INTELLIGENT ROBOTS

CHAPTER 7: KALMAN FILTER

Outline

- Background
- Gaussians
- Kalman Filters
- Extended Kalman Filters

Bayes Filters

1. Algorithm **Bayes_filter**(*Bel(x),d*):
2. $\eta=0$
3. If *d* is a perceptual data item *z* then
4. For all *x* do
5. $Bel'(x) = P(z \mid x) Bel(x)$
6. $\eta = \eta + Bel'(x)$
7. For all *x* do
8. $Bel'(x) = \eta^{-1} Bel'(x)$
9. Else if *d* is an action data item *u* then
10. For all *x* do
11. $Bel'(x) = \int P(x \mid u, x')\, Bel(x')\, dx'$
12. Return *Bel'(x)*

$$Bel(x_t) = \eta\ P(z_t \mid x_t) \int P(x_t \mid u_t, x_{t-1})\, Bel(x_{t-1})\, dx_{t-1}$$

Background

- Bayes filter with **Gaussians**
- Developed in the late 1950's
- Most relevant Bayes filter variant in practice
- Applications range from economics, wheather forecasting, satellite navigation to robotics and many more.

- The Kalman filter "algorithm" is a bunch of **matrix multiplications**!

Outline

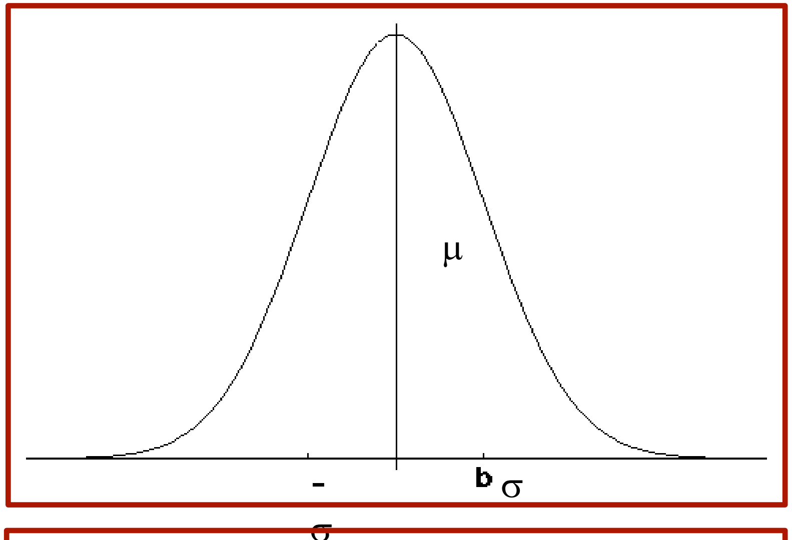
- Problem Statement
- Gaussians
- Kalman Filters
- Extended Kalman Filters

Gaussians

$$p(x) \sim N(\mu, \sigma^2):$$

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}}$$

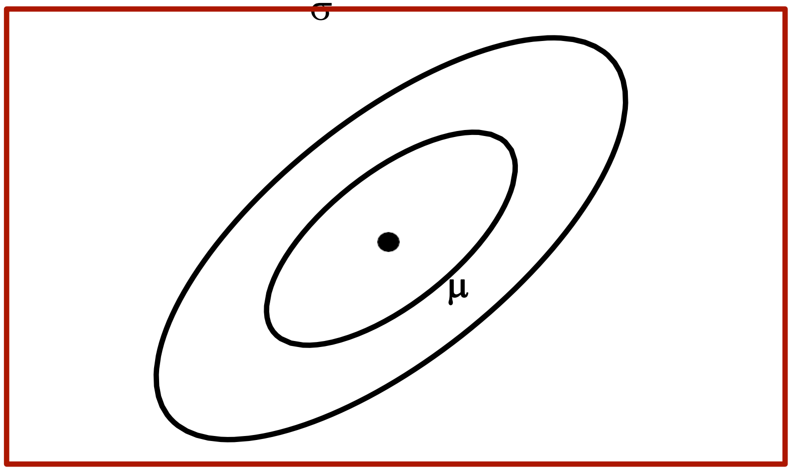
Univariate



$$p(\mathbf{x}) \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma}):$$

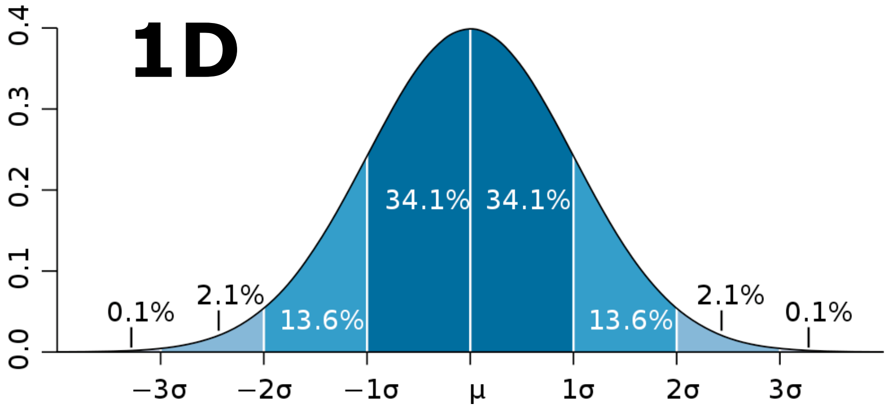
$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^t \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$$

Multivariate



Gaussians

1D



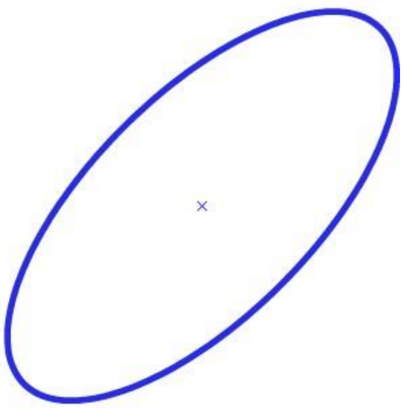
2D

$$C = \begin{bmatrix} 0.020 & 0.013 \\ 0.013 & 0.020 \end{bmatrix}$$

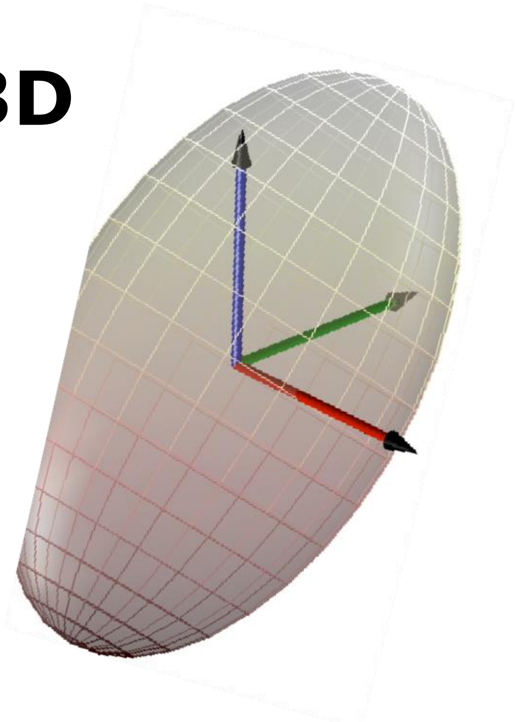
$$\lambda_1 = 0.007$$

$$\lambda_2 = 0.033$$

$$\rho = \sigma_{XY} / \sigma_X \sigma_Y = 0.673$$



3D



Properties of Gaussians

$$\left.\begin{array}{l} X \sim N(\mu, \sigma^2) \\ Y = aX + b \end{array}\right\} \quad \Rightarrow \quad Y \sim N(a\mu + b, a^2\sigma^2)$$

$$\left.\begin{array}{l} X_1 \sim N(\mu_1, \sigma_1^{\,2}) \\ X_2 \sim N(\mu_2, \sigma_2^{\,2}) \end{array}\right\} \Rightarrow p(X_1) \cdot p(X_2) \sim N\left(\frac{\sigma_2^{\,2}}{\sigma_1^{\,2} + \sigma_2^{\,2}} \mu_1 + \frac{\sigma_1^{\,2}}{\sigma_1^{\,2} + \sigma_2^{\,2}} \mu_2, \quad \frac{1}{\sigma_1^{\,-2} + \sigma_2^{\,-2}} \right)$$

Multivariate Gaussians

$$\left.\begin{array}{r} X \sim N(\mu,\Sigma) \\ Y = AX + B \end{array}\right\} \Rightarrow \quad Y \sim N(A\mu + B, A\Sigma A^T)$$

$$\left.\begin{array}{l} X_1 \sim N(\mu_1,\Sigma_1) \\ X_2 \sim N(\mu_2,\Sigma_2) \end{array}\right\} \Rightarrow p(X_1)\cdot p(X_2) \sim N\left(\frac{\Sigma_2}{\Sigma_1+\Sigma_2}\mu_1 + \frac{\Sigma_1}{\Sigma_1+\Sigma_2}\mu_2, \quad \frac{1}{\Sigma_1^{-1}+\Sigma_2^{-1}}\right)$$

(where division "–" denotes matrix inversion)

- We **stay Gaussian** as long as we start with Gaussians and perform only **linear transformations**

Outline

- Problem Statement
- Gaussians
- Kalman Filters
- Extended Kalman Filters

Problem Statement

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

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

with a measurement

$$\boxed{z_t = C_t x_t + \delta_t}$$

Problem Statement

$\boxed{A_t}$ Matrix (nxn) that describes how the state evolves from t to t-1 without controls or noise.

$\boxed{B_t}$ Matrix (nxl) that describes how the control u_t changes the state from t to t-1.

$\boxed{C_t}$ Matrix (kxn) that describes how to map the state x_t to an observation z_t.

$\boxed{\varepsilon_t}$ $\boxed{\delta_t}$ Random variables representing the process and measurement noise that are assumed to be independent and normally distributed with covariance Q_t and R_t respectively.

Bayes Filter: Two Steps

- Prediction

$$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1})\, bel(x_{t-1})\, dx_{t-1}$$

- Correction

$$bel(x_t) = \eta\, p(z_t \mid x_t)\, \overline{bel}(x_t)$$

Problem Statement

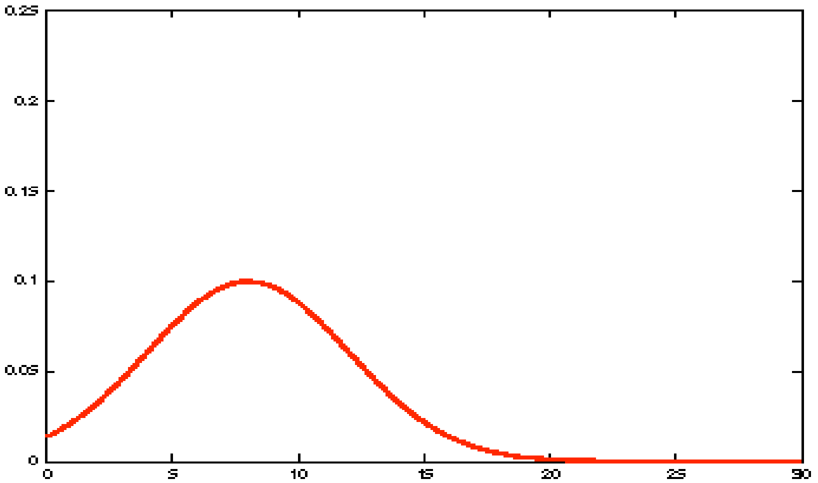
Estimates the state x of a discrete-time controlled process that is governed by the linear stochastic difference equation

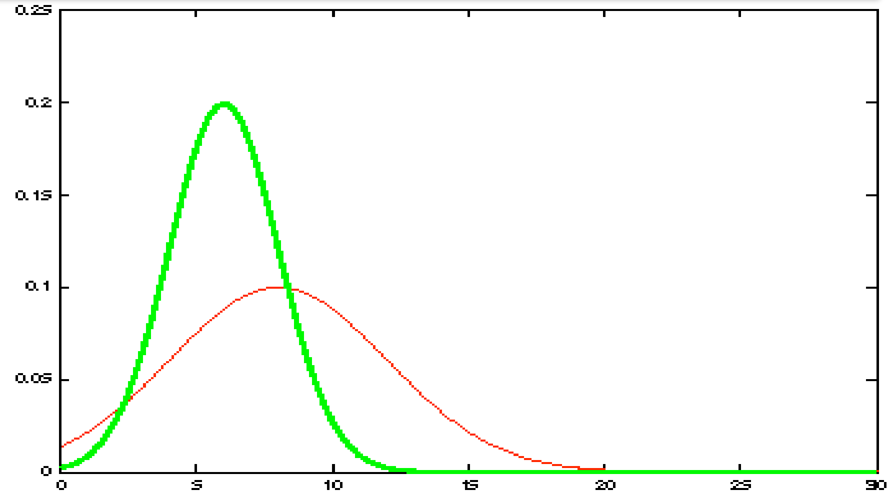
$$\boxed{x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t}$$

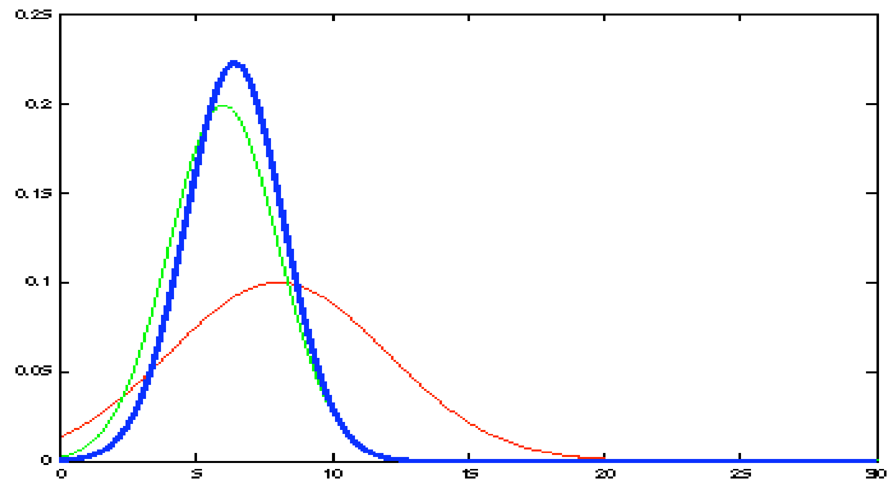
with a measurement

$$\boxed{z_t = C_t x_t + \delta_t}$$

Kalman Filter Update 1D



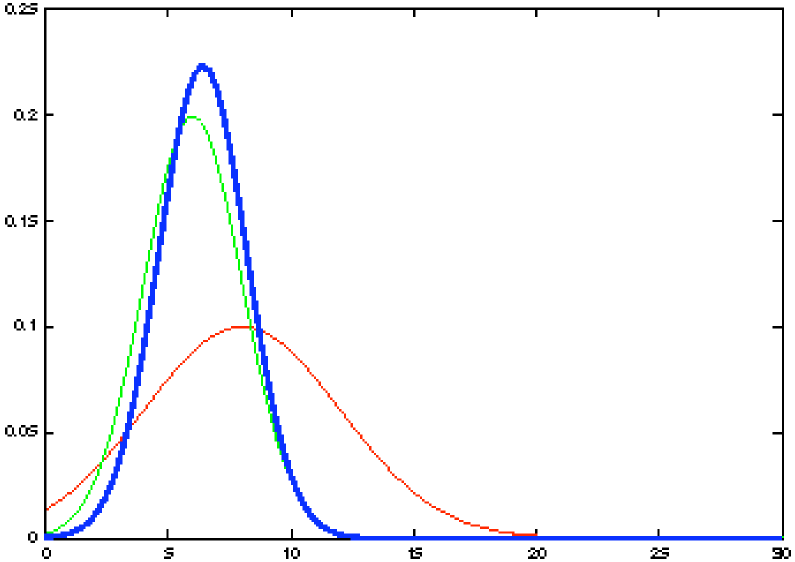




Kalman Filter: Correction

$$bel(x_t) = \begin{cases} \mu_t = \bar{\mu}_t + K_t(z_t - \bar{\mu}_t) \\ \sigma_t^2 = (1 - K_t)\bar{\sigma}_t^2 \end{cases} \quad \text{with} \quad K_t = \frac{\bar{\sigma}_t^2}{\bar{\sigma}_t^2 + \bar{\sigma}_{obs,t}^2}$$

$$bel(x_t) = \begin{cases} \mu_t = \bar{\mu}_t + K_t(z_t - C_t\bar{\mu}_t) \\ \Sigma_t = (I - K_t C_t)\bar{\Sigma}_t \end{cases} \quad \text{with} \quad K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + R_t)^{-1}$$



How to get the blue one?

-> **Kalman correction step**

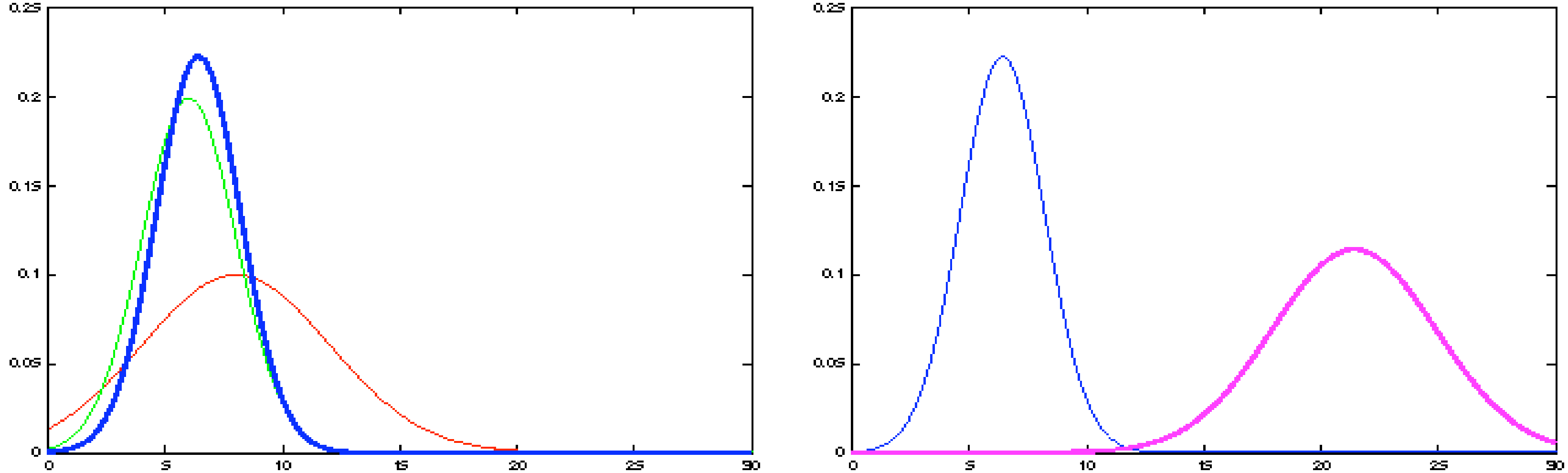
Kalman Filter: Prediction

How to get the magenta one?

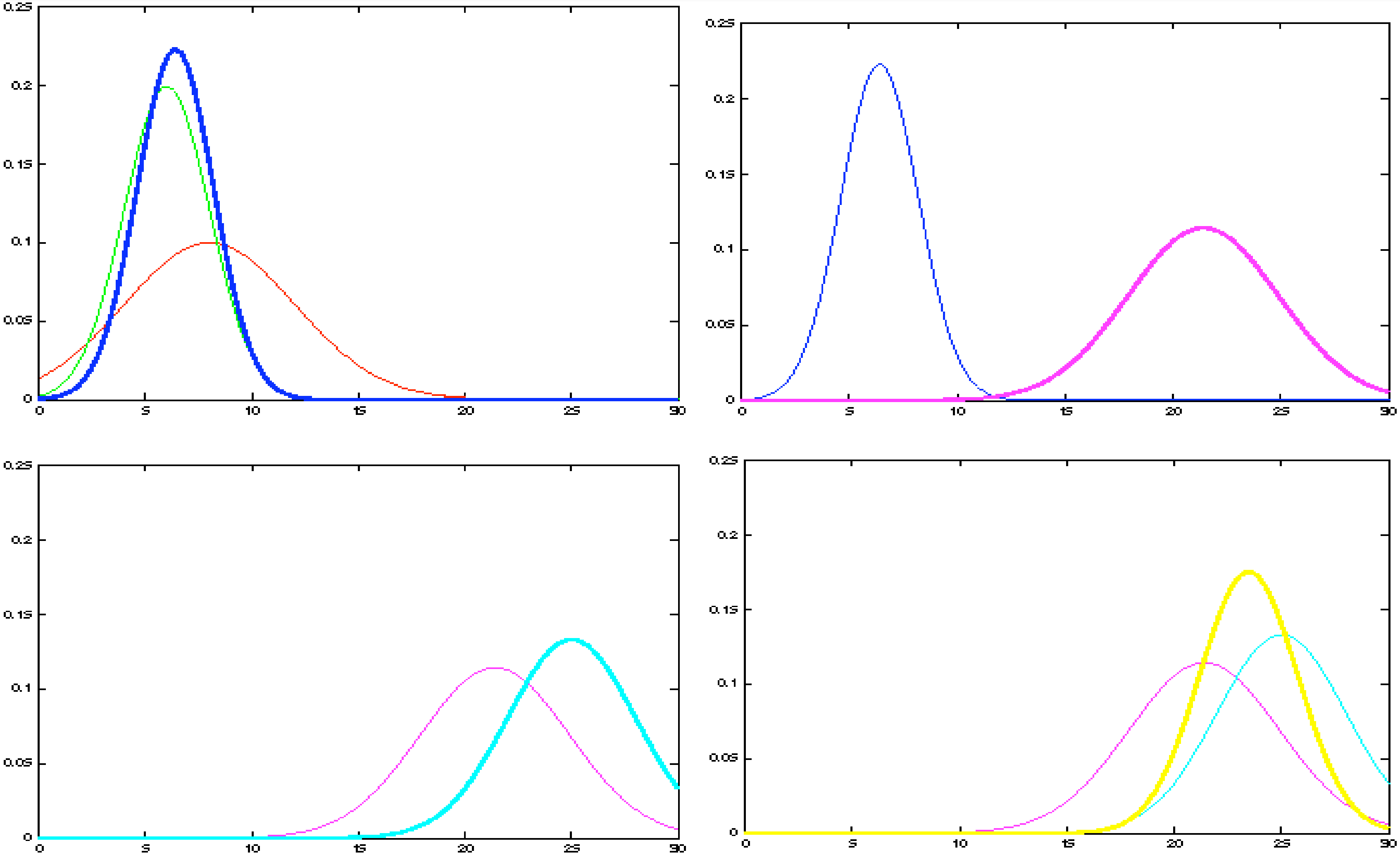
-> **State prediction step**

$$\overline{bel}(x_t) = \begin{cases} \bar{\mu}_t = a_t \mu_{t-1} + b_t u_t \\ \bar{\sigma}_t^2 = a_t^2 \sigma_t^2 + \sigma_{act,t}^2 \end{cases}$$

$$\overline{bel}(x_t) = \begin{cases} \bar{\mu}_t = A_t \mu_{t-1} + B_t u_t \\ \bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t \end{cases}$$



Kalman Filter Update



Linear Gaussian System： Initialization

- Initial belief is normally distributed:

$$\boxed{bel(x_0) = N\left(x_0; \mu_0, \Sigma_0\right)}$$

Linear Gaussian System: Dynamics

- Dynamics are linear function of state and control plus additive noise:

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

$$\boxed{p(x_t \mid u_t, x_{t-1}) = N\left(x_t; A_t x_{t-1} + B_t u_t, Q_t\right)}$$

$$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1}) \qquad bel(x_{t-1})\, dx_{t-1}$$

$$\Downarrow \qquad \Downarrow$$

$$\sim N\left(x_t; A_t x_{t-1} + B_t u_t, Q_t\right) \qquad \sim N\left(x_{t-1}; \mu_{t-1}, \Sigma_{t-1}\right)$$

Linear Gaussian System: Dynamics

$$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1}) \qquad bel(x_{t-1})\, dx_{t-1}$$

$$\Downarrow \qquad\qquad\qquad \Downarrow$$

$$\sim N\left(x_t; A_t x_{t-1} + B_t u_t, Q_t\right) \qquad \sim N\left(x_{t-1}; \mu_{t-1}, \Sigma_{t-1}\right)$$

$$\Downarrow$$

$$\overline{bel}(x_t) = \eta \int \exp\left\{-\frac{1}{2}(x_t - A_t x_{t-1} - B_t u_t)^T Q_t^{-1}(x_t - A_t x_{t-1} - B_t u_t)\right\}$$

$$\exp\left\{-\frac{1}{2}(x_{t-1} - \mu_{t-1})^T \Sigma_{t-1}^{-1}(x_{t-1} - \mu_{t-1})\right\} dx_{t-1}$$

$$\overline{bel}(x_t) = \begin{cases} \bar{\mu}_t = A_t \mu_{t-1} + B_t u_t \\ \bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t \end{cases}$$

Linear Gaussian System: Observations

- Observations are linear function of state plus additive noise:

$$\boxed{z_t = C_t x_t + \delta_t}$$

$$\boxed{p(z_t \mid x_t) = N\left(z_t; C_t x_t, R_t\right)}$$

$$\boxed{\begin{array}{llll} bel(x_t) = & \eta & p(z_t \mid x_t) & \overline{bel}(x_t) \\ & & \Downarrow & \Downarrow \\ & & \sim N\left(z_t; C_t x_t, R_t\right) & \sim N\left(x_t; \bar{\mu}_t, \bar{\Sigma}_t\right) \end{array}}$$

Linear Gaussian System: Observations

$$bel(x_t) = \eta \quad p(z_t \mid x_t) \qquad \overline{bel}(x_t)$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\sim N(z_t; C_t x_t, R_t) \qquad \sim N\left(x_t; \bar{\mu}_t, \bar{\Sigma}_t\right)$$

$$\Downarrow$$

$$bel(x_t) = \eta \exp\left\{-\frac{1}{2}(z_t - C_t x_t)^T R_t^{-1}(z_t - C_t x_t)\right\} \exp\left\{-\frac{1}{2}(x_t - \bar{\mu}_t)^T \bar{\Sigma}_t^{-1}(x_t - \bar{\mu}_t)\right\}$$

$$bel(x_t) = \begin{cases} \mu_t = \bar{\mu}_t + K_t(z_t - C_t \bar{\mu}_t) \\ \Sigma_t = (I - K_t C_t)\bar{\Sigma}_t \end{cases} \quad \text{with} \quad K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + R_t)^{-1}$$

Kalman Filter Algorithm

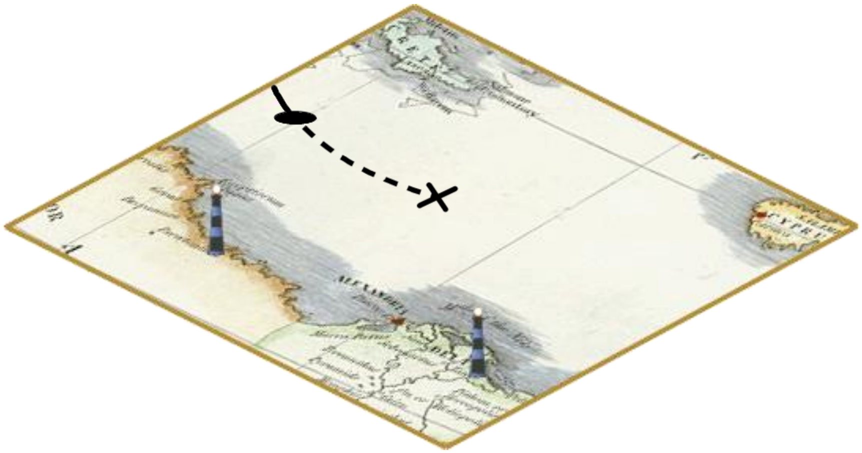
1. Algorithm **Kalman_filter**($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

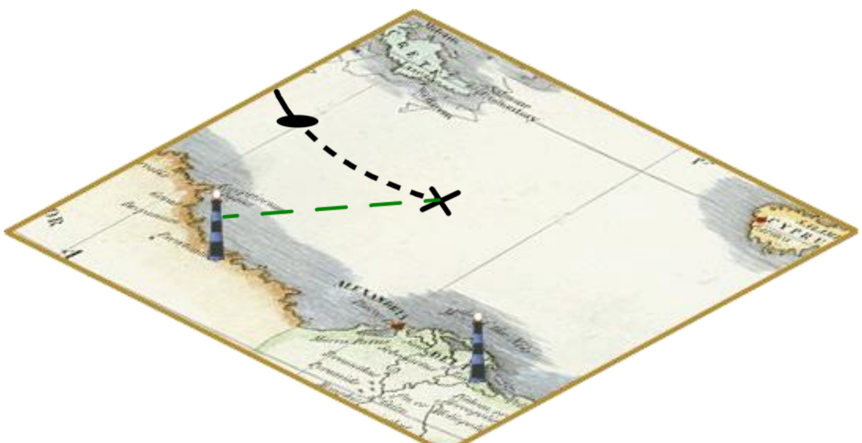
2. Prediction:
3. $\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$
4. $\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t$

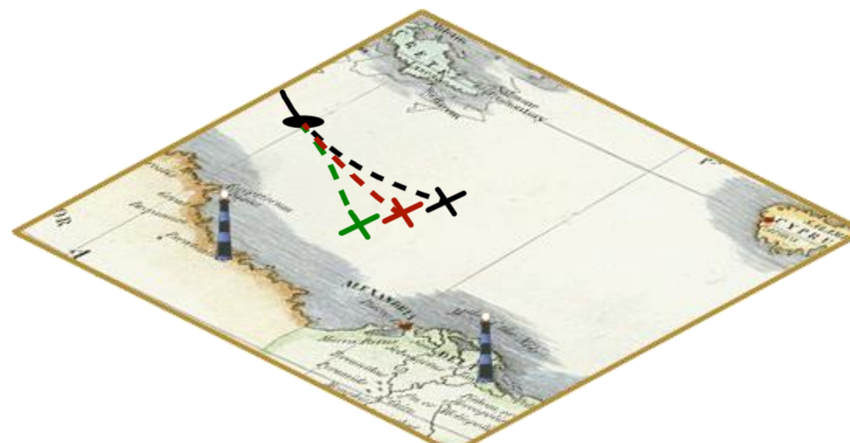
5. Correction:
6. $K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + R_t)^{-1}$
7. $\mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$
8. $\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$

9. Return μ_t, Σ_t

Kalman Filter Algorithm

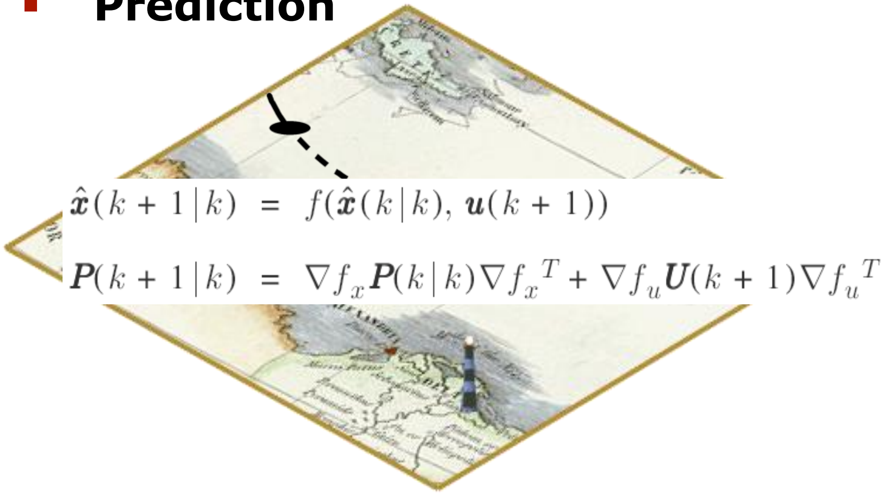






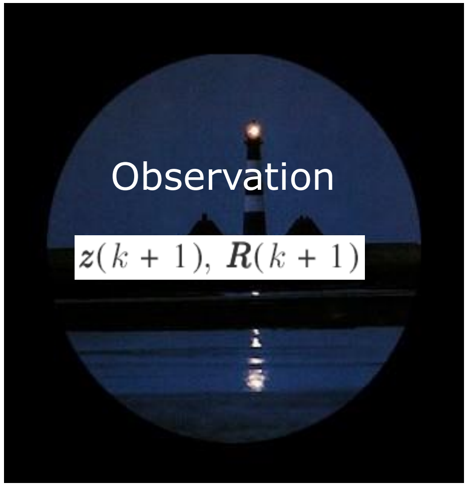
Kalman Filter Algorithm

- **Prediction**

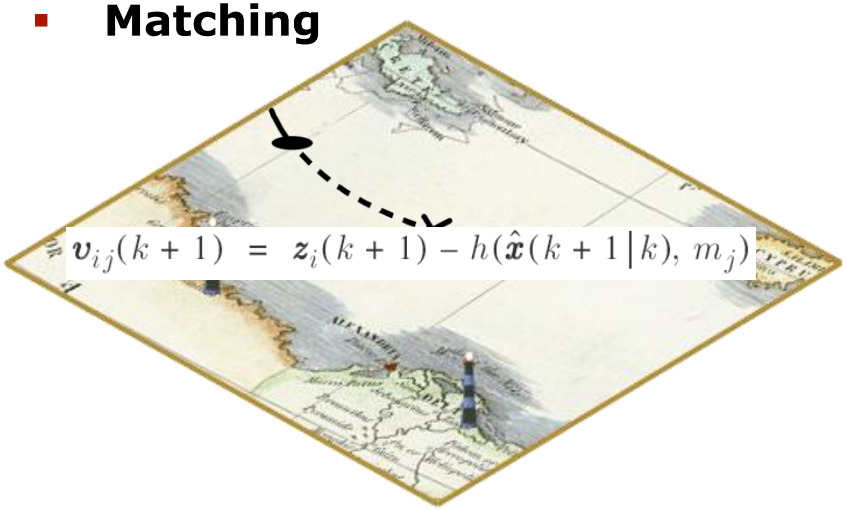


$$\hat{\boldsymbol{x}}(k+1|k) = f(\hat{\boldsymbol{x}}(k|k), \boldsymbol{u}(k+1))$$

$$\boldsymbol{P}(k+1|k) = \nabla f_x \boldsymbol{P}(k|k) \nabla f_x{}^T + \nabla f_u \boldsymbol{U}(k+1) \nabla f_u{}^T$$

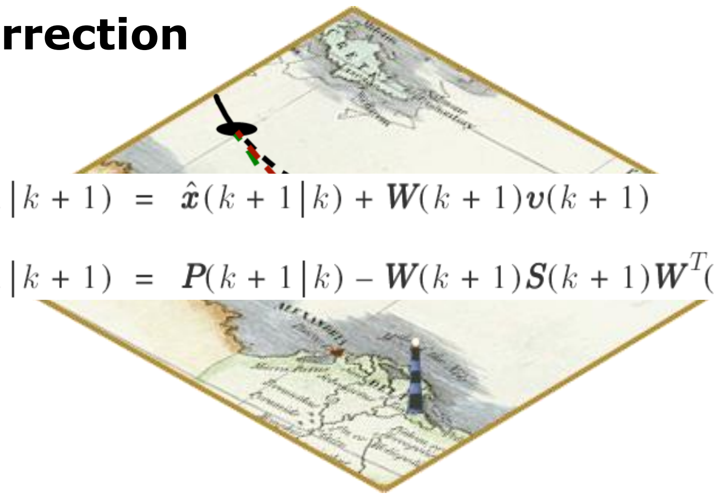


- **Matching**



$$\boldsymbol{v}_{ij}(k+1) = \boldsymbol{z}_i(k+1) - h(\hat{\boldsymbol{x}}(k+1|k), m_j)$$

- **Correction**

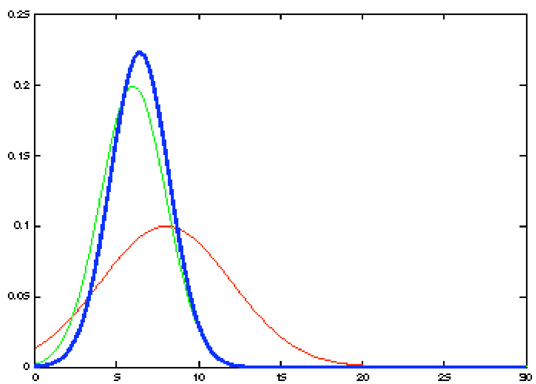


$$\hat{\boldsymbol{x}}(k+1|k+1) = \hat{\boldsymbol{x}}(k+1|k) + \boldsymbol{W}(k+1)\boldsymbol{v}(k+1)$$

$$\boldsymbol{P}(k+1|k+1) = \boldsymbol{P}(k+1|k) - \boldsymbol{W}(k+1)\boldsymbol{S}(k+1)\boldsymbol{W}^T(k+1)$$

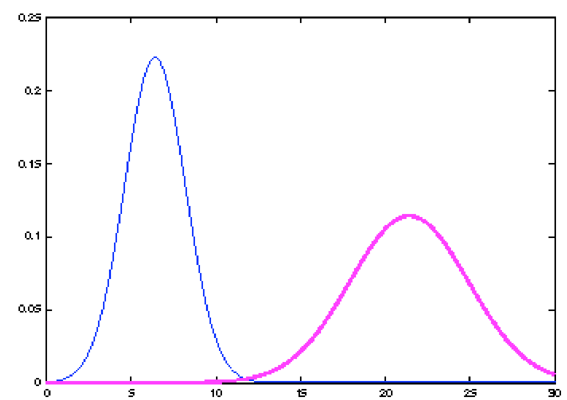
Prediction-Correction-Cycle

Prediction

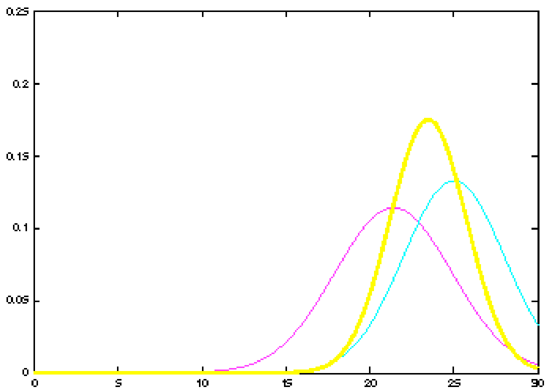


$$\overline{bel}(x_t) = \begin{cases} \overline{\mu}_t = a_t \mu_{t-1} + b_t u_t \\ \sigma_t^2 = a_t^2 \sigma_t^2 + \sigma_{act,t}^2 \end{cases}$$

$$\overline{bel}(x_t) = \begin{cases} \overline{\mu}_t = A_t \mu_{t-1} + B_t u_t \\ \overline{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t \end{cases}$$

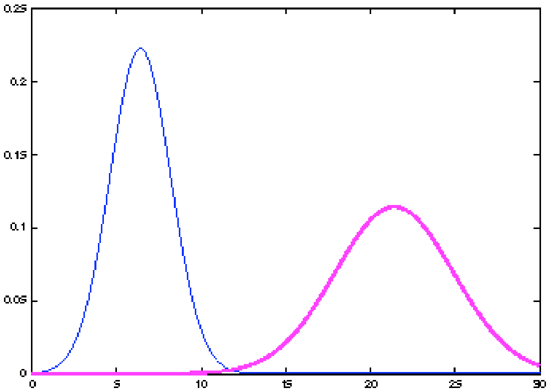


Prediction-Correction-Cycle



$$bel(x_t) = \begin{cases} \mu_t = \bar{\mu}_t + K_t(z_t - \bar{\mu}_t) \\ \sigma_t^2 = (1 - K_t)\bar{\sigma}_t^2 \end{cases}, \quad K_t = \frac{\bar{\sigma}_t^2}{\bar{\sigma}_t^2 + \sigma_{obs,t}^2}$$

$$bel(x_t) = \begin{cases} \mu_t = \bar{\mu}_t + K_t(z_t - C_t\bar{\mu}_t) \\ \Sigma_t = (I - K_t C_t)\bar{\Sigma}_t \end{cases}, \quad K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + R_t)^{-1}$$



Correction

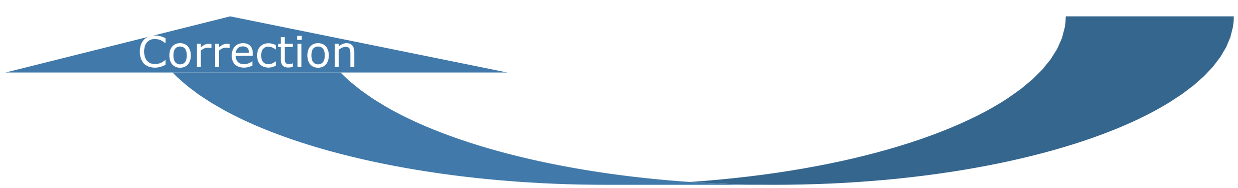
Prediction-Correction-Cycle

$$\overline{bel}(x_t) = \begin{cases} \bar{\mu}_t = a_t \mu_{t-1} + b_t u_t \\ \bar{\sigma}_t^2 = a_t^2 \sigma_t^2 + \sigma_{act,t}^2 \end{cases}$$

$$\overline{bel}(x_t) = \begin{cases} \bar{\mu}_t = A_t \mu_{t-1} + B_t u_t \\ \bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t \end{cases}$$

$$bel(x_t) = \begin{cases} \mu_t = \bar{\mu}_t + K_t (z_t - \bar{\mu}_t) \\ \sigma_t^2 = (1 - K_t)\bar{\sigma}_t^2 \end{cases}, \quad K_t = \frac{\bar{\sigma}_t^2}{\bar{\sigma}_t^2 + \sigma_{obs,t}^2}$$

$$bel(x_t) = \begin{cases} \mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t) \\ \Sigma_t = (I - K_t C_t)\bar{\Sigma}_t \end{cases}, \quad K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + R_t)^{-1}$$



Kalman Filter Summary

- Highly efficient: Polynomial in the measurement dimensionality *k* and state dimensionality *n*:

$$O(k^{2.376} + n^2)$$

- **Optimal for linear Gaussian systems**!

- Most robotics systems are **nonlinear**!

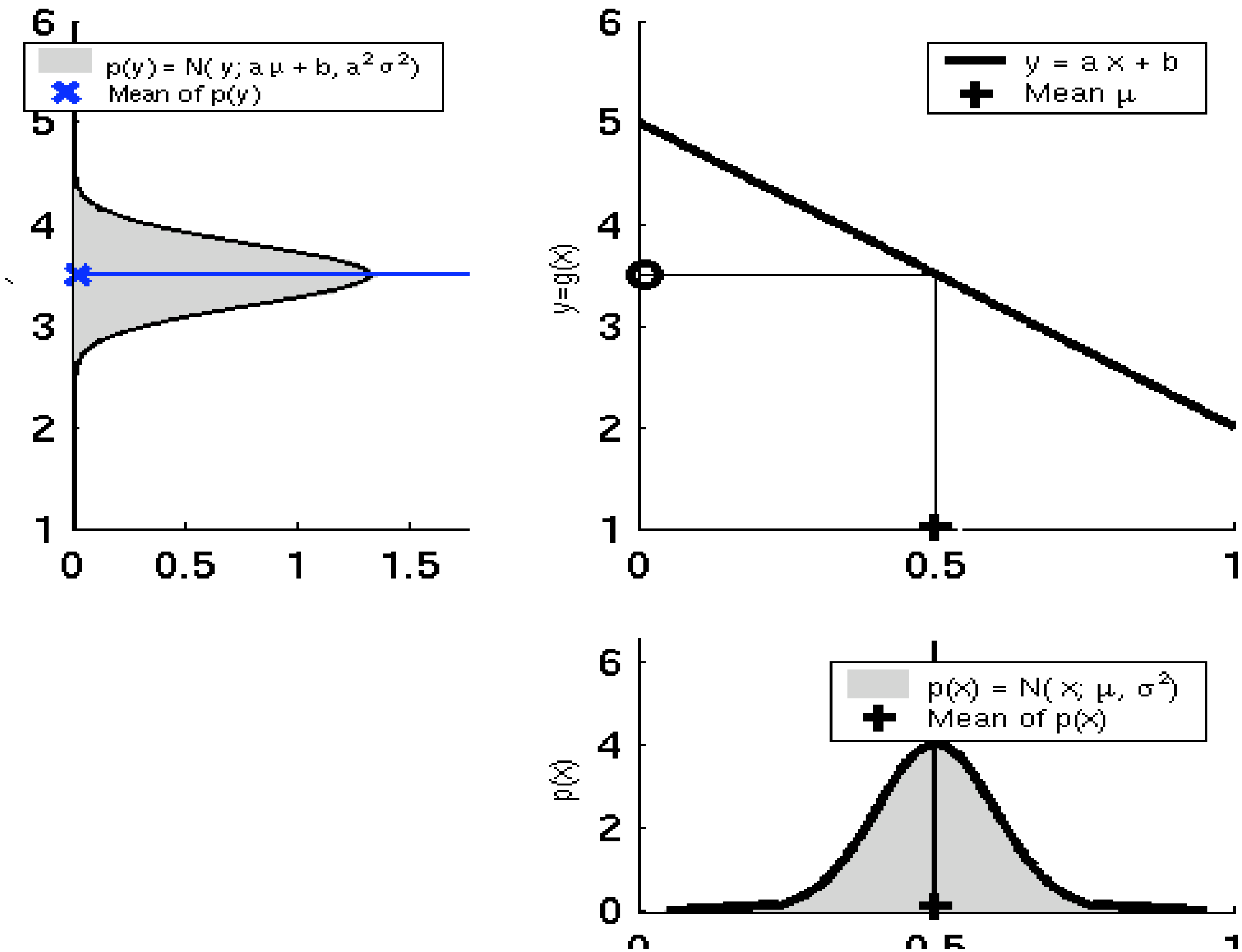
Nonlinear Dynamics

- Most realistic robotic problems involve nonlinear functions

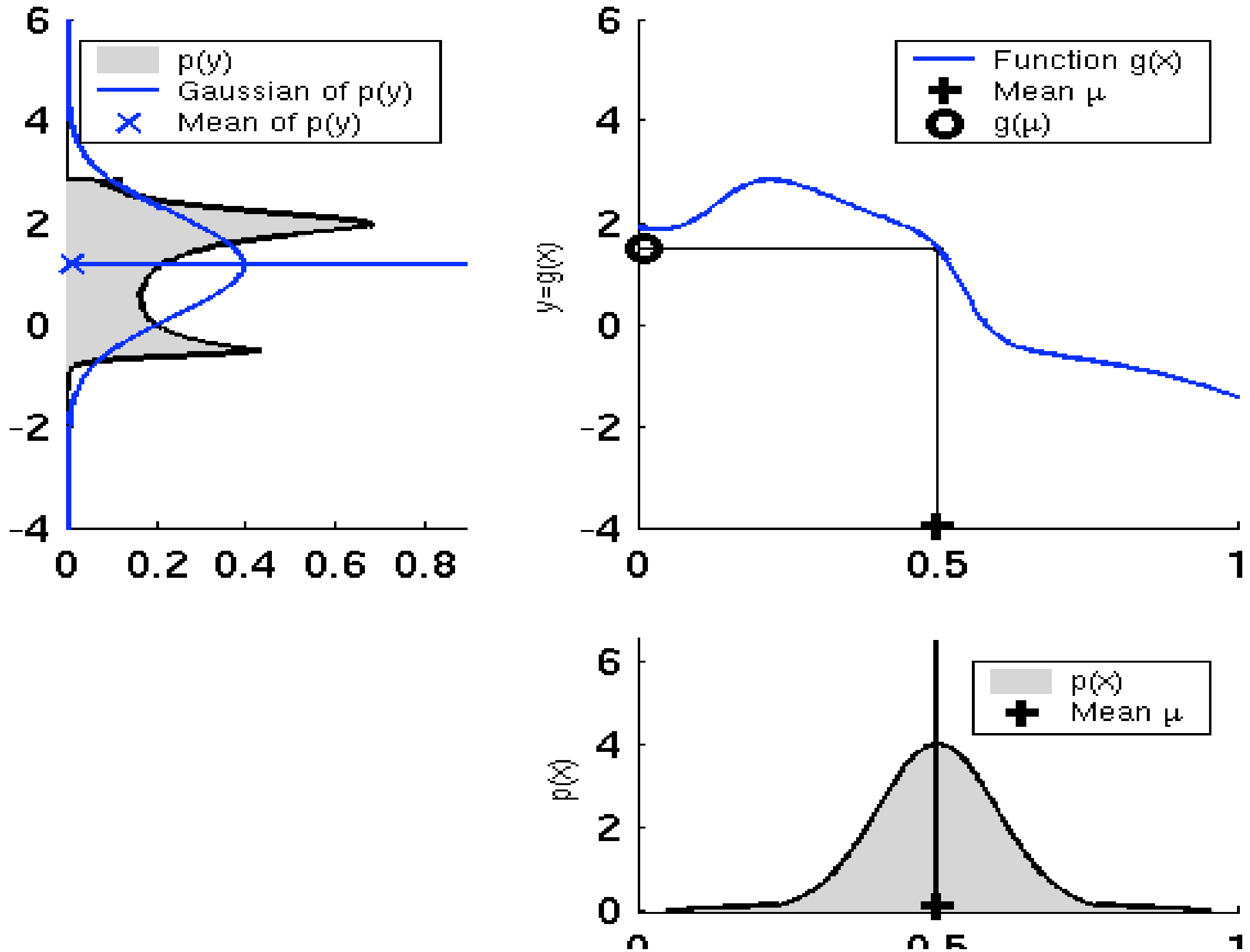
$$\boxed{x_t = g(u_t, x_{t-1})}$$

$$\boxed{z_t = h(x_t)}$$

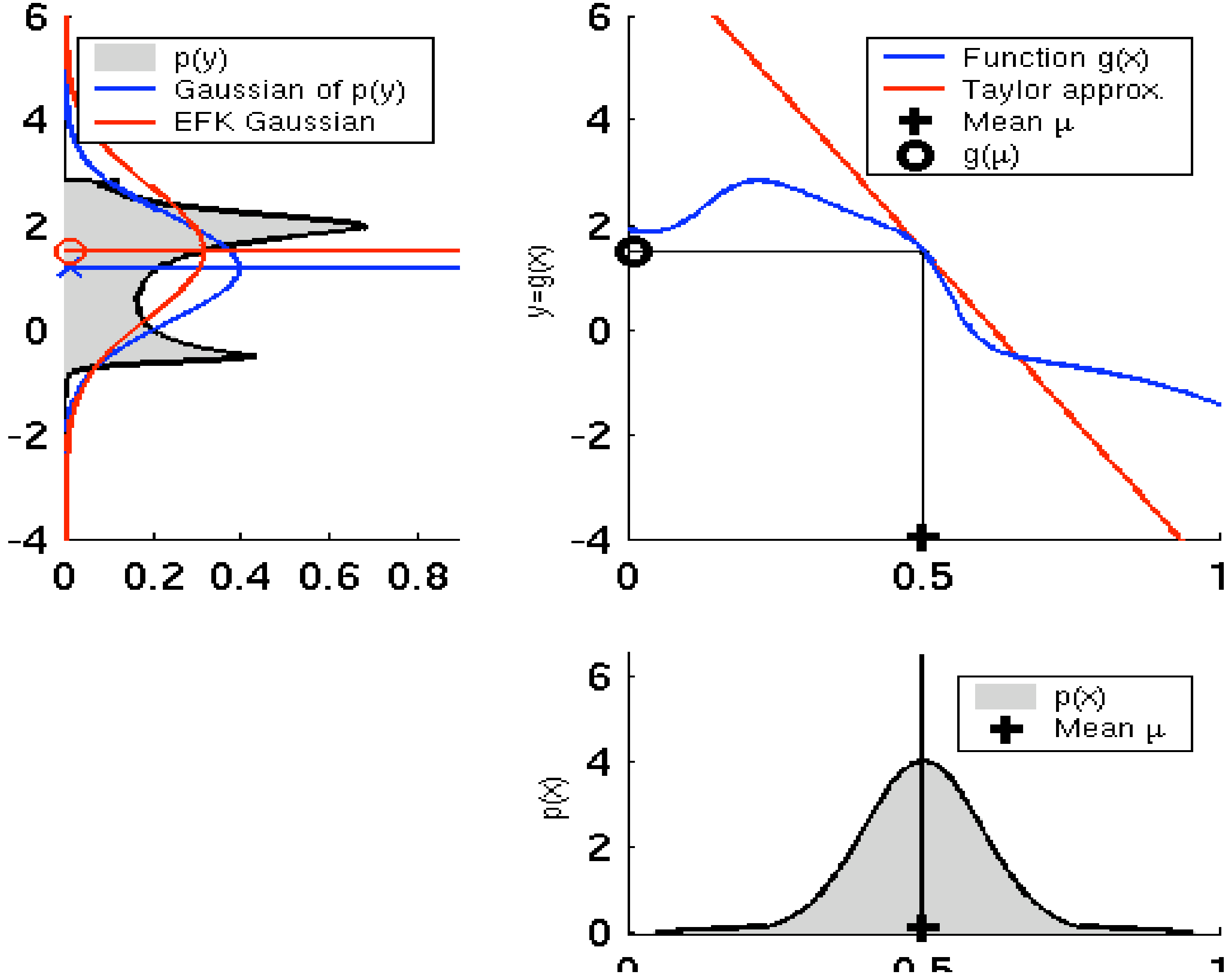
Linear Assumption Revisited



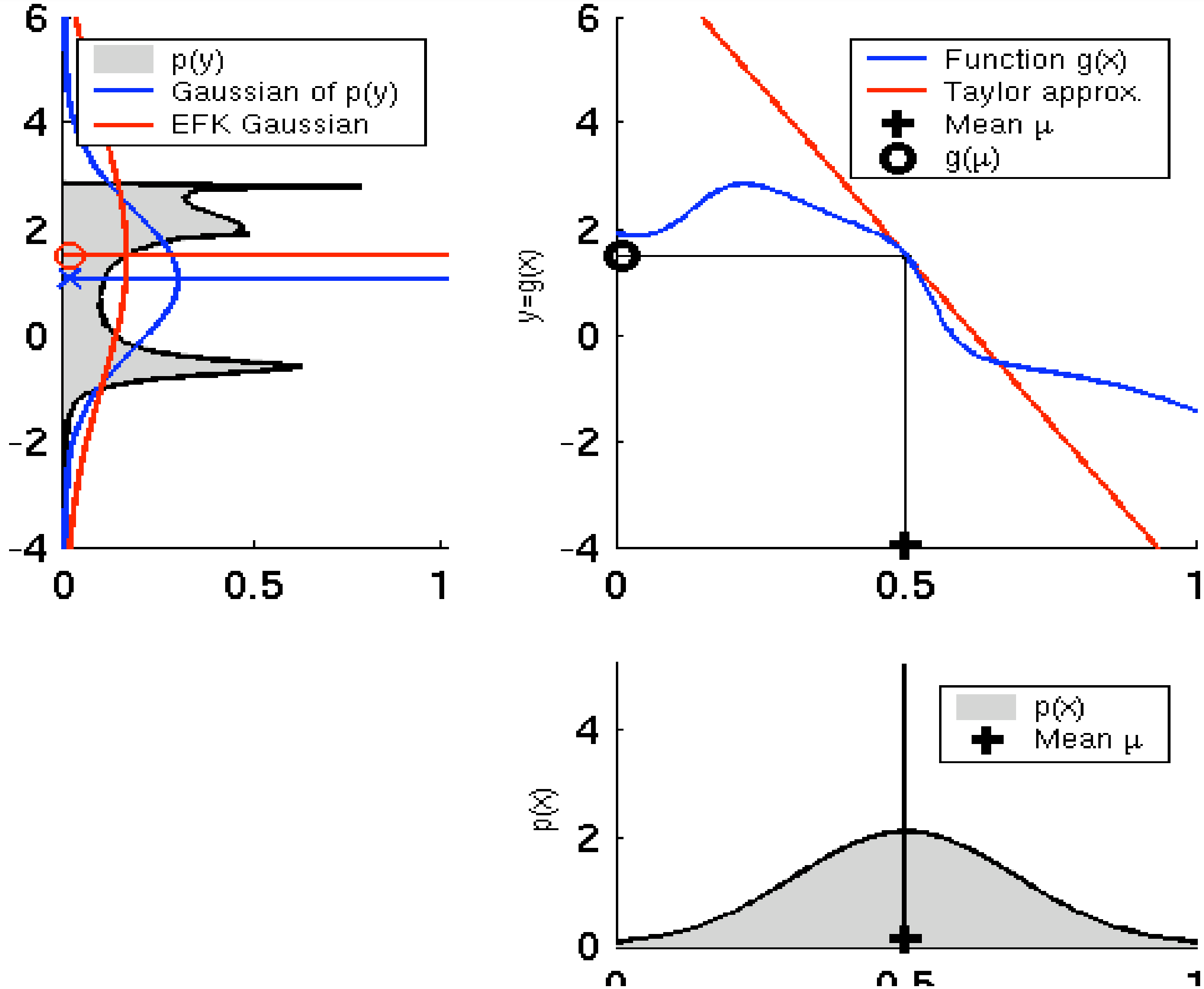
Nonlinear Function



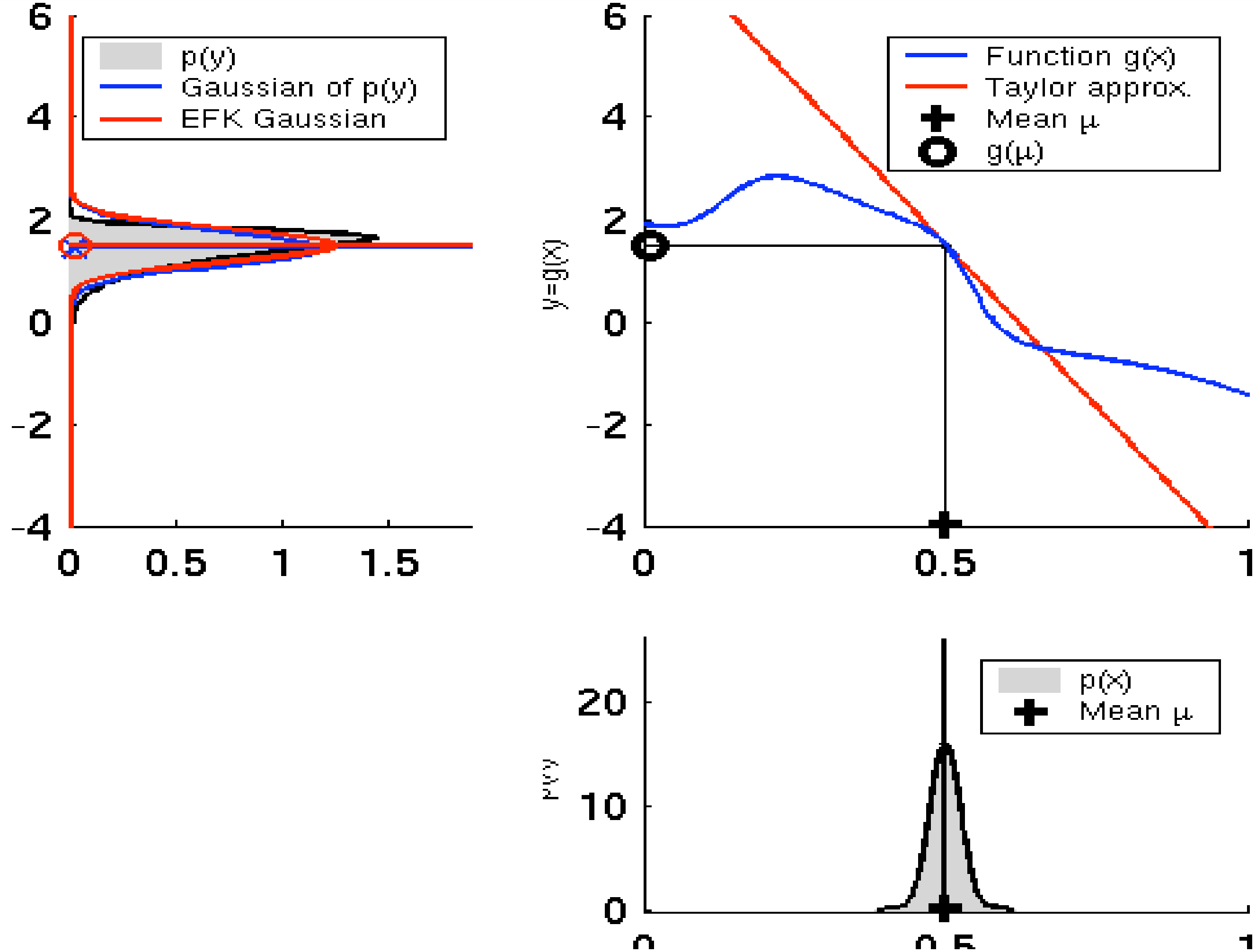
EKF Linearization I



EKF Linearization II



EKF Linearization III



EKF Linearization：First Order

- Prediction:

$$g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} (x_{t-1} - \mu_{t-1})$$

$$g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + G_t \ (x_{t-1} - \mu_{t-1})$$

- Correction:

$$h(x_t) \approx h(\bar{\mu}_t) + \frac{\partial h(\bar{\mu}_t)}{\partial x_t} (x_t - \bar{\mu}_t)$$

$$h(x_t) \approx h(\bar{\mu}_t) + H_t \ (x_t - \bar{\mu}_t)$$

EKF Algorithm

1. **Extended_Kalman_filter**(μ_{t-1}, Σ_{t-1}, u_t, z_t):

2. Prediction:

3. $$\bar{\mu}_t = g(u_t, \mu_{t-1}) \longleftarrow \bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$$

4. $$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + Q_t \longleftarrow \bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t$$

5. Correction:

6. $$K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + R_t)^{-1} \longleftarrow K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + R_t)^{-1}$$

7. $$\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t)) \longleftarrow \mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$$

8. $$\Sigma_t = (I \quad K_t H_t) \bar{\Sigma}_t \longleftarrow \Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$$

9. Return μ_t, Σ_t

$$H_t = \frac{\partial h(\bar{\mu}_t)}{\partial x_t} \qquad G_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}}$$

Extended Kalman Filter Summary

- Highly efficient: Polynomial in measurement dimensionality k and state dimensionality n:

$$O(k^{2.376} + n^2)$$

- Not optimal!
- Can diverge if nonlinearities are large!
- Works surprisingly well even when all assumptions are violated!

Homework 6

Problem 1: Please derive the optimal estimator of x for a given y, where $y = Cx + n$ and $n \sim \text{Gauss}(0, Q)$. What are the mean and variance of such an estimator?

Problem 2: Please derive the optimal estimator of $x(t)$ for a given $x(t\text{-}1)$, where $x(t) = Ax(t\text{-}1) + m(t)$, $x(t\text{-}1) \sim \text{Gauss}(\mu_{t-1}, \Sigma_{t-1},)$, and $m \sim \text{Gauss}(0, R)$. What are the mean and variance of such an estimator?

Problem 3: Please derive the optimal estimator of $x(t)$ for a given $y(t)$ and a given $x(t\text{-}1)$, where $x(t) = Ax(t\text{-}1) + m(t)$, $y(t) = Cx(t) + n(t)$, $x(t\text{-}1) \sim \text{Gauss}(\mu_{t-1}, \Sigma_{t-1},)$, $m \sim \text{Gauss}(0, R)$ and n ~ $\text{Gauss}(0, Q)$. What are the mean and variance of such an estimator? (two solutions with and without using $K(t)$)