

Robot Mapping

Unscented Kalman Filter

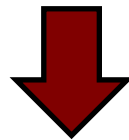
**Gian Diego Tipaldi, Luciano Spinello,
Wolfram Burgard**

KF, EKF and UKF

- Kalman filter requires linear models
- EKF linearizes via Taylor expansion

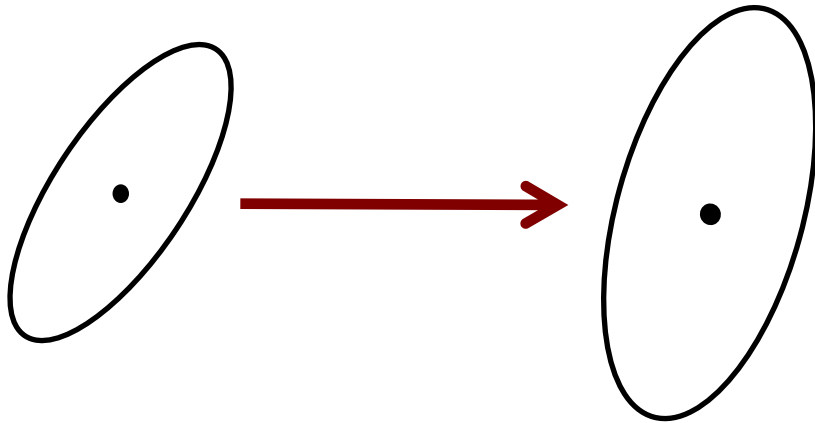
Is there a better way to linearize?

Unscented Transform



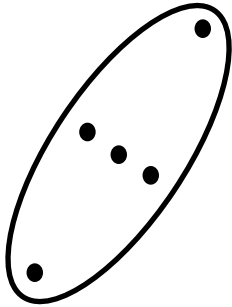
Unscented Kalman Filter (UKF)

Taylor Approximation (EKF)



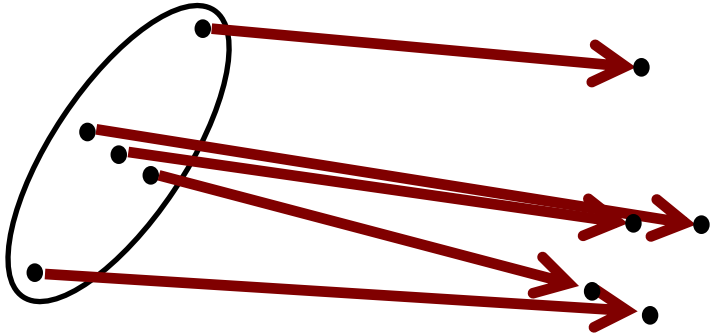
Linearization of the non-linear function through Taylor expansion

Unscented Transform



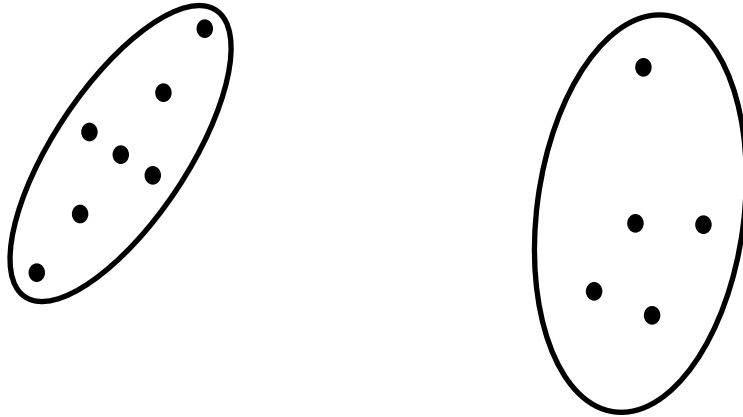
Compute a set of (so-called)
sigma points

Unscented Transform



Transform each sigma point through the non-linear function

Unscented Transform



Compute Gaussian from the transformed and weighted sigma points

Unscented Transform Overview

- Compute a set of sigma points
- Each sigma points has a weight
- Transform the point through the non-linear function
- Compute a Gaussian from weighted points

- Avoids to linearize **around the mean** as Taylor expansion (and EKF) does

Sigma Points

- How to choose the sigma points?
- How to set the weights?

Sigma Points Properties

- How to choose the sigma points?
- How to set the weights?
- Select $\mathcal{X}^{[i]}, w^{[i]}$ so that:

$$\sum_i w^{[i]} = 1$$

$$\mu = \sum_i w^{[i]} \mathcal{X}^{[i]}$$

$$\Sigma = \sum_i w^{[i]} (\mathcal{X}^{[i]} - \mu)(\mathcal{X}^{[i]} - \mu)^T$$

- There is no unique solution for $\mathcal{X}^{[i]}, w^{[i]}$

Sigma Points

- Choosing the sigma points

$$\mathcal{X}^{[0]} = \mu$$

First sigma point is the mean

Sigma Points

- Choosing the sigma points

$$\mathcal{X}^{[0]} = \mu$$

$$\mathcal{X}^{[i]} = \mu + \left(\sqrt{(n + \lambda) \Sigma} \right)_i \quad \text{for } i = 1, \dots, n$$

$$\mathcal{X}^{[i]} = \mu - \left(\sqrt{(n + \lambda) \Sigma} \right)_{i-n} \quad \text{for } i = n + 1, \dots, 2n$$

matrix square
root

column vector

dimensionality

scaling parameter

Matrix Square Root

- Defined as S with $\Sigma = SS$
- Computed via diagonalization

$$\begin{aligned}\Sigma &= VD V^{-1} \\ &= V \begin{pmatrix} d_{11} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & d_{nn} \end{pmatrix} V^{-1} \\ &= V \begin{pmatrix} \sqrt{d_{11}} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \sqrt{d_{nn}} \end{pmatrix} \begin{pmatrix} \sqrt{d_{11}} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \sqrt{d_{nn}} \end{pmatrix} V^{-1}\end{aligned}$$

Matrix Square Root

- Thus, we can define

$$S = V \underbrace{\begin{pmatrix} \sqrt{d_{11}} & \dots & 0 \\ 0 & \ddots & 0 \\ 0 & \dots & \sqrt{d_{nn}} \end{pmatrix}}_{D^{1/2}} V^{-1}$$

- so that

$$SS = (VD^{1/2}V^{-1})(VD^{1/2}V^{-1}) = VDV^{-1} = \Sigma$$

Cholesky Matrix Square Root

- Alternative definition of the matrix square root

$$L \text{ with } \Sigma = LL^T$$

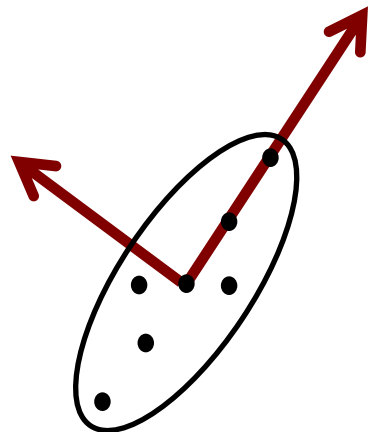
- Result of the Cholesky decomposition
- Numerically stable solution
- Often used in UKF implementations
- L and Σ have the same Eigenvectors

Sigma Points and Eigenvectors

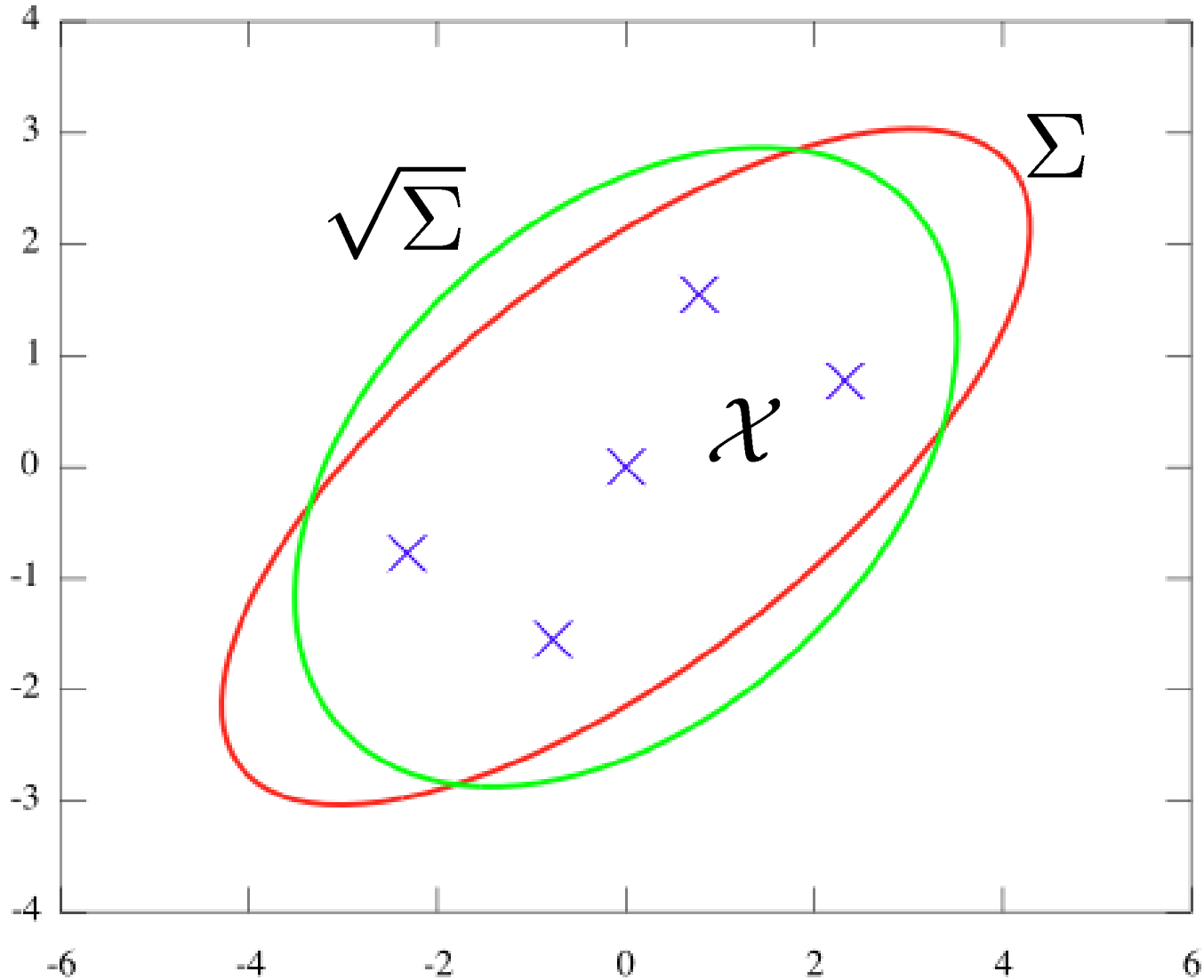
- Sigma point **can** but **do not have to** lie on the main axes of Σ

$$\mathcal{X}^{[i]} = \mu + \left(\sqrt{(n + \lambda) \Sigma} \right)_i \quad \text{for } i = 1, \dots, n$$

$$\mathcal{X}^{[i]} = \mu - \left(\sqrt{(n + \lambda) \Sigma} \right)_{i-n} \quad \text{for } i = n + 1, \dots, 2n$$



Sigma Points Example



Sigma Point Weights

- Weight sigma points

**for computing
the mean**

parameters

$$\begin{aligned} w_m^{[0]} &= \frac{\lambda}{n + \lambda} \\ w_c^{[0]} &= w_m^{[0]} + (1 - \alpha^2 + \beta) \\ w_m^{[i]} &= w_c^{[i]} = \frac{1}{2(n + \lambda)} \quad \text{for } i = 1, \dots, 2n \end{aligned}$$

for computing the covariance

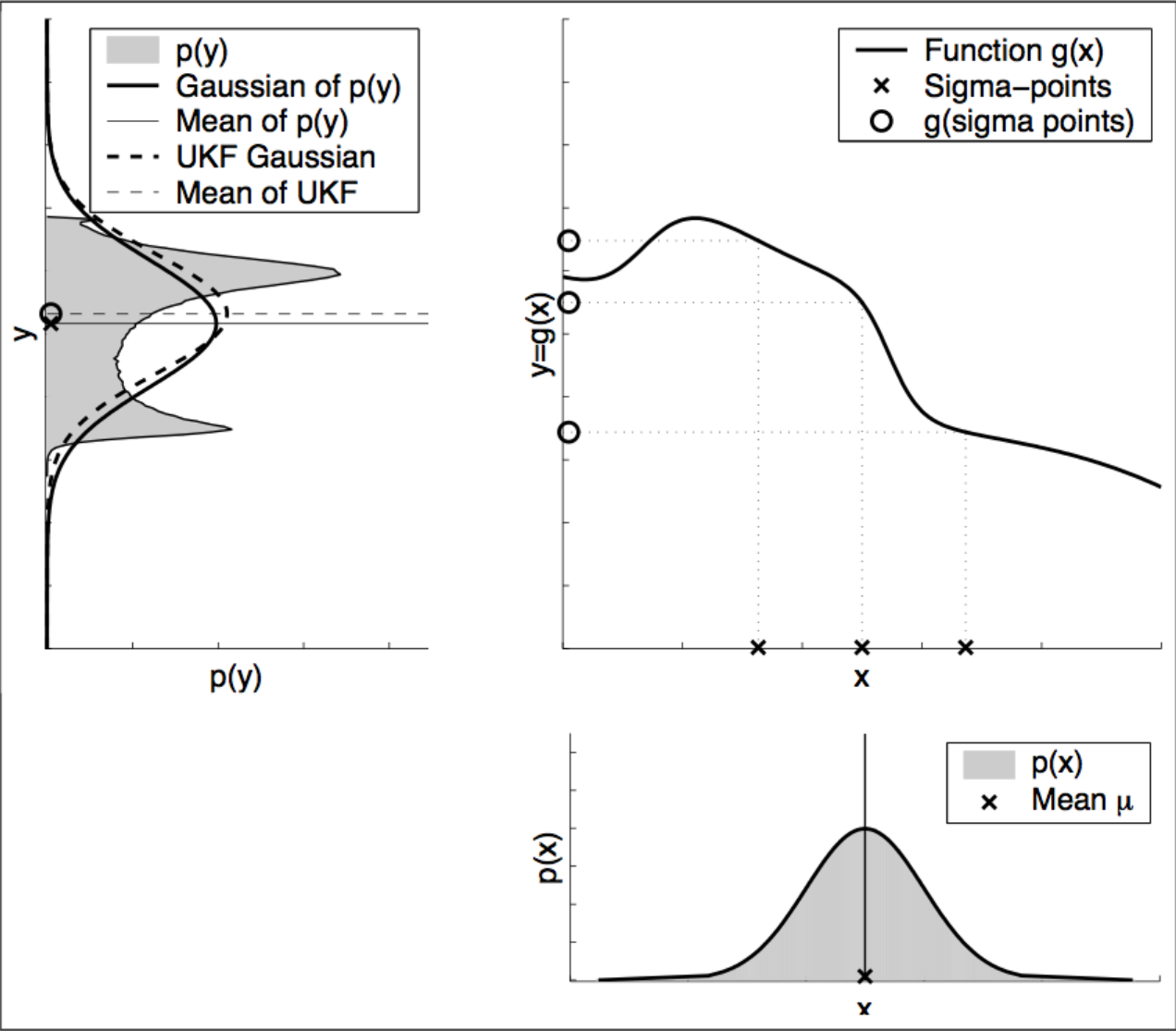
Recover the Gaussian

- Compute Gaussian from weighted and transformed points

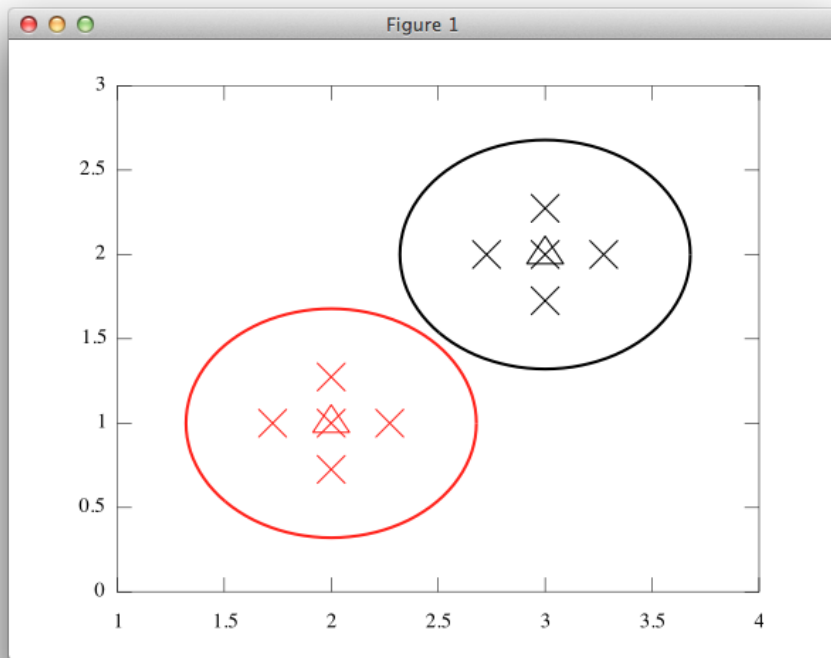
$$\mu' = \sum_{i=0}^{2n} w_m^{[i]} g(\mathcal{X}^{[i]})$$

$$\Sigma' = \sum_{i=0}^{2n} w_c^{[i]} (g(\mathcal{X}^{[i]}) - \mu')(g(\mathcal{X}^{[i]}) - \mu')^T$$

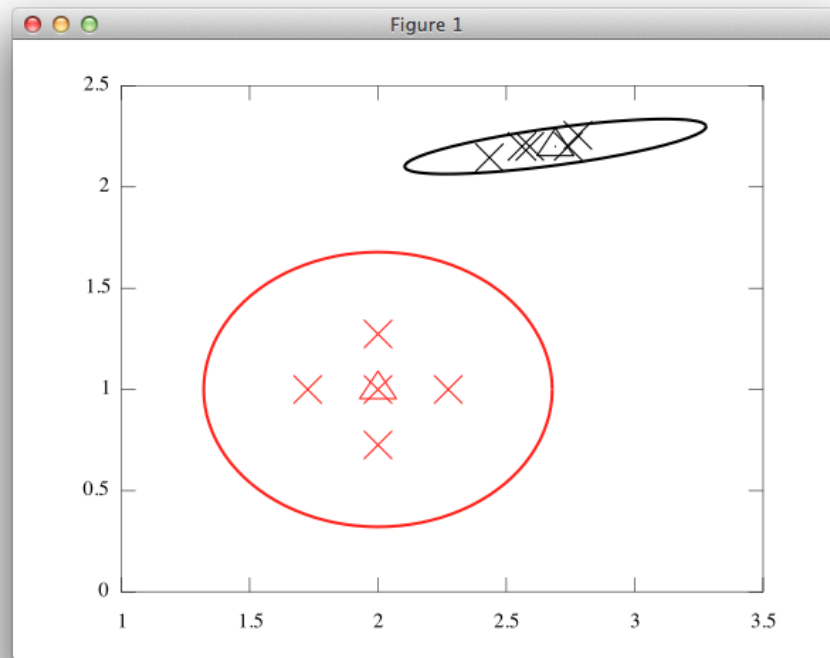
Example



Examples



$$g((x, y)^T) = \begin{pmatrix} x + 1 \\ y + 1 \end{pmatrix}^T$$



$$g((x, y)^T) = \begin{pmatrix} 1 + x + \sin(2x) + \cos(y) \\ 2 + 0.2y \end{pmatrix}^T$$

Unscented Transform Summary

- Sigma points

$$\mathcal{X}^{[0]} = \mu$$

$$\mathcal{X}^{[i]} = \mu + \left(\sqrt{(n + \lambda) \Sigma} \right)_i \quad \text{for } i = 1, \dots, n$$

$$\mathcal{X}^{[i]} = \mu - \left(\sqrt{(n + \lambda) \Sigma} \right)_{i-n} \quad \text{for } i = n + 1, \dots, 2n$$

- Weights

$$w_m^{[0]} = \frac{\lambda}{n + \lambda}$$

$$w_c^{[0]} = w_m^{[0]} + (1 - \alpha^2 + \beta)$$

$$w_m^{[i]} = w_c^{[i]} = \frac{1}{2(n + \lambda)} \quad \text{for } i = 1, \dots, 2n$$

UT Parameters

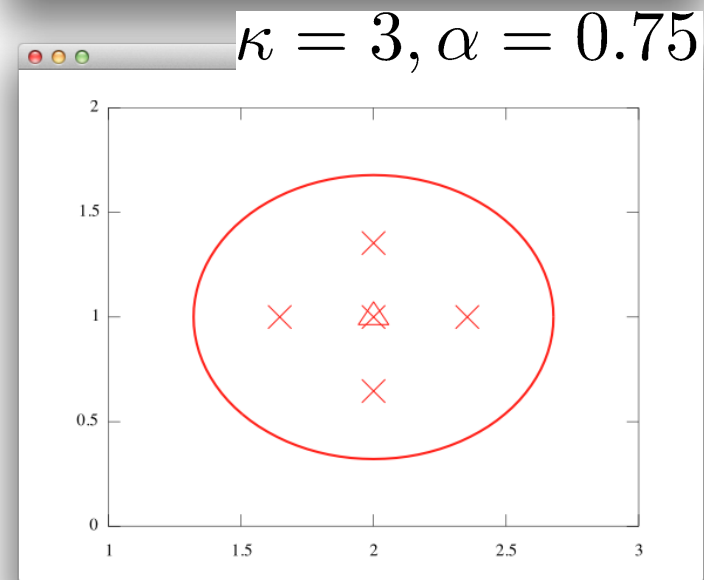
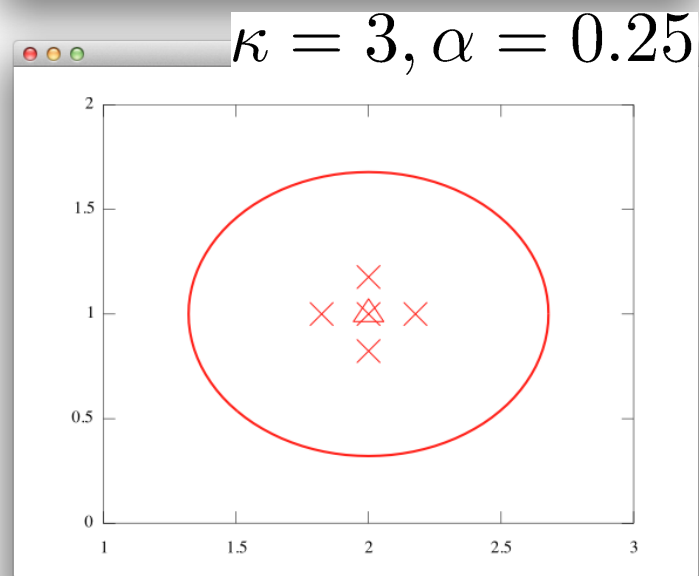
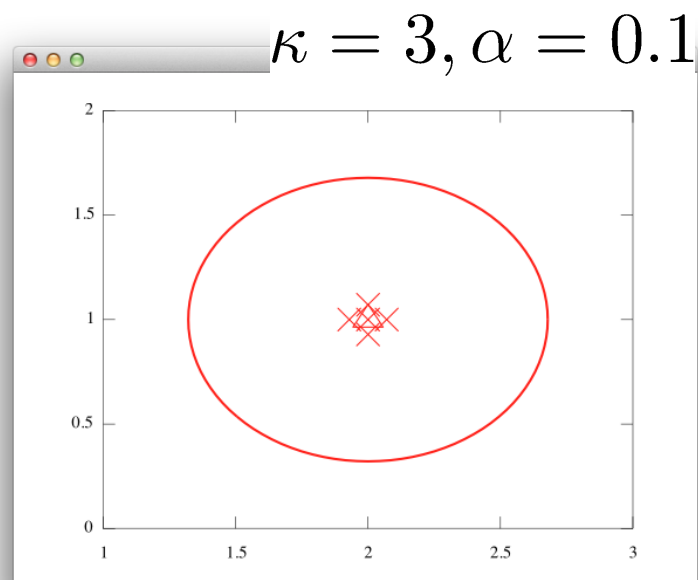
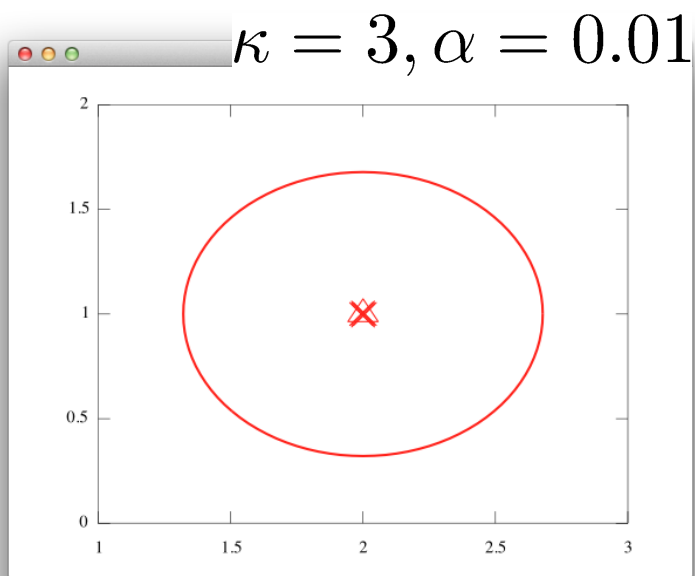
- Free parameters as there is no unique solution
- Scaled Unscented Transform suggests

$\kappa \geq 0$ Influence how far the
sigma points are
 $\alpha \in (0, 1]$ away from the mean

$$\lambda = \alpha^2(n + \kappa) - n$$

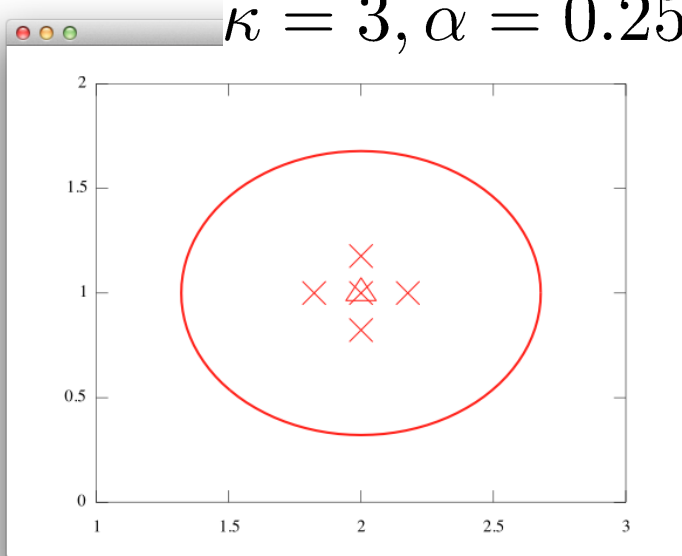
$\beta = 2$ Optimal choice for
Gaussians

Examples

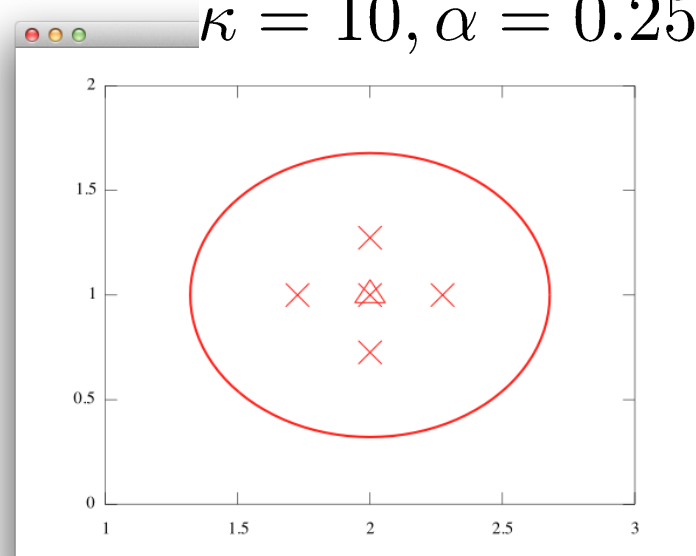


Examples

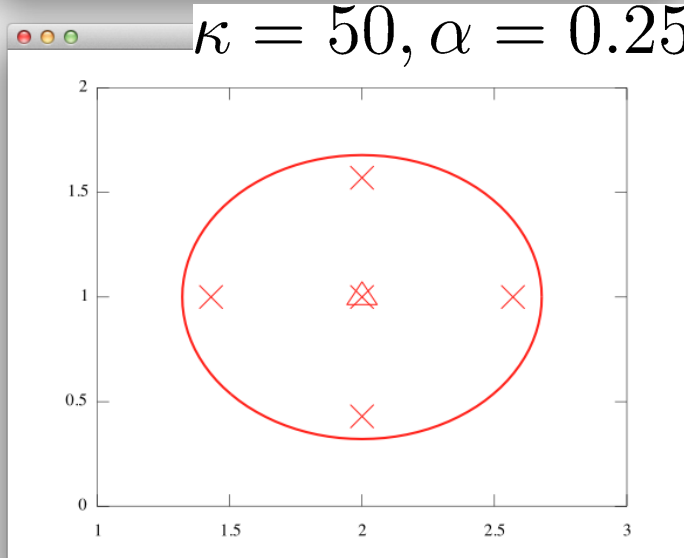
$\kappa = 3, \alpha = 0.25$



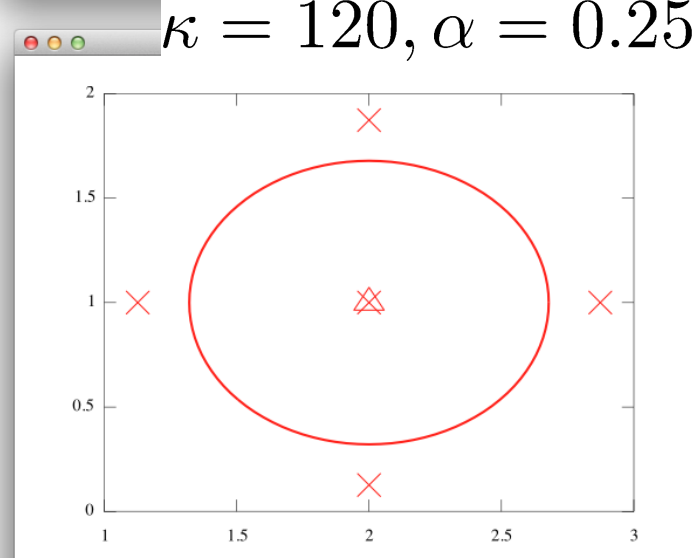
$\kappa = 10, \alpha = 0.25$



$\kappa = 50, \alpha = 0.25$



$\kappa = 120, \alpha = 0.25$



EKF Algorithm

- 1: **Extended_Kalman_filter**($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):
- 2: $\bar{\mu}_t = g(u_t, \mu_{t-1})$
- 3: $\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$
- 4: $K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$
- 5: $\mu_t = \bar{\mu}_t + K_t(z_t - h(\bar{\mu}_t))$
- 6: $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$
- 7: *return* μ_t, Σ_t

EKF to UKF – Prediction

Unscented

1: ~~Extended~~ Kalman filter($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

2: $\bar{\mu}_t =$ replace this by sigma point

3: $\bar{\Sigma}_t =$ propagation of the motion

4: $K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$

5: $\mu_t = \bar{\mu}_t + K_t(z_t - h(\bar{\mu}_t))$

6: $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$

7: return μ_t, Σ_t

UKF Algorithm – Prediction

1: **Unscented_Kalman_filter**($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

2: $\mathcal{X}_{t-1} = (\mu_{t-1} \quad \mu_{t-1} + \sqrt{(n + \lambda)\Sigma_{t-1}} \quad \mu_{t-1} - \sqrt{(n + \lambda)\Sigma_{t-1}})$

3: $\bar{\mathcal{X}}_t^* = g(u_t, \mathcal{X}_{t-1})$

4: $\bar{\mu}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{X}}_t^{*[i]}$

5: $\bar{\Sigma}_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{*[i]} - \bar{\mu}_t)(\bar{\mathcal{X}}_t^{*[i]} - \bar{\mu}_t)^T + R_t$

EKF to UKF – Correction

Unscented

1: ~~Extended~~ Kalman filter($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

2: $\bar{\mu}_t =$ replace this by sigma point

3: $\bar{\Sigma}_t =$ propagation of the motion

use sigma point propagation for the expected observation and Kalman gain

5: $\mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$

6: $\Sigma_t = \bar{\Sigma}_t - K_t S_t K_t^T$

7: return μ_t, Σ_t

UKF Algorithm – Correction (1)

$$6: \quad \bar{\mathcal{X}}_t = (\bar{\mu}_t \quad \bar{\mu}_t + \sqrt{(n + \lambda)\bar{\Sigma}_t} \quad \bar{\mu}_t - \sqrt{(n + \lambda)\bar{\Sigma}_t})$$

$$7: \quad \bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t)$$

$$8: \quad \hat{z}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{Z}}_t^{[i]}$$

$$9: \quad S_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T + Q_t$$

$$10: \quad \bar{\Sigma}_t^{x,z} = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{[i]} - \bar{\mu}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T$$

$$11: \quad K_t = \bar{\Sigma}_t^{x,z} S_t^{-1}$$

UKF Algorithm – Correction (1)

$$6: \quad \bar{\mathcal{X}}_t = (\bar{\mu}_t \quad \bar{\mu}_t + \sqrt{(n + \lambda)\bar{\Sigma}_t} \quad \bar{\mu}_t - \sqrt{(n + \lambda)\bar{\Sigma}_t})$$

$$7: \quad \bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t)$$

$$8: \quad \hat{z}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{Z}}_t^{[i]}$$

$$9: \quad S_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T + Q_t$$

$$10: \quad \bar{\Sigma}_t^{x,z} = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{[i]} - \bar{\mu}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T$$

$$11: \quad K_t = \bar{\Sigma}_t^{x,z} S_t^{-1}$$

$$K_t = \underbrace{\bar{\Sigma}_t^{x,z}}_{\substack{\bar{\Sigma}_t \\ H_t^T}} \underbrace{(H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}}_{S_t} \quad (\text{from EKF})$$

UKF Algorithm – Correction (2)

$$6: \quad \bar{\mathcal{X}}_t = (\bar{\mu}_t \quad \bar{\mu}_t + \sqrt{(n + \lambda)\bar{\Sigma}_t} \quad \bar{\mu}_t - \sqrt{(n + \lambda)\bar{\Sigma}_t})$$

$$7: \quad \bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t)$$

$$8: \quad \hat{z}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{Z}}_t^{[i]}$$

$$9: \quad S_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T + Q_t$$

$$10: \quad \bar{\Sigma}_t^{x,z} = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{[i]} - \bar{\mu}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T$$

$$11: \quad K_t = \bar{\Sigma}_t^{x,z} S_t^{-1}$$

$$12: \quad \mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$$

$$13: \quad \Sigma_t = \bar{\Sigma}_t - K_t S_t K_t^T$$

$$14: \quad \text{return } \mu_t, \Sigma_t$$

UKF Algorithm – Correction (2)

$$6: \quad \bar{\mathcal{X}}_t = (\bar{\mu}_t \quad \bar{\mu}_t + \sqrt{(n + \lambda)\bar{\Sigma}_t} \quad \bar{\mu}_t - \sqrt{(n + \lambda)\bar{\Sigma}_t})$$

$$7: \quad \bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t)$$

$$8: \quad \hat{z}_t = \sum_{i=0}^{2n} w_m^{[i]} \bar{\mathcal{Z}}_t^{[i]}$$

$$9: \quad S_t = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T + Q_t$$

$$10: \quad \bar{\Sigma}_t^{x,z} = \sum_{i=0}^{2n} w_c^{[i]} (\bar{\mathcal{X}}_t^{[i]} - \bar{\mu}_t)(\bar{\mathcal{Z}}_t^{[i]} - \hat{z}_t)^T$$

$$11: \quad K_t = \bar{\Sigma}_t^{x,z} S_t^{-1}$$

$$12: \quad \mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$$

$$13: \quad \Sigma_t = \bar{\Sigma}_t - K_t S_t K_t^T$$

$$14: \quad \text{return } \mu_t, \Sigma_t$$

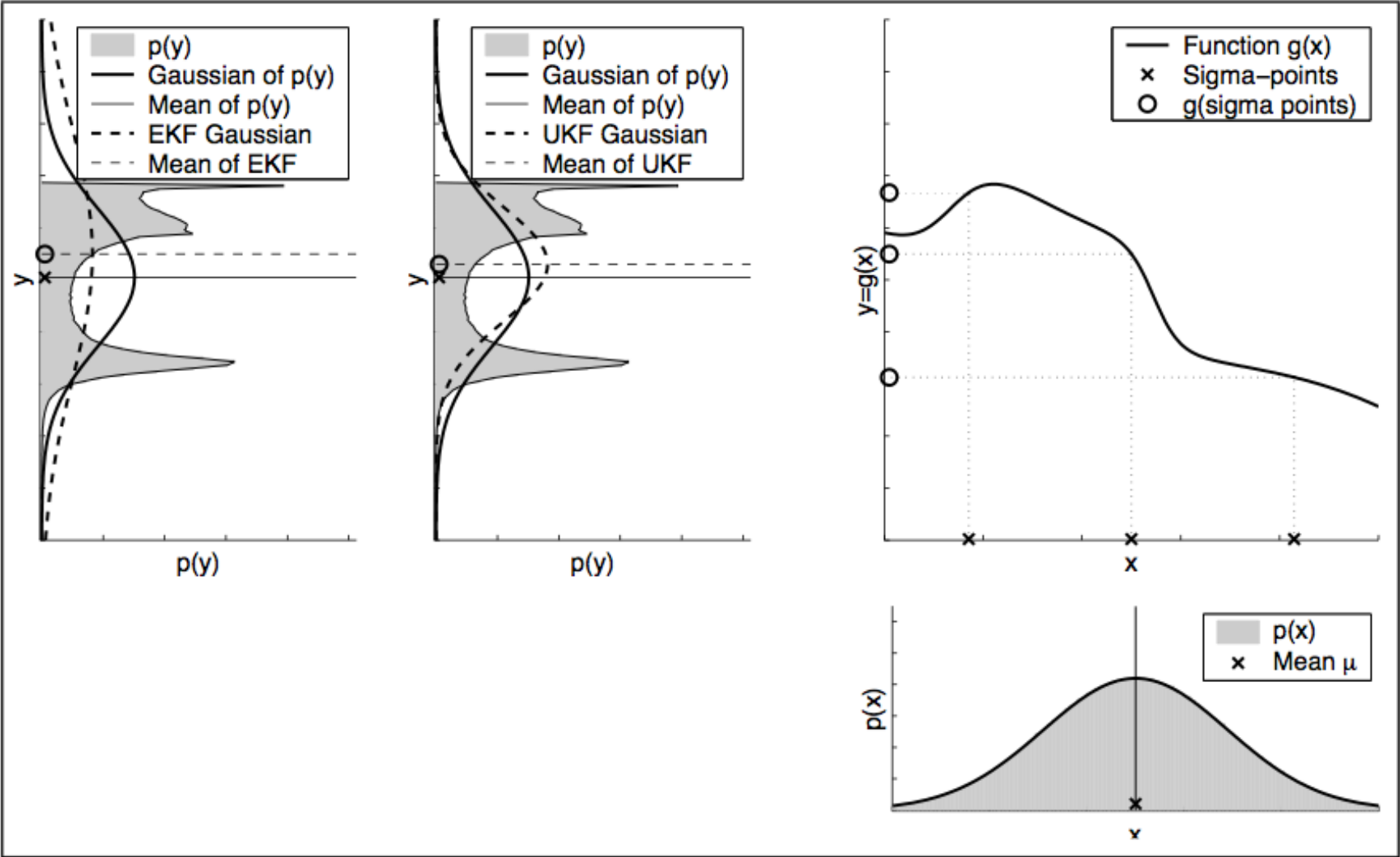
$$\begin{aligned} \Sigma_t &= (I - K_t H_t) \bar{\Sigma}_t \\ &= \bar{\Sigma}_t - K_t H_t \bar{\Sigma}_t \\ &= \bar{\Sigma}_t - K_t (\Sigma^{x,z})^T \\ &= \bar{\Sigma}_t - K_t (\Sigma^{x,z} s_t^{-1} S_t)^T \\ &= \bar{\Sigma}_t - K_t (K_t S_t)^T \\ &= \bar{\Sigma}_t - K_t S_t^T K_t^T \\ &= \bar{\Sigma}_t - K_t S_t K_t^T \end{aligned}$$

(see next slide)

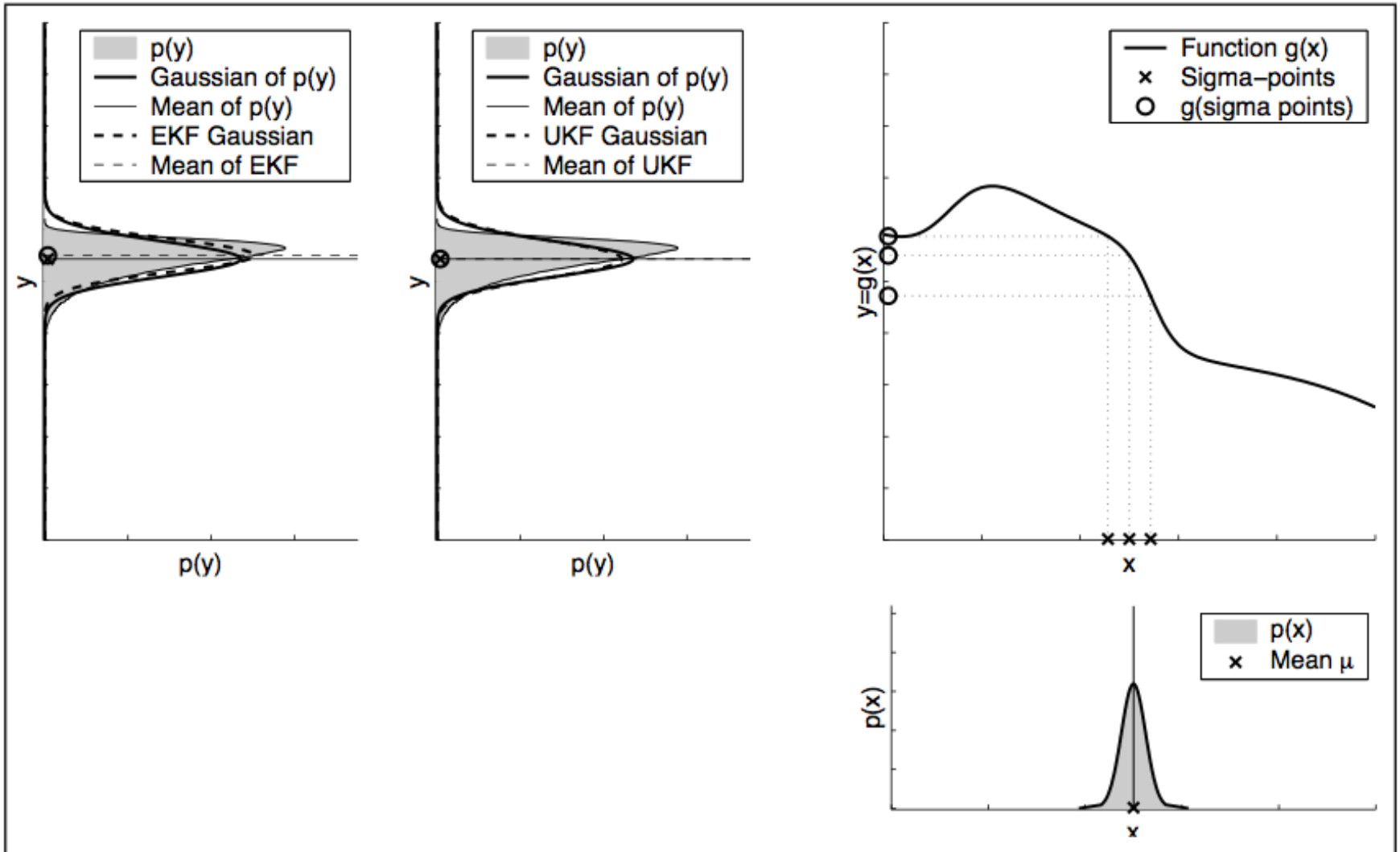
From EKF to UKF – Computing the Covariance

$$\begin{aligned}\Sigma_t &= (I - K_t H_t) \bar{\Sigma}_t \\ &= \bar{\Sigma}_t - K_t \underline{H_t \bar{\Sigma}_t} \\ &= \bar{\Sigma}_t - K_t \left(\bar{\Sigma}^{x,z} \right)^T \\ &= \bar{\Sigma}_t - K_t \left(\underline{\bar{\Sigma}^{x,z} S_t^{-1} S_t} \right)^T \\ &= \bar{\Sigma}_t - K_t \left(\underline{K_t S_t} \right)^T \\ &= \bar{\Sigma}_t - K_t S_t^T K_t^T \\ &= \bar{\Sigma}_t - K_t S_t K_t^T\end{aligned}$$

UKF vs. EKF

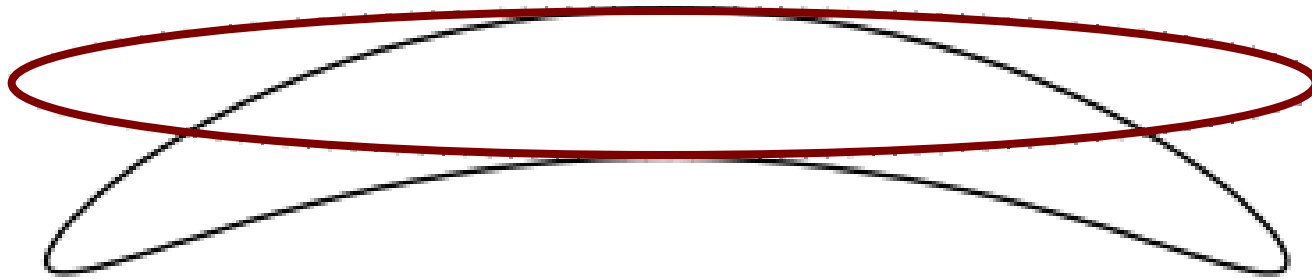


UKF vs. EKF (Small Covariance)

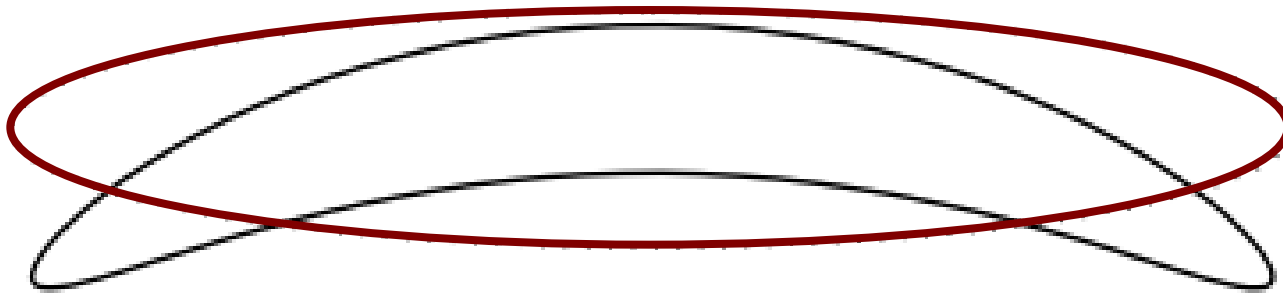


UKF vs. EKF – Banana Shape

EKF approximation

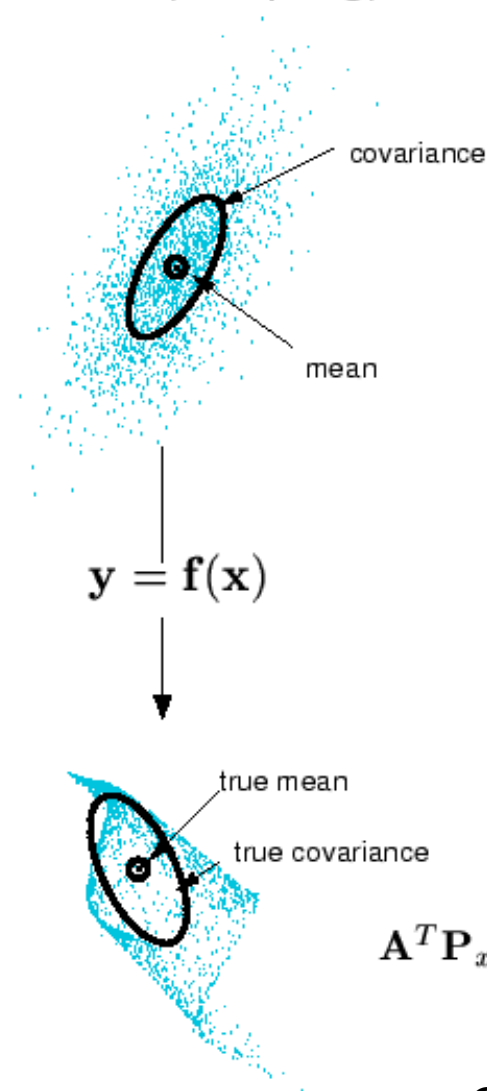


UKF approximation

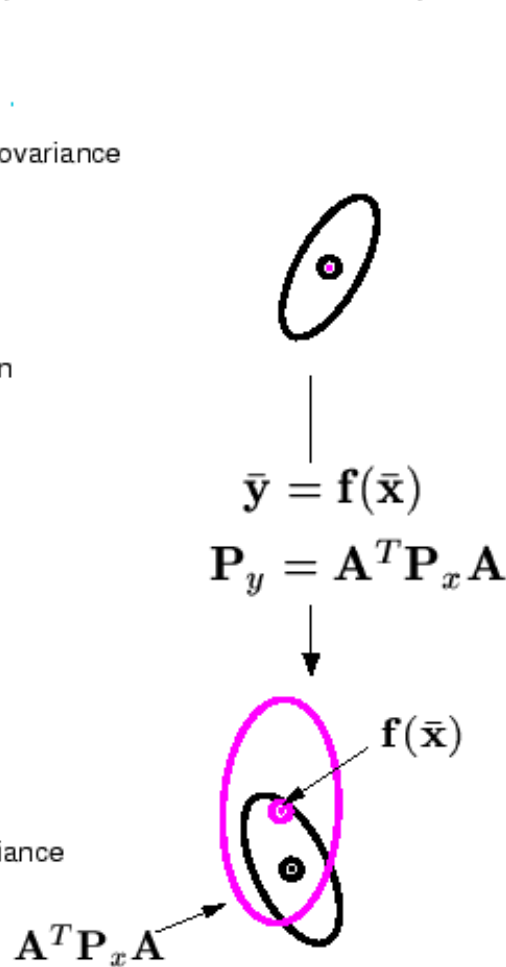


UKF vs. EKF

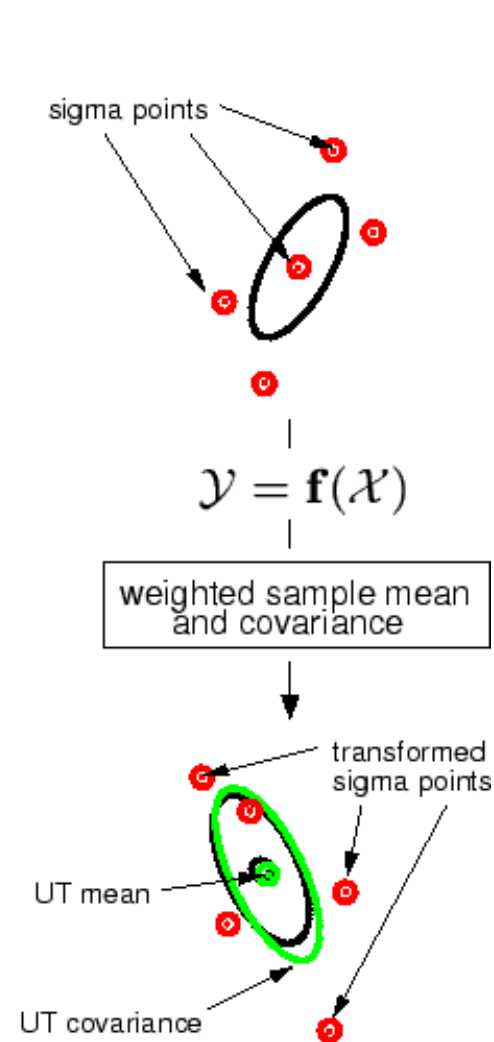
Actual (sampling)



Linearized (EKF)



UT



UT/UKF Summary

- Unscented transforms as an alternative to linearization
- UT is a better approximation than Taylor expansion
- UT uses sigma point propagation
- Free parameters in UT
- UKF uses the UT in the prediction and correction step

UKF vs. EKF

- Same results as EKF for linear models
- Better approximation than EKF for non-linear models
- Differences often “somewhat small”
- No Jacobians needed for the UKF
- Same complexity class
- Slightly slower than the EKF
- Still restricted to Gaussian distributions

Literature

Unscented Transform and UKF

- Thrun et al.: “Probabilistic Robotics”, Chapter 3.4
- “A New Extension of the Kalman Filter to Nonlinear Systems” by Julier and Uhlmann, 1995
- “Dynamische Zustandsschätzung” by Fränken, 2006, pages 31-34

Slide Information

- These slides have been created by Cyrill Stachniss as part of the robot mapping course taught in 2012/13 and 2013/14. I created this set of slides partially extending existing material of Edwin Olson, Pratik Agarwal, and myself.
- I tried to acknowledge all people that contributed image or video material. In case I missed something, please let me know. If you adapt this course material, please make sure you keep the acknowledgements.
- Feel free to use and change the slides. If you use them, I would appreciate an acknowledgement as well. To satisfy my own curiosity, I appreciate a short email notice in case you use the material in your course.
- My video recordings are available through YouTube:
http://www.youtube.com/playlist?list=PLgnQpQtFTOGQrZ4O5QzbIHgl3b1JHimN_&feature=g-list