

Notes on Kalman Filter (KF, EKF, ESKF, IEKF, IESKF)

Gyubeom Edward Im*

February 9, 2024

Contents

1 Recursive Bayes Filter	2
1.1 Derivation of Recursive Bayes Filter	3
1.2 Gaussian Belief Case	4
2 Kalman Filter (KF)	5
2.1 Prediction step	6
2.2 Correction step	6
2.3 Summary	6
3 Extended Kalman Filter (EKF)	7
3.1 Prediction step	8
3.2 Correction step	8
3.3 Summary	9
4 Error-state kalman filter (ESKF)	9
4.1 Prediction step	11
4.2 Correction step	12
4.2.1 Reset	12
4.3 Summary	12
5 Iterated extended kalman filter (IEKF)	13
5.1 Compare to EKF	13
5.1.1 Commonality 1	13
5.1.2 Commonality 2	13
5.1.3 Commonality 3	13
5.1.4 Difference 1	13
5.1.5 Difference 2	14
5.2 Prediction step	14
5.3 Correction step	14
5.4 Summary	14
6 Iterated error-state kalman filter (IESKF)	15
6.1 Compare to ESKF	15
6.1.1 Commonality 1	15
6.1.2 Commonality 2	16
6.1.3 Commonality 3	16
6.1.4 Difference (MAP-based derivation)	16
6.2 Prediction step	18
6.3 Correction step	18
6.4 Summary	18

*blog: alida.tistory.com, email: criterion.im@gmail.com

7 MAP, GN, and EKF relationship	19
7.1 Traditional EKF derivation	19
7.2 MAP-based EKF derivation	19
7.2.1 Start from MAP estimator	19
7.2.2 MLE of new observation function	20
7.2.3 Gauss-Newton Optimization	20
8 Wrap-up	22
8.1 Kalman Filter (KF)	23
8.2 Extended Kalman Filter (EKF)	23
8.3 Error-state Kalman Filter (ESKF)	24
8.4 Iterated Extended Kalman Filter (IEKF)	24
8.5 Iterated Error-state Kalman Filter (IESKF)	25
9 Reference	25
10 Revision log	26

1 Recursive Bayes Filter

State estimation problems generally refer to estimating the current state (e.g., position, speed, rotation) based on given control inputs and observations. Given control inputs and observations, the degree of reliability of the current state \mathbf{x}_t , or the Belief $\text{bel}(\mathbf{x}_t)$ about \mathbf{x}_t , is defined as follows:

$$\text{bel}(\mathbf{x}_t) = p(\mathbf{x}_t \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \quad (1)$$

- \mathbf{x}_t : State variable at time t
- $\mathbf{z}_{1:t} = \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$: Observations from time 1 to t
- $\mathbf{u}_{1:t} = \{\mathbf{u}_1, \dots, \mathbf{u}_t\}$: Control inputs from time 1 to t
- $\text{bel}(\mathbf{x}_t)$: Belief of \mathbf{x}_t , which represents the probability (degree of belief) that the robot is at \mathbf{x}_t based on sensor observations $\mathbf{z}_{1:t}$ and control inputs $\mathbf{u}_{1:t}$ from the start time to t seconds.

$\text{bel}(\cdot)$ is expressed and developed according to the Bayesian rule, so it is also called the Bayes filter. Using the Markov assumption and the Bayesian rule, a recursive filter can be induced, and this is called the recursive Bayes filter.

$$\begin{aligned} \text{bel}(\mathbf{x}_t) &= \eta \cdot p(\mathbf{z}_t \mid \mathbf{x}_t) \overline{\text{bel}}(\mathbf{x}_t) \\ \overline{\text{bel}}(\mathbf{x}_t) &= \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_t) \text{bel}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1} \end{aligned} \quad (2)$$

- $\eta = 1/p(\mathbf{z}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t})$: A normalization factor to maintain the definition of the probability distribution by normalizing its width to 1
- $p(\mathbf{z}_t \mid \mathbf{x}_t)$: Observation model
- $\int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_t) d\mathbf{x}_{t-1}$: Motion model

The Recursive Bayes filter is thus called a recursive filter because it calculates the current step's $\text{bel}(\mathbf{x}_t)$ from the previous step's $\text{bel}(\mathbf{x}_{t-1})$.

1.1 Derivation of Recursive Bayes Filter

The formulation of the Recursive Bayes Filter is derived as follows:

$$\begin{aligned}
\text{bel}(\mathbf{x}_t) &= p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \text{bel}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \overline{\text{bel}}(\mathbf{x}_{t-1})
\end{aligned} \tag{3}$$

Steps applied include Bayesian rule, Markov Assumption, and Marginalization to derive the recursive filter process.

Step 1:

$$\begin{aligned}
\text{bel}(\mathbf{x}_t) &= p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t})
\end{aligned} \tag{4}$$

Apply the Bayesian rule: $p(x|y) = \frac{p(y|x)p(x)}{p(y)}$

Step 2:

$$\begin{aligned}
\text{bel}(\mathbf{x}_t) &= p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t})
\end{aligned} \tag{5}$$

Apply the Markov Assumption that the current state depends only on the immediately previous state.

Step 3:

$$\begin{aligned}
\text{bel}(\mathbf{x}_t) &= p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1}
\end{aligned} \tag{6}$$

Apply the Law of Total Probability or Marginalization: $p(x) = \int_y p(x|y) \cdot p(y) dy$

Step 4:

$$\begin{aligned}
\text{bel}(\mathbf{x}_t) &= p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1}
\end{aligned} \tag{7}$$

Apply the Markov Assumption that the current state depends only on the immediately previous state.

Step 5:

$$\begin{aligned}
\text{bel}(\mathbf{x}_t) &= p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}) d\mathbf{x}_{t-1}
\end{aligned} \tag{8}$$

Apply the Markov Assumption that the current state depends only on the immediately previous state.

Step 6:

$$\begin{aligned}
\text{bel}(\mathbf{x}_t) &= p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot \text{bel}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1}
\end{aligned} \tag{9}$$

$p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1})$ is substituted by $\text{bel}(\mathbf{x}_{t-1})$ as they are equivalent.

Step 7:

$$\begin{aligned}
\text{bel}(\mathbf{x}_t) &= p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot p(\mathbf{x}_t | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot p(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot \text{bel}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1} \\
&= \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \cdot \overline{\text{bel}}(\mathbf{x}_{t-1})
\end{aligned} \tag{10}$$

The integral $\int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) \cdot \text{bel}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1}$ is replaced by $\overline{\text{bel}}(\mathbf{x}_{t-1})$.

1.2 Gaussian Belief Case

If $\text{bel}(\mathbf{x}_t)$ follows a Gaussian distribution, this is specifically referred to as the Kalman filter.

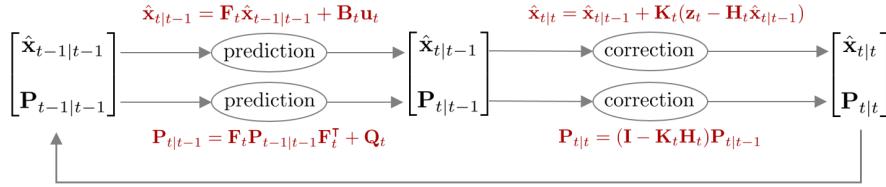
$$\begin{aligned}
\overline{\text{bel}}(\mathbf{x}_t) &\sim \mathcal{N}(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}) \quad (\text{Kalman Filter Prediction}) \\
\text{bel}(\mathbf{x}_t) &\sim \mathcal{N}(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}) \quad (\text{Kalman Filter Correction})
\end{aligned} \tag{11}$$

Mean and covariance are also denoted as $(\hat{\mathbf{x}}, \mathbf{P})$ or $(\hat{\mu}, \Sigma)$, representing the same concept with different notations.

2 Kalman Filter (KF)

NOMENCLATURE of Kalman Filter

- Scalars are denoted by lowercase letters, e.g., a
- Vectors are denoted by bold lowercase letters, e.g., \mathbf{a}
- Matrices are denoted by bold uppercase letters, e.g., \mathbf{R}
- prediction: $\overline{\text{bel}}(\mathbf{x}_t) \sim \mathcal{N}(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1})$
 - $\hat{\mathbf{x}}_{t|t-1}$: Mean at step t given the correction value at step $t - 1$. Some literature also denotes this as \mathbf{x}_t^- .
 - $\hat{\mathbf{P}}_{t|t-1}$: Covariance at step t given the correction value at step $t - 1$. Some literature also denotes this as \mathbf{P}_t^- .
- correction: $\text{bel}(\mathbf{x}_t) \sim \mathcal{N}(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t})$
 - $\hat{\mathbf{x}}_{t|t}$: Mean at step t given the prediction value at step t . Some literature also denotes this as \mathbf{x}_t^+ .
 - $\hat{\mathbf{P}}_{t|t}$: Covariance at step t given the prediction value at step t . Some literature also denotes this as \mathbf{P}_t^+ .



At time t , the robot's position is denoted by \mathbf{x}_t , the measurements from the robot's sensor by \mathbf{z}_t , and the robot's control input by \mathbf{u}_t . Using these, we can define the motion model and observation model. The models are constrained by the requirement that they must be linear.

$$\begin{array}{ll} \text{Motion Model:} & \mathbf{x}_t = \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{B}_t \mathbf{u}_t \\ \text{Observation Model:} & \mathbf{z}_t = \mathbf{H}_t \mathbf{x}_t + \mathbf{v}_t \end{array} \quad (12)$$

- \mathbf{x}_t : Model state variable
- \mathbf{u}_t : Model input
- \mathbf{z}_t : Model measurement
- \mathbf{F}_t : Model state transition matrix
- \mathbf{B}_t : Matrix that transforms input \mathbf{u}_t to the state variable when given
- \mathbf{H}_t : Model observation matrix
- $\mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$: Motion model noise. \mathbf{Q}_t denotes the covariance matrix of \mathbf{w}_t .
- $\mathbf{v}_t \sim \mathcal{N}(0, \mathbf{R}_t)$: Observation model noise. \mathbf{R}_t denotes the covariance matrix of \mathbf{v}_t .

Assuming all random variables follow Gaussian distributions, $p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t), p(\mathbf{z}_t | \mathbf{x}_t)$ can be represented as follows.

$$\begin{aligned} p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) &\sim \mathcal{N}(\mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{B}_t \mathbf{u}_t, \mathbf{Q}_t) \\ &= \frac{1}{\sqrt{\det(2\pi \mathbf{Q}_t)}} \exp \left(-\frac{1}{2} (\mathbf{x}_t - \mathbf{F}_t \mathbf{x}_{t-1} - \mathbf{B}_t \mathbf{u}_t)^T \mathbf{Q}_t^{-1} (\mathbf{x}_t - \mathbf{F}_t \mathbf{x}_{t-1} - \mathbf{B}_t \mathbf{u}_t) \right) \end{aligned} \quad (13)$$

$$\begin{aligned} p(\mathbf{z}_t | \mathbf{x}_t) &\sim \mathcal{N}(\mathbf{H}_t \mathbf{x}_t, \mathbf{R}_t) \\ &= \frac{1}{\sqrt{\det(2\pi \mathbf{R}_t)}} \exp \left(-\frac{1}{2} (\mathbf{z}_t - \mathbf{H}_t \mathbf{x}_t)^T \mathbf{R}_t^{-1} (\mathbf{z}_t - \mathbf{H}_t \mathbf{x}_t) \right) \end{aligned} \quad (14)$$

Next, the $\overline{\text{bel}}(\mathbf{x}_t)$, $\text{bel}(\mathbf{x}_t)$ that need to be computed through the Kalman filter can be represented as follows.

$$\begin{aligned}\overline{\text{bel}}(\mathbf{x}_t) &= \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{u}_t) \text{bel}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1} \sim \mathcal{N}(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}) \\ \text{bel}(\mathbf{x}_t) &= \eta \cdot p(\mathbf{z}_t \mid \mathbf{x}_t) \overline{\text{bel}}(\mathbf{x}_t) \sim \mathcal{N}(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t})\end{aligned}\quad (15)$$

As seen in (15), the Kalman filter operates by first computing the predicted value $\overline{\text{bel}}(\mathbf{x}_t)$ using the previous step's value and motion model in the prediction phase, then obtaining the corrected value $\text{bel}(\mathbf{x}_t)$ using the measurement and observation model in the correction phase. By substituting (13) and (14) into (15), we can derive the means and covariances for the prediction and correction steps $(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1})$, $(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t})$ respectively. For a detailed derivation process, refer to Section 3.2.4 "Mathematical Derivation of the KF" in [[13]].

The initial value $\text{bel}(\mathbf{x}_0)$ is given as follows.

$$\text{bel}(\mathbf{x}_0) \sim \mathcal{N}(\hat{\mathbf{x}}_0, \mathbf{P}_0) \quad (16)$$

- $\hat{\mathbf{x}}_0$: Typically set to 0
- \mathbf{P}_0 : Typically set to a small value ($< 1e-2$).

2.1 Prediction step

Prediction involves computing $\overline{\text{bel}}(\mathbf{x}_t)$. Since $\overline{\text{bel}}(\mathbf{x}_t)$ follows a Gaussian distribution, we can compute the mean $\hat{\mathbf{x}}_{t|t-1}$ and variance $\mathbf{P}_{t|t-1}$ as follows.

$$\boxed{\begin{aligned}\hat{\mathbf{x}}_{t|t-1} &= \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} + \mathbf{B}_t \mathbf{u}_t \\ \mathbf{P}_{t|t-1} &= \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{Q}_t\end{aligned}} \quad (17)$$

2.2 Correction step

Correction involves computing $\text{bel}(\mathbf{x}_t)$. Since $\text{bel}(\mathbf{x}_t)$ also follows a Gaussian distribution, we can compute the mean $\hat{\mathbf{x}}_{t|t}$ and variance $\mathbf{P}_{t|t}$ as follows. Here, \mathbf{K}_t represents the Kalman gain.

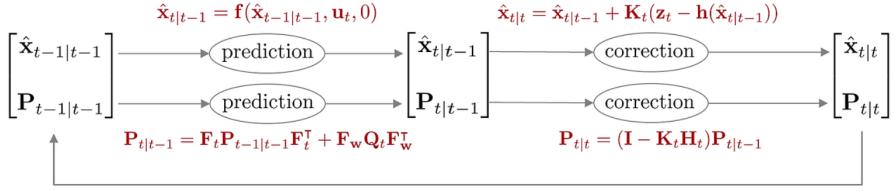
$$\boxed{\begin{aligned}\mathbf{K}_t &= \mathbf{P}_{t|t-1} \mathbf{H}_t^\top (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^\top + \mathbf{R}_t)^{-1} \\ \hat{\mathbf{x}}_{t|t} &= \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{z}_t - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1}) \\ \mathbf{P}_{t|t} &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1}\end{aligned}} \quad (18)$$

2.3 Summary

The Kalman Filter can be expressed as a function as follows.

$$\boxed{\begin{aligned}\text{KalmanFilter}(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{P}_{t-1|t-1}, \mathbf{u}_t, \mathbf{z}_t) \{ & \\ & \text{(Prediction Step)} \\ & \hat{\mathbf{x}}_{t|t-1} = \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} + \mathbf{B}_t \mathbf{u}_t \\ & \mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{Q}_t \\ & \\ & \text{(Correction Step)} \\ & \mathbf{K}_t = \mathbf{P}_{t|t-1} \mathbf{H}_t^\top (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^\top + \mathbf{R}_t)^{-1} \\ & \hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{z}_t - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1}) \\ & \mathbf{P}_{t|t} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1} \\ & \text{return } \hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t} \\ \}\end{aligned}} \quad (19)$$

3 Extended Kalman Filter (EKF)



Kalman filters assume that both the motion model and the observation model are linear to estimate the state. However, since most phenomena in the real world are modeled non-linearly, applying the Kalman filter as previously defined would not work properly. The extended Kalman filter (EKF) was proposed to use the Kalman filter even in non-linear motion and observation models. **EKF uses the first-order Taylor approximation to approximate the non-linear model to a linear model before applying the Kalman filter.** The motion model and observation model of the EKF are as follows.

$$\begin{aligned} \text{Motion Model: } & \mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_t, \mathbf{w}_t) \\ \text{Observation Model: } & \mathbf{z}_t = \mathbf{h}(\mathbf{x}_t, \mathbf{v}_t) \end{aligned} \quad (20)$$

- \mathbf{x}_t : state variable of the model
- \mathbf{u}_t : input of the model
- \mathbf{z}_t : measurement of the model
- $\mathbf{f}(\cdot)$: non-linear motion model function
- $\mathbf{h}(\cdot)$: non-linear observation model function
- $\mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$: noise in the motion model. \mathbf{Q}_t represents the covariance matrix of \mathbf{w}_t
- $\mathbf{v}_t \sim \mathcal{N}(0, \mathbf{R}_t)$: noise in the observation model. \mathbf{R}_t represents the covariance matrix of \mathbf{v}_t

In the above formula, $\mathbf{f}(\cdot)$ means a non-linear motion model and $\mathbf{h}(\cdot)$ means a non-linear observation model. When the first-order Taylor approximation is performed on $\mathbf{f}(\cdot), \mathbf{h}(\cdot)$, it becomes as follows.

$$\begin{aligned} \mathbf{x}_t &\approx \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0) + \mathbf{F}_t (\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1|t-1}) + \mathbf{F}_w \mathbf{w}_t \\ \mathbf{z}_t &\approx \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0) + \mathbf{H}_t (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) + \mathbf{H}_v \mathbf{v}_t \end{aligned} \quad (21)$$

At this time, \mathbf{F}_t represents the Jacobian matrix of the motion model calculated at $\hat{\mathbf{x}}_{t-1|t-1}$, and \mathbf{H}_t represents the Jacobian matrix of the observation model calculated at $\hat{\mathbf{x}}_{t|t-1}$. And $\mathbf{F}_w, \mathbf{H}_v$ represent the Jacobian matrix of the noise when $\mathbf{w}_t = 0, \mathbf{v}_t = 0$ respectively. For more information on the Jacobian, refer to this post.

$$\mathbf{F}_t = \frac{\partial \mathbf{f}}{\partial \mathbf{x}_{t-1}} \Big|_{\mathbf{x}_{t-1}=\hat{\mathbf{x}}_{t-1|t-1}} \quad \mathbf{F}_w = \frac{\partial \mathbf{f}}{\partial \mathbf{w}_t} \Big|_{\substack{\mathbf{x}_{t-1}=\hat{\mathbf{x}}_{t-1|t-1} \\ \mathbf{w}_t=0}} \quad (22)$$

$$\mathbf{H}_t = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_t} \Big|_{\mathbf{x}_t=\hat{\mathbf{x}}_{t|t-1}} \quad \mathbf{H}_v = \frac{\partial \mathbf{h}}{\partial \mathbf{v}_t} \Big|_{\substack{\mathbf{x}_t=\hat{\mathbf{x}}_{t|t-1} \\ \mathbf{v}_t=0}} \quad (23)$$

(21) If the formula is expanded, it becomes as follows.

$$\begin{aligned} \mathbf{x}_t &= \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0) - \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} + \mathbf{F}_w \mathbf{w}_t \\ &= \mathbf{F}_t \mathbf{x}_{t-1} + \tilde{\mathbf{u}}_t + \tilde{\mathbf{w}}_t \end{aligned} \quad (24)$$

- $\tilde{\mathbf{u}}_t = \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0) - \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1}$ - $\tilde{\mathbf{w}}_t = \mathbf{F}_w \mathbf{w}_t \sim \mathcal{N}(0, \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top)$

$$\begin{aligned} \mathbf{z}_t &= \mathbf{H}_t \mathbf{x}_t + \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0) - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1} + \mathbf{H}_v \mathbf{v}_t \\ &= \mathbf{H}_t \mathbf{x}_t + \tilde{\mathbf{z}}_t + \tilde{\mathbf{v}}_t \end{aligned} \quad (25)$$

- $\tilde{\mathbf{z}}_t = \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0) - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1}$ - $\tilde{\mathbf{v}}_t = \mathbf{H}_v \mathbf{v}_t \sim \mathcal{N}(0, \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)$

Assuming all random variables follow a Gaussian distribution, $p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_t), p(\mathbf{z}_t|\mathbf{x}_t)$ can be expressed as follows.

$$\begin{aligned} p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_t) &\sim \mathcal{N}(\mathbf{F}_t \mathbf{x}_{t-1} + \tilde{\mathbf{u}}_t, \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top) \\ &= \frac{1}{\sqrt{\det(2\pi \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top)}} \exp\left(-\frac{1}{2}(\mathbf{x}_t - \mathbf{F}_t \mathbf{x}_{t-1} - \tilde{\mathbf{u}}_t)^\top (\mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top)^{-1} (\mathbf{x}_t - \mathbf{F}_t \mathbf{x}_{t-1} - \tilde{\mathbf{u}}_t)\right) \end{aligned} \quad (26)$$

$$\begin{aligned} p(\mathbf{z}_t|\mathbf{x}_t) &\sim \mathcal{N}(\mathbf{H}_t \mathbf{x}_t + \tilde{\mathbf{z}}_t, \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top) \\ &= \frac{1}{\sqrt{\det(2\pi \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)}} \exp\left(-\frac{1}{2}(\mathbf{z}_t - \mathbf{H}_t \mathbf{x}_t - \tilde{\mathbf{z}}_t)^\top (\mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)^{-1} (\mathbf{z}_t - \mathbf{H}_t \mathbf{x}_t - \tilde{\mathbf{z}}_t)\right) \end{aligned} \quad (27)$$

$\mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top$ represents the noise in the linearized motion model, and $\mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top$ represents the noise in the linearized observation model. Next, $\overline{\text{bel}}(\mathbf{x}_t)$, $\text{bel}(\mathbf{x}_t)$ that need to be found through the Kalman filter can be expressed as follows.

$$\begin{aligned} \overline{\text{bel}}(\mathbf{x}_t) &= \int p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_t) \text{bel}(\mathbf{x}_{t-1}) d\mathbf{x}_{t-1} \sim \mathcal{N}(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}) \\ \text{bel}(\mathbf{x}_t) &= \eta \cdot p(\mathbf{z}_t|\mathbf{x}_t) \overline{\text{bel}}(\mathbf{x}_t) \sim \mathcal{N}(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}) \end{aligned} \quad (28)$$

EKF operates in a similar manner to KF, using the values from the previous step and the motion model in the prediction to first obtain the predicted value $\overline{\text{bel}}(\mathbf{x}_t)$, and then using the observed values and observation model in the correction to obtain the corrected value $\text{bel}(\mathbf{x}_t)$. Substitute $p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_t)$, $p(\mathbf{z}_t|\mathbf{x}_t)$ into the above equation to obtain the mean and covariance of the prediction and correction steps $(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}), (\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t})$ respectively. Refer to section 3.2.4 "Mathematical Derivation of the KF" in [[13]] for detailed derivation.

The initial value $\text{bel}(\mathbf{x}_0)$ is given as follows.

$$\text{bel}(\mathbf{x}_0) \sim \mathcal{N}(\hat{\mathbf{x}}_0, \mathbf{P}_0) \quad (29)$$

- $\hat{\mathbf{x}}_0$: Typically set to 0
- \mathbf{P}_0 : Typically set to a small value ($< 1e-2$)

3.1 Prediction step

Prediction involves calculating $\overline{\text{bel}}(\mathbf{x}_t)$. The covariance matrix is obtained using the linearized Jacobian matrix \mathbf{F}_t .

$$\boxed{\begin{aligned} \hat{\mathbf{x}}_{t|t-1} &= \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0) \\ \mathbf{P}_{t|t-1} &= \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top \end{aligned}} \quad (30)$$

3.2 Correction step

Correction involves calculating $\text{bel}(\mathbf{x}_t)$. The Kalman gain and covariance matrix are obtained using the linearized Jacobian matrix \mathbf{H}_t .

$$\boxed{\begin{aligned} \mathbf{K}_t &= \mathbf{P}_{t|t-1} \mathbf{H}_t^\top (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^\top + \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)^{-1} \\ \hat{\mathbf{x}}_{t|t} &= \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0)) \\ \mathbf{P}_{t|t} &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1} \end{aligned}} \quad (31)$$

3.3 Summary

The Extended Kalman Filter can be expressed as a function as follows.

```

ExtendedKalmanFilter( $\hat{\mathbf{x}}_{t-1|t-1}$ ,  $\mathbf{P}_{t-1|t-1}$ ,  $\mathbf{u}_t$ ,  $\mathbf{z}_t$ ){
    (Prediction Step)
     $\hat{\mathbf{x}}_{t|t-1} = \mathbf{f}_t(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0)$ 
     $\mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top$ 

    (Correction Step)
     $\mathbf{K}_t = \mathbf{P}_{t|t-1} \mathbf{H}_t^\top (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t + \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v) ^{-1}$ 
     $\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}_t(\hat{\mathbf{x}}_{t|t-1}, 0))$ 
     $\mathbf{P}_{t|t} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1}$ 
    return  $\hat{\mathbf{x}}_{t|t}$ ,  $\mathbf{P}_{t|t}$ 
}

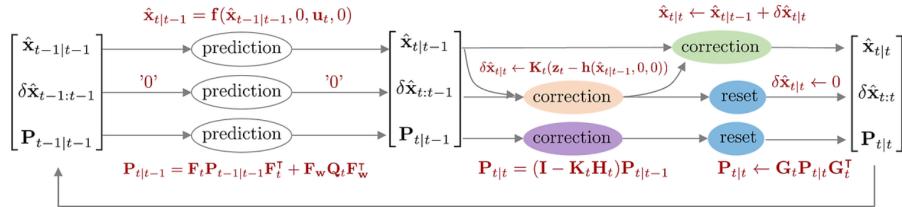
```

(32)

4 Error-state kalman filter (ESKF)

NOMENCLATURE of error-state kalman filter

- prediction: $\overline{\text{bel}}(\delta\mathbf{x}_t) \sim \mathcal{N}(\delta\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1})$
 - $\delta\hat{\mathbf{x}}_{t|t-1}$: $t-1$ 스텝의 correction 값이 주어졌을 때 t 스텝의 평균. 일부 문헌은 $\delta\mathbf{x}_t^-$ 로도 표기함.
 - $\hat{\mathbf{P}}_{t|t-1}$: $t-1$ 스텝의 correction 값이 주어졌을 때 t 스텝의 공분산. 일부 문헌은 \mathbf{P}_t^- 로도 표기함.
- correction: $\text{bel}(\delta\mathbf{x}_t) \sim \mathcal{N}(\delta\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t})$
 - $\delta\hat{\mathbf{x}}_{t|t}$: t 스텝의 prediction 값이 주어졌을 때 t 스텝의 평균. 일부 문헌은 $\delta\mathbf{x}_t^+$ 로도 표기함.
 - $\hat{\mathbf{P}}_{t|t}$: t 스텝의 prediction 값이 주어졌을 때 t 스텝의 공분산. 일부 문헌은 \mathbf{P}_t^+ 로도 표기함.



에러상태 칼만필터(error-state kalman filter, ESKF)는 기존의 상태 변수 \mathbf{x}_t 의 평균과 분산을 추정하는 EKF와 달리 에러 상태 변수 $\delta\mathbf{x}_t$ 의 평균과 분산을 추정하는 칼만 필터 알고리즘을 말한다. 상태 변수를 바로 추정하지 않고 에러상태를 통해 추정하기 때문에 indirect kalman filter라고도 불린다. 또 다른 이름으로는 error-state extended kalman filter(ES-EKF)로도 불린다. ESKF는 기존 상태변수를 true 상태 변수라고 부르며 이를 다음과 같이 명목(nominal) 상태와 에러(error) 상태의 합으로 표현한다.

$$\mathbf{x}_{\text{true},t} = \mathbf{x}_t + \delta\mathbf{x}_t \quad (33)$$

- $\mathbf{x}_{\text{true},t}$: 기존 KF, EKF에서 업데이트된 t 스텝의 true 상태변수
- \mathbf{x}_t : t 스텝의 명목(nominal) 상태 변수. 에러가 없는 상태를 의미한다
- $\delta\mathbf{x}_t$: t 스텝의 에러(error) 상태 변수

위 식을 해석하면 실제(true) 추정하고자 하는 상태 변수 \mathbf{x}_t 는 에러가 없는 일반(또는 명목, nominal) 상태 $\hat{\mathbf{x}}_t$ 과 모델 및 센서 노이즈로부터 발생하는 에러 상태 $\delta\mathbf{x}_t$ 의 합으로 나타낼 수 있다는 의미이다. 이 때, nominal 상태는 (상대적으로) 큰 값을 가지며 비선형성을 가진다. 반면에 에러 상태는 0 근처의 작은 값을 가지고 선형성을 가진다. **기존의 EKF는 비선형성이 큰 true (nominal + error) 상태 변수를 선형화하여 필터링하기 때문에 속도가 느리고 시간이 지날수록 에러 값이 누적되는 반면에, ESKF는 에러 상태만을 선형화하여 필터링하기 때문에 속도 및 정확성이 더욱 빠른 장점이 있다.** 기존 EKF와 비교했을 때 ESKF 이 가지는 장점들을 정리하면 다음과 같다(Madyastha et al., 2011):

- 방향(orientation)에 대한 여러 상태 표현법이 최소한의 파라미터를 가진다. 즉, 자유도만큼의 최소 파라미터를 가지기 때문에 over-parameterized로 인해 발생하는 특이점(signularity) 같은 현상이 발생하지 않는다.
- 여러 상태 시스템은 항상 원점(origin) 근처에서만 동작하기 때문에 선형화하기 용이하다. 따라서 점 별락 같은 파라미터 특이점 현상이 발생하지 않으며 항상 선형화를 수행할 수 있다.
- 여러 상태는 일반적으로 값이 작기 때문에 2차항 이상의 값들은 무시할 수 있다. 이는 자코비안 연산을 쉽고 빠르게 수행할 수 있도록 도와준다. 몇몇 자코비안은 상수화하여 사용하기도 한다.

다만 ESKF는 prediction 속도는 빠르지만 nominal 상태에서 일반적으로 비선형을 가진 큰 값들이 처리되므로 nominal 상태가 처리되는 correction 스텝의 속도가 느린 편이다.

ESKF의 모션 모델과 관측 모델은 다음과 같다.

$$\begin{aligned} \text{Error-state Motion Model: } \quad \mathbf{x}_t + \delta\mathbf{x}_t &= \mathbf{f}(\mathbf{x}_{t-1}, \delta\mathbf{x}_{t-1}, \mathbf{u}_t, \mathbf{w}_t) \\ \text{Error-state Observation Model: } \quad \mathbf{z}_t &= \mathbf{h}(\mathbf{x}_t, \delta\mathbf{x}_t, \mathbf{v}_t) \end{aligned} \quad (34)$$

- \mathbf{x}_t : 모델의 nominal 상태 변수
- $\delta\mathbf{x}_t$: 모델의 에러 상태 변수
- \mathbf{u}_t : 모델의 입력(input)
- \mathbf{z}_t : 모델의 관측값(measurement)
- $\mathbf{f}(\cdot)$: 비선형 모션(motion) 모델 함수
- $\mathbf{h}(\cdot)$: 비선형 관측(observation) 모델 함수
- $\mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$: 에러 상태 모델의 노이즈. \mathbf{Q}_t 는 \mathbf{w}_t 의 공분산 행렬을 의미
- $\mathbf{v}_t \sim \mathcal{N}(0, \mathbf{R}_t)$: 관측 모델의 노이즈. \mathbf{R}_t 는 \mathbf{v}_t 의 공분산 행렬을 의미

$\mathbf{f}(\cdot), \mathbf{h}(\cdot)$ 에 각각 1차 테일러 근사를 수행하면 다음과 같다. 해당 전개는 [5]를 참고하여 작성하였다.

$$\begin{aligned} \mathbf{x}_t + \delta\mathbf{x}_t &\approx \mathbf{f}(\mathbf{x}_{t-1|t-1}, \hat{\delta}\mathbf{x}_{t-1|t-1}, \mathbf{u}_t, 0) + \mathbf{F}_t(\delta\mathbf{x}_{t-1} - \hat{\delta}\mathbf{x}_{t-1|t-1}) + \mathbf{F}_w\mathbf{w}_t \\ \mathbf{z}_t &\approx \mathbf{h}(\mathbf{x}_{t|t-1}, \hat{\delta}\mathbf{x}_{t|t-1}, 0) + \mathbf{H}_t(\delta\mathbf{x}_t - \hat{\delta}\mathbf{x}_{t|t-1}) + \mathbf{H}_v\mathbf{v}_t \end{aligned} \quad (35)$$

이 때, 두 자코비안 $\mathbf{F}_t, \mathbf{H}_t$ 모두 true 상태 $\mathbf{x}_{true,t}$ 가 아닌 에러 상태 $\delta\mathbf{x}_t$ 에 대한 자코비안임에 유의한다. 해당 자코비안 부분이 EKF와 가장 다른 부분이다

$$\boxed{\mathbf{F}_t = \frac{\partial \mathbf{f}}{\partial \delta\mathbf{x}_{t-1}} \Big|_{\delta\mathbf{x}_{t-1} = \hat{\delta}\mathbf{x}_{t-1|t-1}} \quad \mathbf{F}_w = \frac{\partial \mathbf{f}}{\partial \mathbf{w}_t} \Big|_{\delta\mathbf{x}_{t-1} = \hat{\delta}\mathbf{x}_{t-1|t-1}, \mathbf{w}_t = 0}} \quad (36)$$

$$\boxed{\mathbf{H}_t = \frac{\partial \mathbf{h}}{\partial \delta\mathbf{x}_t} \Big|_{\delta\mathbf{x}_t = \hat{\delta}\mathbf{x}_{t|t-1}} \quad \mathbf{H}_v = \frac{\partial \mathbf{h}}{\partial \mathbf{v}_t} \Big|_{\delta\mathbf{x}_t = \hat{\delta}\mathbf{x}_{t|t-1}, \mathbf{v}_t = 0}} \quad (37)$$

\mathbf{H}_t 는 다음과 같이 연쇄법칙을 통해 표현할 수 있다.

$$\mathbf{H}_t = \frac{\partial \mathbf{h}}{\partial \delta\mathbf{x}_t} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_{true,t}} \frac{\partial \mathbf{x}_{true,t}}{\partial \delta\mathbf{x}_t} \quad (38)$$

이 중 앞 부분 $\frac{\partial \mathbf{h}}{\partial \mathbf{x}_{true,t}}$ 은 EKF에서 구한 자코비안과 동일하지만 에러 상태 변수에 대한 자코비안 $\frac{\partial \mathbf{x}_{true,t}}{\partial \delta\mathbf{x}_t}$ 이 추가되었다. 이에 대한 자세한 내용은 Quaternion kinematics for the error-state Kalman filter 내용 정리 포스트를 참고하면 된다.

(35) 식을 전개하면 다음과 같다.

$$\mathbf{x}_t + \delta\mathbf{x}_t = \mathbf{F}_t \delta\mathbf{x}_{t-1} + \mathbf{f}(\mathbf{x}_{t-1|t-1}, \hat{\delta}\mathbf{x}_{t-1|t-1}, \mathbf{u}_t, 0) - \mathbf{F}_t \hat{\delta}\mathbf{x}_{t-1|t-1} + \mathbf{F}_w \mathbf{w}_t \quad (39)$$

이전 correction 스텝에서 항상 $\hat{\delta}\mathbf{x}_{t-1|t-1} = 0$ 으로 초기화되므로 관련된 항에 0을 대입한다.

$$\mathbf{x}_t + \delta\mathbf{x}_t = \mathbf{F}_t \delta\mathbf{x}_{t-1} + \mathbf{f}(\mathbf{x}_{t-1|t-1}, 0, \mathbf{u}_t, 0) + \mathbf{F}_w \mathbf{w}_t \quad (40)$$

nominal 상태 변수 \mathbf{x}_t 는 정의 상 노이즈가 없는 $\mathbf{f}(\mathbf{x}_{t-1}, 0, \mathbf{u}_t, 0)$ 과 동일하므로 서로 소거된다.

$$\begin{aligned}\delta \mathbf{x}_t &= \mathbf{F}_t \delta \mathbf{x}_{t-1} + \mathbf{F}_w \mathbf{w}_t \\ &= \mathbf{F}_t \delta \mathbf{x}_{t-1} + \tilde{\mathbf{w}}_t \\ &= 0 + \tilde{\mathbf{w}}_t\end{aligned}\quad (41)$$

- $\tilde{\mathbf{w}}_t = \mathbf{F}_w \mathbf{w}_t \sim \mathcal{N}(0, \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top)$

관측 모델 함수도 동일하게 예상 상태 변수의 값을 0으로 치환하면 아래와 같이 전개된다.

$$\begin{aligned}\mathbf{z}_t &= \mathbf{H}_t \delta \mathbf{x}_t + \mathbf{h}(\mathbf{x}_{t|t-1}, \delta \hat{\mathbf{x}}_{t-1|t-1}, 0) - \mathbf{H}_t \delta \hat{\mathbf{x}}_{t|t-1} + \mathbf{H}_v \mathbf{v}_t \\ &= \mathbf{h}(\mathbf{x}_{t|t-1}, 0, 0) + \mathbf{H}_v \mathbf{v}_t \\ &= \tilde{\mathbf{z}}_t + \tilde{\mathbf{v}}_t\end{aligned}\quad (42)$$

- $\tilde{\mathbf{z}}_t = \mathbf{h}(\mathbf{x}_{t|t-1}, 0, 0)$

- $\tilde{\mathbf{v}}_t = \mathbf{H}_v \mathbf{v}_t \sim \mathcal{N}(0, \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)$

학률 변수가 모두 가우시안 분포를 따른다고 가정하면 $p(\delta \mathbf{x}_t | \delta \mathbf{x}_{t-1}, \mathbf{u}_t), p(\mathbf{z}_t | \delta \mathbf{x}_t)$ 는 다음과 같이 나타낼 수 있다.

$$\begin{aligned}p(\delta \mathbf{x}_t | \delta \mathbf{x}_{t-1}, \mathbf{u}_t) &\sim \mathcal{N}(0, \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top) \\ &= \frac{1}{\sqrt{\det(2\pi \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top)}} \exp\left(-\frac{1}{2}(\delta \mathbf{x}_t)^\top (\mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top)^{-1} (\delta \mathbf{x}_t)\right)\end{aligned}\quad (43)$$

$$\begin{aligned}p(\mathbf{z}_t | \delta \mathbf{x}_t) &\sim \mathcal{N}(\tilde{\mathbf{z}}_t, \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top) \\ &= \frac{1}{\sqrt{\det(2\pi \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)}} \exp\left(-\frac{1}{2}(\mathbf{z}_t - \tilde{\mathbf{z}}_t)^\top (\mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)^{-1} (\mathbf{z}_t - \tilde{\mathbf{z}}_t)\right)\end{aligned}\quad (44)$$

$\mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top$ 은 선형화된 예상 상태 모션 모델의 노이즈를 의미하며 $\mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top$ 은 선형화된 관측 모델의 노이즈를 의미한다. 다음으로 칼만 필터를 통해 구해야 하는 $\overline{\text{bel}}(\delta \mathbf{x}_t), \text{bel}(\delta \mathbf{x}_t)$ 은 아래와 같이 나타낼 수 있다.

$$\begin{aligned}\overline{\text{bel}}(\delta \mathbf{x}_t) &= \int p(\delta \mathbf{x}_t | \delta \mathbf{x}_{t-1}, \mathbf{u}_t) \text{bel}(\delta \mathbf{x}_{t-1}) d\mathbf{x}_{t-1} \sim \mathcal{N}(\delta \hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}) \\ \text{bel}(\delta \mathbf{x}_t) &= \eta \cdot p(\mathbf{z}_t | \delta \mathbf{x}_t) \overline{\text{bel}}(\delta \mathbf{x}_t) \sim \mathcal{N}(\delta \hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t})\end{aligned}\quad (45)$$

ESKF 또한 EKF와 동일하게 prediction에서 이전 스텝의 값과 모션 모델을 사용하여 예측값 $\overline{\text{bel}}(\delta \mathbf{x}_t)$ 을 먼저 구한 후 correction에서 관측값과 관측 모델을 사용하여 보정된 값 $\text{bel}(\delta \mathbf{x}_t)$ 을 구하는 방식으로 동작한다. $p(\delta \mathbf{x}_t | \delta \mathbf{x}_{t-1}, \mathbf{u}_t), p(\mathbf{z}_t | \delta \mathbf{x}_t)$ 를 위 식에 대입하여 정리하면 prediction, correction 스텝의 평균과 공분산 $(\delta \hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}), (\delta \hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t})$ 을 각각 구할 수 있다. 자세한 유도 과정은 [[13]]의 3.2.4 "Mathematical Derivation of the KF" 섹션을 참조하면 된다.

초기값 $\text{bel}(\delta \mathbf{x}_0)$ 은 다음과 같이 주어진다.

$$\text{bel}(\delta \mathbf{x}_0) \sim \mathcal{N}(0, \mathbf{P}_0) \quad (46)$$

- $\delta \hat{\mathbf{x}}_0 = 0$: 항상 0의 값을 가진다

- \mathbf{P}_0 : 일반적으로 작은 값(<1e-2)으로 설정한다.

4.1 Prediction step

Prediction은 $\overline{\text{bel}}(\delta \mathbf{x}_t)$ 를 구하는 과정을 말한다. 공분산 행렬을 구할 때 선형화된 자코비안 행렬 \mathbf{F}_t 가 사용된다.

$$\begin{aligned}\delta \hat{\mathbf{x}}_{t|t-1} &= \mathbf{F}_t \delta \hat{\mathbf{x}}_{t-1|t-1} = 0 \quad \leftarrow \text{Always } 0 \\ \hat{\mathbf{x}}_{t|t-1} &= \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, 0, \mathbf{u}_t, 0) \\ \mathbf{P}_{t|t-1} &= \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top\end{aligned}\quad (47)$$

위 식에서 \mathbf{F}_t 는 예상 상태에 대한 자코비안임에 유의한다. 예상 상태 변수 $\delta \hat{\mathbf{x}}$ 는 때 correction 스텝이 끝나면 0으로 초기화된다. 여기에 선형 자코비안 \mathbf{F}_t 를 곱해도 값은 0이 되기 때문에 prediction 스텝에서 $\delta \hat{\mathbf{x}}$ 값은 항상 0이 된다. 따라서 예상 상태 $\delta \hat{\mathbf{x}}_{t|t-1}$ 는 prediction 스텝에서는 변하지 않고 nominal 상태 $\hat{\mathbf{x}}_{t|t-1}$ 와 예상 상태의 공분산 $\mathbf{P}_{t|t-1}$ 만 업데이트된다.

4.2 Correction step

Correction은 $\text{bel}(\delta\mathbf{x}_t)$ 를 구하는 과정을 말한다. 칼만 게인과 공분산 행렬을 구할 때 선형화된 자코비안 행렬 \mathbf{H}_t 가 사용된다.

$$\begin{aligned}
 \mathbf{K}_t &= \mathbf{P}_{t|t-1} \mathbf{H}_t^\top (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^\top + \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v)^{-1} \\
 \delta\hat{\mathbf{x}}_{t|t} &= \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0, 0)) \\
 \hat{\mathbf{x}}_{t|t} &= \hat{\mathbf{x}}_{t|t-1} + \delta\hat{\mathbf{x}}_{t|t} \\
 \mathbf{P}_{t|t} &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1} \\
 \text{reset } \delta\hat{\mathbf{x}}_{t|t} \\
 \delta\hat{\mathbf{x}}_{t|t} &\leftarrow 0 \\
 \mathbf{P}_{t|t} &\leftarrow \mathbf{G} \mathbf{P}_{t|t} \mathbf{G}^\top
 \end{aligned} \tag{48}$$

위 식에서 \mathbf{H}_t 는 관측 모델에 대한 true 상태 $\mathbf{x}_{\text{true},t}$ 의 자코비안이 아닌 에러 상태 $\delta\mathbf{x}_t$ 의 자코비안임에 유의한다. $\mathbf{P}_{t|t-1}, \mathbf{K}_t$ 또한 EKF와 기호만 같은 뿐 실제 값은 다름에 유의한다. 즉, 전체적인 공식은 EKF와 동일하지만 $\mathbf{F}, \mathbf{H}, \mathbf{P}, \mathbf{K}$ 행렬이 에러 상태 $\delta\mathbf{x}_t$ 에 대한 값을 의미하는 점이 다르다.

4.2.1 Reset

nominal 상태가 정상적으로 업데이트되면 다음으로 에러 상태의 값을 0으로 리셋해야 한다. 리셋을 하는 이유는 새로운 nominal 상태에 대한 새로운 에러(new error)를 표현해야 하기 때문이다. 리셋으로 인해 에러 상태의 공분산 $\mathbf{P}_{t|t}$ 이 업데이트된다.

리셋 함수를 $\mathbf{g}(\cdot)$ 라고 하면 이는 다음과 같이 나타낼 수 있다. 이에 대한 자세한 내용은 Quaternion kinematics for the error-state Kalman filter 내용 정리 포스트의 챕터 6 내용을 참고하면 된다.

$$\delta\mathbf{x} \leftarrow \mathbf{g}(\delta\mathbf{x}) = \delta\mathbf{x} - \delta\hat{\mathbf{x}}_{t|t-1} \tag{49}$$

ESKF의 리셋 과정은 다음과 같다.

$$\begin{aligned}
 \delta\hat{\mathbf{x}}_{t|t} &\leftarrow 0 \\
 \mathbf{P}_{t|t} &\leftarrow \mathbf{G} \mathbf{P}_{t|t} \mathbf{G}^\top
 \end{aligned} \tag{50}$$

\mathbf{G} 는 다음과 같이 정의된 리셋에 대한 자코비안을 의미한다.

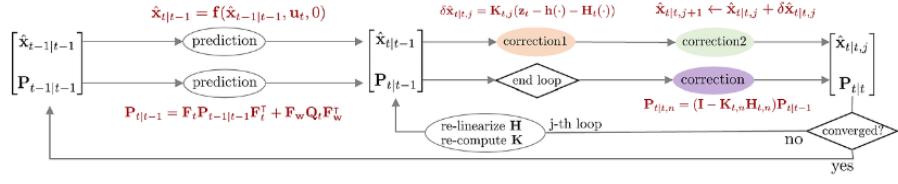
$$\mathbf{G} = \frac{\partial \mathbf{g}}{\partial \delta\mathbf{x}} \Big|_{\delta\mathbf{x}_t=\delta\hat{\mathbf{x}}_{t|t}} \tag{51}$$

4.3 Summary

ESKF를 함수로 표현하면 다음과 같다.

$$\begin{aligned}
 \text{ErrorStateKalmanFilter}(\hat{\mathbf{x}}_{t-1|t-1}, \delta\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{P}_{t-1|t-1}, \mathbf{u}_t, \mathbf{z}_t) \{ \\
 &\quad \text{(Prediction Step)} \\
 &\quad \delta\hat{\mathbf{x}}_{t|t-1} = \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} = 0 \quad \leftarrow \text{Always 0} \\
 &\quad \hat{\mathbf{x}}_{t|t-1} = \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, 0, \mathbf{u}_t, 0) \\
 &\quad \mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top \\
 \\
 &\quad \text{(Correction Step)} \\
 &\quad \mathbf{K}_t = \mathbf{P}_{t|t-1} \mathbf{H}_t^\top (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^\top + \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v)^{-1} \\
 &\quad \delta\hat{\mathbf{x}}_{t|t} = \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0, 0)) \\
 &\quad \hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \delta\hat{\mathbf{x}}_{t|t} \\
 &\quad \mathbf{P}_{t|t} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1} \\
 &\quad \text{reset } \delta\hat{\mathbf{x}}_{t|t} \\
 &\quad \delta\hat{\mathbf{x}}_{t|t} \leftarrow 0 \\
 &\quad \mathbf{P}_{t|t} \leftarrow \mathbf{G} \mathbf{P}_{t|t} \mathbf{G}^\top \\
 &\quad \text{return } \delta\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}
 \}
 \end{aligned} \tag{52}$$

5 Iterated extended kalman filter (IEKF)



Iterated extended kalman filter (IEKF)는 EKF에서 correction 스텝 부분을 반복적으로 수행하는 알고리즘이다. EKF는 비선형 함수를 선형화하여 상태 변수를 추정하기 때문에 선형화 과정에서 필연적으로 오차가 발생할 수밖에 없다. IEKF는 이러한 선형화 오차를 줄이기 위해서 correction 스텝이 종료된 후 업데이트 변화량 $\delta\hat{x}_{t|t,j}$ 이 충분히 크다면 다시 선형화를 수행하여 반복적으로 correction 스텝을 진행한다.

이 때, $\delta\hat{x}$ 는 문맥 상 여러 상태 변수가 아닌 업데이트 변화량으로 해석된다. 즉 $\hat{x}_{t|t,j+1} \leftarrow \hat{x}_{t|t,j} + \delta\hat{x}_{t|t,j}$ 와 같이 j번째 posterior 값에 더해져서 j+1 번째 posterior 값을 업데이트하는 용도로만 사용된다. 다시 말하면, 상태 추정의 대상이 아니다.

5.1 Compare to EKF

5.1.1 Commonality 1

IEKF에서 모션 모델과 관측 모델은 다음과 같다. 이는 EKF와 완전히 동일하다.

$$\begin{aligned} \text{Motion Model: } \mathbf{x}_t &= \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_t, \mathbf{w}_t) \\ \text{Observation Model: } \mathbf{z}_t &= \mathbf{h}(\mathbf{x}_t, \mathbf{v}_t) \end{aligned} \quad (53)$$

- \mathbf{x}_t : 모델의 상태 변수(state variable) - \mathbf{u}_t : 모델의 입력(input) - \mathbf{z}_t : 모델의 관측값(measurement) - $\mathbf{f}(\cdot)$: 비선형 모션(motion) 모델 함수 - $\mathbf{h}(\cdot)$: 비선형 관측(observation) 모델 함수 - $\mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$: 모션 모델의 노이즈. \mathbf{Q}_t 는 \mathbf{w}_t 의 공분산 행렬을 의미 - $\mathbf{v}_t \sim \mathcal{N}(0, \mathbf{R}_t)$: 관측 모델의 노이즈. \mathbf{R}_t 는 \mathbf{v}_t 의 공분산 행렬을 의미

5.1.2 Commonality 2

다음으로 선형화 과정도 EKF의 (24), (25)와 완전히 동일하다.

$$\boxed{\begin{aligned} \mathbf{x}_t &= \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0) - \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} + \mathbf{F}_w \mathbf{w}_t \\ &= \mathbf{F}_t \mathbf{x}_{t-1} + \tilde{\mathbf{u}}_t + \tilde{\mathbf{w}}_t \end{aligned}} \quad (54)$$

- $\tilde{\mathbf{u}}_t = \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0) - \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1}$ - $\tilde{\mathbf{w}}_t = \mathbf{F}_w \mathbf{w}_t \sim \mathcal{N}(0, \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top)$

$$\boxed{\begin{aligned} \mathbf{z}_t &= \mathbf{H}_t \mathbf{x}_t + \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0) - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1} + \mathbf{H}_v \mathbf{v}_t \\ &= \mathbf{H}_t \mathbf{x}_t + \tilde{\mathbf{z}}_t + \tilde{\mathbf{v}}_t \end{aligned}} \quad (55)$$

- $\tilde{\mathbf{z}}_t = \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0) - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1}$ - $\tilde{\mathbf{v}}_t = \mathbf{H}_v \mathbf{v}_t \sim \mathcal{N}(0, \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)$

5.1.3 Commonality 3

IEKF는 EKF와 동일하게 $\mathbf{F}, \mathbf{H}, \mathbf{K}$ 행렬이 true 상태 변수 \mathbf{x}_{true} 에 대한 행렬을 사용한다.

5.1.4 Difference 1

EKF: prediction 값으로부터 한 번에 correction 값이 나온다.

IEKF: correction 값이 다시 prediction 값이 되어 correction 스텝을 반복적으로 진행한다.

$$\boxed{\begin{aligned} \text{EKF Correction : } \hat{\mathbf{x}}_{t|t-1} &\rightarrow \hat{\mathbf{x}}_{t|t} \\ \text{IEKF Correction : } \hat{\mathbf{x}}_{t|t,j} &\leftarrow \hat{\mathbf{x}}_{t|t,j+1} \end{aligned}} \quad (56)$$

- j : j-th iteration

5.1.5 Difference 2

(55) 식을 전개하여 innovation term \mathbf{r}_t 을 만들어보면 다음과 같다.

$$\mathbf{r}_t = \mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0) - \mathbf{H}_t(\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) \quad (57)$$

EKF의 경우 correction 스텝 (31)의 두 번째 줄을 보면 칼만 게인 \mathbf{K}_t 에 \mathbf{r}_t 가 곱해져서 posterior를 계산하는 것을 볼 수 있다.

$$\begin{aligned} \text{EKF correction : } \hat{\mathbf{x}}_{t|t} &= \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t(\mathbf{r}_t) \\ &= \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t(\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0)) \end{aligned} \quad (58)$$

EKF: $\mathbf{H}_t(\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})$ 부분은 $\mathbf{x}_t = \hat{\mathbf{x}}_{t|t-1}$ 이 대입되어 소거되고 나머지 부분만 사용된다.

IEKF: 매 순간 correction 스텝을 반복하면서 새로운 상태(new operating point)에 대한 선형화를 수행하기 때문에 해당 부분이 소거되지 않는다.

EKF innovation term : $\mathbf{r}_t = \mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t t-1}, 0)$		(59)
IEKF innovation term : $\mathbf{r}_{t,j} = \mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t t,j}, 0) - \mathbf{H}_{t,j}(\hat{\mathbf{x}}_{t t-1} - \hat{\mathbf{x}}_{t t,j})$		

만약 첫 번째 iteration $j = 0$ 의 경우 $\hat{\mathbf{x}}_{t|t,0} = \hat{\mathbf{x}}_{t|t-1}$ 이므로 소거되어 EKF와 동일한 식이 되지만 $j = 1$ 이상부터는 다른 값이 되므로 소거되지 않는다.

5.2 Prediction step

Prediction은 $\overline{\text{bel}}(\mathbf{x}_t)$ 를 구하는 과정을 말한다. 공분산 행렬을 구할 때 선형화된 자코비안 행렬 \mathbf{F}_t 가 사용된다. 이는 EKF의 prediction 스텝과 완전히 동일하다.

$\hat{\mathbf{x}}_{t t-1} = \mathbf{f}(\hat{\mathbf{x}}_{t-1 t-1}, \mathbf{u}_t, 0)$		(60)
$\mathbf{P}_{t t-1} = \mathbf{F}_t \mathbf{P}_{t-1 t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top$		

5.3 Correction step

Correction은 $\text{bel}(\mathbf{x}_t)$ 를 구하는 과정을 말한다. 칼만 게인과 공분산 행렬을 구할 때 선형화된 자코비안 행렬 \mathbf{H}_t 가 사용된다. IEKF의 correction 스텝은 업데이트 변화량 $\delta\hat{\mathbf{x}}_{t|t,j}$ 가 충분히 작아질 때까지 반복적(iterative)으로 수행한다.

set ϵ start j-th loop $\mathbf{K}_{t,j} = \mathbf{P}_{t t-1} \mathbf{H}_{t,j}^\top (\mathbf{H}_{t,j} \mathbf{P}_{t t-1} \mathbf{H}_{t,j}^\top + \mathbf{H}_{v,j} \mathbf{R}_t \mathbf{H}_{v,j}^\top)^{-1}$ $\delta\hat{\mathbf{x}}_{t t,j} = \mathbf{K}_{t,j} (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t t,j}, 0) - \mathbf{H}_t(\hat{\mathbf{x}}_{t t-1} - \hat{\mathbf{x}}_{t t,j}))$ $\hat{\mathbf{x}}_{t t,j+1} = \hat{\mathbf{x}}_{t t,j} + \delta\hat{\mathbf{x}}_{t t,j}$ iterate until $\delta\hat{\mathbf{x}}_{t t,j} < \epsilon$. end loop $\mathbf{P}_{t t,n} = (\mathbf{I} - \mathbf{K}_{t,n} \mathbf{H}_{t,n}) \mathbf{P}_{t t-1}$		(61)
--	--	------

5.4 Summary

IEKF를 함수로 표현하면 다음과 같다.

```

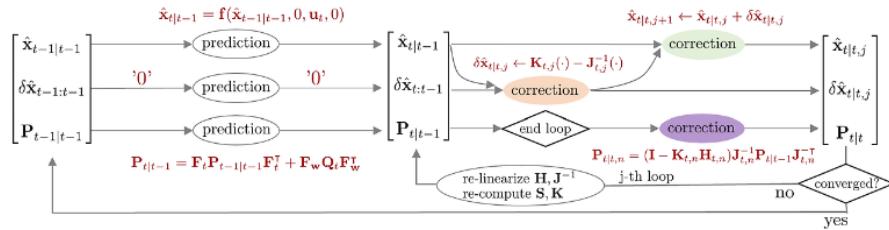
IteratedExtendedKalmanFilter( $\hat{\mathbf{x}}_{t-1|t-1}$ ,  $\mathbf{P}_{t-1|t-1}$ ,  $\mathbf{u}_t$ ,  $\mathbf{z}_t$ ) {
    (Prediction Step)
     $\hat{\mathbf{x}}_{t|t-1} = \mathbf{f}_t(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0)$ 
     $\mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top$ 

    (Correction Step)
    set  $\epsilon$ 
    start j-th loop
         $\mathbf{K}_{t,j} = \mathbf{P}_{t|t-1} \mathbf{H}_{t,j}^\top (\mathbf{H}_{t,j} \mathbf{P}_{t|t-1} \mathbf{H}_{t,j} + \mathbf{H}_{v,j} \mathbf{R}_t \mathbf{H}_{v,j}^\top)^{-1}$ 
         $\delta\hat{\mathbf{x}}_{t|t,j} = \mathbf{K}_{t,j} (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}, 0) - \mathbf{H}_t(\hat{\mathbf{x}}_{t|t-1} - \hat{\mathbf{x}}_{t|t,j}))$ 
         $\hat{\mathbf{x}}_{t|t,j+1} = \hat{\mathbf{x}}_{t|t,j} + \delta\hat{\mathbf{x}}_{t|t,j}$ 
        iterate until  $\delta\hat{\mathbf{x}}_{t|t,j} < \epsilon$ .
    end loop
     $\mathbf{P}_{t|t,n} = (\mathbf{I} - \mathbf{K}_{t,n} \mathbf{H}_{t,n}) \mathbf{P}_{t|t-1}$ 
    return  $\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}$ 
}

```

(62)

6 Iterated error-state kalman filter (IESKF)



IEKF가 correction 스텝에서 true 상태 변수 \mathbf{x}_{true} 를 반복적으로 추정한다면 IESKF는 에러 상태 변수 $\delta\mathbf{x}_t$ 를 반복적으로 추정하는 알고리즘이다. 두 상태 변수 사이의 관계는 다음과 같다.

$$\mathbf{x}_{true,t} = \mathbf{x}_t + \delta\mathbf{x}_t \quad (63)$$

- $\mathbf{x}_{true,t}$: 기존 KF, EKF에서 업데이트된 t 스텝의 true 상태변수
- \mathbf{x}_t : t 스텝의 명목(nominal) 상태 변수. 에러가 없는 상태를 의미한다
- $\delta\mathbf{x}_t$: t 스텝의 에러(error) 상태 변수

반복적(iterative)으로 값이 업데이트 되는 IESKF 과정에서 $\mathbf{x}_{true,t}$ 는 업데이트 된 이후 상태, nominal \mathbf{x}_t 는 업데이트 되기 전 상태, 에러 상태 변수 $\delta\mathbf{x}_t$ 는 업데이트 변화량으로 해석하면 된다.

6.1 Compare to ESKF

6.1.1 Commonality 1

IESKF의 모션 모델과 관측 모델은 다음과 같다. 이는 ESKF와 완전히 동일하다.

$$\begin{aligned}
 \text{Error-state Motion Model: } \mathbf{x}_t + \delta\mathbf{x}_t &= \mathbf{f}(\mathbf{x}_{t-1}, \delta\mathbf{x}_{t-1}, \mathbf{u}_t, \mathbf{w}_t) \\
 \text{Error-state Observation Model: } \mathbf{z}_t &= \mathbf{h}(\mathbf{x}_t, \delta\mathbf{x}_t, \mathbf{v}_t)
 \end{aligned} \quad (64)$$

- \mathbf{x}_t : 모델의 nominal 상태 변수
- $\delta\mathbf{x}_t$: 모델의 에러 상태 변수
- \mathbf{u}_t : 모델의 입력(input)
- \mathbf{z}_t : 모델의 관측값(measurement)

- $\mathbf{f}(\cdot)$: 비선형 모션(motion) 모델 함수
- $\mathbf{h}(\cdot)$: 비선형 관측(observation) 모델 함수
- $\mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$: 에러 상태 모델의 노이즈. \mathbf{Q}_t 는 \mathbf{w}_t 의 공분산 행렬을 의미
- $\mathbf{v}_t \sim \mathcal{N}(0, \mathbf{R}_t)$: 관측 모델의 노이즈. \mathbf{R}_t 는 \mathbf{v}_t 의 공분산 행렬을 의미

6.1.2 Commonality 2

IESKF의 두 자코비안 $\mathbf{F}_t, \mathbf{H}_t$ 모두 true 상태 $\mathbf{x}_{\text{true},t}$ 가 아닌 ESKF와 동일하게 에러 상태 $\delta\mathbf{x}_t$ 에 대한 자코비안을 의미한다.

$$\boxed{\mathbf{F}_t = \frac{\partial \mathbf{f}}{\partial \delta\mathbf{x}_{t-1}} \Big|_{\delta\mathbf{x}_{t-1}=\delta\hat{\mathbf{x}}_{t-1|t-1}} \quad \mathbf{F}_w = \frac{\partial \mathbf{f}}{\partial \mathbf{w}_t} \Big|_{\substack{\delta\mathbf{x}_{t-1}=\delta\hat{\mathbf{x}}_{t-1|t-1} \\ \mathbf{w}_t=0}}} \quad (65)$$

$$\boxed{\mathbf{H}_t = \frac{\partial \mathbf{h}}{\partial \delta\mathbf{x}_t} \Big|_{\delta\mathbf{x}_t=\delta\hat{\mathbf{x}}_{t|t-1}} \quad \mathbf{H}_v = \frac{\partial \mathbf{h}}{\partial \mathbf{v}_t} \Big|_{\substack{\delta\mathbf{x}_t=\delta\hat{\mathbf{x}}_{t|t-1} \\ \mathbf{v}_t=0}}} \quad (66)$$

\mathbf{H}_t 는 다음과 같이 연쇄법칙을 통해 표현할 수 있다.

$$\mathbf{H}_t = \frac{\partial \mathbf{h}}{\partial \delta\mathbf{x}_t} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_{\text{true},t}} \frac{\partial \mathbf{x}_{\text{true},t}}{\partial \delta\mathbf{x}_t} \quad (67)$$

6.1.3 Commonality 3

선형화된 에러 상태 변수 $\hat{\mathbf{x}}_t$ 또한 ESKF와 동일하게 구할 수 있다.

$$\boxed{\begin{aligned} \delta\mathbf{x}_t &= \mathbf{F}_t \delta\mathbf{x}_{t-1} + \mathbf{F}_w \mathbf{w}_t \\ &= \mathbf{F}_t \delta\mathbf{x}_{t-1} + \tilde{\mathbf{w}}_t \\ &= 0 + \tilde{\mathbf{w}}_t \end{aligned}} \quad (68)$$

- $\tilde{\mathbf{w}}_t = \mathbf{F}_w \mathbf{w}_t \sim \mathcal{N}(0, \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top)$

6.1.4 Difference (MAP-based derivation)

ESKF derivation:

ESKF의 correction 스텝은 아래 수식을 전개하여 평균 $\delta\hat{\mathbf{x}}_{t|t}$ 와 공분산 $\mathbf{P}_{t|t}$ 를 유도한다.

$$\text{bel}(\delta\mathbf{x}_t) = \eta \cdot p(\mathbf{z}_t \mid \delta\mathbf{x}_t) \overline{\text{bel}}(\delta\mathbf{x}_t) \sim \mathcal{N}(\delta\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}) \quad (69)$$

- $p(\mathbf{z}_t \mid \delta\mathbf{x}_t) \sim \mathcal{N}(\tilde{\mathbf{z}}_t, \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)$: (??) 참조
- $\overline{\text{bel}}(\delta\mathbf{x}_t) \sim \mathcal{N}(\delta\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1})$

$$\boxed{\begin{aligned} \delta\hat{\mathbf{x}}_{t|t} &= \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0, 0)) \\ \mathbf{P}_{t|t} &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1} \end{aligned}} \quad (70)$$

IESKF derivation:

IESKF는 maximum a posteriori(MAP)을 사용하여 correction 스텝을 유도한다. 이 때, MAP 추정을 위해 Gauss-Newton 최적화 방식이 사용된다. 보다 자세한 내용은 섹션 7을 참조하면 된다. 해당 유도 과정은 [6],[7],[8],[9],[10],[11] 을 참고하여 작성하였다. 이 중 [6]이 가장 자세하게 IESKF 유도 과정에 대해 설명하고 있다.

EKF를 MAP 방식으로 유도하면 최종적으로 아래와 같은 식을 최적화해야 한다.

$$\arg \min_{\mathbf{x}_t} \|\mathbf{z}_t - \mathbf{h}(\mathbf{x}_{\text{true},t}, 0)\|_{\mathbf{H}_v \mathbf{R}^{-1} \mathbf{H}_v^\top} + \|\mathbf{x}_{\text{true},t} - \hat{\mathbf{x}}_{t|t-1}\|_{\mathbf{P}_{t|t-1}^{-1}} \quad (71)$$

위 식을 nominal 상태 \mathbf{x}_t 와 에러 상태 $\delta\mathbf{x}_t$ 에 대한 식으로 표현하면 다음과 같다.

$$\arg \min_{\delta\hat{\mathbf{x}}_{t|t}} \|\mathbf{z}_t - \mathbf{h}(\mathbf{x}_t, \delta\mathbf{x}_t, 0)\|_{\mathbf{H}_v \mathbf{R}^{-1} \mathbf{H}_v^\top} + \|\delta\hat{\mathbf{x}}_{t|t-1}\|_{\mathbf{P}_{t|t-1}^{-1}} \quad (72)$$

Tip

Prediction 에러 상태 변수 $\delta\hat{\mathbf{x}}_{t|t-1}$ 은 다음과 같이 정의한다.

$$\begin{aligned}\delta\hat{\mathbf{x}}_{t|t-1} &= \mathbf{x}_{\text{true},t} - \hat{\mathbf{x}}_{t|t-1} \\ &\sim \mathcal{N}(0, \mathbf{P}_{t|t-1})\end{aligned}\quad (73)$$

Posterior(=correction) 에러 상태 변수 $\delta\hat{\mathbf{x}}_{t|t}$ 는 다음과 같이 정의한다.

$$\delta\hat{\mathbf{x}}_{t|t} = \mathbf{x}_{\text{true},t|t} - \hat{\mathbf{x}}_{t|t} \quad (74)$$

위 식을 이항하면 아래 공식이 성립한다.

$$\mathbf{x}_{\text{true},t|t} = \hat{\mathbf{x}}_{t|t} + \delta\hat{\mathbf{x}}_{t|t} \quad (75)$$

(72)의 앞 부분은 (42) 선형화와 동일하게 진행하면 된다.

$$\mathbf{z}_t - \mathbf{h}(\mathbf{x}_{t|t-1}, \delta\hat{\mathbf{x}}_{t|t-1}, 0) \approx \mathbf{z}_t - \mathbf{h}(\mathbf{x}_{t|t-1}, 0, 0) - \mathbf{H}_t \delta\mathbf{x}_t \quad (76)$$

위 식에서 $\delta\mathbf{x}_t$ 는 ESKF에서는 항상 0이므로 제거하였으나 IESKF에서는 0이 아닌 값을 가지므로 제거하지 않는다. 다음으로 (72)의 뒷 부분에 (75)를 대입하면 아래와 같이 전개된다.

$$\begin{aligned}\delta\hat{\mathbf{x}}_{t|t-1} &= \mathbf{x}_{\text{true},t} - \hat{\mathbf{x}}_{t|t-1} \\ &= (\hat{\mathbf{x}}_{t|t} + \delta\hat{\mathbf{x}}_{t|t}) - \hat{\mathbf{x}}_{t|t-1} \\ &\approx \hat{\mathbf{x}}_{t|t} - \hat{\mathbf{x}}_{t|t-1} + \mathbf{J}_t \delta\hat{\mathbf{x}}_{t|t} \\ &\sim \mathcal{N}(0, \mathbf{P}_{t|t-1})\end{aligned}\quad (77)$$

이 때, \mathbf{J}_t 는 다음과 같이 정의된다.

$$\mathbf{J}_t = \frac{\partial}{\partial \delta\hat{\mathbf{x}}_{t|t}} \left((\hat{\mathbf{x}}_{t|t} + \delta\hat{\mathbf{x}}_{t|t}) - \hat{\mathbf{x}}_{t|t-1} \right) \Big|_{\delta\hat{\mathbf{x}}_{t|t}=0} \quad (78)$$

(77)의 세 번째 줄과 네 번째 줄을 이항한 후 정리하면 아래와 같은 공식을 얻는다.

$$\delta\hat{\mathbf{x}}_{t|t} \sim \mathcal{N}(-\mathbf{J}_t^{-1}(\hat{\mathbf{x}}_{t|t} - \hat{\mathbf{x}}_{t|t-1}), \mathbf{J}_t^{-1} \mathbf{P}_{t|t-1} \mathbf{J}_t^{-\top}) \quad (79)$$

지금까지 계산한 (76), (77)를 (72)에 대입하면 아래와 같은 MAP 추정 문제가 된다.

$$\begin{aligned}\arg \min_{\delta\hat{\mathbf{x}}_{t|t}} \quad & \| \mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0, 0) - \mathbf{H}_t \delta\hat{\mathbf{x}}_{t|t} \|_{\mathbf{H}_v \mathbf{R}^{-1} \mathbf{H}_v^\top} \\ & + \| \hat{\mathbf{x}}_{t|t} - \hat{\mathbf{x}}_{t|t-1} + \mathbf{J}_t \delta\hat{\mathbf{x}}_{t|t} \|_{\mathbf{P}_{t|t-1}^{-1}}\end{aligned}\quad (80)$$

IESKF는 이를 반복적으로(iterative)하게 추정함으로써 예상 상태 변수 $\delta\hat{\mathbf{x}}_{t|t,j}$ 가 특정 값 ϵ 이하로 수렴할 때까지 반복한다. j 번째 iteration에 대한 표현은 다음과 같다.

$$\boxed{\delta\hat{\mathbf{x}}_{t|t,j} \sim \mathcal{N}(-\mathbf{J}_{t,j}^{-1}(\hat{\mathbf{x}}_{t|t,j} - \hat{\mathbf{x}}_{t|t-1}), \mathbf{J}_{t,j}^{-1} \mathbf{P}_{t|t-1} \mathbf{J}_{t,j}^{-\top})} \quad (81)$$

$$\begin{aligned}\arg \min_{\delta\hat{\mathbf{x}}_{t|t,j}} \quad & \| \mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}, 0, 0) - \mathbf{H}_t \delta\hat{\mathbf{x}}_{t|t,j} \|_{\mathbf{H}_v \mathbf{R}^{-1} \mathbf{H}_v^\top} \\ & + \| \hat{\mathbf{x}}_{t|t,j} - \hat{\mathbf{x}}_{t|t-1} + \mathbf{J}_{t,j} \delta\hat{\mathbf{x}}_{t|t,j} \|_{\mathbf{P}_{t|t-1}^{-1}}\end{aligned}\quad (82)$$

6.2 Prediction step

Prediction은 $\overline{\text{bel}}(\delta \mathbf{x}_t)$ 를 구하는 과정을 말한다. 공분산 행렬을 구할 때 선형화된 자코비안 행렬 \mathbf{F}_t 가 사용된다. 이는 ESKF와 완전히 동일하다.

$$\boxed{\begin{aligned}\delta \hat{\mathbf{x}}_{t|t-1} &= \mathbf{F}_t \delta \hat{\mathbf{x}}_{t-1|t-1} = 0 && \leftarrow \text{Always 0} \\ \hat{\mathbf{x}}_{t|t-1} &= \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, 0, \mathbf{u}_t, 0) \\ \mathbf{P}_{t|t-1} &= \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top\end{aligned}} \quad (83)$$

6.3 Correction step

Correction은 $\text{bel}(\mathbf{x}_t)$ 를 구하는 과정을 말한다. 칼만 게인과 공분산 행렬을 구할 때 선형화된 자코비안 행렬 \mathbf{H}_t 가 사용된다. IESKF의 correction 스텝은 업데이트 변화량 $\delta \hat{\mathbf{x}}_{t|t,j}$ 가 충분히 작아질 때까지 반복적 (iterative)으로 수행한다. 앞서 유도한 (82)을 미분한 후 0으로 설정하여 풀면 아래와 같은 공식을 얻는다.

$$\boxed{\begin{aligned}&\text{set } \epsilon \\ &\text{start j-th loop} \\ &\quad \mathbf{S}_{t,j} = \mathbf{H}_{t,j} \mathbf{J}_{t,j}^{-1} \mathbf{P}_{t|t-1} \mathbf{J}_{t,j}^{-\top} \mathbf{H}_{t,j}^\top + \mathbf{H}_{v,j} \mathbf{R}_t \mathbf{H}_{v,j}^\top \\ &\quad \mathbf{K}_{t,j} = \mathbf{J}_{t,j}^{-1} \mathbf{P}_{t|t-1} \mathbf{J}_{t,j}^{-\top} \mathbf{H}_{t,j}^\top \mathbf{S}_{t,j}^{-1} \\ &\quad \delta \hat{\mathbf{x}}_{t|t,j} = \mathbf{K}_{t,j} \left(\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}, 0, 0) + \mathbf{H}_{t,j} \mathbf{J}_{t,j}^{-1} (\hat{\mathbf{x}}_{t|t,j} - \hat{\mathbf{x}}_{t|t-1}) \right) - \mathbf{J}_{t,j}^{-1} (\hat{\mathbf{x}}_{t|t,j} - \hat{\mathbf{x}}_{t|t-1}) \\ &\quad \hat{\mathbf{x}}_{t|t,j+1} = \hat{\mathbf{x}}_{t|t,j} \boxplus \delta \hat{\mathbf{x}}_{t|t,j} \\ &\quad \text{iterate until } \delta \hat{\mathbf{x}}_{t|t,j} < \epsilon. \\ &\text{end loop} \\ &\mathbf{P}_{t|t,n} = (\mathbf{I} - \mathbf{K}_{t,n} \mathbf{H}_{t,n}) \mathbf{J}_{t,n}^{-1} \mathbf{P}_{t|t-1} \mathbf{J}_{t,n}^{-\top}\end{aligned}} \quad (84)$$

6.4 Summary

IESKF를 함수로 표현하면 다음과 같다.

$$\boxed{\begin{aligned}&\text{IteratedErrorStateKalmanFilter}(\hat{\mathbf{x}}_{t-1|t-1}, \delta \hat{\mathbf{x}}_{t-1|t-1}, \mathbf{P}_{t-1|t-1}, \mathbf{u}_t, \mathbf{z}_t) \{ \\ &\quad \text{(Prediction Step)} \\ &\quad \delta \hat{\mathbf{x}}_{t|t-1} = \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} = 0 && \leftarrow \text{Always 0} \\ &\quad \hat{\mathbf{x}}_{t|t-1} = \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, 0, \mathbf{u}_t, 0) \\ &\quad \mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top \\ \\ &\quad \text{(Correction Step)} \\ &\quad \text{set } \epsilon \\ &\quad \text{start j-th loop} \\ &\quad \quad \mathbf{S}_{t,j} = \mathbf{H}_{t,j} \mathbf{J}_{t,j}^{-1} \mathbf{P}_{t|t-1} \mathbf{J}_{t,j}^{-\top} \mathbf{H}_{t,j}^\top + \mathbf{H}_{v,j} \mathbf{R}_t \mathbf{H}_{v,j}^\top \\ &\quad \quad \mathbf{K}_{t,j} = \mathbf{J}_{t,j}^{-1} \mathbf{P}_{t|t-1} \mathbf{J}_{t,j}^{-\top} \mathbf{H}_{t,j}^\top \mathbf{S}_{t,j}^{-1} \\ &\quad \quad \delta \hat{\mathbf{x}}_{t|t,j} = \mathbf{K}_{t,j} \left(\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}, 0, 0) + \mathbf{H}_{t,j} \mathbf{J}_{t,j}^{-1} (\hat{\mathbf{x}}_{t|t,j} - \hat{\mathbf{x}}_{t|t-1}) \right) - \mathbf{J}_{t,j}^{-1} (\hat{\mathbf{x}}_{t|t,j} - \hat{\mathbf{x}}_{t|t-1}) \\ &\quad \quad \hat{\mathbf{x}}_{t|t,j+1} = \hat{\mathbf{x}}_{t|t,j} \boxplus \delta \hat{\mathbf{x}}_{t|t,j} \\ &\quad \quad \text{iterate until } \delta \hat{\mathbf{x}}_{t|t,j} < \epsilon. \\ &\quad \text{end loop} \\ &\quad \mathbf{P}_{t|t,n} = (\mathbf{I} - \mathbf{K}_{t,n} \mathbf{H}_{t,n}) \mathbf{J}_{t,n}^{-1} \mathbf{P}_{t|t-1} \mathbf{J}_{t,n}^{-\top} \\ &\quad \text{return } \hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}\end{aligned}} \quad (85)$$

7 MAP, GN, and EKF relationship

7.1 Traditional EKF derivation

EKF의 관측 모델 함수가 다음과 같이 주어졌다고 하자. 전개의 편의를 위해 관측 노이즈 \mathbf{v}_t 를 밖으로 위치하였다.

$$\text{Observation Model: } \mathbf{z}_t = \mathbf{h}(\mathbf{x}_t) + \mathbf{v}_t \quad (86)$$

EKF의 correction 스텝은 아래 수식을 전개하여 평균 $\hat{\mathbf{x}}_{t|t}$ 와 공분산 $\mathbf{P}_{t|t}$ 를 유도한다.

$$\text{bel}(\mathbf{x}_t) = \eta \cdot p(\mathbf{z}_t | \mathbf{x}_t) \overline{\text{bel}}(\mathbf{x}_t) \sim \mathcal{N}(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}) \quad (87)$$

$$\begin{aligned} - p(\mathbf{z}_t | \mathbf{x}_t) &\sim \mathcal{N}(\mathbf{h}_t(\mathbf{x}_t), \mathbf{R}_t) \\ - \overline{\text{bel}}(\mathbf{x}_t) &\sim \mathcal{N}(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}) \end{aligned}$$

$$\boxed{\begin{aligned} \hat{\mathbf{x}}_{t|t} &= \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1})) \\ \mathbf{P}_{t|t} &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1} \end{aligned}} \quad (88)$$

7.2 MAP-based EKF derivation

7.2.1 Start from MAP estimator

Correction 스텝을 유도하는 방법으로 posterior의 확률을 최대화하는 maximum a posteriori(MAP) 추정을 사용할 수 있다. 자세한 내용은 [12]를 참고하여 작성하였다.

$$\begin{aligned} \hat{\mathbf{x}}_{t|t} &= \arg \max_{\mathbf{x}_t} \text{bel}(\mathbf{x}_t) \\ &= \arg \max_{\mathbf{x}_t} p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \quad \cdots \text{posterior} \\ &\propto \arg \max_{\mathbf{x}_t} p(\mathbf{z}_t | \mathbf{x}_t) \overline{\text{bel}}(\mathbf{x}_t) \quad \cdots \text{likelihood} \cdot \text{prior} \\ &\propto \arg \max_{\mathbf{x}_t} \exp \left(-\frac{1}{2} \left[(\mathbf{z}_t - \mathbf{h}(\mathbf{x}_t))^T \mathbf{R}_t^{-1} (\mathbf{z}_t - \mathbf{h}(\mathbf{x}_t)) \right. \right. \\ &\quad \left. \left. + (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T \mathbf{P}_{t|t-1}^{-1} (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) \right] \right) \end{aligned} \quad (89)$$

$$\begin{aligned} - p(\mathbf{z}_t | \mathbf{x}_t) &\sim \mathcal{N}(\mathbf{h}_t(\mathbf{x}_t), \mathbf{R}_t) \\ - \text{bel}(\mathbf{x}_t) &\sim \mathcal{N}(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}) \end{aligned}$$

마이너스 부호를 제거하면 최대화(maximization) 문제가 최소화(minimization) 문제로 변하고 다음과 같은 최적화 식으로 정리할 수 있다.

$$\begin{aligned} \hat{\mathbf{x}}_{t|t} &\propto \arg \min_{\mathbf{x}_t} \exp \left((\mathbf{z}_t - \mathbf{h}(\mathbf{x}_t))^T \mathbf{R}_t^{-1} (\mathbf{z}_t - \mathbf{h}(\mathbf{x}_t)) \right. \\ &\quad \left. + (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T \mathbf{P}_{t|t-1}^{-1} (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) \right) \end{aligned} \quad (90)$$

$$\boxed{\hat{\mathbf{x}}_{t|t} = \arg \min_{\mathbf{x}_t} \| \mathbf{z}_t - \mathbf{h}(\mathbf{x}_t) \|_{\mathbf{R}_t^{-1}} + \| \mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1} \|_{\mathbf{P}_{t|t-1}^{-1}}} \quad (91)$$

- $\| \mathbf{a} \|_{\mathbf{B}} = \mathbf{a}^T \mathbf{B} \mathbf{a}$: Mahalanobis norm

(91) 내부의 식을 전개하고 cost function \mathbf{C}_t 라고 정의하면 다음과 같다. 이 때, $\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}$ 의 순서를 바꿔도 전체 값에는 영향을 주지 않는다.

$$\mathbf{C}_t = (\mathbf{z}_t - \mathbf{h}(\mathbf{x}_t))^T \mathbf{R}_t^{-1} (\mathbf{z}_t - \mathbf{h}(\mathbf{x}_t)) + (\hat{\mathbf{x}}_{t|t-1} - \mathbf{x}_t)^T \mathbf{P}_{t|t-1}^{-1} (\hat{\mathbf{x}}_{t|t-1} - \mathbf{x}_t) \quad (92)$$

이를 행렬 형태로 표현하면 다음과 같다.

$$\mathbf{C}_t = \begin{bmatrix} \hat{\mathbf{x}}_{t|t-1} - \mathbf{x}_t \\ \mathbf{z}_t - \mathbf{h}(\mathbf{x}_t) \end{bmatrix}^T \begin{bmatrix} \mathbf{P}_{t|t-1}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_t^{-1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_{t|t-1} - \mathbf{x}_t \\ \mathbf{z}_t - \mathbf{h}(\mathbf{x}_t) \end{bmatrix} \quad (93)$$

7.2.2 MLE of new observation function

위 식을 만족하는 새로운 관측 함수를 다음과 같이 정의할 수 있다.

$$\boxed{\begin{aligned} \mathbf{y}_t &= \mathbf{g}(\mathbf{x}_t) + \mathbf{e}_t \\ &\sim \mathcal{N}(\mathbf{g}(\mathbf{x}_t), \mathbf{P}_e) \end{aligned}} \quad (94)$$

- $\mathbf{y}_t = \begin{bmatrix} \hat{\mathbf{x}}_{t|t-1} \\ \mathbf{z}_t \end{bmatrix}$
- $\mathbf{g}(\mathbf{x}_t) = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{h}(\mathbf{x}_t) \end{bmatrix}$
- $\mathbf{e}_t \sim \mathcal{N}(0, \mathbf{P}_e)$
- $\mathbf{P}_e = \begin{bmatrix} \mathbf{P}_{t|t-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_t \end{bmatrix}$

비선형 함수 $\mathbf{g}(\mathbf{x}_t)$ 는 다음과 같이 선형화할 수 있다.

$$\begin{aligned} \mathbf{g}(\mathbf{x}_t) &\approx \mathbf{g}(\hat{\mathbf{x}}_{t|t-1}) + \mathbf{J}_t(\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) \\ &= \mathbf{g}(\hat{\mathbf{x}}_{t|t-1}) + \mathbf{J}_t \delta \hat{\mathbf{x}}_{t|t-1} \end{aligned} \quad (95)$$

자코비안 \mathbf{J}_t 는 다음과 같다.

$$\begin{aligned} \mathbf{J}_t &= \frac{\partial \mathbf{g}}{\partial \mathbf{x}_t} \Big|_{\mathbf{x}_t=\hat{\mathbf{x}}_{t|t-1}} \\ &= \frac{\partial \begin{bmatrix} \mathbf{x}_t \\ \mathbf{h}(\mathbf{x}_t) \end{bmatrix}}{\partial \mathbf{x}_t} \Big|_{\mathbf{x}_t=\hat{\mathbf{x}}_{t|t-1}} \\ &= \frac{\partial \begin{bmatrix} \mathbf{x}_t \\ \mathbf{h}(\hat{\mathbf{x}}_{t|t-1}) + \mathbf{H}_t(\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) \end{bmatrix}}{\partial \mathbf{x}_t} \Big|_{\mathbf{x}_t=\hat{\mathbf{x}}_{t|t-1}} \\ &= \begin{bmatrix} \mathbf{I} \\ \mathbf{H}_t \end{bmatrix} \end{aligned} \quad (96)$$

따라서 다음과 같이 \mathbf{y}_t 에 대한 likelihood를 전개할 수 있다.

$$\begin{aligned} p(\mathbf{y}_t | \mathbf{x}_t) &\sim \mathcal{N}(\mathbf{g}(\mathbf{x}_t), \mathbf{P}_e) \\ &= \eta \cdot \exp \left(-\frac{1}{2} (\mathbf{y}_t - \mathbf{g}(\mathbf{x}_t))^T \mathbf{P}_e^{-1} (\mathbf{y}_t - \mathbf{g}(\mathbf{x}_t)) \right) \end{aligned} \quad (97)$$

즉, 기존의 $\text{bel}(\mathbf{x}_t)$ 에 대한 maximum a posteriori(MAP) 문제는 $p(\mathbf{y}_t | \mathbf{x}_t)$ 에 대한 maximum likelihood estimation(MLE) 문제를 푸는 것으로 귀결된다. (97) 식을 MLE로 풀면 다음과 같다.

$$\begin{aligned} \hat{\mathbf{x}}_{t|t} &= \arg \max_{\mathbf{x}_t} p(\mathbf{y}_t | \mathbf{x}_t) \\ &\propto \arg \min_{\mathbf{x}_t} -\ln p(\mathbf{y}_t | \mathbf{x}_t) \\ &\propto \arg \min_{\mathbf{x}_t} \frac{1}{2} (\mathbf{y}_t - \mathbf{g}(\mathbf{x}_t))^T \mathbf{P}_e^{-1} (\mathbf{y}_t - \mathbf{g}(\mathbf{x}_t)) \\ &\propto \arg \min_{\mathbf{x}_t} \|\mathbf{y}_t - \mathbf{g}(\mathbf{x}_t)\|_{\mathbf{P}_e^{-1}} \end{aligned} \quad (98)$$

7.2.3 Gauss-Newton Optimization

(98) 식은 최소제곱법의 형태를 지닌다. 특히 가중치 \mathbf{P}_e^{-1} 가 중간에 곱해지므로 weighted least squares(WLS)라고도 부른다. 식을 선형화한 후 다시 정리하면 아래와 같다.

$$\begin{aligned}
\hat{\mathbf{x}}_{t|t} &= \arg \min_{\mathbf{x}_t} \|\mathbf{y}_t - \mathbf{g}(\mathbf{x}_t)\|_{\mathbf{P}_{\mathbf{e}}^{-1}} \\
&= \arg \min_{\mathbf{x}_t} \|\mathbf{y}_t - \mathbf{g}(\hat{\mathbf{x}}_{t|t-1}) - \mathbf{J}_t \delta \hat{\mathbf{x}}_{t|t-1}\|_{\mathbf{P}_{\mathbf{e}}^{-1}} \\
&= \arg \min_{\mathbf{x}_t} \|\mathbf{J}_t \delta \hat{\mathbf{x}}_{t|t-1} - (\mathbf{y}_t - \mathbf{g}(\hat{\mathbf{x}}_{t|t-1}))\|_{\mathbf{P}_{\mathbf{e}}^{-1}} \\
&= \arg \min_{\mathbf{x}_t} \|\mathbf{J}_t \delta \hat{\mathbf{x}}_{t|t-1} - \mathbf{r}_t\|_{\mathbf{P}_{\mathbf{e}}^{-1}}
\end{aligned} \tag{99}$$

- $\delta \hat{\mathbf{x}}_{t|t-1} = \mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}$: 표현의 편의를 위해 \mathbf{x}_t 를 true 상태로 표현

선형화된 residual term \mathbf{r}_t 는 다음과 같다.

$$\begin{aligned}
\mathbf{r}_t &= \mathbf{y}_t - \mathbf{g}(\hat{\mathbf{x}}_{t|t-1}) \\
&= \mathbf{J}_t \delta \hat{\mathbf{x}}_{t|t-1} + \mathbf{e} \\
&\sim \mathcal{N}(0, \mathbf{P}_{\mathbf{e}})
\end{aligned} \tag{100}$$

GN의 정규방정식으로 통해 해를 구하면 다음과 같다.

$$(\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t) \delta \hat{\mathbf{x}}_{t|t-1} = \mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{r}_t \tag{101}$$

$$\therefore \delta \hat{\mathbf{x}}_{t|t-1} = (\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t)^{-1} \mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{r}_t \tag{102}$$

위 식에서 $(\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t)$ 부분을 일반적으로 근사 해시안(approximate hessian) 행렬 $\tilde{\mathbf{H}}$ 이라고 부른다.

Posterior covariance matrix $\mathbf{P}_{t|t}$:

$\mathbf{P}_{t|t}$ 는 다음과 같이 구할 수 있다.

$$\begin{aligned}
\mathbf{P}_{t|t} &= \mathbb{E}[(\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})(\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T] \\
&= \mathbb{E}(\delta \hat{\mathbf{x}}_{t|t-1} \delta \hat{\mathbf{x}}_{t|t-1}^T) \\
&= \mathbb{E}\left[(\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t)^{-1} \mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{r}_t \mathbf{r}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t (\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t)^{-1}\right] \\
&= (\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t)^{-1} \mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbb{E}(\mathbf{r}_t \mathbf{r}_t^T) \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t (\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t)^{-1} \quad \leftarrow \mathbb{E}(\mathbf{r}_t \mathbf{r}_t^T) = \mathbf{P}_{\mathbf{e}} \\
&= (\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t)^{-1} \\
&= \left(\begin{bmatrix} \mathbf{I} & \mathbf{H}_t^T \end{bmatrix} \begin{bmatrix} \mathbf{P}_{t|t-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_t \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{I} \\ \mathbf{H}_t \end{bmatrix} \right)^{-1} \\
&= \left(\mathbf{P}_{t|t-1}^{-1} + \mathbf{H}_t^T \mathbf{R}_t^{-1} \mathbf{H}_t \right)^{-1} \\
&= \mathbf{P}_{t|t-1} - \mathbf{P}_{t|t-1} \mathbf{H}_t^T (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^T + \mathbf{R}_t)^{-1} \mathbf{H}_t \mathbf{P}_{t|t-1} \quad \leftarrow \text{matrix inversion lemmas} \\
&= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1}
\end{aligned} \tag{103}$$

위 식의 다섯번 째 줄에서 보다시피 GN을 통해 구한 근사 해시안 행렬의 역함수 $\tilde{\mathbf{H}}^{-1}$ 와 EKF의 posterior 공분산 $\mathbf{P}_{t|t}$ 는 같은 값을 가지는 것을 알 수 있다.

$$\tilde{\mathbf{H}}^{-1} = (\mathbf{J}_t^T \mathbf{P}_{\mathbf{e}}^{-1} \mathbf{J}_t)^{-1} = \mathbf{P}_{t|t} \tag{104}$$

Posterior mean $\mathbf{x}_{t|t}$:

GN을 반복적으로 수행함에 따라 j 번째 $\mathbf{x}_{t|t,j}$ 는 다음과 같이 구할 수 있다.

$$\begin{aligned}
\hat{\mathbf{x}}_{t|t,j+1} &= \hat{\mathbf{x}}_{t|t,j} + \delta \hat{\mathbf{x}}_{t|t,j} \\
&= \hat{\mathbf{x}}_{t|t,j} + (\mathbf{J}_t^\top \mathbf{P}_{t|t-1}^{-1} \mathbf{J}_t)^{-1} (\mathbf{J}_t^\top \mathbf{P}_{t|t-1}^{-1} \mathbf{r}_t) \\
&= (\mathbf{J}_t^\top \mathbf{P}_{t|t-1}^{-1} \mathbf{J}_t)^{-1} \mathbf{J}_t^\top \mathbf{P}_{t|t-1}^{-1} (\mathbf{y}_t - \mathbf{g}(\hat{\mathbf{x}}_{t|t,j}) + \mathbf{J}_t \hat{\mathbf{x}}_{t|t,j}) \quad \leftarrow \mathbf{r}_t = \mathbf{y}_t - \mathbf{g}(\hat{\mathbf{x}}_{t|t,j}) \\
&= \left(\mathbf{P}_{t|t-1}^{-1} + \mathbf{H}_t^\top \mathbf{R}_t^{-1} \mathbf{H}_t \right)^{-1} \begin{bmatrix} \mathbf{P}_{t|t-1}^{-1} & \mathbf{H}_t^\top \mathbf{R}_t^{-1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_{t|t-1} \\ \mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}) + \mathbf{H}_t \hat{\mathbf{x}}_{t|t,j} \end{bmatrix} \quad \leftarrow \text{expand } \mathbf{J}_t \\
&= \left(\mathbf{P}_{t|t-1}^{-1} + \mathbf{H}_t^\top \mathbf{R}_t^{-1} \mathbf{H}_t \right)^{-1} \left(\mathbf{H}_t^\top \mathbf{R}_t^{-1} (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}) + \mathbf{H}_t \hat{\mathbf{x}}_{t|t,j}) + \mathbf{P}_{t|t-1}^{-1} \hat{\mathbf{x}}_{t|t-1} \right) \\
&= \left(\mathbf{P}_{t|t-1}^{-1} + \mathbf{H}_t^\top \mathbf{R}_t^{-1} \mathbf{H}_t \right)^{-1} \left(\mathbf{H}^\top \mathbf{R}_t^{-1} (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}) - \mathbf{H}_t(\hat{\mathbf{x}}_{t|t-1} - \hat{\mathbf{x}}_{t|t,j}) + \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1}) + \mathbf{P}_{t|t-1}^{-1} \hat{\mathbf{x}}_{t|t-1} \right) \\
&= \left(\mathbf{P}_{t|t-1}^{-1} + \mathbf{H}_t^\top \mathbf{R}_t^{-1} \mathbf{H}_t \right)^{-1} \left(\mathbf{H}^\top \mathbf{R}_t^{-1} (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}) - \mathbf{H}_t(\hat{\mathbf{x}}_{t|t-1} - \hat{\mathbf{x}}_{t|t,j})) + \underbrace{(\mathbf{P}_{t|t-1}^{-1} + \mathbf{H}_t^\top \mathbf{R}_t^{-1} \mathbf{H}_t)}_{\hat{\mathbf{x}}_{t|t-1}} \hat{\mathbf{x}}_{t|t-1} \right) \\
&= \hat{\mathbf{x}}_{t|t-1} + \left(\mathbf{P}_{t|t-1}^{-1} + \mathbf{H}_t^\top \mathbf{R}_t^{-1} \mathbf{H}_t \right)^{-1} \mathbf{H}_t^\top \mathbf{R}_t^{-1} (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}) - \mathbf{H}_t(\hat{\mathbf{x}}_{t|t-1} - \hat{\mathbf{x}}_{t|t,j})) \\
&= \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t,j}) - \mathbf{H}_t(\hat{\mathbf{x}}_{t|t-1} - \hat{\mathbf{x}}_{t|t,j}))
\end{aligned} \tag{105}$$

위 식은 IEKF의 (61) 식과 동일하다. 마지막 식에서 보다시피 Gauss-Newton을 통해 EKF의 해를 추정하는 것과 IEKF를 통해 해를 추정하는 것은 동일한 의미를 지닌다. 만약 처음 iteration $j = 0$ 인 경우 $\hat{\mathbf{x}}_{t|t,0} = \hat{\mathbf{x}}_{t|t-1}$ 이 되어서 식은 다음과 같이 정리된다.

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1})) \tag{106}$$

이는 EKF의 해와 동일하다. 즉, EKF는 GN iteration=1과 동일한 의미를 지니며 IEKF는 GN과 동일한 연산을 수행하는 것을 알 수 있다.

8 Wrap-up

지금까지 설명한 KF, EKF, ESKF, IEKF, IESKF를 한 장의 슬라이드로 표현하면 다음과 같다. 클릭하면 큰 그림으로 볼 수 있다.

8.1 Kalman Filter (KF)

Kalman Filter(KF)

Motion Model: $\mathbf{x}_t = \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{B}_t \mathbf{u}_t + \mathbf{w}_t$
 Observation Model: $\mathbf{z}_t = \mathbf{H}_t \mathbf{x}_t + \mathbf{v}_t$

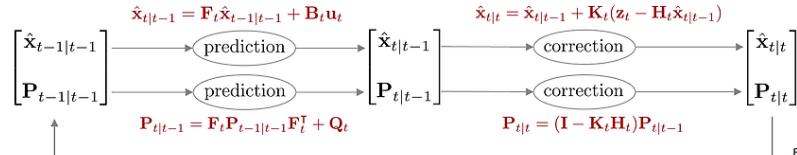
$$\begin{aligned}\overline{\text{bel}}(\mathbf{x}_t) &\sim \mathcal{N}(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}) \\ \text{bel}(\mathbf{x}_t) &\sim \mathcal{N}(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}) \\ \mathbf{w}_t &\sim \mathcal{N}(0, \mathbf{Q}_t) \\ \mathbf{v}_t &\sim \mathcal{N}(0, \mathbf{R}_t)\end{aligned}$$

KF Prediction:

$$\begin{aligned}\hat{\mathbf{x}}_{t|t-1} &= \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} + \mathbf{B}_t \mathbf{u}_t \\ \mathbf{P}_{t|t-1} &= \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{Q}_t\end{aligned}$$

KF Correction:

$$\begin{aligned}\mathbf{K}_t &= \mathbf{P}_{t|t-1} \mathbf{H}_t^\top (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^\top + \mathbf{R}_t)^{-1} \\ \hat{\mathbf{x}}_{t|t} &= \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{z}_t - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1}) \\ \mathbf{P}_{t|t} &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1}\end{aligned}$$



For more information, please visit alida.tistory.com

8.2 Extended Kalman Filter (EKF)

Extended Kalman Filter(EKF)

Motion Model: $\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_t, \mathbf{w}_t) \simeq \mathbf{F}_t \mathbf{x}_{t-1} + \tilde{\mathbf{u}}_t + \tilde{\mathbf{w}}_t$
 Observation Model: $\mathbf{z}_t = \mathbf{h}(\mathbf{x}_t, \mathbf{v}_t) \simeq \mathbf{H}_t \mathbf{x}_t + \tilde{\mathbf{z}}_t + \tilde{\mathbf{v}}_t$

$$\begin{aligned}\overline{\text{bel}}(\mathbf{x}_t) &\sim \mathcal{N}(\hat{\mathbf{x}}_{t|t-1}, \mathbf{P}_{t|t-1}) \\ \text{bel}(\mathbf{x}_t) &\sim \mathcal{N}(\hat{\mathbf{x}}_{t|t}, \mathbf{P}_{t|t}) \\ \tilde{\mathbf{w}}_t &\sim \mathcal{N}(0, \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top) \\ \tilde{\mathbf{v}}_t &\sim \mathcal{N}(0, \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)\end{aligned}$$

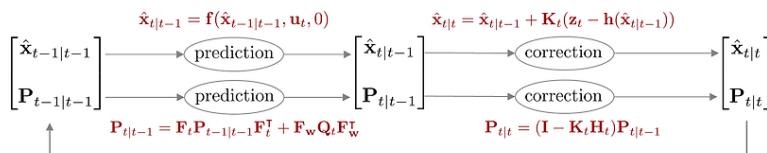
$$\begin{aligned}\mathbf{F}_t &= \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}_{t-1}} \right|_{\mathbf{x}_{t-1}=\hat{\mathbf{x}}_{t-1|t-1}} \\ \mathbf{F}_w &= \left. \frac{\partial \mathbf{f}}{\partial \mathbf{w}_t} \right|_{\mathbf{x}_{t-1}=\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{w}_t=0} \\ \mathbf{H}_t &= \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}_t} \right|_{\mathbf{x}_t=\hat{\mathbf{x}}_{t|t-1}} \\ \mathbf{H}_v &= \left. \frac{\partial \mathbf{h}}{\partial \mathbf{v}_t} \right|_{\mathbf{x}_t=\hat{\mathbf{x}}_{t|t-1}, \mathbf{v}_t=0}\end{aligned}$$

EKF Prediction:

$$\begin{aligned}\hat{\mathbf{x}}_{t|t-1} &= \mathbf{f}(\hat{\mathbf{x}}_{t-1|t-1}, \mathbf{u}_t, 0) \\ \mathbf{P}_{t|t-1} &= \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^\top + \mathbf{F}_w \mathbf{Q}_t \mathbf{F}_w^\top\end{aligned}$$

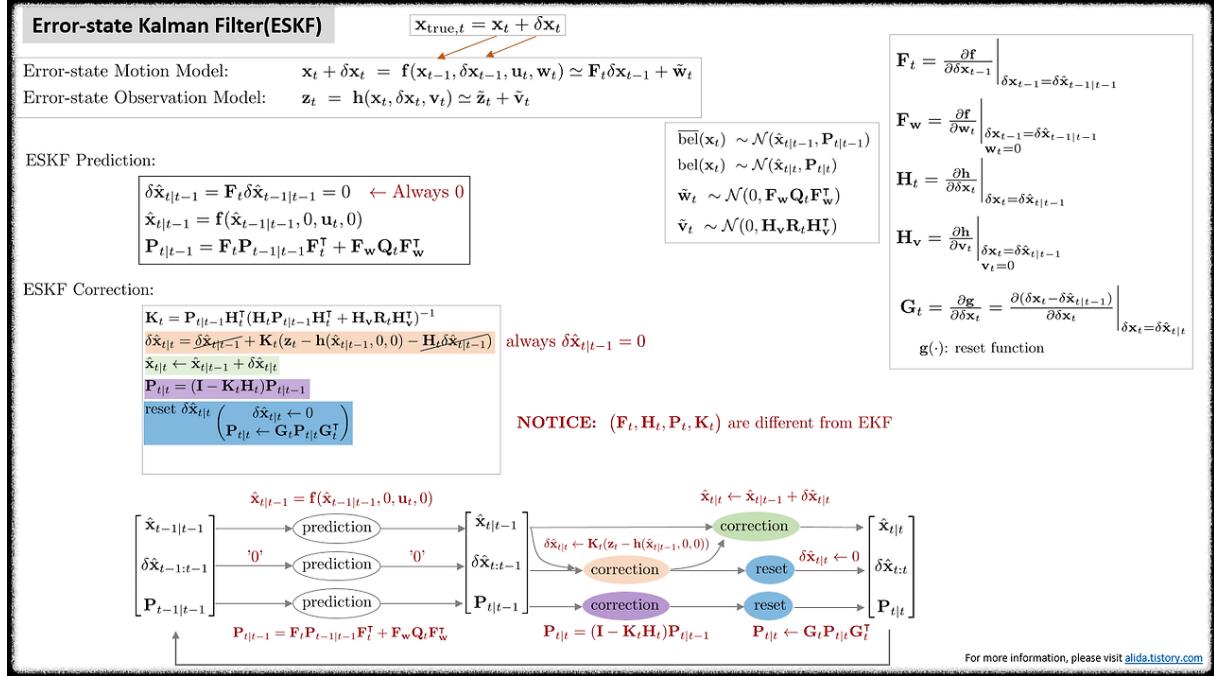
EKF Correction:

$$\begin{aligned}\mathbf{K}_t &= \mathbf{P}_{t|t-1} \mathbf{H}_t^\top (\mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^\top + \mathbf{H}_v \mathbf{R}_t \mathbf{H}_v^\top)^{-1} \\ \hat{\mathbf{x}}_{t|t} &= \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{z}_t - \mathbf{h}(\hat{\mathbf{x}}_{t|t-1})) \\ \mathbf{P}_{t|t} &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1}\end{aligned}$$

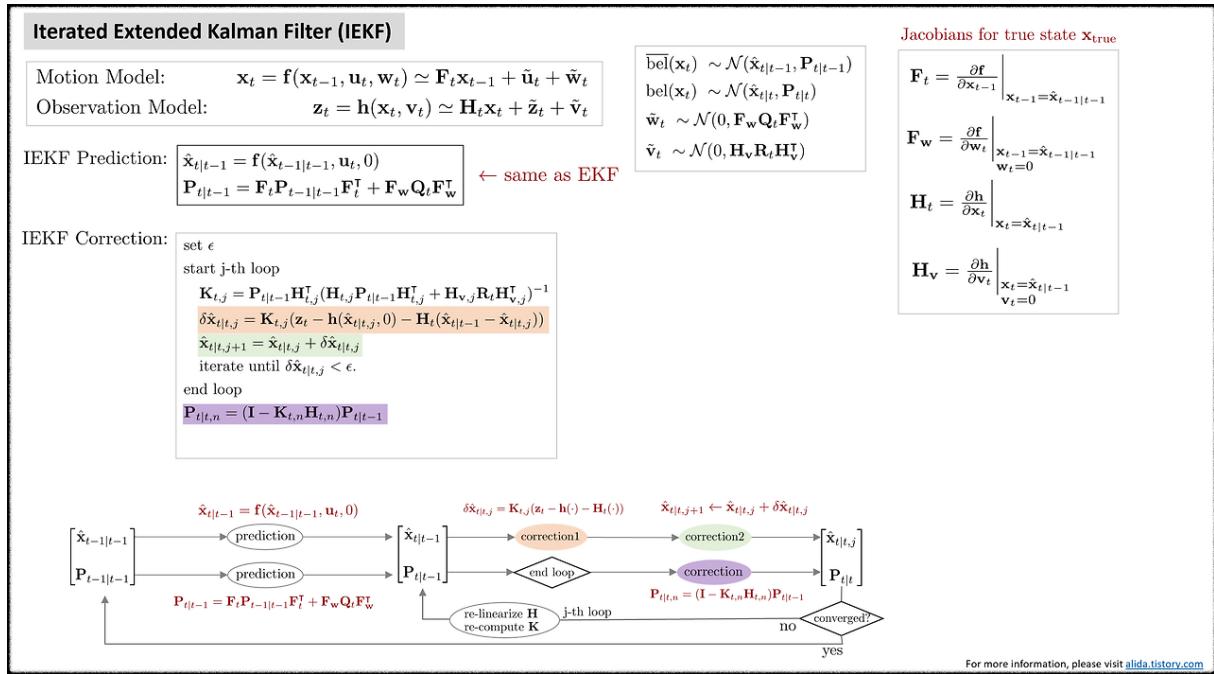


For more information, please visit alida.tistory.com

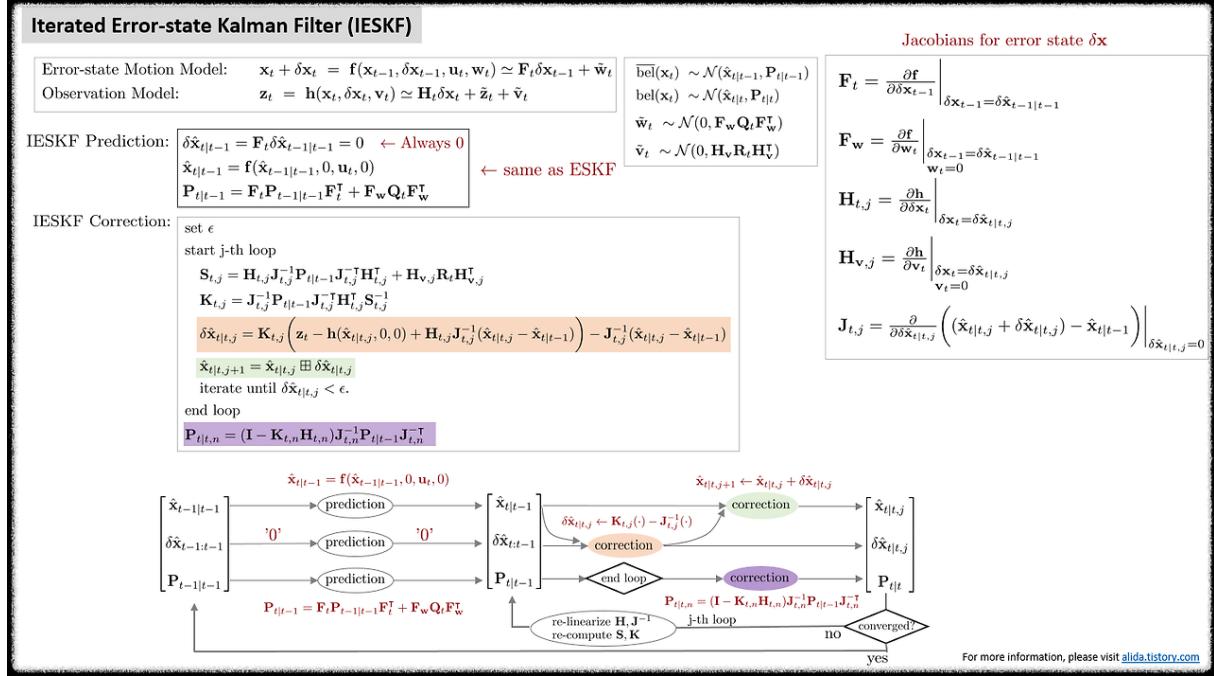
8.3 Error-state Kalman Filter (ESKF)



8.4 Iterated Extended Kalman Filter (IEKF)



8.5 Iterated Error-state Kalman Filter (IESKF)



9 Reference

- [1] (Wiki) Kalman Filter
- [2] (Paper) Sola, Joan. "Quaternion kinematics for the error-state Kalman filter." arXiv preprint arXiv:1711.02508 (2017).
- [3] (Youtube) Robot Mapping Coure - Freiburg Univ
- [4] (Blog) [SLAM] Kalman filter and EKF(Extended Kalman Filter) - Jinyong Jeong
- [5] (Blog) Error-State Kalman Filter understanding and formula derivation - CSDN
- [6] (Paper) He, Dongjiao, Wei Xu, and Fu Zhang. "Kalman filters on differentiable manifolds." arXiv preprint arXiv:2102.03804 (2021).
- [7] (Book) SLAM in Autonomous Driving book (SAD book)
- [8] (Paper) Xu, Wei, and Fu Zhang. "Fast-llo: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter." IEEE Robotics and Automation Letters 6.2 (2021): 3317-3324.
- [9] (Paper) Huai, Jianzhu, and Xiang Gao. "A Quick Guide for the Iterated Extended Kalman Filter on Manifolds." arXiv preprint arXiv:2307.09237 (2023).
- [10] (Paper) Bloesch, Michael, et al. "Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback." The International Journal of Robotics Research 36.10 (2017): 1053-1072.
- [11] (Paper) Skoglund, Martin A., Gustaf Hendeby, and Daniel Axehill. "Extended Kalman filter modifications based on an optimization view point." 2015 18th International Conference on Information Fusion (Fusion). IEEE, 2015.
- [12] (Blog) From MAP, MLE, OLS, GN to IEKF, EKF
- [13] (Book) Thrun, Sebastian. "Probabilistic robotics." Communications of the ACM 45.3 (2002): 52-57.

10 Revision log

- 1st: 2020-06-23
- 2nd: 2020-06-24
- 3rd: 2020-06-26
- 4th: 2023-01-21
- 5th: 2023-01-31
- 6th: 2023-02-02
- 7th: 2023-02-04
- 8th: 2024-02-08
- 9th: 2024-02-09