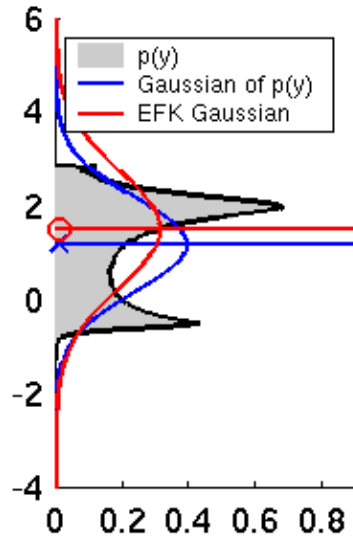
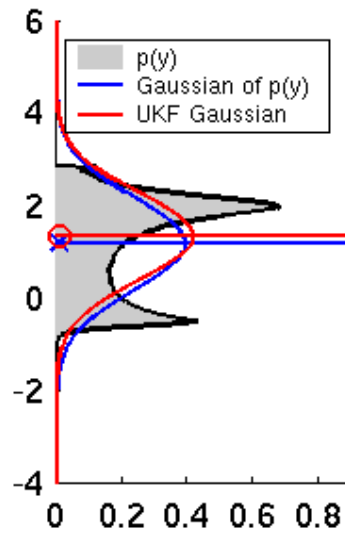


# Unscented Kalman Filter

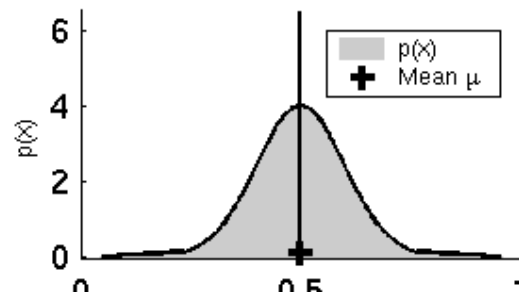
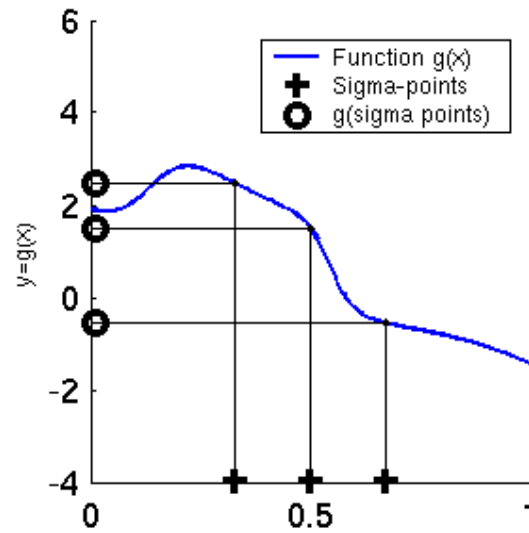
# Linearization via Unscented Transform



EKF



UKF



# Unscented Transform

- intuition: it should be easier to approximate a given distribution than it is to approximate an arbitrary non-linear function
  - it is easy to transform a point through a non-linear function
  - use a set of points that capture the mean and covariance of the distribution, transform the points through the non-linear function, then compute the (weighted) mean and covariance of the transformed points

# Empirical transformation of a Gaussian random variable

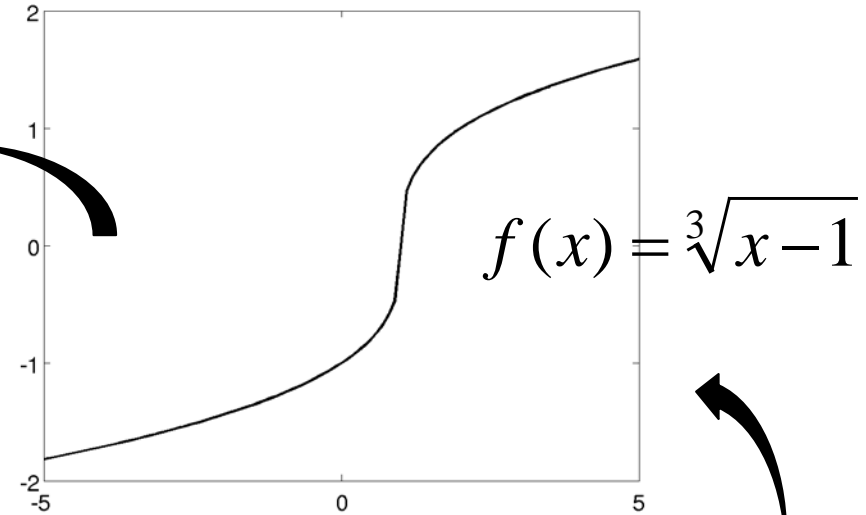
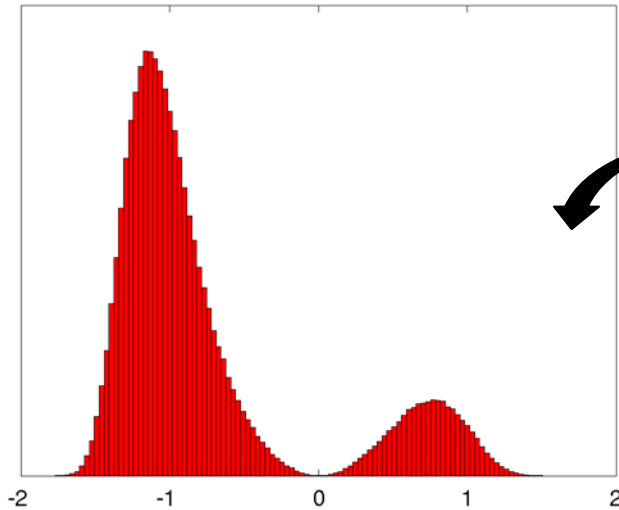
```
% generate 500,000 samples from N(0, 1)
x = randn(1, 500000);
```

```
% draw the histogram of x
xc = -5:0.2:5;
nx = hist(x, xc);
bar(xc, nx, 1);
```

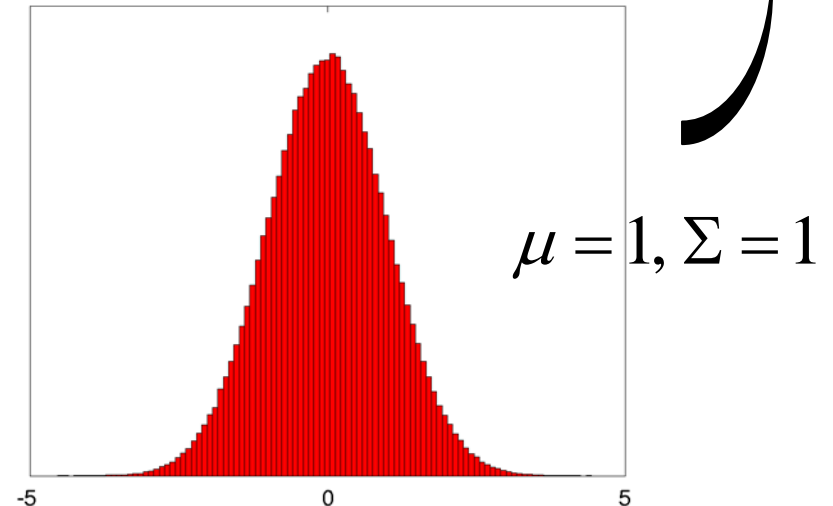
```
% transform each sample by f(x)
y = nthroot(x - 1, 3);
```

```
% draw the histogram of y
xc = -2:0.1:2;
ny = hist(y, xc);
bar(xc, ny, 1);
```

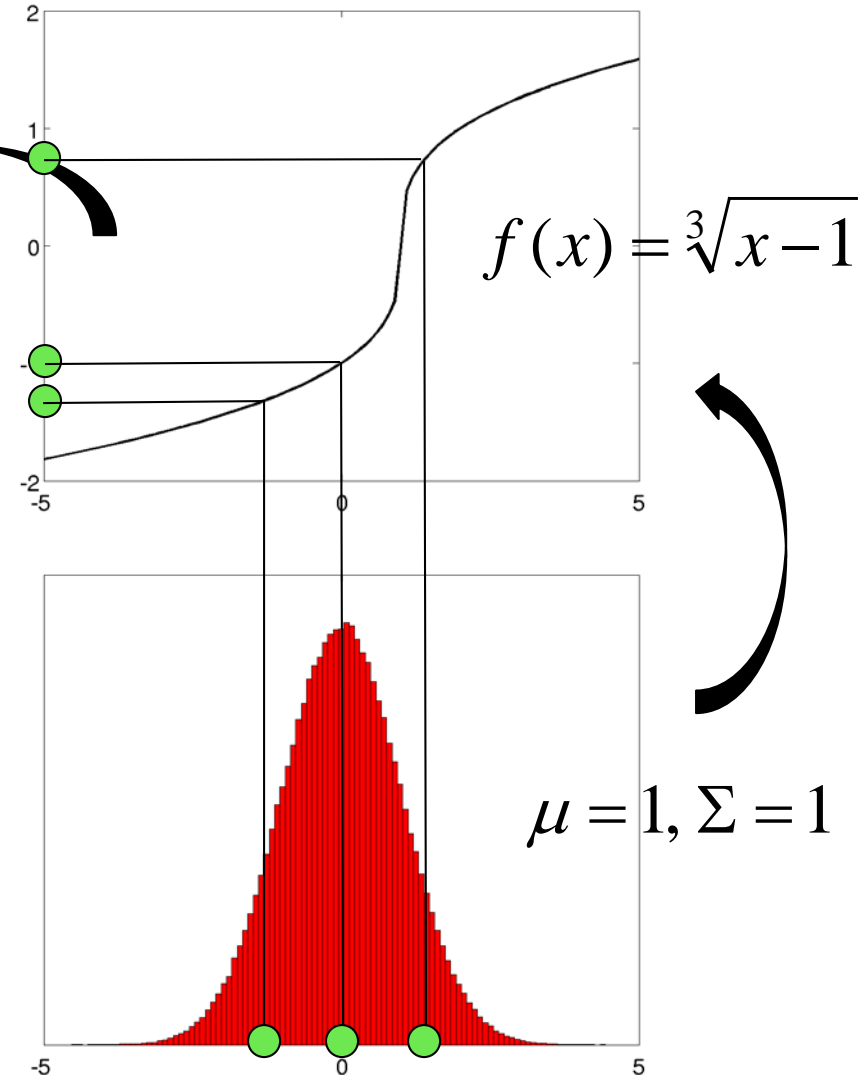
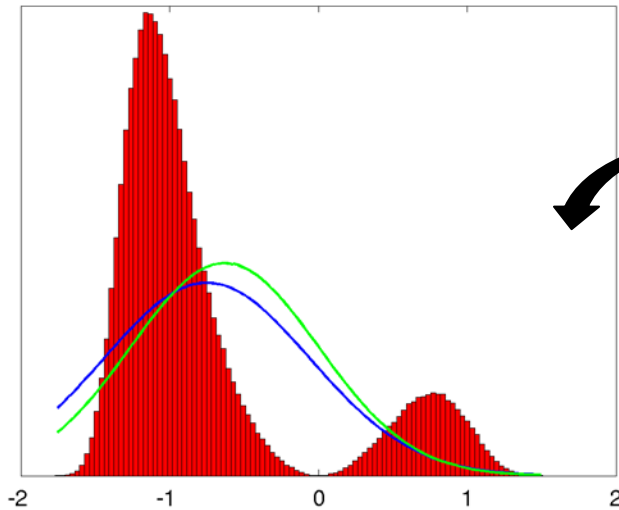
# Empirical transformation of a Gaussian random variable



-transform 500,000 random samples through a non-linear function

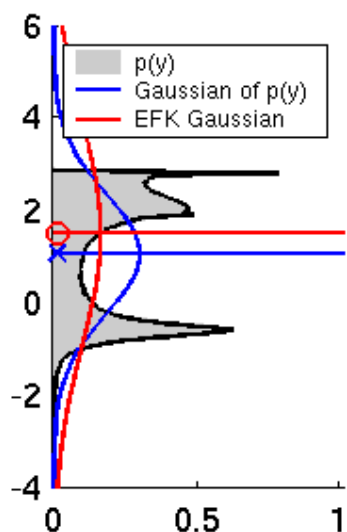


# Unscented Transform

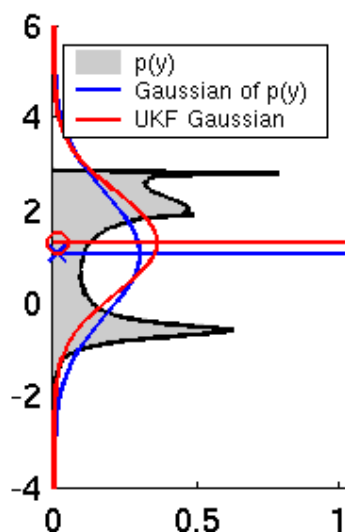


- transform 3 carefully chosen samples through a non-linear function
- samples are called sigma points

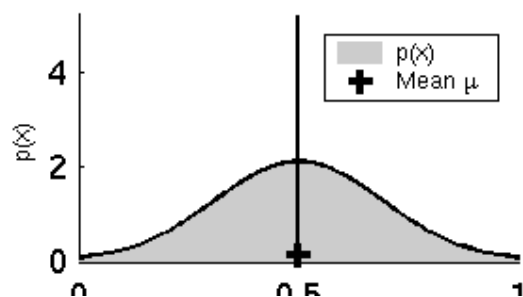
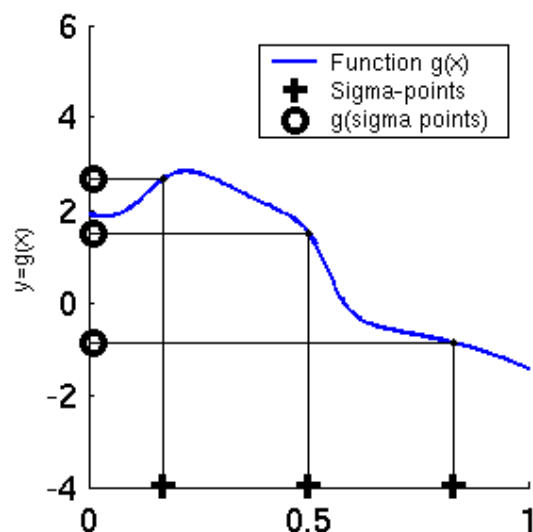
# UKF Sigma-Point Estimate (2)



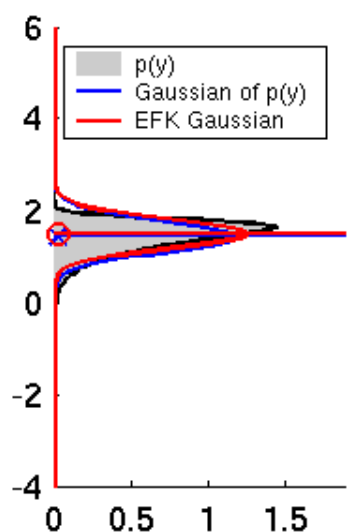
EKF



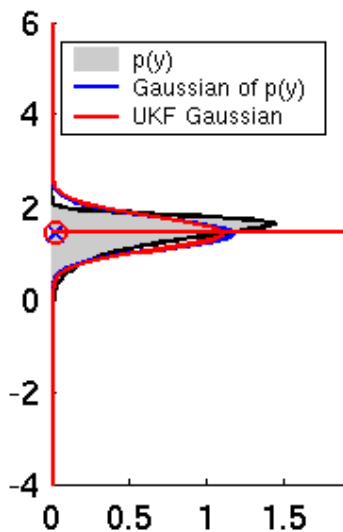
UKF



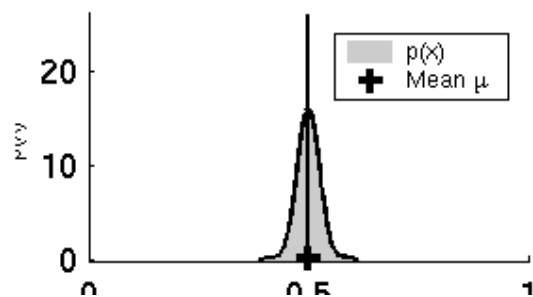
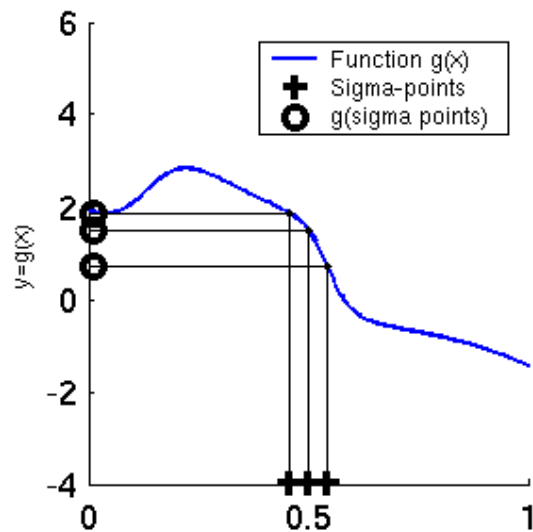
# UKF Sigma-Point Estimate (3)



EKF

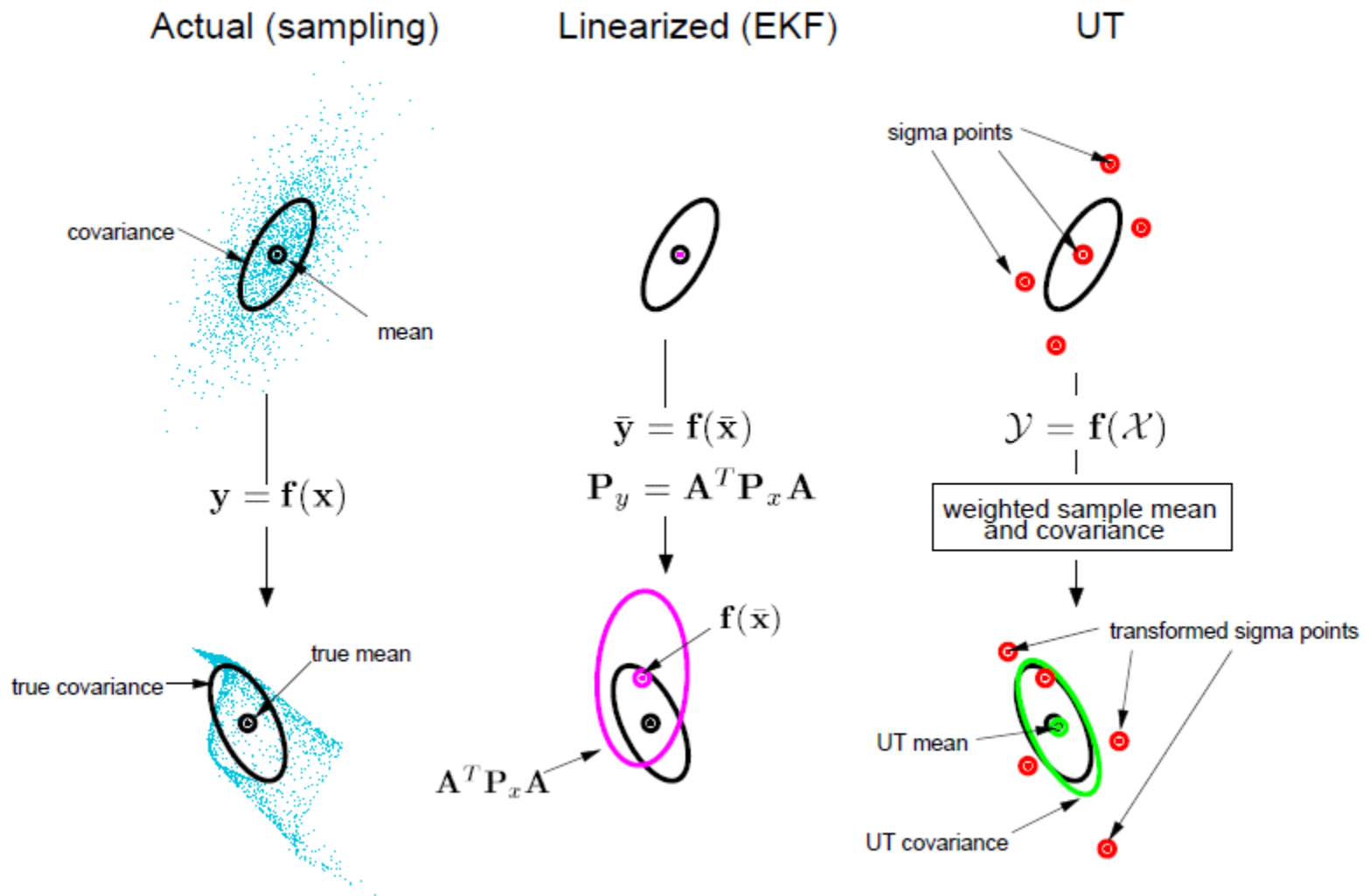


UKF





# UKF Sigma-Point Estimate (4)



# Unscented Transform

- for an n-dimensional Gaussian with mean  $\mu$  and covariance  $\Sigma$ , the unscented transform uses  $2n+1$  sigma points (and associated weights)

Sigma points

$$\chi^0 = \mu$$

$$\chi^i = \mu \pm \left( \sqrt{(n + \lambda) \Sigma} \right)_i$$

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

Weights

$$w_m^0 = \frac{\lambda}{n + \lambda} \quad w_c^0 = \frac{\lambda}{n + \lambda} + (1 - \alpha^2 + \beta)$$

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

# Unscented Transform

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

- choose  $\kappa \geq 0$  to guarantee a “reasonable” covariance matrix
  - value is not critical, so choose  $\kappa = 0$  by default
- choose  $0 \leq \alpha \leq 1$ 
  - controls the spread of the sigma point distribution; should be small when nonlinearities are strong
- choose  $\beta \geq 0$ 
  - $\beta = 2$  is optimal if distribution is Gaussian

# Unscented Transform

Pass sigma points through nonlinear function

$$\psi^i = g(\chi^i)$$

Recover mean and covariance

$$\mu' = \sum_{i=0}^{2n} w_m^i \psi^i$$

$$\Sigma' = \sum_{i=0}^{2n} w_c^i (\psi^i - \mu)(\psi^i - \mu)^T$$

## UKF\_localization ( $\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, m$ ):

### Prediction:

$$R_t = \begin{pmatrix} (\alpha_1 v_t^2 + \alpha_2 \omega_t^2)^2 & 0 \\ 0 & (\alpha_3 v_t^2 + \alpha_4 \omega_t^2)^2 \end{pmatrix}$$

Control noise

$$Q_t = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_r^2 \end{pmatrix}$$

Measurement noise

$$\mu_{t-1}^a = \begin{pmatrix} \mu_{t-1}^T & (0 \ 0)^T & (0 \ 0)^T \end{pmatrix}$$

Augmented state mean

$$\Sigma_{t-1}^a = \begin{pmatrix} \Sigma_{t-1} & 0 & 0 \\ 0 & R_t & 0 \\ 0 & 0 & Q_t \end{pmatrix}$$

Augmented covariance

$$\chi_{t-1}^a = \begin{pmatrix} \mu_{t-1}^a & \mu_{t-1}^a + \gamma \sqrt{\Sigma_{t-1}^a} & \mu_{t-1}^a - \gamma \sqrt{\Sigma_{t-1}^a} \end{pmatrix}$$

Sigma points

$$\bar{\chi}_t^x = g(u_t + \chi_t^u, \chi_{t-1}^x)$$

Prediction of sigma points

$$\bar{\mu}_t = \sum_{i=0}^{2L} w_m^i \chi_{i,t}^x$$

Predicted mean

$$\bar{\Sigma}_t = \sum_{i=0}^{2L} w_c^i (\chi_{i,t}^x - \bar{\mu}_t)(\chi_{i,t}^x - \bar{\mu}_t)^T$$

Predicted covariance

## UKF\_localization ( $\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, m$ ):

### Correction:

$$\bar{Z}_t = h(\chi_t^x) + \chi_t^z$$

Measurement sigma points

$$\hat{z}_t = \sum_{i=0}^{2L} w_m^i \bar{Z}_{i,t}$$

Predicted measurement mean

$$S_t = \sum_{i=0}^{2L} w_c^i (\bar{Z}_{i,t} - \hat{z}_t)(\bar{Z}_{i,t} - \hat{z}_t)^T$$

Pred. measurement covariance

$$\Sigma_t^{x,z} = \sum_{i=0}^{2L} w_c^i (\bar{\chi}_{i,t}^x - \bar{\mu}_t)(\bar{Z}_{i,t} - \hat{z}_t)^T$$

Cross-covariance

$$K_t = \Sigma_t^{x,z} S_t^{-1}$$

Kalman gain

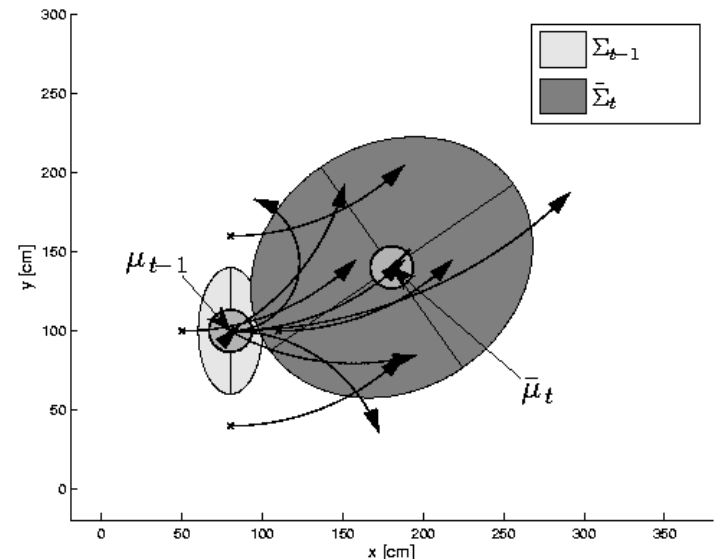
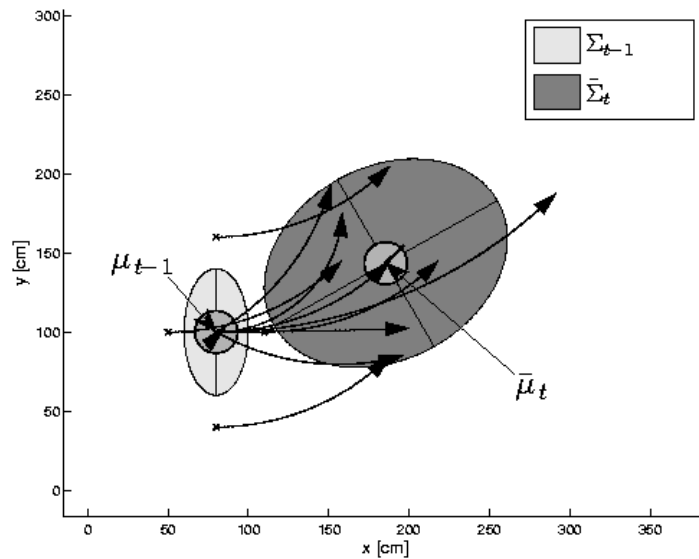
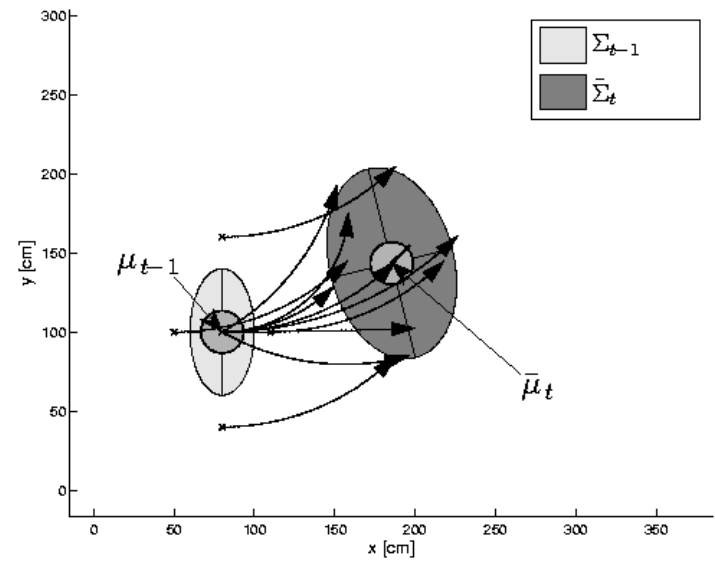
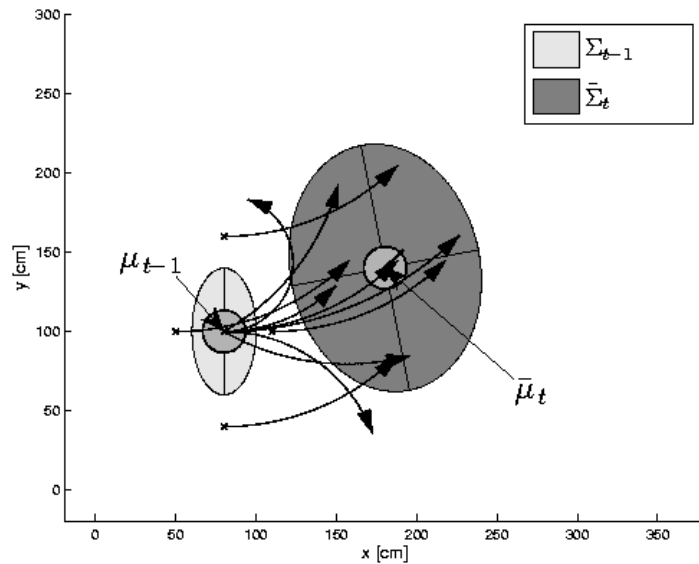
$$\mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$$

Updated mean

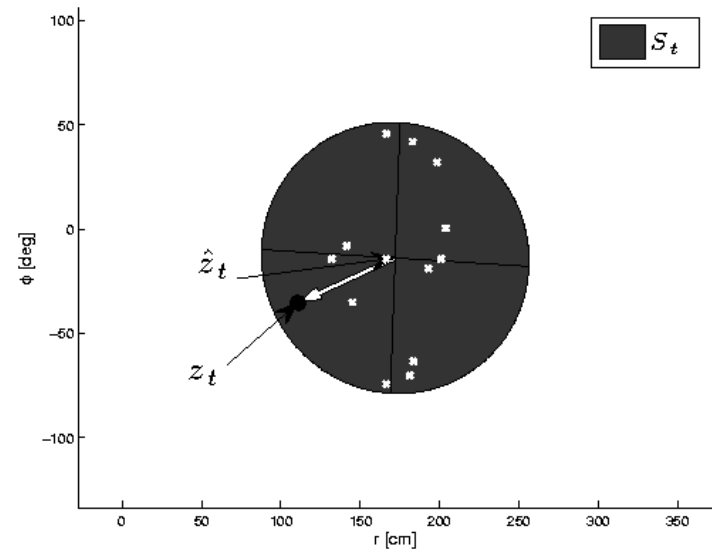
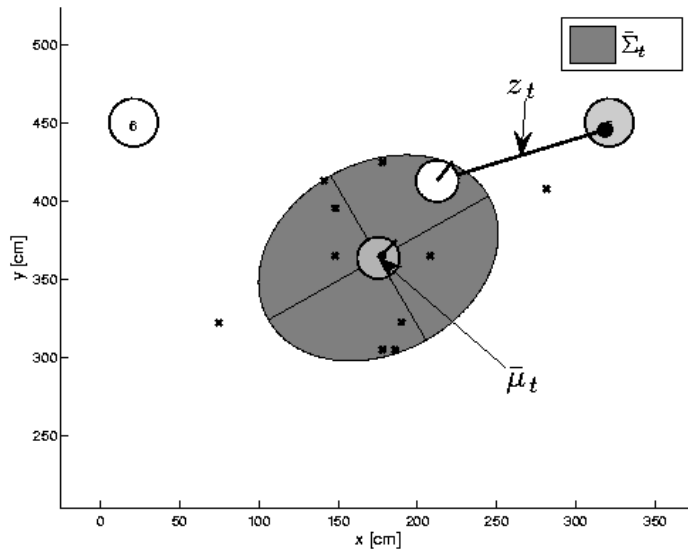
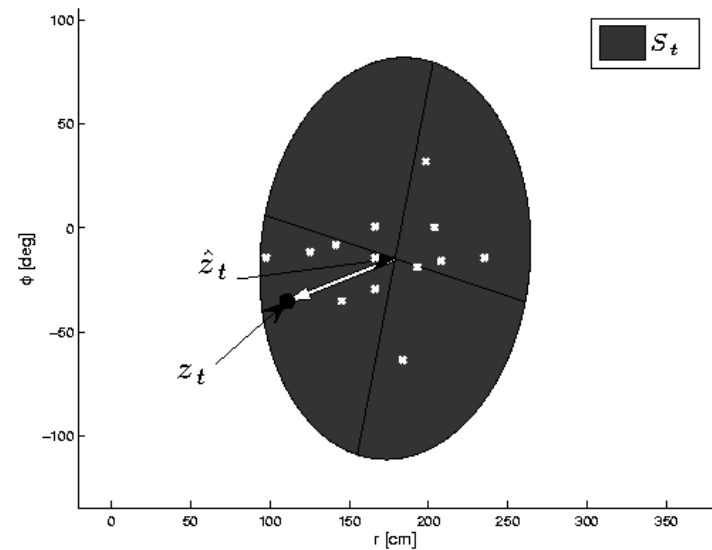
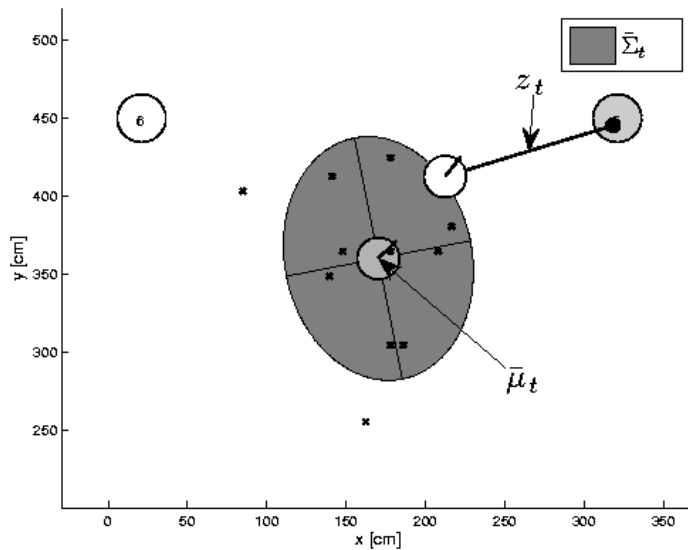
$$\Sigma_t = \bar{\Sigma}_t - K_t S_t K_t^T$$

Updated covariance

# UKF Prediction Step

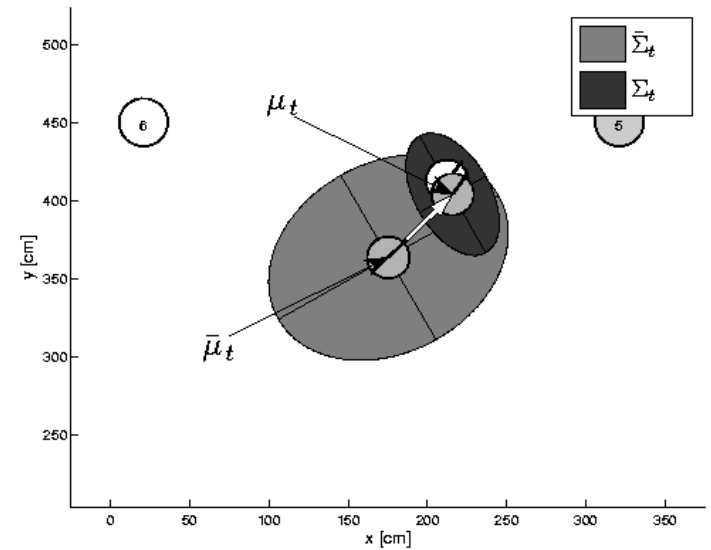
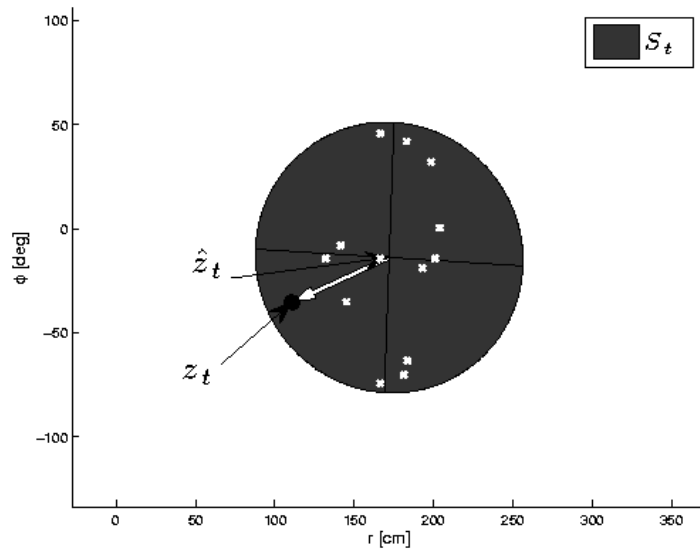
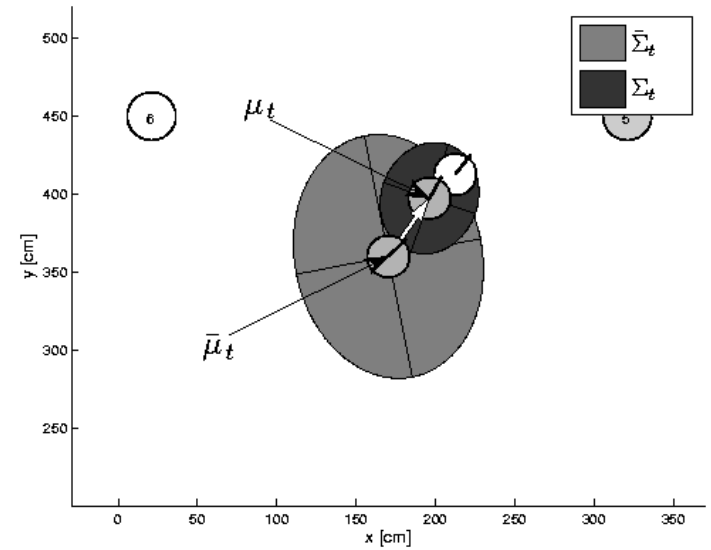
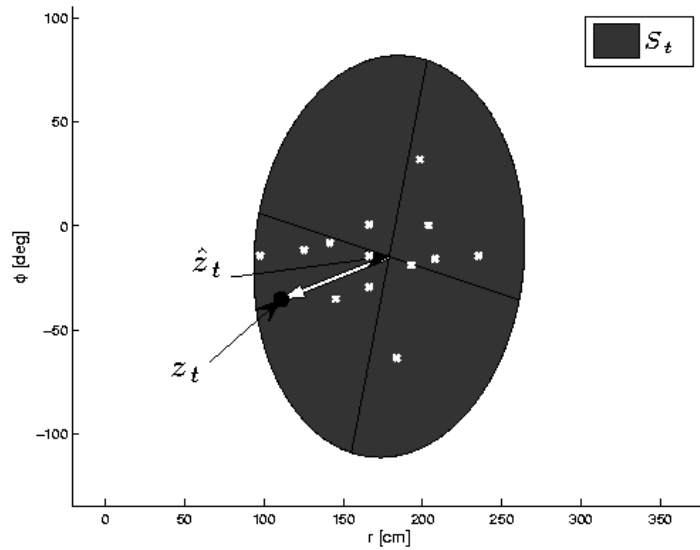


# UKF Observation Prediction Step

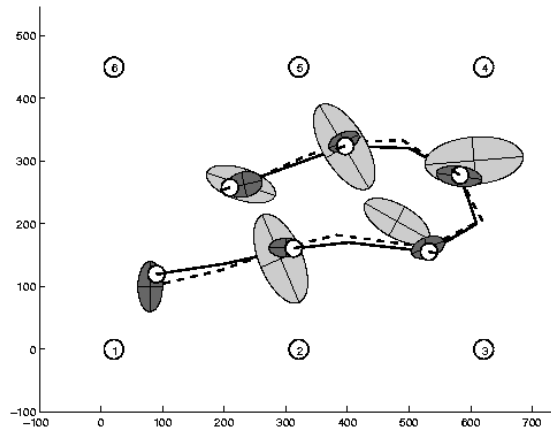




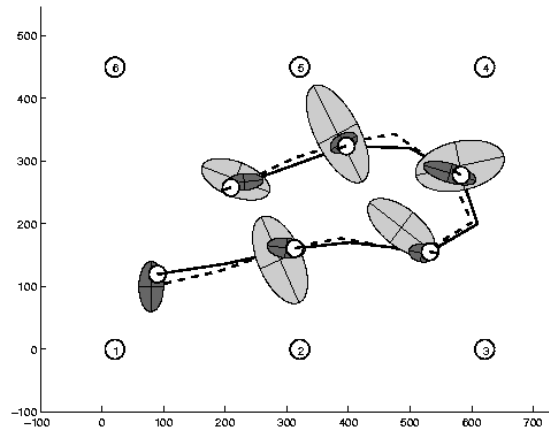
# UKF Correction Step



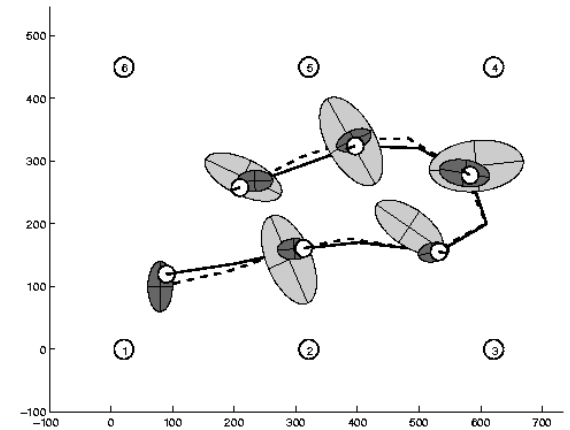
# Estimation Sequence



EKF

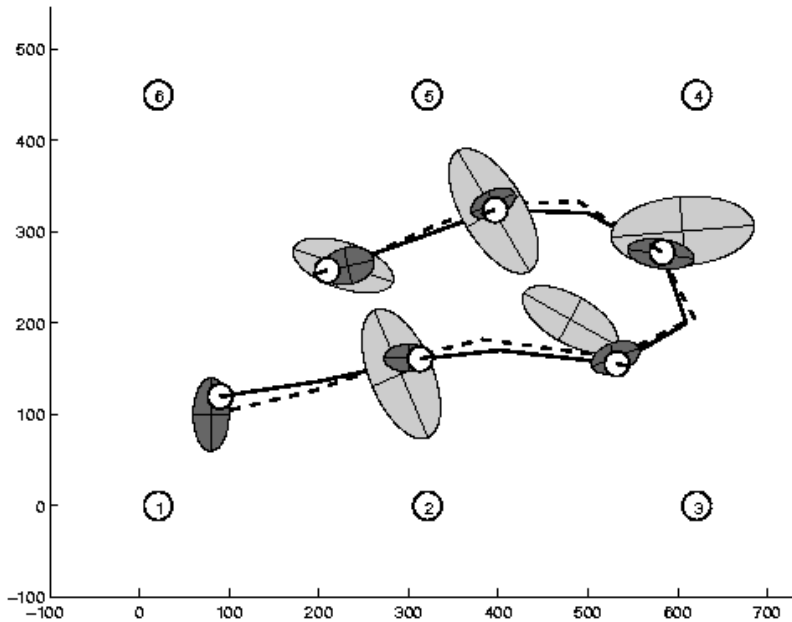


PF

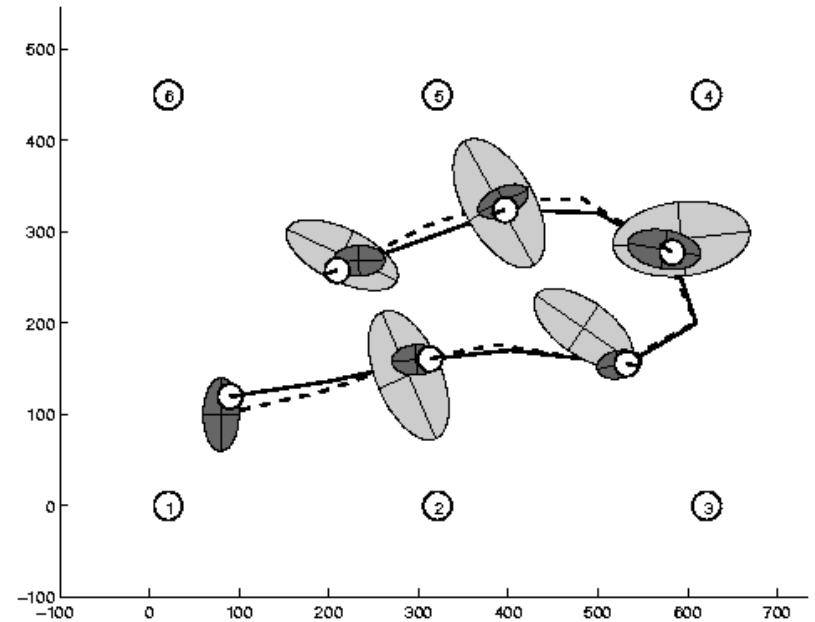


UKF

# Estimation Sequence



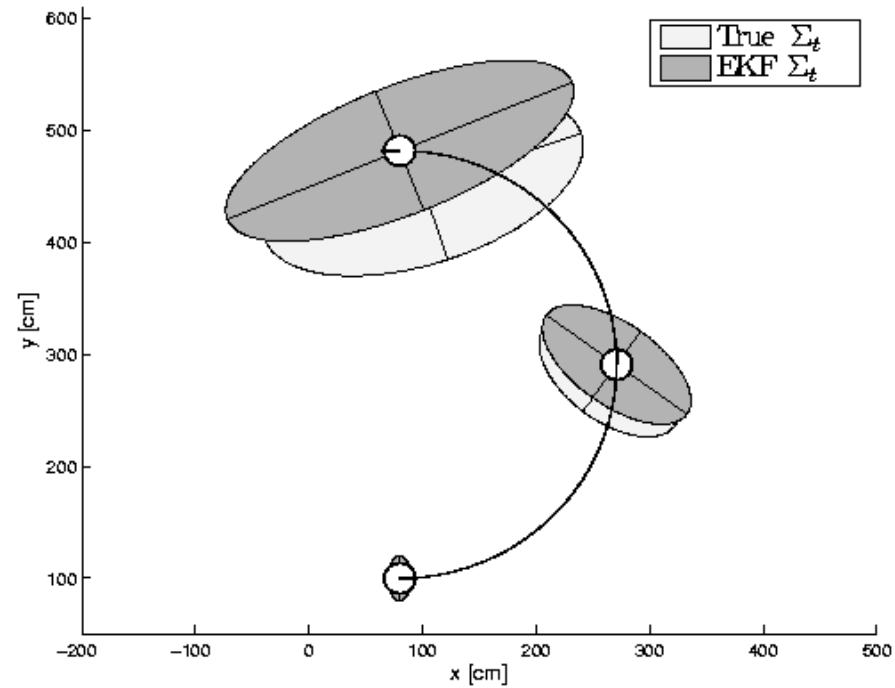
EKF



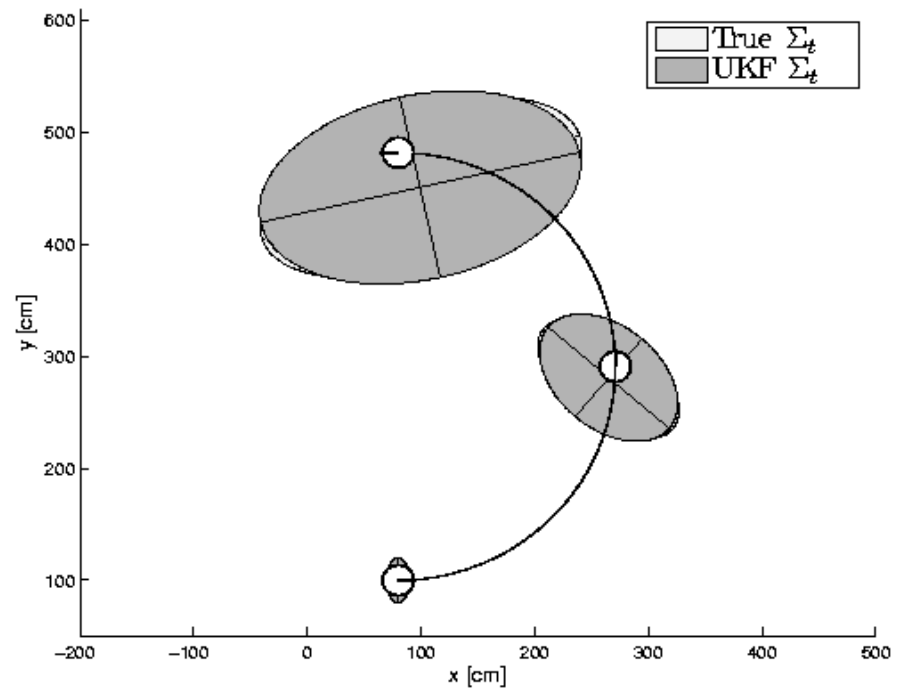
UKF

# Prediction Quality

velocity\_motion\_model



EKF



UKF

# UKF Summary

- **Highly efficient:** Same complexity as EKF, with a constant factor slower in typical practical applications
- **Better linearization than EKF:** Accurate in first two terms of Taylor expansion (EKF only first term)
- **Derivative-free:** No Jacobians needed
- **Still not optimal!**