Università degli studi di Roma “Tor Vergata”

Bachelor degree in Engineering Sciences

UNSCENTED AND EXTENDED KALMAN FILTER

Supervisor
Ing. Francesco Martinelli

Candidate
Antonio Loquercio

Table of contents

1. Extended Kalman Filter
2. Unscented Kalman Filter
3. Graphical Comparison
4. Application to a mobile robot motion

Extended Kalman Filter

- Implementation through linearization of *f()* and *h()* around the estimates

System:

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1})$$

$$y_k = h(x_k) + v_k$$

$\hat{x}_{k-1}$ → Expected Value & Covariance

P_{k-1}

Q_k → Covariance of w_k

R_k → Covariance of v_k

Prediction step:

$$f(x_{k-1}, u_{k-1}, w_{k-1}) \approx f(\widehat{x_{k-1}}, u_{k-1}, 0) + \frac{\partial f}{\partial x}_{(\widehat{x_{k-1}}, u_{k-1}, 0)} (x_{k-1} - \widehat{x_{k-1}}) + \frac{\partial f}{\partial w}_{(\widehat{x_{k-1}}, u_{k-1}, 0)} (w_{k-1})$$

$$\widehat{x_k}^- = E[f(x_{k-1}, u_{k-1}, w_{k-1})] \approx f(\widehat{x_{k-1}}, u_{k-1}, 0) + \frac{\partial f}{\partial x}_{(\widehat{x_{k-1}}, u_{k-1}, 0)} E(x_{k-1} - \widehat{x_{k-1}}) + \frac{\partial f}{\partial w}_{(\widehat{x_{k-1}}, u_{k-1}, 0)} E(w_{k-1}) \approx f(\widehat{x_{k-1}}, u_{k-1}, 0)$$

Defining: $F_{x,k-1} \triangleq \frac{\partial f}{\partial x}_{(\widehat{x_{k-1}},u_{k-1},0)}$ $F_{w,k-1} \triangleq \frac{\partial f}{\partial w}_{(\widehat{x_{k-1}},u_{k-1},0)}$

We obtain the covariance matrix before the kth measurement:

$$P_k^- = E[(x_k - \widehat{x_k^-})(x_k - \widehat{x_k^-})'] = F_{x,k-1}P_{k-1}F'_{x,k-1} + F_{w,k-1}Q_{k-1}F'_{w,k-1}$$

Correction step:

$$h(x_k) \approx h(\widehat{x_k^-}) + \frac{\partial h}{\partial x}_{(\widehat{x_{k-1}},u_{k-1},0)}(x_{k-1} - \widehat{x_{k-1}})$$

$$E[h(x_k)] \approx h(\widehat{x_k^-})$$

Defining: $H_{x,k} \triangleq \frac{\partial h}{\partial x}_{(\widehat{x_{k-1}},u_{k-1},0)}$

Substituting these parameters in the LSQ estimator formula, the algorithm of Kalman filter can be summarized as:

Prediction step:

$$\widehat{x_k}^- = f(\widehat{x_{k-1}}, u_{k-1}, 0)$$

$$P_k^- = F_{x,k-1} P_{k-1} F'_{x,k-1} + F_{w,k-1} Q_{k-1} F'_{w,k-1}$$

Correction step:

$$\widehat{x_k} = \widehat{x_k^-} + K_k(y_k - h(\widehat{x_k^-}))$$

$$P_k = (I - K_k H_{x,k})\, P_k^-$$

$$K_k = P_k^- H'_{x,k} \left(H_{x,k} P_k^- H'_{x,k} + R_k\right)^{-1}$$

Where the last parameter is called **Kalman Gain**

UNSCENTED KALMAN FILTER

- Implementation through Unscented Transformation

The Unscented Transformation

- A trick to propagate mean and covariance of random variables (or vectors) through nonlinear transformations

X -> random variable $f(x)$ -> non linear function

3 STEP PROCESS

Produce 2N+1 Sigma Points, v_i, with weight w_i

↓

Map these points through the nonlinear transformation $f(x)$

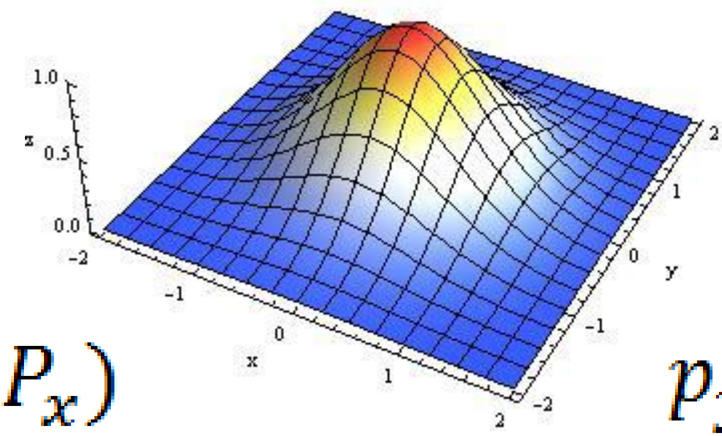
↓

Approximate, from these points, $p_y()$ by a Gaussian

$$p_y(\cdot) \cong \mathcal{N}(\hat{y}, P_y)$$

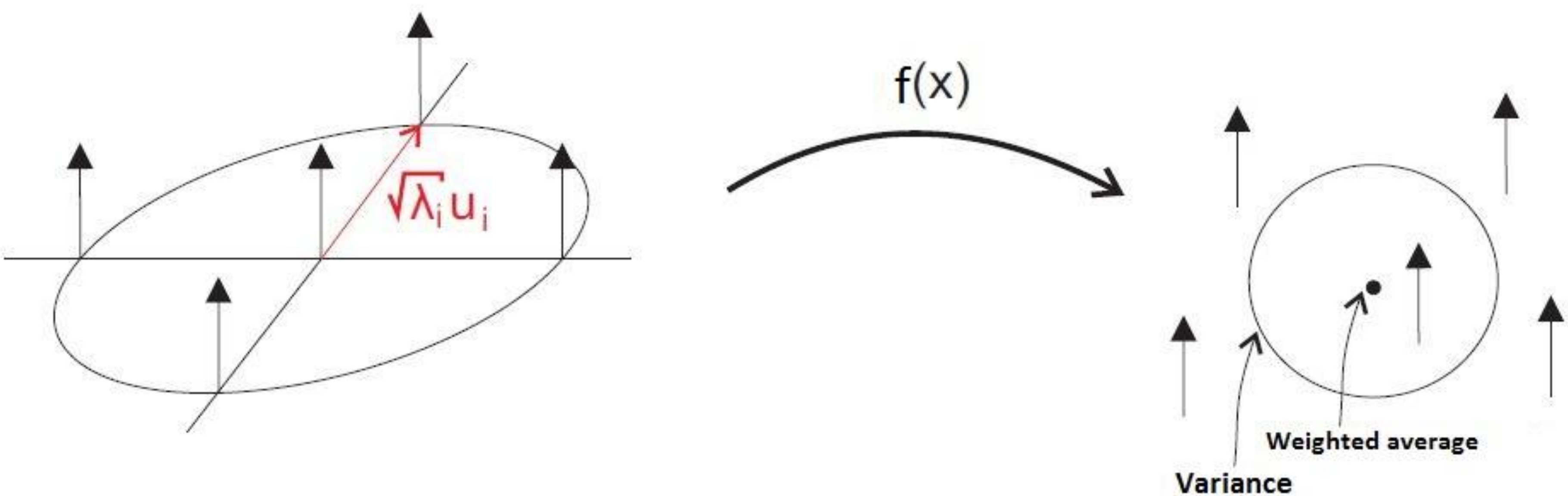
Graphical example with n = 2

Gaussian Distribution



$$p_x(\cdot) \sim \mathcal{N}(\hat{x}, P_x)$$

$$p_y(\cdot) \cong \mathcal{N}(\hat{y}, P_y)$$



How to compute Sigma Points?

$$v_0 = \hat{x}$$
$$v_1 = \hat{x} + \propto \sigma_1$$
$$v_2 = \hat{x} - \propto \sigma_1$$
$$\vdots$$
$$v_{2N-1} = \hat{x} + \propto \sigma_N$$
$$v_{2N} = \hat{x} - \propto \sigma_N$$

$\propto$ →

$$\sum_{i=0}^{2N} w_i (v_i - x)(v_i - x)' = P_x$$

$$\sum_{i=0}^{2N} w_i v_i = \hat{x}$$

σ_i → $\sqrt{\lambda_i} u_i$

The Unscented Filter

The Unscented Transformation is used to establish the following algorithmic procedure

$$x_{a,k-1} = \begin{bmatrix} x_{k-1} \\ w_{k-1} \end{bmatrix}$$

$$f_a(x_a, u) = f(x, u, w)$$

Consequently we will have

$$\hat{x}_{a,k-1} = \begin{bmatrix} \hat{x}_{k-1} \\ 0 \end{bmatrix}$$

$$P_{a,k-1} = \begin{bmatrix} P_{k-1} & 0 \\ 0 & Q_{k-1} \end{bmatrix}$$

Prediction Step Procedure

Compute Sigma Points

$$[W, V_{k-1}] = SigmaPoints(\hat{x}_{a,k-1}, P_{a,k-1})$$

Apply the dynamics to each Sigma Point

$$v_{i,k}^{-} = f_a(v_{i,k-1}, u_{i,k-1})$$

Get a priori estimate and its covariance

$$\hat{x}_k^{-} = \sum_{i=0}^{2n_a} w_i v_{i,k}^{-}$$

$$P_k^{-} = \sum_{i=0}^{2n_a} w_i (v_{i,k}^{-} - \hat{x}_k^{-})(v_{i,k}^{-} - \hat{x}_k^{-})'$$

Correction Step Procedure

For each transformed sigma point, compute the measurement vector

$$y_{i,k} = h(v_{i,k}^-)$$

Compute the a priori expected measurement vector

$$y_k^- = \sum_{i=0}^{2n_a} w_i y_{i,k}$$

Compute covariance matrices and Kalman Gain

$$P_y = \sum_{i=0}^{2n_a} w_i (y_{i,k} - y_k^-)(y_{i,k} - y_k^-)' + R_k$$

$$P_{xy} = \sum_{i=0}^{2n_a} w_i (v_{j,k}^- - \hat{x}_k^-)(y_{i,k} - y_k^-)'$$

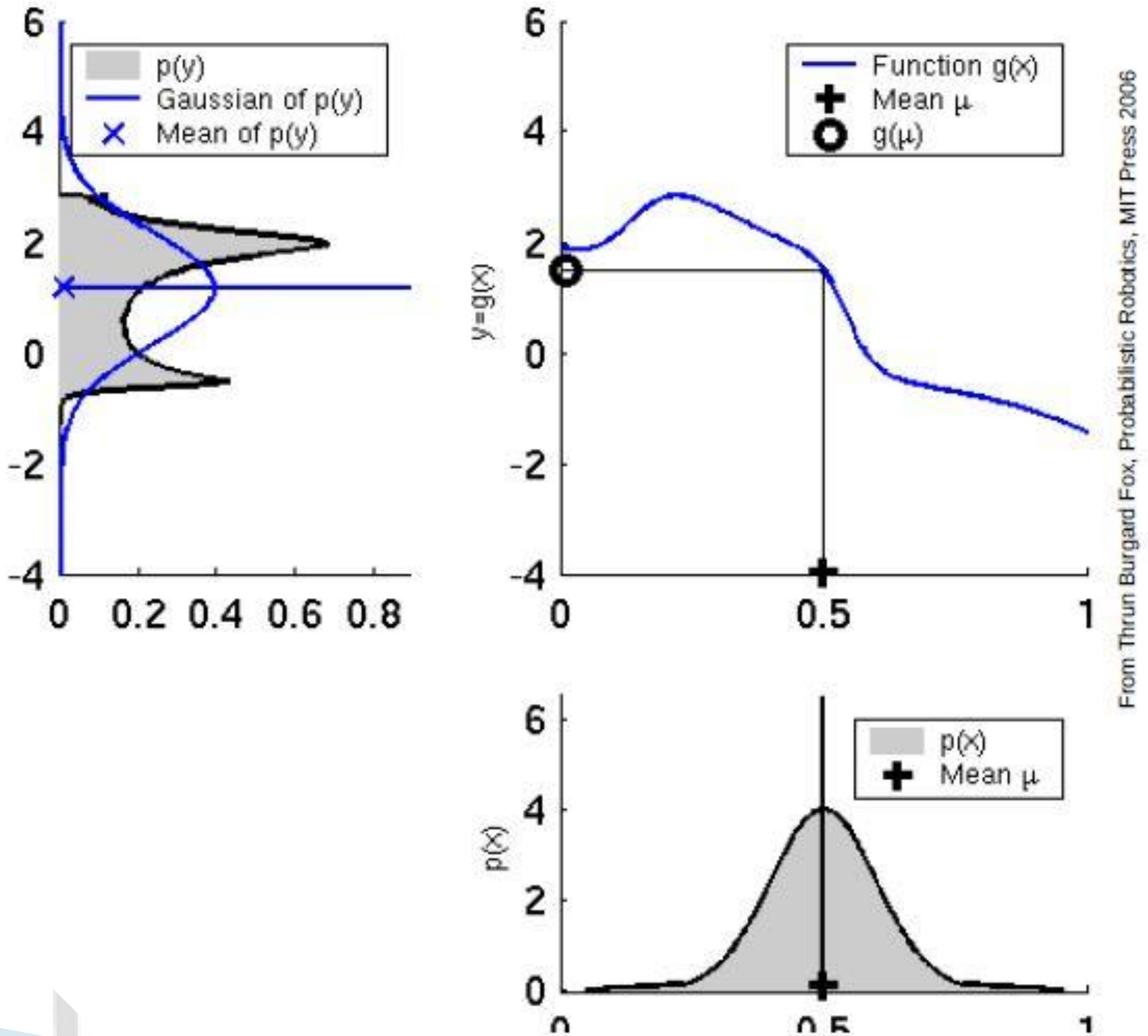
$$K_k = P_{xy} P_y^{-1}$$

Compute the estimate and update the covariance matrix

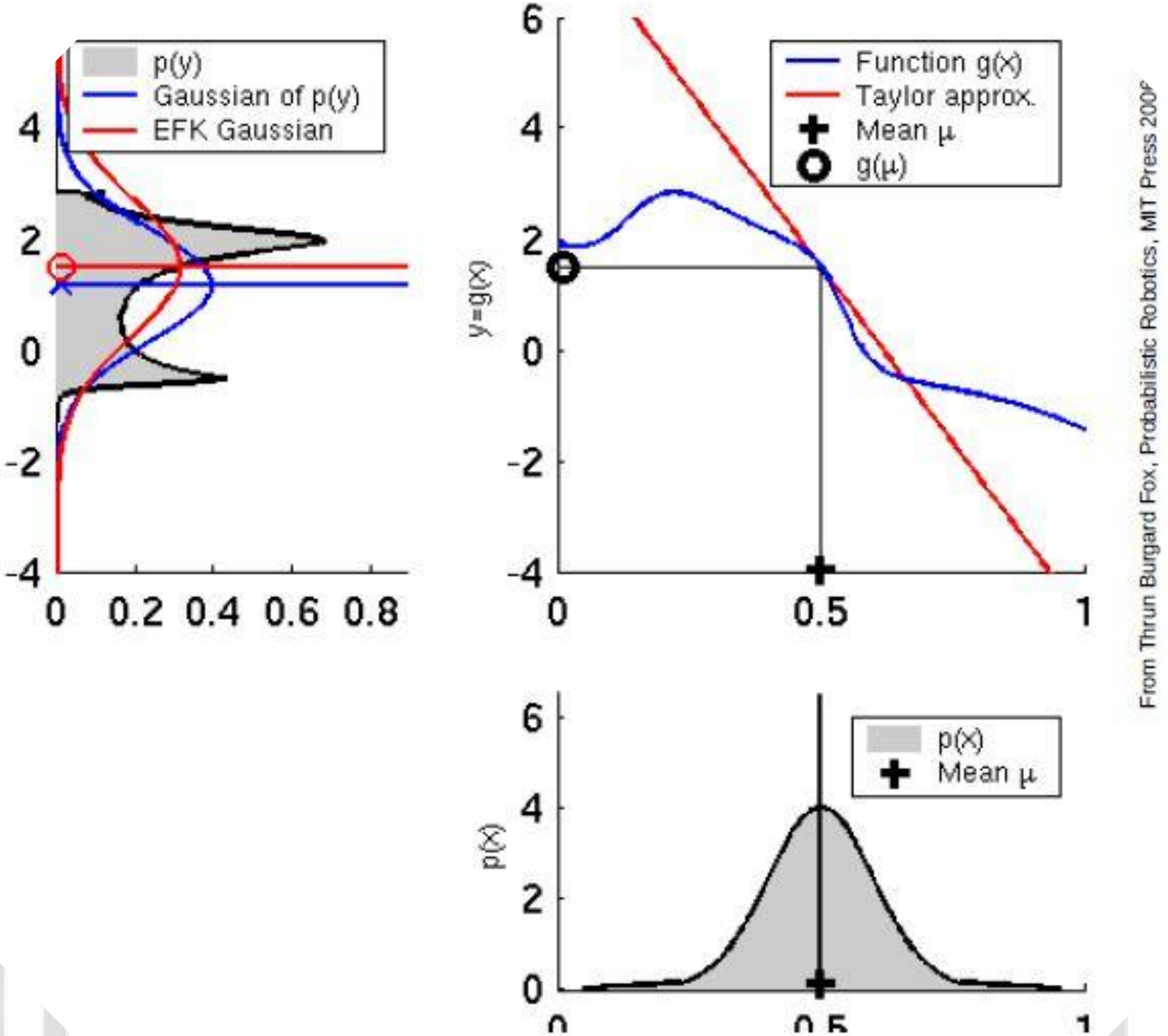
$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - \hat{y}_k^-)$$

$$P_k = P_k^- - K_k P_y K_k'$$

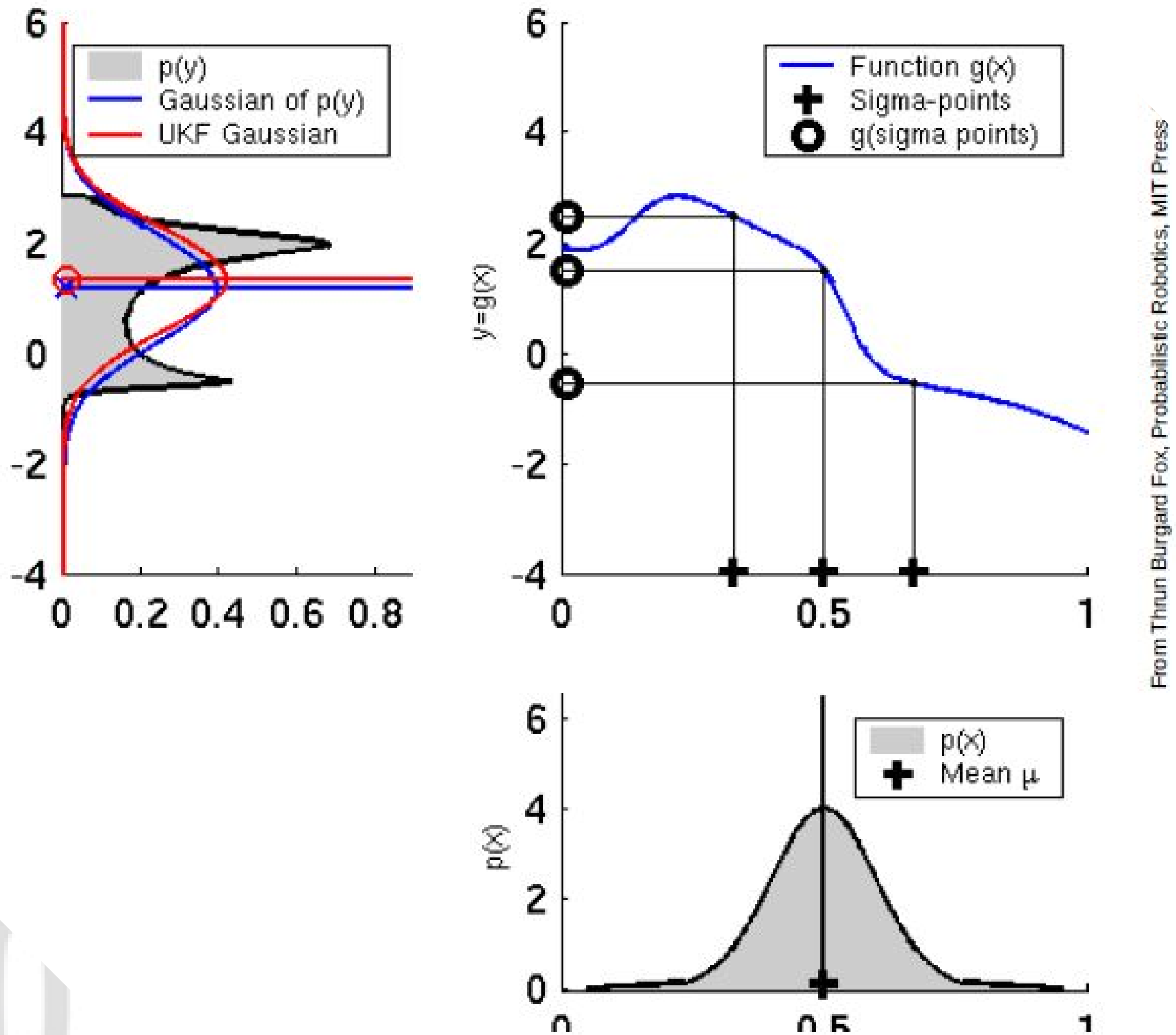
Nonlinear transformation of a Gaussian variable



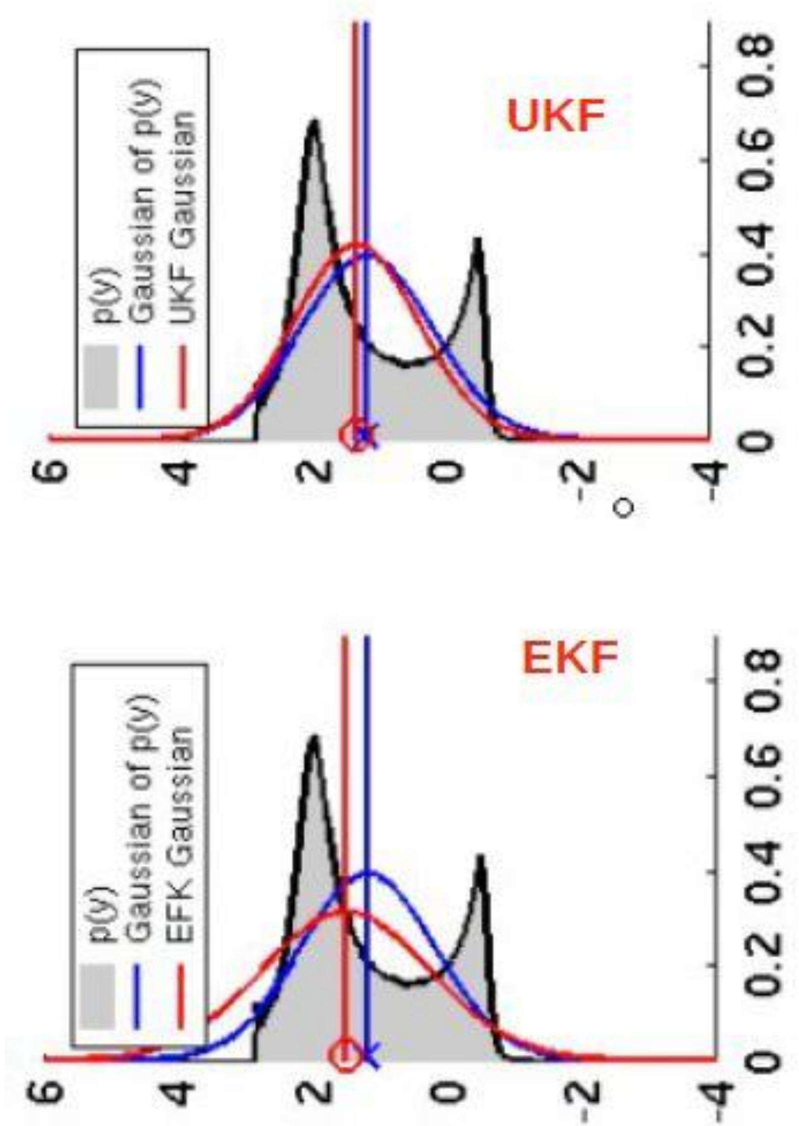
EKF Approximation



UKF Approximation



EKF VS UKF



From Thrun Burgard Fox, Probabilistic Robotics, MIT Press 2006

Localization as an estimation problem

The Robot must infer its position and orientation from available data

Motion Information

- *Proprioceptive sensors* (encoders, accelerometers, etc.)

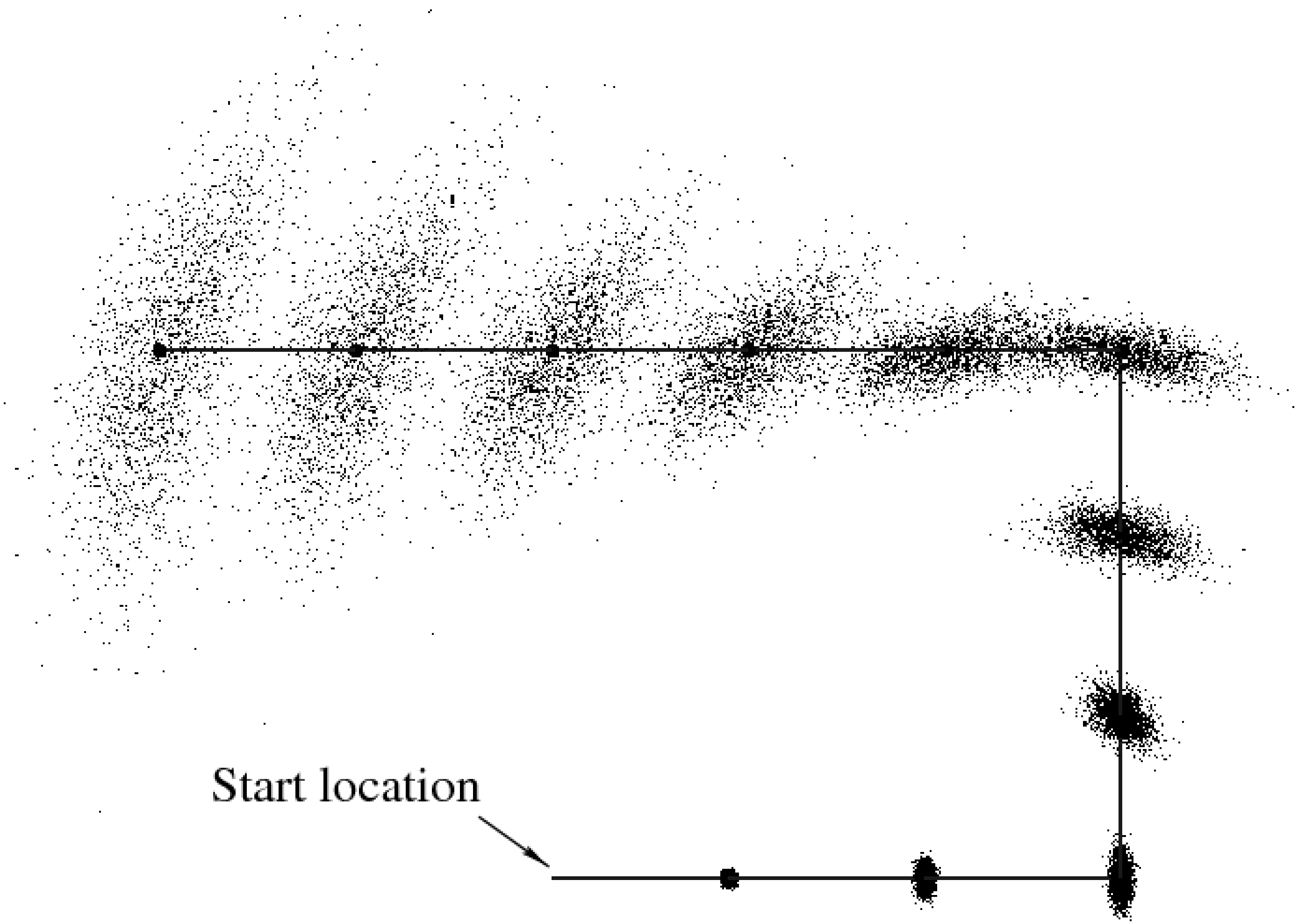
Environment Measurements

- *Exteroceptive sensors* (e.g. laser, sonar, IR, GPS, camera, RFID, etc.)

A filtering approach is required to merge information

Considering the **Observability Issue**

IF ONLY MOTION INFORMATION ARE USED, ERRORS WILL BE CUMULATIVE



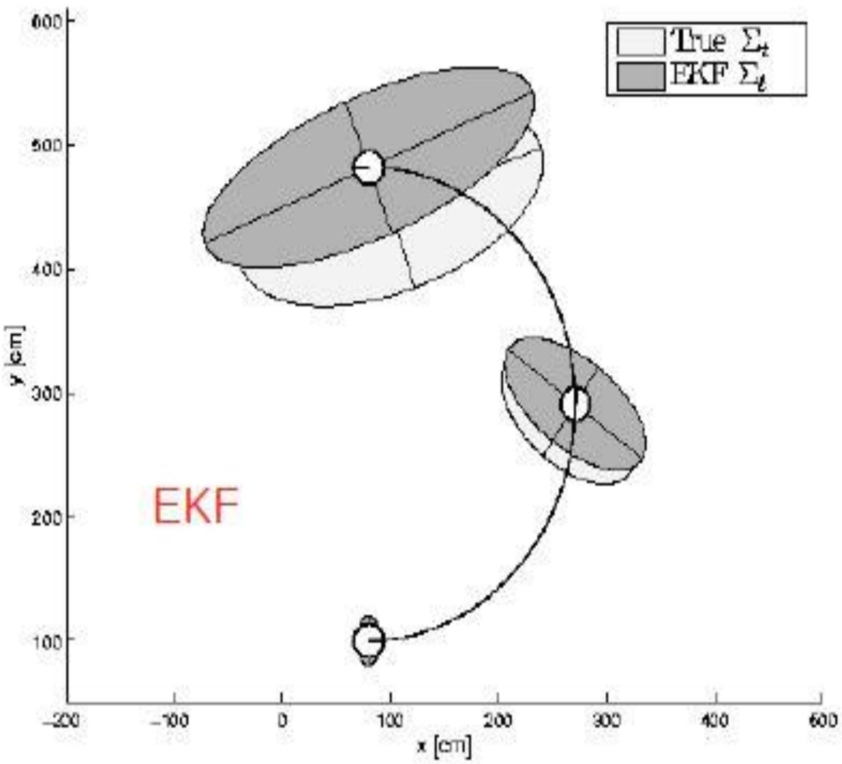
Mathematical approximation of mobile robot motion

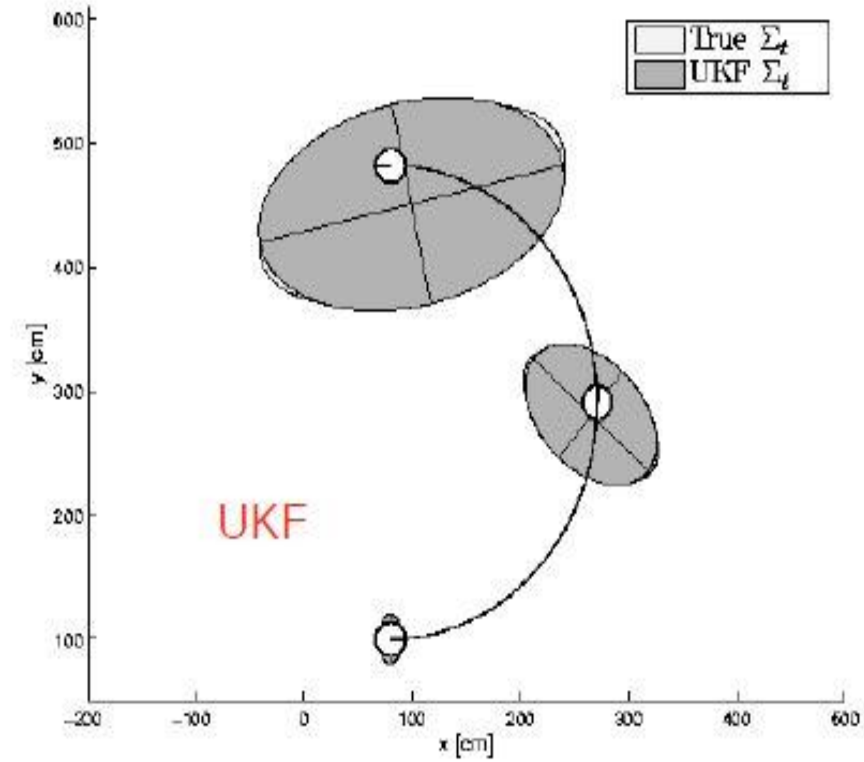
$$x \to x_r = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \qquad u \to \begin{bmatrix} u_R^e \\ u_L^e \end{bmatrix} \qquad w \to \begin{bmatrix} n_R \\ n_L \end{bmatrix}$$

$$f(x, u+n) \to f(x_r, [u_R^e, u_L^e]^T + [n_R, n_L]^T) := \begin{cases} x + \frac{u_R^e + n_R + u_L^e + n_L}{2} \cos(\theta) \\ y + \frac{u_R^e + n_R + u_L^e + n_L}{2} \sin(\theta) \\ \theta + \frac{u_R^e + n_R - u_L^e - n_L}{d} \end{cases}$$

$$z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} = \begin{bmatrix} h_1(x) + \nu_1 \\ h_2(x) + \nu_2 \\ \vdots \\ h_m(x) + \nu_m \end{bmatrix} = h(x) + \nu$$

Quality of approximation in EKF and UKF





From Thrun Burgard Fox, Probabilistic Robotics, MIT Press 2006

Conclusions

Common characteristics

- Implementation goal: Guassian Fitting of nonlinear systems pdfs
- Based on affine Least SQuare (LSQ) estimator
- Used to approximate unimodal pdfs

Differences

EXTENDED FILTER

- Tool: linearization of nonlinear functions
- Fixed formula, based on Jacobians of given functions
- Required computation of Jacobians for linearization

UNSCENTED FILTER

- Tool: Sigma Point transformation
- Flexible formula: some parameters could be adapted
- Non linear functions are fully exploited

Bibliography

1) Fox T. B. , *Probabilistic Robotics*, MIT Press, 2006
2) Julier S.J., Uhlmann J.K., *Unscented Filtering and Nonlinear Estimation,* USA, 2003
3) Martinelli F., *Mobile robot localization. Lecture notes of the course Robotica con Laboratorio*, University of Rome Tor Vergata, 2014
4) Martinelli F., *Derivation of the EKF and of the UKF from the affine LSQ estimator. Lecture notes of the course on Filtering and Mobile Robot Localization*, University of Rome Tor Vergata, 2010