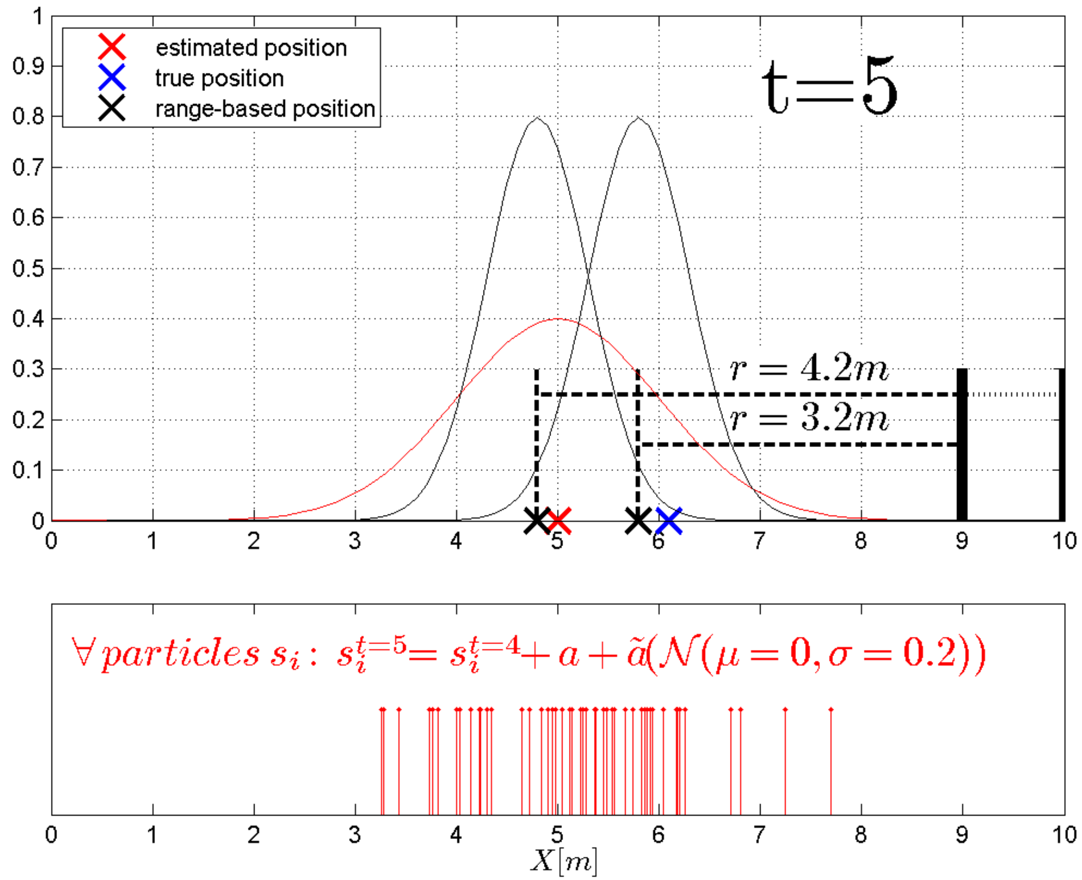
Particle Filter Localization



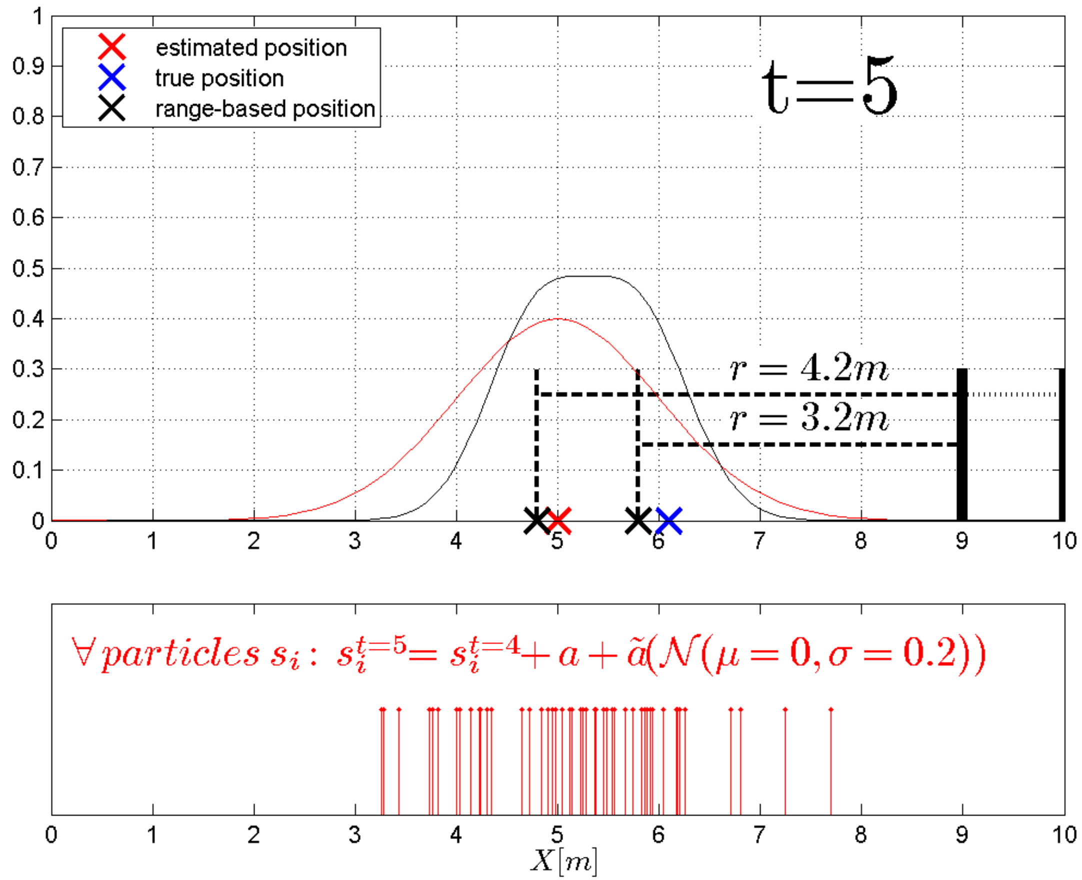
Particle Filter Localization



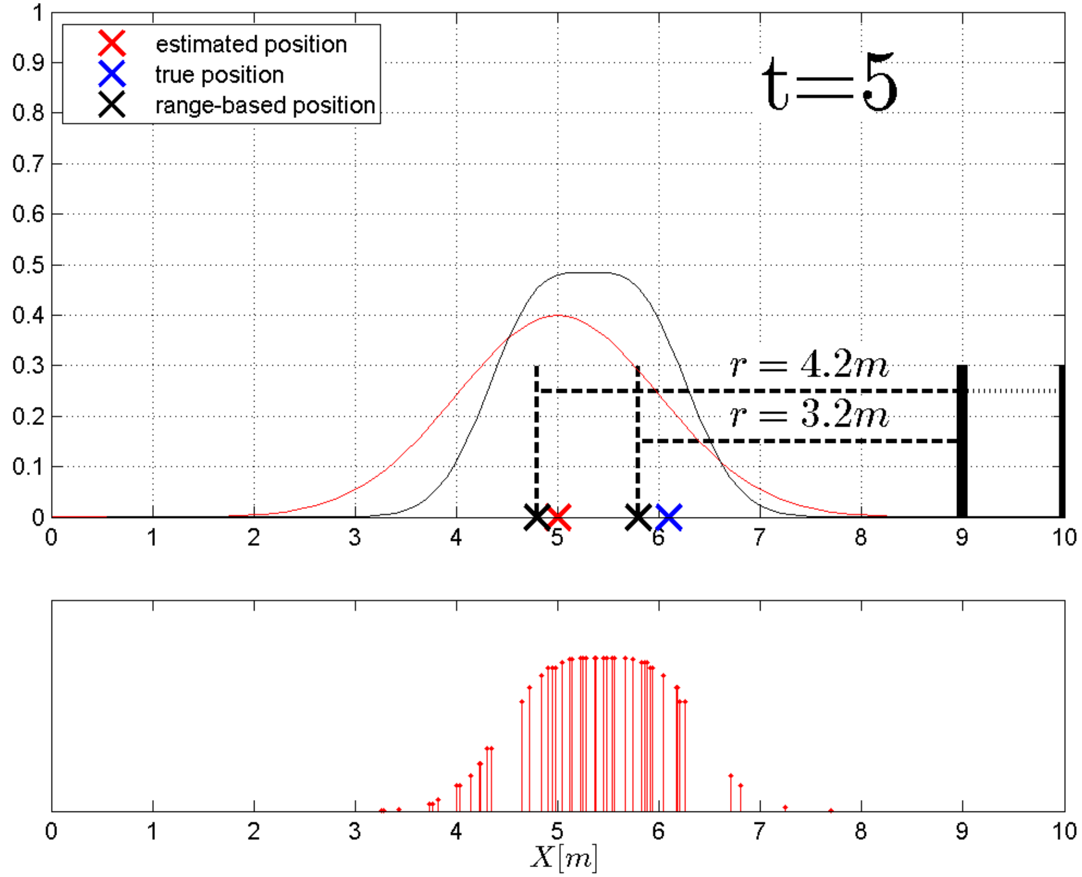
Particle Filter Localization



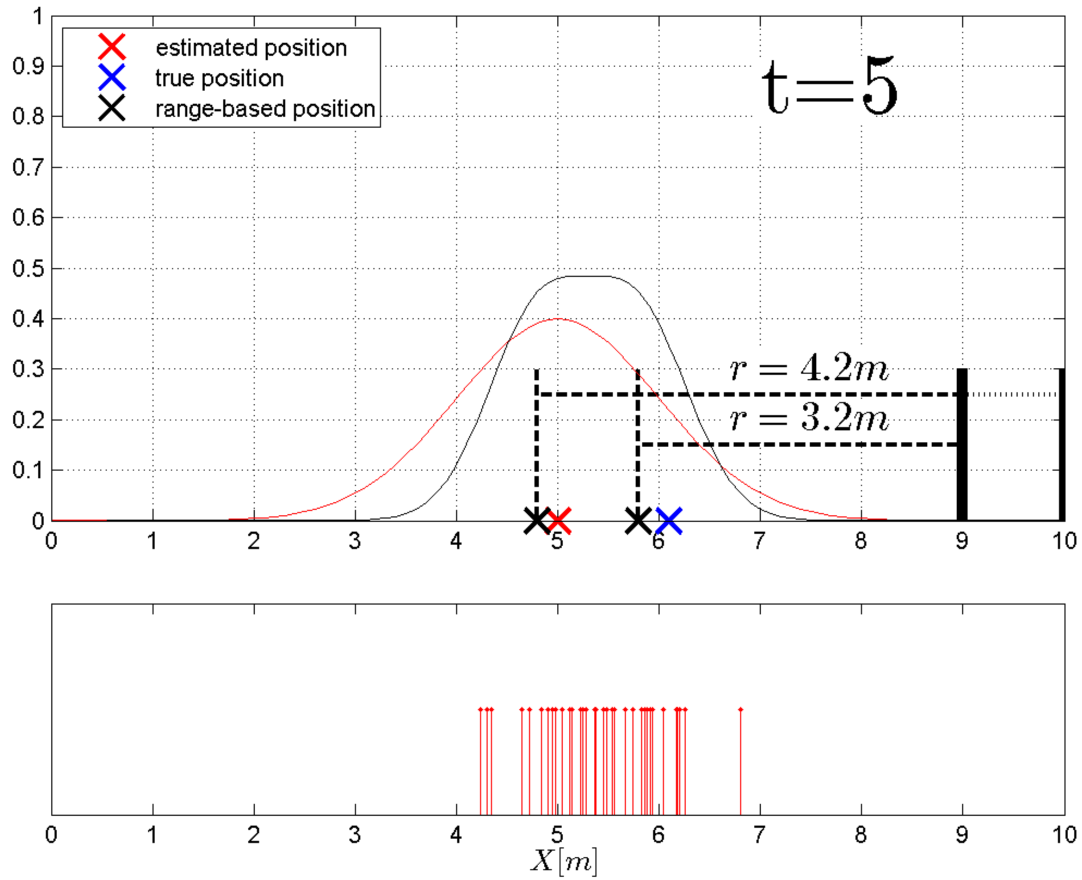
Particle Filter Localization



Particle Filter Localization



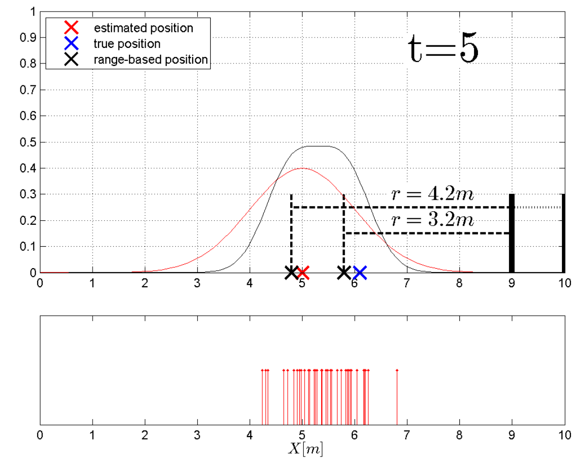
Particle Filter Localization



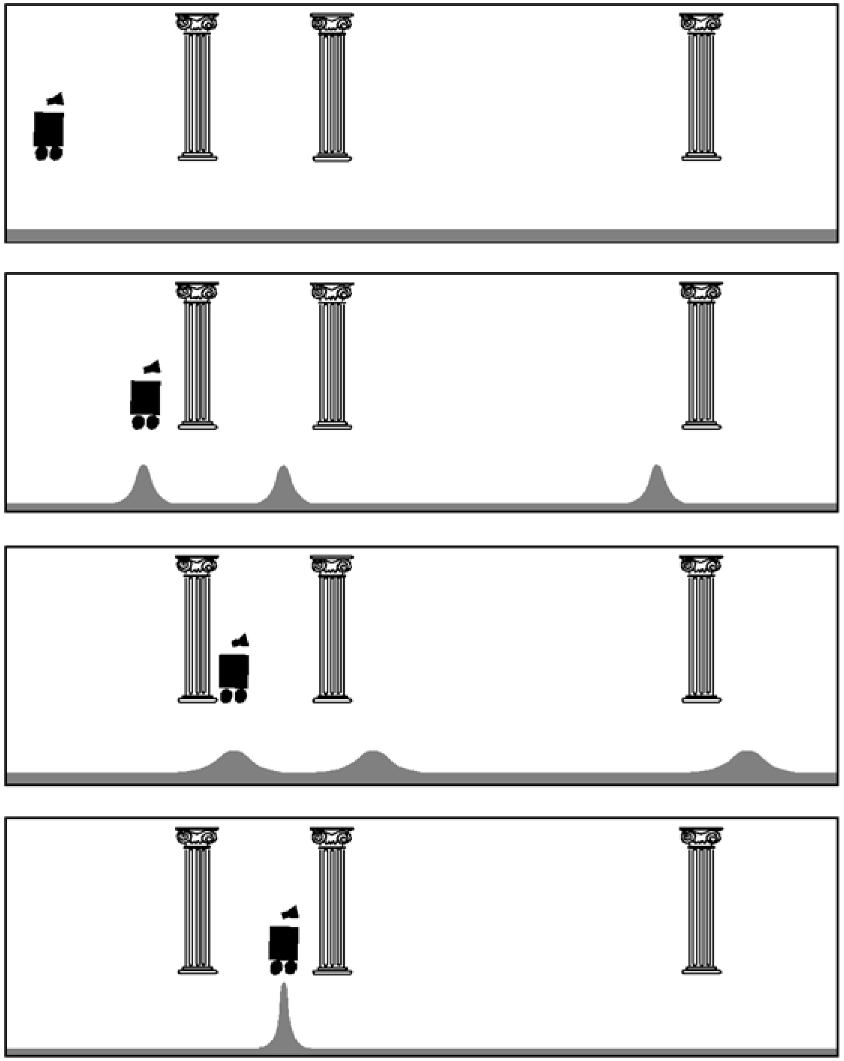
Particle Filter Localization

Belief representation through particle distribution

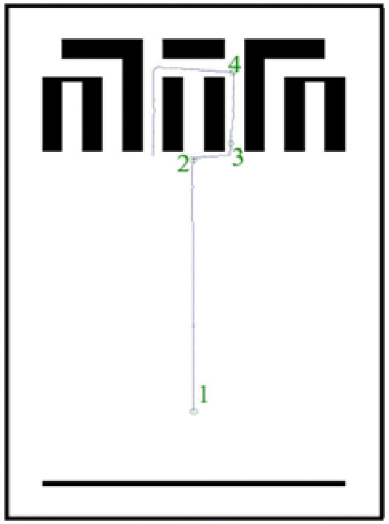
- Advantages:
  - Can model arbitrary beliefs
  - No assumptions on noise characteristic
- Disadvantages:
  - No unique solution
  - Not continuous
  - Computationally expensive
  - Tuning required

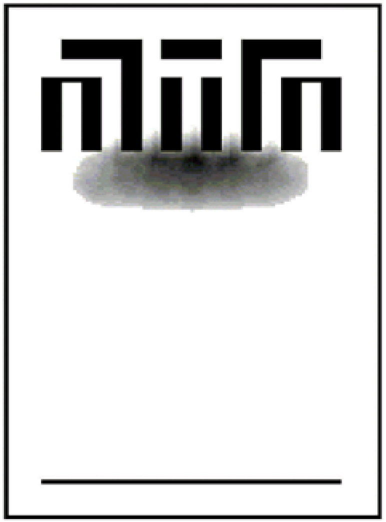


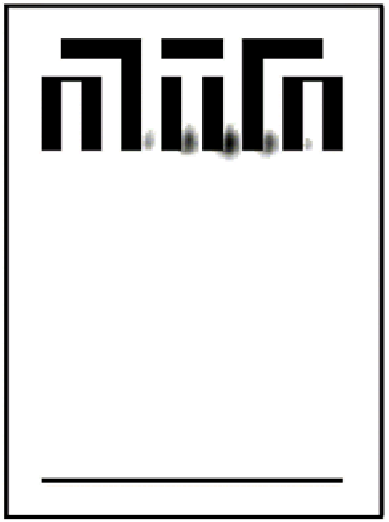
Particle Filter Localization

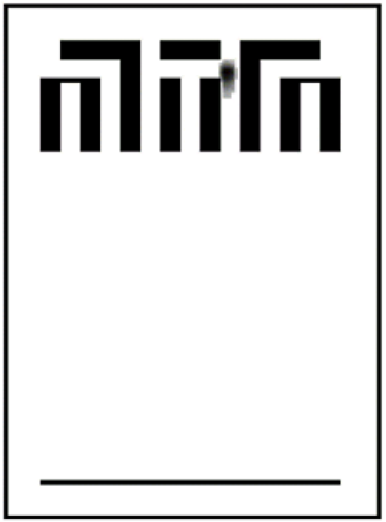


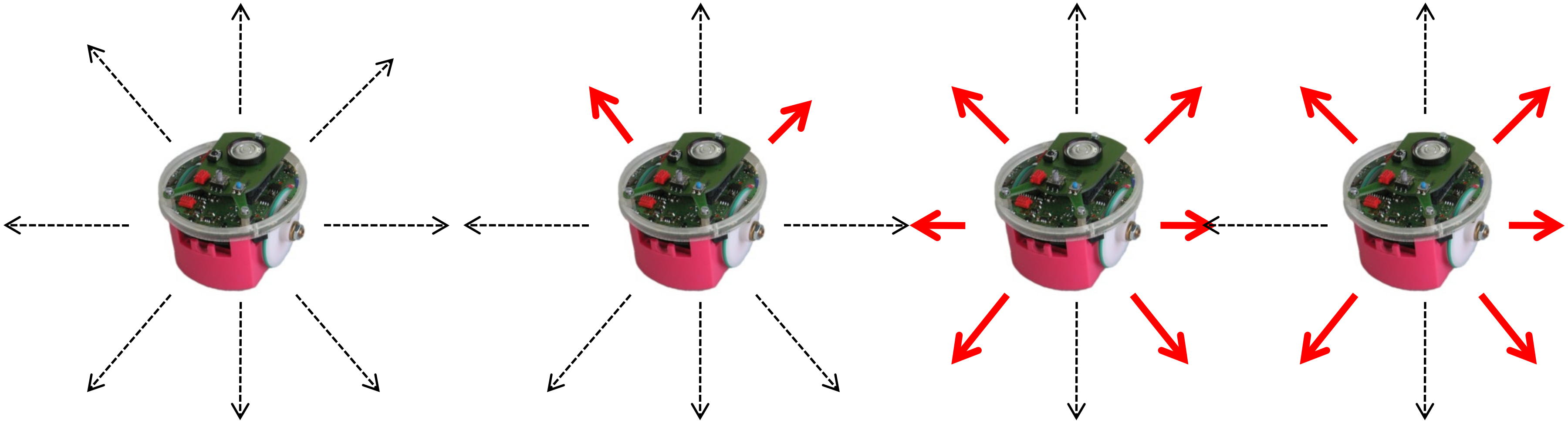
Particle Filter Localization











Conclusion

Take Home Messages

- Estimation methods, including Kalman filter techniques are applicable to a large variety of problems involving noisy processes and noisy sensing, not just localization
- Feature-based localization is a way to compensate odometry limitations by leveraging exteroceptive sensors in addition to proprioceptive ones
- A Kalman filter is a computationally efficient, optimal recursive data processing algorithm that allows fusion of multiple estimates coming from either process or sensing models
- A Kalman filter assumes linear motion and sensor models characterized by white Gaussian noise: many problems in robot localization do not fulfill these assumptions
- Particle filters allow for working around limitations of Kalman filters (in localization problems and beyond) at the price of additional complexity and computational cost

Additional Literature – Week 10

Pointers

http://www.probabilistic-robotics.org/

Books

- Siegwart R., Nourbakhsh I., and Scaramuzza D., “Introduction to Autonomous Mobile Robots, second Edition”, MIT Press, 2011.
- Thrun S., Burgard W., and Fox D., Probabilistic Robotics, MIT Press, 2005.
- Choset H., Lynch K. M., Hutchinson S., Kantor G., Burgard W., Kavraki L., and Thrun S., “Principles of Robot Motion”. MIT Press, 2005.
- Borenstein J., Everett H. R., and Feng L. “Navigating Mobile Robots: Systems and Techniques”, A. K. Peters, Ltd., 1996.