Localization 2: The Kalman Filter

Algorithm

Kalman filter

The **Kalman filter** is a passive, local, probabilistic localization algorithm that computes the Gaussian probability density over states in an LG system.

Inputs:

- μ_k
- Σ_k
- u_k
- y_k
- A, B, C, G, H
- $\Sigma_\theta, \Sigma_\psi$

Outputs:

- μ_{k+1}
- Σ_{k+1}

Kalman update rule

To execute the Kalman filter, there are four steps.

1. Compute Σ'_{k+1}, an intermediate covariance that accounts for u_k but not y_k:

$$\Sigma'_{k+1} = A\Sigma_k A^\top + G\Sigma_\theta G^\top$$

2. Compute L_{k+1}, a partial result that makes the equations simpler:

$$L_{k+1} = \Sigma'_{k+1} C^\top (C\Sigma'_{k+1} C^\top + H\Sigma_\psi H)^{-1}$$

Kalman update rule

To execute the Kalman filter, there are four steps.

3. Compute μ_{k+1}:

$$\mu_{k+1} = A\mu_k + Bu_k + L_{k+1}(y_k - C(A\mu_k + Bu_k))$$

4. Compute Σ_{k+1}:

$$\Sigma_{k+1} = (I - L_{k+1}C)\Sigma'_{k+1}$$

Extended Kalman filter

What happens if we don't have a linear system?

$$x_{k+1} = f(x_k, u_k) + \theta_k$$

$$y_k = h(x_k) + \psi_k$$

Extended Kalman filter

What happens if we don't have a linear system?

$$x_{k+1} = f(x_k, u_k) + \theta_k$$

$$y_k = h(x_k) + \psi_k$$

Good news: If we make a linear approximation the system (by taking partial derivatives of f and h), then we can still use the Kalman filter updates.

This is called the **Extended Kalman Filter (EKF)**.

Extended Kalman filter

What happens if we don't have a linear system?

$$x_{k+1} = f(x_k, u_k) + \theta_k$$

$$y_k = h(x_k) + \psi_k$$

Bad news: If we do this, we lose the guarantee that we are representing $P(x_k \mid u_1, \ldots, u_k, y_1, \ldots, y_k)$ exactly. In statistical terms, we no longer have an **optimal estimator**.

Extended Kalman filter

What happens if we don't have a linear system?

$$x_{k+1} = f(x_k, u_k) + \theta_k$$

$$y_k = h(x_k) + \psi_k$$

Additional good news: In spite of this lack of guarantees, the EKF generally works well in practice, especially when the non-linearities are not too great.