Bayesian Filters

Jon Bosque

CMK Group

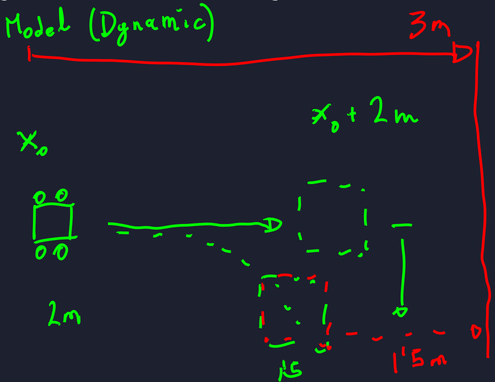
November 03, 2025

Bayesian Filters

Problem: State Estimation

System state is not accesible, and needs to be inferred from sensors or from past **actuations** by propagating an **initial state** using a **model**.

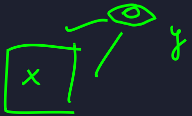
- Odometry drift
- Sensor noise
- … -Lidar -IMU …



Bayesian Approach

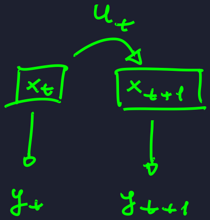
- Use probability distributions to represent uncertainty in state.
- Sequential estimation: update state estimate at each time step.

Bayesian Filter Framework



Given:

- State: x_t
- Control/input: u_t
- Observation: y_t
- **Model of the system:** (state transition and observation models)
  - Motion Model $p(x_{t+1}|x_t, u_t)$ Dynamics
  - Observation Model $p(y|x)$



Goal: Estimate $bel(x_t) = p(x_t|y_{1:t}, u_{1:t})$

$bel(x_0)$

$x_t : \mathbb{R}^n$

$$bel(x_t) = p(x_t|x_0, u_{1:t}, y_{1:t}) = \eta p(y_t|x_t) \int p(x_t|x_{t-1}, u_t) p(x_{t-1}|bel(x_0), u_{1:t-1}, y_{1:t-1}) dx_{t-1}$$

1. Prediction:

$$\overline{bel}(x_t) = \int \underbrace{p(x_t|x_{t-1}, u_t)}_{\text{Dynamics}} bel(x_{t-1}) dx_{t-1}$$

2. Correction:

$$bel(x_t) = \underset{\text{Normalizatio}}{\eta} \underbrace{p(y_t|x_k)}_{\text{Obs Model}} \overline{bel}(x_t)$$

$$\int_{-\infty}^{\infty} bel(x_t) = 1$$

η :

Kalman Filter: Overview

- **Kalman filter** is a closed-form solution to the Bayes filter for $\underbrace{\textbf{Linear}}_{\text{Dynamic Obs}}$-$\underbrace{\textbf{Gaussian}}_{\text{Noise Distrib}}$ systems.
- Represents belief as a Gaussian: $bel(x_t) = \mathcal{N}(x_t; \mu_t, \Sigma_t)$.
- Two main steps each iteration:
  - **Prediction**: Project state forward using the motion model.
  - **Correction**: Update belief with new sensor measurement.

Actuation

Motion model (state transition):

$$x_t = A_t x_{t-1} + B_t u_t + w_t, \quad w_t \sim \mathcal{N}(0, R_t)$$

Noise Random Dynamics

Measurement model:

$$y_t = C_t x_t + v_t, \quad v_t \sim \mathcal{N}(0, Q_t)$$

$\mathbb{R}^{n_y}$ $\mathbb{R}^{n_x}$ Obs Unreliable Sensors

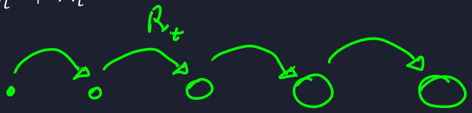
- A_t: state transition matrix
- B_t: control matrix
- C_t: observation matrix
- R_t, Q_t: process and measurement noise covariances

Kalman Filter: Prediction Step

Predict the next state (prior):

$$\overline{\mu}_t = A_t \mu_{t-1} + B_t u_t$$

$$\overline{\Sigma}_t = A_t \Sigma_{t-1} A_t^\top + R_t$$



- $\overline{\mu}_t$: predicted mean of x_t
- $\overline{\Sigma}_t$: predicted covariance of x_t

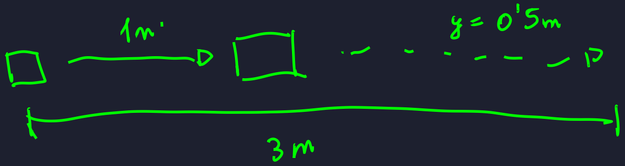
Kalman Filter: How to update the measurement?

- Minimizing this weighted sum of errors:

$$\underset{x_t}{\textbf{argmin}} \quad (y_t - C_t x_t)^\top Q_t^{-1} (y_t - C_t x_t) + (x_t - \overline{\mu}_t)^\top \overline{\Sigma}_t^{-1} (x_t - \overline{\mu}_t)$$

Sensor

- Q_t weights the new observation according to its reliability.
- $\overline{\Sigma}_t$ (which includes R_t from the prediction) weights the prior.



Correct the prediction with the new observation:

$$K_t = \overline{\Sigma}_t C_t^\top (C_t \overline{\Sigma}_t C_t^\top + Q_t)^{-1}$$

$$\mu_t = \overline{\mu}_t + K_t(y_t - C_t \overline{\mu}_t)$$

$$\Sigma_t = (I - K_t C_t)\overline{\Sigma}_t$$

- K_t: Kalman gain
- μ_t, Σ_t: updated mean and covariance of x_t

Kalman

Linear
Gaussian } Optimal!

Close Form

Extended Kalman Filter (EKF)

- The Kalman filter assumes **linear** motion and measurement models.
- The **Extended Kalman Filter (EKF)** generalizes to **nonlinear models**:

$$x_t = \underbrace{f(x_{t-1}, u_t)}_{\text{Dyn}} + w_t, \qquad y_t = \underbrace{h(x_t)}_{\text{Obs}} + v_t$$

- $f(\cdot)$: nonlinear state transition function
- $h(\cdot)$: nonlinear measurement function
- w_t, v_t: process and measurement Gaussian noise

EKF: Linearization via Taylor Series

- EKF approximates $f(\cdot)$ and $h(\cdot)$ **locally by their first-order Taylor expansion**.
- Linearize at current state estimates:

$$F_t = \left.\frac{\partial f}{\partial x}\right|_{x=\mu_{t-1},\,u=u_t} \qquad H_t = \left.\frac{\partial h}{\partial x}\right|_{x=\bar{\mu}_t}$$

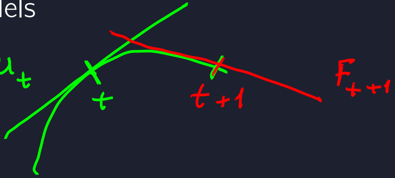
- F_t, H_t: Jacobians of motion and measurement models

$A_t \; B_t \;|\; F_t$

$C_t \;|\; H_t$

$x_t = F_{At} \cdot x_{t-1} + F_{Bt} \cdot u_t$

$y_t = H_t \cdot x_t$



EKF: Summary

- EKF recursively applies the Kalman filter update, but each time **linearizes** f and h using the Taylor expansion at the current mean estimate.
- Works well if models are nearly linear or noise is small.
- For highly nonlinear systems, linearization errors can accumulate and lead to inconsistency.

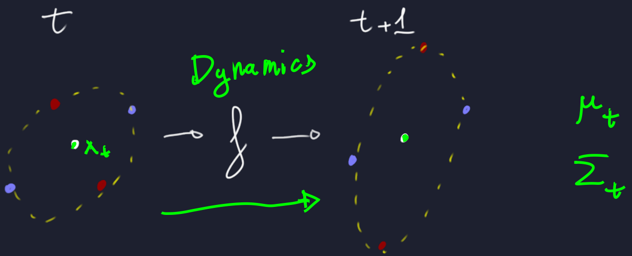
Unscented Kalman Filter (UKF)

- It avoids linearization by using the **unscented transform**, a method to propagate means and covariances through nonlinear functions.
- UKF works with the same nonlinear models as EKF:

$$x_t = \underbrace{f(x_{t-1}, u_t)} + \underset{\substack{\uparrow \\ \text{Gauss}}}{w_t}, \qquad y_t = \underbrace{h(x_t)} + \underset{\substack{\uparrow \\ \text{Gauss}}}{v_t}$$

UKF: The Unscented Transform

- The UKF uses deterministically chosen sigma points to represent and propagate the state distribution through the nonlinear dynamics.
- The new mean and covariance are computed from the transformed sigma points using moment-matching weights.



Unscented Kalman Filter (UKF): Summary

1. **No linearization:** Uses deterministic sigma points, not Jacobians, to handle nonlinear process and measurement models.
2. **Efficient and robust:** Captures the true mean and covariance, typically providing better estimates than EKF for nonlinear problems.
3. **Still Gaussian:** Multimodal dynamics or strongly non gaussian behavior cannot be properly captured.

Summary: KF, EKF, and UKF Comparison

2 strategies

Filter	Model type	Mean/Covariance Update
KF	Linear-Gaussian	Exact analytic formulas
EKF	Nonlinear-Gaussian	Linearize model at x_t (Taylor expansion)
UKF	Nonlinear-Gaussian	Unscented transform (sigma points)

Multi modal

- **EKF:** Updates mean/covariance by first-order Taylor series of the model at the estimate.
- **UKF:** Propagates sigma points through the model and reconstructs mean/covariance.

Coming Soon: Non Gaussian