Bayes Filter Reminder

- **Prediction**
  \[ \overline{bel}(x_t) = \int p(x_t | u_t, x_{t-1}) \, bel(x_{t-1}) \, dx_{t-1} \]

- **Correction**
  \[ bel(x_t) = \eta \, p(z_t | x_t) \, \overline{bel}(x_t) \]

---

**Gaussians**

Univariate

- \( p(x) \sim N(\mu, \sigma^2) \):
  \[ p(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \]

Multivariate

- \( p(x) \sim N(\mu, \Sigma) \):
  \[ p(x) = \frac{1}{(2\pi)^{d/2}\left|\Sigma\right|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)} \]

---

**Properties of Gaussians**

\[ X \sim N(\mu, \sigma^2) \]
\[ Y = aX + b \]

\[ \Rightarrow \quad Y \sim N(a\mu + b, a^2\sigma^2) \]

\[ \begin{align*}
X_1 & \sim N(\mu_1, \sigma_1^2) \\
X_2 & \sim N(\mu_2, \sigma_2^2)
\end{align*} \]

\[ \Rightarrow p(X_1) \cdot p(X_2) \sim N\left(\frac{\sigma_1^2 \mu_1 + \sigma_2^2 \mu_2}{\sigma_1^2 + \sigma_2^2}, \frac{1}{\sigma_1^2 + \sigma_2^2}\right) \]
**Multivariate Gaussians**

\[ X \sim N(\mu, \Sigma) \]
\[ Y = AX + B \]

\[ \Rightarrow \quad Y \sim N(A\mu + B, A\Sigma A^T) \]

\[ X_1 \sim N(\mu_1, \Sigma_1) \]
\[ X_2 \sim N(\mu_2, \Sigma_2) \]

\[ \Rightarrow \quad p(X_1) \cdot p(X_2) \sim N\left( \frac{\Sigma_1}{\Sigma_1 + \Sigma_2} \mu_1 + \frac{\Sigma_2}{\Sigma_1 + \Sigma_2} \mu_2, \frac{1}{\Sigma_1 + \Sigma_2} \right) \]

- We stay in the “Gaussian world” as long as we start with Gaussians and perform only linear transformations.

**Discrete Kalman Filter**

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

\[ x_t = A_t x_{t-1} + B_t u_t + \epsilon_t \]

with a measurement

\[ z_t = C_t x_t + \delta_t \]
Kalman Filter Updates in 1D

\[ bel(x_t) = \left\{ \begin{array}{l}
\mu_t = \mu_{t-1} + K_t(z_t - \mu_t) \\
\sigma_t^2 = (1 - K_t)\sigma_{t-1}^2
\end{array} \right. \]

with \[ K_t = \frac{\sigma_t^2}{\sigma_t^2 + \sigma_{\text{obs}}^2} \]

\[ bel(x_t) = \left\{ \begin{array}{l}
\mu_t = \mu_{t-1} + K_t(z_t - C_t\mu_t) \\
\Sigma_t = (I - K_t C_t)\Sigma_{t-1}
\end{array} \right. \]

with \[ K_t = \frac{\Sigma_t C_t^T (C_t \Sigma_t C_t^T + Q_t)^{-1}}{\sigma_t^2 + \sigma_{\text{obs}}^2} \]

Kalman Filter Updates in 1D

\[ bel(x_t) = \left\{ \begin{array}{l}
\mu_t = a_t \mu_{t-1} + b_t u_t \\
\sigma_t^2 = a_t^2 \sigma_{t-1}^2 + \sigma_u^2
\end{array} \right. \]

\[ bel(x_t) = \left\{ \begin{array}{l}
\mu_t = \Lambda_t \mu_{t-1} + B_t u_t \\
\Sigma_t = \Lambda_t R_{t-1} \Lambda_t^T + R_t
\end{array} \right. \]

Linear Gaussian Systems: Initialization

- Initial belief is normally distributed:

\[ bel(x_0) = N(x_0; \mu_0, \Sigma_0) \]
Linear Gaussian Systems: Dynamics

- Dynamics are linear function of state and control plus additive noise:
\[ x_t = A_t x_{t-1} + B_t u_t + \epsilon_t \]
\[ p(x_t \mid u_t, x_{t-1}) = N(x_t; A_t x_{t-1} + B_t u_t, R_t) \]
\[ \text{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1}) \, dx_{t-1} \quad \text{bel}(x_{t-1}) \, dx_{t-1} \]
\[ \downarrow \quad \downarrow \]
\[ \sim N(x_t; A_t x_{t-1} + B_t u_t, R_t) \sim N(x_{t-1}; \mu_{t-1}, \Sigma_{t-1}) \]

Linear Gaussian Systems: Observations

- Observations are linear function of state plus additive noise:
\[ z_t = C_t x_t + \delta_t \]
\[ p(z_t \mid x_t) = N(z_t; C_t x_t, Q_t) \]
\[ \text{bel}(x_t) = \eta \quad p(z_t \mid x_t) \quad \text{bel}(x_t) \]
\[ \downarrow \quad \downarrow \]
\[ \sim N(z_t; C_t x_t, Q_t) \sim N(x_t; \mu_t, \Sigma_t) \]

Linear Gaussian Systems: Dynamics

\[ \text{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1}) \, bel(x_{t-1}) \, dx_{t-1} \]
\[ \downarrow \]
\[ \sim N(x_t; A_t x_{t-1} + B_t u_t + \epsilon_t, R_t) \sim N(x_{t-1}; \mu_{t-1}, \Sigma_{t-1}) \]

\[ \text{bel}(x_t) = \eta \int \exp \left\{ -\frac{1}{2} (x_t - A_t x_{t-1} - B_t u_t)^T R_t^{-1}(x_t - A_t x_{t-1} - B_t u_t) \right\} \]
\[ \exp \left\{ -\frac{1}{2} (x_{t-1} - \mu_{t-1})^T \Sigma_{t-1}^{-1}(x_{t-1} - \mu_{t-1}) \right\} \, dx_{t-1} \]

\[ \text{bel}(x_t) = \left[ \begin{array}{c} \mu_t = A_t \mu_{t-1} + B_t u_t \\ \Sigma_t = A_t \Sigma_{t-1} A_t^T + R_t \end{array} \right] \]

Linear Gaussian Systems: Observations

\[ \text{bel}(x_t) = \eta \cdot p(z_t \mid x_t) \quad \text{bel}(x_t) \]
\[ \downarrow \quad \downarrow \]
\[ \sim N(z_t; C_t x_t, Q_t) \sim N(x_t; \mu_t, \Sigma_t) \]

\[ \text{bel}(x_t) = \eta \exp \left\{ -\frac{1}{2} (z_t - C_t x_t)^T Q_t^{-1}(z_t - C_t x_t) \right\} \exp \left\{ -\frac{1}{2} (x_t - \mu_t)^T \Sigma_t^{-1}(x_t - \mu_t) \right\} \]

\[ \text{bel}(x_t) = \left[ \begin{array}{c} \mu_t = \mu_t + K_t (z_t - C_t \mu_t) \\ \Sigma_t = (I - K_t C_t) \Sigma_t \end{array} \right] \text{ with } K_t = \Sigma_t C_t^T (C_t \Sigma_t C_t^T + Q_t)^{-1} \]
Kalman Filter Algorithm

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

2. Prediction:
   - μ_t = A_t μ_{t-1} + B_t u_t
   - Σ_t = A_t Σ_{t-1} A_t^T + R_t

3. Correction:
   - K_t = Σ_t C_t^T (C_t Σ_t C_t^T + Q_t)^{-1}
   - μ_t = μ_t + K_t (z_t - C_t μ_t)
   - Σ_t = (I - K_t C_t) Σ_t
   - Return μ_t, Σ_t
Kalman Filter Summary

- Highly efficient: Polynomial in measurement dimensionality $k$ and state dimensionality $n$:
  \[ O(k^{2.376} + n^2) \]
- Optimal for linear Gaussian systems!
- Most robotics systems are nonlinear!

Nonlinear Dynamic Systems

- Most realistic robotic problems involve nonlinear functions
  \[ x_t = g(u_t, x_{t-1}) \]
  \[ z_t = h(x_t) \]

Linearity Assumption Revisited

Non-linear Function
EKF Linearization (1)

EKF Linearization (2)

EKF Linearization (3)

EKF Linearization: First Order Taylor Series Expansion

- **Prediction:**
  \[ g(u_t, x_{t-1}) = g(u_t, \mu_{t-1}) + \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} (x_{t-1} - \mu_{t-1}) \]
  \[ g(u_t, x_{t-1}) = g(u_t, \mu_{t-1}) + G_{t} (x_{t-1} - \mu_{t-1}) \]

- **Correction:**
  \[ h(x_t) = h(\bar{\mu}_t) + \frac{\partial h(\bar{\mu}_t)}{\partial x_t} (x_t - \bar{\mu}_t) \]
  \[ h(x_t) = h(\bar{\mu}_t) + H_t (x_t - \bar{\mu}_t) \]
**EKF Algorithm**

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

2. Prediction:
   3. \( \bar{\mu}_t = g(u_t, \mu_{t-1}) \)
   4. \( \bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t \)

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

9. Return \( \mu_t, \Sigma_t \)

   \[ H_t = \frac{\partial h(\bar{\mu}_t)}{\partial \mu} \]
   \[ G_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial \mu_{t-1}} \]

**Localization**

"Using sensory information to locate the robot in its environment is the most fundamental problem to providing a mobile robot with autonomous capabilities." [Cox '91]

- **Given**
  - Map of the environment.
  - Sequence of sensor measurements.

- **Wanted**
  - Estimate of the robot’s position.

- **Problem classes**
  - Position tracking
  - Global localization
  - Kidnapped robot problem (recovery)

**Landmark-based Localization**

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

   **Prediction:**

   \( G_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial \mu_{t-1}} \)

   Jacobian of g w.r.t location

2. \( V_t = \frac{\partial \xi(u_t, \mu_{t-1})}{\partial u_t} \)

   Jacobian of g w.r.t control

3. \( M_t = \begin{pmatrix} 0^T & \alpha \end{pmatrix} \begin{pmatrix} \alpha^T \end{pmatrix} \)

   Motion noise

4. \( \bar{\mu}_t = g(u_t, \mu_{t-1}) \)

5. \( \bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + V_t M_t V_t^T \)

   Predicted mean

   Predicted covariance
1. \textbf{EKF Localization} \,(\mu_{\text{v}}, \Sigma_{\text{v}}, \mu_{\text{w}}, \Sigma_{\text{w}}, \mu_{\text{a}}, \Sigma_{\text{a}}):

2. \hat{z}_t = \begin{pmatrix} \sqrt{\|\mathbf{y}_t - \mathbf{H}_t \mu_t \|^2 + \Sigma_t} \\ \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \end{pmatrix} \end{pmatrix}, \quad \text{Predicted measurement, mean}

3. \quad \mathbf{H}_t = \frac{\partial \hat{z}_t}{\partial \mathbf{x}_t}, \quad \text{Jacobian of w.r.t. location}

4. \quad \mathbf{Q}_t = \begin{pmatrix} 0 & \Sigma_v \\ \Sigma_v & 0 \end{pmatrix}, \quad \text{Pred. measurement covariance}

5. \quad \mathbf{S}_t = \begin{pmatrix} \Sigma_v & \Sigma_v \end{pmatrix}, \quad \text{Kalman gain}

6. \quad \mathbf{S}_t = \Sigma_v + \mathbf{H}_t \mathbf{S}_t \mathbf{H}_t^T, \quad \mathbf{K}_t

7. \quad \hat{\mathbf{x}}_t = \mathbf{H}_t \mathbf{S}_t \mathbf{H}_t^T \mathbf{x}_t + \mathbf{S}_t \mathbf{K}_t (\hat{z}_t - \hat{z}_t)

8. \quad \Sigma_t = (I - \mathbf{K}_t \mathbf{H}_t) \Sigma_t, \quad \text{Updated mean and covariance}

\textbf{EKF Correction Step}
Estimation Sequence (1)

Estimation Sequence (2)

Comparison to GroundTruth

EKF 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!
Linearization via Unscented Transform

EKF  UKF

UUKF Sigma-Point Estimate (2)

EKF  UKF

UUKF Sigma-Point Estimate (3)

EKF  UKF

Unscented Transform

Sigma points

$$\chi^0 = \mu$$

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

Weights

$$w^0_m = \frac{\lambda}{n+\lambda}$$

$$w^0_c = \frac{\lambda}{n+\lambda} + (1-\alpha^2 + \beta)$$

$$w^i_m = w^i_c = \frac{1}{2(n+\lambda)}$$

for $$i = 1, ..., 2n$$

Pass sigma points through nonlinear function

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

Recover mean and covariance

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

$$\Sigma' = \sum_{i=0}^{2n} w^i_c (\psi^i - \mu)(\psi^i - \mu)^T$$
**UKF_localization** \( \mu_{t-1}, \Sigma_{t-1}, u_t, z_t, m \):

**Prediction:**

\[
M_{t} = \begin{pmatrix}
\sigma_{t}^{1} v_{t-1} + \sigma_{a}^{1} a_{t} & 0 \\
0 & \sigma_{t}^{1} v_{t-1} + \sigma_{a}^{1} a_{t}
\end{pmatrix}
\]

Motion noise

\[
Q_{t} = \begin{pmatrix}
\sigma_{t}^{2} & 0 \\
0 & \sigma_{t}^{2}
\end{pmatrix}
\]

Measurement noise

\[
\mu_{t-1} = (0 0)^{T}
\]

Augmented state mean

\[
\Sigma_{t-1} = \begin{pmatrix}
0 & 0 \\
0 & 0
\end{pmatrix}
\]

Augmented covariance

\[
\chi_{t-1} = \begin{pmatrix}
\mu_{t-1}^{x} \\
\mu_{t-1}^{y}
\end{pmatrix} + \gamma \begin{pmatrix}
\Sigma_{t-1}^{x} \\
\Sigma_{t-1}^{y}
\end{pmatrix}
\]

Sigma points

\[
\chi_{t-1}^{x} = g(u_{t} + \chi_{t-1}^{x}, \chi_{t-1}^{y})
\]

Prediction of sigma points

\[
\bar{\mu}_{t} = \sum_{i=0}^{2L} w_{i} \chi_{t-1}^{x}
\]

Predicted mean

\[
\bar{\Sigma}_{t} = \sum_{i=0}^{2L} w_{i}^{2} (\chi_{t-1}^{x} - \bar{\mu}_{t}) (\chi_{t-1}^{x} - \bar{\mu}_{t})^{T}
\]

Predicted covariance

**UKF_localization** \( \mu_{t-1}, \Sigma_{t-1}, u_t, z_t, m \):

**Correction:**

\[
\tilde{z}_t = h(\chi_{t-1}^{x}) + \chi_{t-1}^{y}
\]

Measurement sigma points

\[
\tilde{z}_t = \sum_{i=0}^{2L} w_{i} \bar{z}_{i}
\]

Predicted measurement mean

\[
S_{t} = \sum_{i=0}^{2L} w_{i} (\bar{z}_{i} - \tilde{z}_{t}) (\bar{z}_{i} - \tilde{z}_{t})^{T}
\]

Pred. measurement covariance

\[
S_{t}^{x} = \sum_{i=0}^{2L} w_{i} (\bar{z}_{i}^{x} - \bar{\mu}_{t}) (\bar{z}_{i}^{x} - \bar{\mu}_{t})^{T}
\]

Cross-covariance

\[
K_{t} = S_{t}^{-1} S_{t}
\]

Kalman gain

\[
\mu_{t} = \bar{\mu}_{t} + K_{t} (z_{t} - \tilde{z}_{t})
\]

Updated mean

\[
\Sigma_{t} = \bar{\Sigma}_{t} - K_{t} S_{t} K_{t}^{T}
\]

Updated covariance

---

1. **EKF_localization** \( \mu_{t-1}, \Sigma_{t-1}, u_t, z_t, m \):

**Correction:**

2. \[
\tilde{z}_t = \begin{pmatrix}
\sqrt{(m_x - \bar{\mu}_{t})^2 + (m_y - \bar{\mu}_{t})^2} \\
\text{atan 2}(m_y - \bar{\mu}_{t}, m_x - \bar{\mu}_{t}) - \bar{\mu}_{t}
\end{pmatrix}
\]

Predicted measurement mean

3. \[
H_{t} = \frac{\partial h(\bar{\mu}_{t}, m)}{\partial \bar{\mu}_{t}} = \begin{pmatrix}
\frac{\partial h}{\partial \bar{\mu}_{x}} \\
\frac{\partial h}{\partial \bar{\mu}_{y}}
\end{pmatrix}
\]

Jacobian of \( h \) w.r.t location

4. \[
Q_{t} = \begin{pmatrix}
\sigma_{t}^{2} & 0 \\
0 & \sigma_{t}^{2}
\end{pmatrix}
\]

5. \[
S_{t} = H_{t} \Sigma_{t-1} H_{t}^{T} + Q_{t}
\]

Pred. measurement covariance

6. \[
K_{t} = \Sigma_{t} H_{t}^{T} S_{t}^{-1}
\]

Kalman gain

7. \[
\mu_{t} = \bar{\mu}_{t} + K_{t} (z_{t} - \tilde{z}_{t})
\]

Updated mean

8. \[
\Sigma_{t} = (I - K_{t} H_{t}) \Sigma_{t}
\]

Updated covariance

---

**UKF Prediction Step**

- \( \bar{\mu}_{t-1} \)
- \( \bar{\Sigma}_{t-1} \)
- \( h(\chi_{t-1}^{x}) \)
- \( \chi_{t-1}^{y} \)
- \( \tilde{z}_{t} \)
- \( S_{t} \)
- \( K_{t} \)
- \( \mu_{t} \)
- \( \Sigma_{t} \)
**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!**

**Kalman Filter-based System**

- [Arras et al. 98]:
  - Laser range-finder and vision
  - High precision (<1cm accuracy)
**Localization With MHT**

- Belief is represented by multiple hypotheses
- Each hypothesis is tracked by a Kalman filter

**Additional problems:**
- **Data association:** Which observation corresponds to which hypothesis?
- **Hypothesis management:** When to add / delete hypotheses?
- Huge body of literature on target tracking, motion correspondence etc.

---

**MHT: Implemented System (1)**

- Hypotheses are extracted from LRF scans
- Each hypothesis has probability of being the correct one:
  \[ H_i = \{ \tilde{x}_i, \Sigma_i, P(H_i) \} \]
- Hypothesis probability is computed using Bayes’ rule
  \[ P(H_i | s) = \frac{P(s | H_i) P(H_i)}{P(s)} \]
- Hypotheses with low probability are deleted.
- New candidates are extracted from LRF scans.
  \[ C_j = \{ z_j, R_j \} \]

[Jensfelt et al. ‘00]

---

**MHT: Implemented System (2)**
MHT: Implemented System (3)
Example run

Map and trajectory

# hypotheses vs. time

# hypotheses

$P(H_{best})$

Courtesy of P. Jensfelt and S. Kristensen