Master's Thesis of Engineering

Federated Invariant EKF for Multi-sensor Navigation System

다중 센서 항법시스템을 위한 연합형 불변 확장칼만필터

February 2022

Graduate School of Engineering
Seoul National University
Aerospace Engineering Major

Jeong Ho Hwang

Federated Invariant EKF for Multi-sensor Navigation System

Advised by Professor Chan Gook Park

Submitting a master's thesis of Engineering

December 2021

Graduate School of Engineering

Seoul National University

Aerospace Engineering Major

Jeong Ho Hwang

Confirming the master's thesis written by
Jeong Ho Hwang
December 2021

Chair _____(Seal)

Vice Chair _____(Seal)

Examiner _____(Seal)

Abstract

Federated Invariant EKF for Multi-sensor Navigation System

Jeong Ho Hwang
Department of Aerospace Engineering
The Graduate School
Seoul National University

This thesis presents the federated invariant extended Kalman filter (IEKF) using multiple measurements. IEKF has superior estimation performance compared to EKF through the definition of state variables on matrix Lie group while using the framework of the EKF. The IEKF enables trajectory independent estimation when left- or right-invariant measurements are used with proper invariant error selection. As a result, the IEKF ensures the convergence and accuracy of estimation, even when the estimation error is large. Most IEKF studies assumed the use of single aiding measurement. However, navigation systems often use multiple aiding sensors to improve estimation performance in applications. When left- and right-invariant measurements are used simultaneously, implementing the LIEKF or RIEKF with a centralized filter structure causes some terms of the measurement matrix dependent on the current estimates, which results in IEKF losing its *trajectory independent* advantage. On the other hand, when a decentralized filter structure, especially a federated filter structure, is applied, the estimation becomes *trajectory independent*

through separate update of each measurement in the local filters. This thesis proposes a fusion method of IEKF using the federated filter structure for simultaneous use of left- and right-invariant measurements. The performance of the proposed fusion method is validated through simulations. The error convergence and accuracy of the proposed method and the centralized IEKF are compared.

Keywords: Invariant extended Kalman filter, Nonlinear filtering, Multi sensor navigation, Federated filter

Student number: 2020-24160

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

The extended Kalman filter (EKF) is the most common and widely used type of filter in navigation [1,2,3]. It is used to solve the problems in various applications by linearizing the nonlinear problem. Calculating the Jacobian matrix in the linearization process might be difficult for inexperienced engineers, but it generally has computational efficiency over other Kalman filter techniques.

The biggest problem with the EKF is that it uses an estimate as a linearization point. Since the linearization process is performed under the small error assumption, divergence is a concerning problem as the estimation error increases [4]. Due to this fact, the EKF has a disadvantage that its convergence is not guaranteed. Various techniques have been proposed to compensate for the shortcoming of EKF. The unscented Kalman filter (UKF) replaced the linearization process of the EKF with an unscented transform [5,6,7]. It is possible to solve the linearization error problem by using the nonlinear function itself. In addition, the UKF is easy to implement compared to the EKF. However, since an unscented transform uses multiple points for the propagation, computation time increases compared to the EKF. Also, it is still a second-order approximation. Like the UKF, the particle filter (PF) can reflect the nonlinearity well because it uses many particles [8]. Still, there are problems such as

degeneracy, impoverishment, and the curse-of-dimensionality [9].

Recently, another variant of the EKF, the Invariant EKF (IEKF), has been proposed and applied to various fields [10]. The IEKF differs in that it defines the state variables in the matrix Lie group while maintaining the framework of the EKF. Also, the definition of error and correction are changed. The advantages of the IEKF by defining the state variables in the matrix Lie group are as follows. 1) Error propagation is autonomous. 2) For the specific measurement models, the innovation model is also independent of the estimate. 3) In the deterministic sense, the estimate is not included in the calculation of Kalman gain. 4) The IEKF guarantees the convergence of estimates, which EKF does not guarantee [11].

Using the IEKF, error propagation and measurement update process differ depending on the selection between left- and right-invariant errors. The selection of the invariant error is decided by the measurement used to compensate the error of state estimates calculated from the reference system. If a measurement has the form of the left-invariant, the left-invariant error is chosen to implement the left-IEKF (LIEKF). Likewise, if a measurement has the form of the right-invariant, the right-invariant error is chosen to design the right-IEKF (RIEKF). If a measurement model does not belongs to either of the two invariant forms, it is recommended to compare the estimation performance of the LIEKF and the RIEKF. For the multi-sensor system, same rule is exploited. If all the aiding sensors have the left-invariant measurement form, the centralized LIEKF is a proper choice. In the opposite case, the centralized RIEKF is a proper choice.

However, when both the left- and the right-invariant measurements are used, the centralized filter structure implementation of the LIEKF and the RIEKF encounters a problem. The measurement matrix of the centralized LIEKF and

RIEKF would contain state estimates due to the mismatched measurements. It means that the IEKF calculates the Kalman gain under the influence of the state estimate. The IEKF is no more *trajectory independent*. It might lead to the deterioration of estimation accuracy and convergence speed. Therefore, it is necessary to derive a method to ensure that each measurement is used without losing the advantage of the IEKF.

1.2 Objectives and contributions

In this thesis, the implementation method of the federated filter structure of the IEKF is proposed. It is shown that the centralized filter estimation performance can be improved in a system that uses the left-/right-invariant measurements simultaneously. Since the IEKF is being studied widely in navigation field, this thesis uses a navigation example to assess the performance of the proposed method. The main contributions of this thesis are as follows:

- For the first time among studies on the IEKF, this thesis proposes a solution when the left- and right-invariant measurements are used simultaneously. A federated filter structure is exploited to process the left- and right-invariant measurements separately in the local filters. Then the outputs of each local filter are fused in the master filter. Since each measurement is processed with proper IEKF, the proposed structure guarantees the *trajectory independent* estimation.
- Since the centralized LIEKF and RIEKF use different measurement matrices, the performance difference between the two techniques is compared. From the simulation results that are consistent with the intuitive

prediction from the measurement matrix, the reason for applying the centralized RIEKF in the previous studies is revealed.

- Previously, there is a study on the fusion of several poses defined in the right-invariant form. This thesis extends the method to fuse the poses when some of the poses are defined with the left-invariant form and the rest of the poses are defined with the right-invariant form. It is done by transforming the left-invariant covariance into the right-invariant covariance through the adjoint matrix.

Chapter 2

Related Works

2.1 Invariant extended Kalman filter (IEKF)

The IEKF is firstly introduced in [12]. It is a new version of the extended Kalman filter (EKF). It uses the invariant state error and geometrically adapted correction rather than the linear state error and the linear error correction. In the problem of estimating attitude, velocity, gyro bias, and accelerometer scale factor using 9-axis measurements of IMU and GPS velocity measurements, the left-IEKF (LIEKF) and the right-IEKF (RIEKF) are derived in [13]. In [14], the IEKF was implemented in the helicopter UAV system, and it showed a superior performance to the conventional EKF.

The IEKF differs from the conventional EKF because it defines the state in the matrix Lie group. As a result, the definition of the error state and the distribution of the uncertainty are changed. For the state variables defined on matrix Lie group, the linearization error is the logarithm of invariant error. This linearization error has the Gaussian distribution in vector space. As a result, the uncertainty distribution of the state variable becomes a concentrated Gaussian distribution. In [15], the concentrated Gaussian distribution is used for the modeling. The concentrated Gaussian distribution expresses the uncertainty of the actual system better than the conventional Gaussian distribution in the Euclidean space. Contrary to the

conventional assumption, it is widely known that the uncertainty due to the noise of the actual system has the banana-shaped distribution. [16,17] showed that the concentrated Gaussian distribution captures the distribution better than the conventional assumption.

In [11], the system to which the IEKF can be applied has been extended to the group affine system. Moreover, they demonstrated that the invariant errors propagate autonomously for the group affine systems. It means that nonlinear problems can be transformed into linear problems in the error propagation stage. If the measurement model belongs to either the left- or right-invariant measurement model, the convergence of the filter is proven in the paper. Also, it was shown that the definition of the state variable could be extended from existing $\mathbb{SE}(3)$ to $\mathbb{SE}_2(3)$. This extension enabled the application of the IEKF to the IMU states consisting of position, velocity, and attitude.

The IEKF has been widely exploited in the simultaneous localization and mapping (SLAM) problems. In [18,19,20], it has been shown that the IEKF can solve the inconsistency problem presented in existing EKF-based SLAM. Especially in [21], the IEKF based SLAM achieved estimation performance close to the iSAM, a state-of-the-art method. Most IEKF studies assume use of a single aiding sensor. However, [22] uses the position and velocity measured by GPS and the heading angle estimated by the magnetometer as measurements. The GPS position and velocity measurements are the left-invariant measurements and the magnetometer measurement is the right-invariant measurement. The study used a centralized RIEKF for the mixed measurements. Similarly, [23] implemented a centralized RIEKF for navigation of the autonomous underwater vehicles (AUVs) equipped with a doppler velocity log (DVL) and a pressure sensor. Since DVL is a right-

invariant measurement and pressure sensor is a left-invariant measurement, choosing a centralized RIEKF is not the best choice.

2.2 Federated filter

For the decentralized filtering depicted in Figure 2.1, the standard Kalman filter is divided into two stages: local filters and a master filter. Local filters are set as many as the number of aiding sensors used by the system and process each sensor's measurement. The outputs of the local filters are fused in the master filter. Since the calculation of each local filter can be performed in parallel, the calculation efficiency is improved. The most significant advantage of decentralized filtering is that it can configure a fault-tolerant system. Since individual local filters process each sensor, classifying sensor failures and isolating them is a structural advantage.

The federated filter is distinguished from the general decentralized filter in that it uses an information-sharing factor. It was proposed by Carlson [25] to improve the fault-tolerant properties. The master filter uses a sensor with a high sampling rate as a reference sensor and puts estimates to the local filter. In each local filter, the measurement update is performed using the dedicated sensor outputs. Therefore, the state variable of each local filter is set according to the dedicated sensor, and the state variable and model of the master filter may be a little more complicated. The overall structure is depicted in Figure 2.2.

When designing a federated filter, it is a matter of choice whether to initialize the local filter using the master filter's estimate or not. For the No Reset (NR) mode, the initialization of the local filter is not done. In this mode, the fused value of the estimate in the master filter is not optimal. Instead, the fault detectability is increased.

On the other hand, for the Fusion Reset (FR) mode, the estimate of state and the error covariance is initialized by the value of the master filter. In this case, the estimates are globally optimal, but the fault detectability decreases.

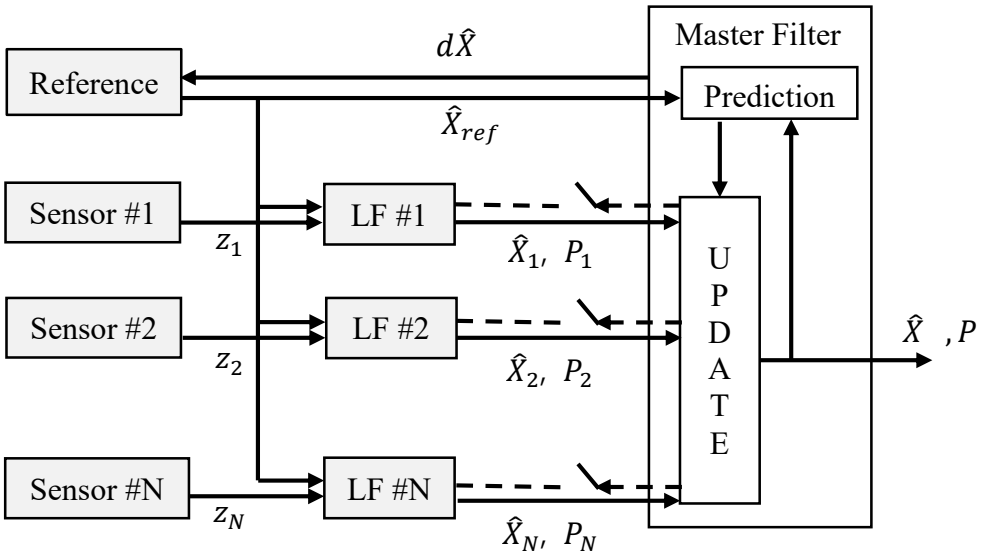

Figure 2.1 Decentralized filter structure

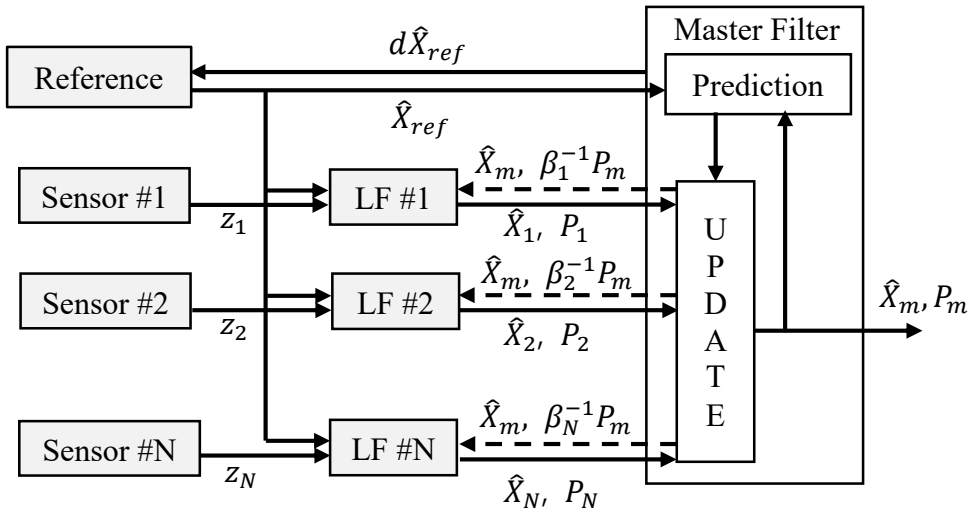

Figure 2.2 Federated filter structure

Chapter 3

Framework of invariant EKF

In this chapter, the overall framework of IEKF is depicted. First, several definitions required to use IEKF are briefly summarized. Then, the framework of the IEKF is depicted and equations are derived for the general system. Next, the advantages of the IEKF are confirmed once again by deriving detailed equations for the IMU state variable, the INS dynamic, and various possible measurement models.

3.1 Mathematical preliminaries

The matrix Lie group $\mathcal{G}$ is a set of square invertible matrices in which matrix multiplication and inversion are defined inside the set [26]. The matrix Lie group satisfies the following properties

$$\mathrm{Id} \in \mathcal{G}, \tag{3.1}$$

$$\forall X \in \mathcal{G},\ \mathrm{X}^{-1} \in \mathcal{G}, \tag{3.2}$$

$$\forall X, Y \in \mathcal{G},\ XY \in \mathcal{G}, \tag{3.3}$$

The Lie algebra $\mathfrak{g}$ is associated with the Lie group $\mathcal{G}$ is a tangent space of $\mathcal{G}$ at the identity. The Lie algebra has the same dimension as the Lie group. The mapping between the Euclidean vector space into the Lie algebra space is defined as follows

$$(\cdot)^{\wedge} : \mathbb{R}^{\dim \mathfrak{g}} \longrightarrow \mathfrak{g}, \tag{3.4}$$

$$(\cdot)^{\vee} : \mathfrak{g} \longrightarrow \mathbb{R}^{\dim \mathfrak{g}}, \tag{3.5}$$

The exponential map of the Lie group exactly transfers the elements of the Lie algebra to the Lie group.

$$\exp(\xi) = \exp_{\mathrm{m}}(\xi^{\wedge}). \tag{3.6}$$

Here, $\exp_{\mathrm{m}}(\cdot)$ is usual exponential of square matrices. To calculate the vector ξ, the logarithm is used as follows

$$\log(\exp(\xi))^{\vee} = \xi. \tag{3.7}$$

The adjoint of $\mathcal{G}$ at X, denoted as Ad_{X}, is as follows

$$\mathrm{Ad}_{\mathrm{X}} : \ \mathfrak{g} \longrightarrow \mathfrak{g}; \ \ \xi^{\wedge} \ \longrightarrow \mathrm{Ad}_{\mathrm{X}}(\xi^{\wedge}) = \mathrm{X}\xi^{\wedge}\mathrm{X}^{-1}. \tag{3.8}$$

Since the adjoint is linear, an equivalent matrix operator, adjoint matrix Ad_X, can be computed as

$$Ad_X = (\mathrm{X}\xi^{\wedge}\mathrm{X}^{-1})^{\vee}. \tag{3.9}$$

3.2 States and model

3.2.1 Matrix Lie group states

Three matrix Lie groups are usually used for the navigation problems: the group of rotation matrices $\mathbb{SO}(3)$, the group of direct spatial isometries $\mathbb{SE}(3)$, and the group of double direct isometries $\mathbb{SE}_2(3)$. If the state variable which concern is the attitude, the attitude is defined on the group of rotation matrices

$$X = R \in \mathbb{SO}(3), \tag{3.10}$$

where R is a 3×3 rotation matrix that defines the rotation between the navigation frame and the body frame. [12] used $\mathbb{SO}(3)$ for the attitude estimation problem.

If the state variable is the attitude and the position, the group of direct spatial isometries is used

$$X = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \in \mathbb{SE}(3). \tag{3.11}$$

Since R is a 3×3 matrix and the navigation frame position p is a 3×1 vector, X is a 4×4 matrix. This 6D pose state is commonly used in the SLAM studies [18]. Actually, [18] augmented other state variables, but the essential state variable is composed of the attitude and position of the vehicle.

Finally, if the system uses an inertial measurement unit (IMU), velocity should be estimated to calculate position. In this case, the state variable defined in the group of double direct isometries is given by

$$X = \begin{bmatrix} R & v & p \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix} \in \mathbb{SE}_2(3), \tag{3.12}$$

where the navigation frame v is a 3×1 vector. As shown in (3.12), X is a 5×5 matrix. Like [21], researchers studying robotics often define the state variables in a form where velocity and position are swapped in (3.10). However, in this thesis, (3.12) will be used.

In addition to (3.10), (3.11), and (3.12), other state variables can be defined using various matrix Lie groups. For example, in [27], the position of the landmarks are augmented by

$$X = \begin{bmatrix} R & v & p & l_1 & \cdots & l_n \\ & 0_{(2+n)\times3} & & & I_{(2+n)\times(2+n)} & \end{bmatrix} \in \mathbb{SE}_{2+n}(3), \tag{3.13}$$

where l_i is the ith landmark's position, and n is the number of landmarks. There is no problem in using such augmented states because all state variables are defined on matrix Lie group. However, not all the state variables are always perfectly defined on matrix Lie group. As can be checked from (3.11), (3.12), and (3.13), all state variables augmented next to the attitude matrix are variables in the navigation/global frame. It means that the state variables on the body frame cannot be defined on matrix Lie group. For instance, in the precision navigation using IMU, bias of the accelerometer and gyroscope needs to be estimated. In this case, the IMU states' biases are augmented [28]. As a result, in [17,23,29], biases are augmented to the Lie group states, defined in Euclidean space.

3.2.2 Process model

A state process model evolving on matrix Lie group with the state at time t, $X_t \in \mathcal{G}$, is denoted by

$$\frac{d}{dt}X_t = f_{u_t}(X_t). \tag{3.14}$$

Letting X_t is a true state variable and $\hat{X}_t$ is an estimate of the state X_t, the state estimation error is defined by the following two forms

$$\eta_t^L = X_t^{-1}\hat{X}_t = (LX_t)^{-1}\left(L\hat{X}_t\right), \tag{3.15}$$

$$\eta_t^R = \hat{X}_t X_t^{-1} = \left(\hat{X}_t L\right)(X_t L)^{-1}, \tag{3.16}$$

where L is an arbitrary matrix that belongs to the Lie group. (3.15) is called the left-invariant error since it is invariant to the left multiplication of an arbitrary matrix. In the same way, (3.16) is called the right-invariant error since it is invariant to the right multiplication.

If the function f_{u_t} in (3.14) satisfies the following property for all $t > 0$, the process model is said to be *group affine system.*

$$A, B \in \mathcal{G} \qquad f_{u_t}(AB) = f_{u_t}(A)B + Af_{u_t}(B) - Af_{u_t}(Id)B, \tag{3.17}$$

where $Id \in \mathcal{G}$ is the group identity element. For the group affine system, the invariant error dynamics are *trajectory independent* and satisfy following

$$\frac{\mathrm{d}}{\mathrm{dt}}\eta_t^L = g_{u_t}^L(\eta_t^L) \quad \text{where } g_{u_t}^L(\eta_t^L) = f_{u_t}(\eta_t^L) - f_{u_t}(I_d)\eta_t^L, \tag{3.18}$$

$$\frac{\mathrm{d}}{\mathrm{dt}}\eta_t^R = g_{u_t}^R(\eta_t^R) \quad \text{where } g_{u_t}^R(\eta_t^R) = f_{u_t}(\eta_t^R) - \eta_t^R f_{u_t}(I_d). \tag{3.19}$$

(3.18) and (3.19) mean that invariant error dynamics are only related to the invariant itself and not the current state estimates. It is often called "state-*trajectory independent* propagation [11]" or "autonomous error dynamics [17]." The independence of the error propagation is proved in [11].

Let a matrix $A_t^i \in \mathbb{R}^{\dim \mathfrak{g} \times \dim \mathfrak{g}}$ satisfies following

$$g_{u_t}^i\left(\exp\left(\xi_t^i\right)\right) \triangleq \left(A_t^i \xi_t^i\right)^\wedge + \mathcal{O}(\left\|\xi_t^i\right\|^2), \tag{3.20}$$

where $i \in \{L, R\}$. If ξ_t^i is defined for all $t > 0$ and is the solution of the linear differential equation

$$\frac{d}{dt}\xi_t^i = A_t^i \ \xi_t^i, \tag{3.21}$$

and the initial error $\xi_0^i \in \mathbb{R}^{\dim \mathfrak{g}}$ defines the initial invariant error $\exp\left(\xi_0^i\right) = \eta_0^i$, then for all $t > 0$,

$$\eta_t^i = \exp(\xi_t^i), \tag{3.22}$$

holds. (3.22) is called "Log-linear property of the error" and it means that the

nonlinear estimation error η_t^i can be exactly recovered from the time-varying linear differential equation (3.21). Although it can be seen that (3.21) is a second-order approximation, the true error can be recovered from ξ_t^i with no approximation error. Note that no matter how big the difference between X_t and $\hat{X}_t$ is, the linearized error ξ_t^i follows the linear differential equation. Also, since the invariant error can be exactly recovered from the solution of (3.21), A_t^i is not the typical linearized Jacobian matrix.

(3.14) only considered deterministic elements of the process model. Usually, the process model also contains the stochastic elements and noise in the system. Considering the noises, the process model becomes

$$\frac{\mathrm{d}}{\mathrm{dt}} X_t = f_{u_t}(X_t) + X_t(w_t^\wedge), \tag{3.23}$$

$$\frac{\mathrm{d}}{\mathrm{dt}} X_t = f_{u_t}(X_t) + (w_t^\wedge) X_t, \tag{3.24}$$

where choosing between (3.23) and (3.24) is decided by applications [30]. Using (3.23), the invariant error dynamics are changed from (3.18) and (3.19) to

$$\frac{d}{dt}\eta_t^L = g_{u_t}^L(\eta_t^L) - (w_t^\wedge)\eta_t^L \text{ and} \tag{3.25}$$

$$\frac{d}{dt}\eta_t^R = g_{u_t}^R(\eta_t^R) - \left(Ad_{\hat{X}_t} w_t^\wedge\right)\eta_t^R, \tag{3.26}$$

where $g_{u_t}^L(\eta_t^L)$ and $g_{u_t}^R(\eta_t^R)$ are not changed. As a result, the linearized error differential equation becomes

$$\frac{d}{dt}\xi_t^i = A_t^i \ \xi_t^i + B_t^i w_t. \tag{3.27}$$

B_t^i is the Jacobian matrix for process noise vector.

Since the stochastic noises are introduced, they might affect the convergence

speed of the observer. However, the convergence property of the IEKF is still guaranteed [30].

3.2.3 Measurement model

The time propagated state estimates are corrected by the sensor measurements in the correction step. In the EKF, the measurement matrix can be calculated through the linearization without any special requirements for the measurement model to be satisfied. In the case of the IEKF, various measurement models can be used in the same way, but in the case of using a specific type of measurement, more advantages can be exploited.

There are two particular types of measurement models as

$$z_t^L = X_t d + n_t \text{ and} \tag{3.28}$$

$$z_t^R = X_t^{-1} d + n_{t,} \tag{3.29}$$

where d is a known constant vector and n_t is the vector of white Gaussian noise. If the measurement satisfies the form (3.28), it is called a left-invariant measurement. If the measurement satisfies the form (3.29), it is called a right-invariant measurement. If a measurement belongs to either left- or right-invariant, the measurement model and the innovation will be autonomous [11,17].

Since the error definition has changed, the innovation also defined differently from the conventional EKF as

$$V_t^L = \hat{X}_t^{-1}(\hat{z}_t^L - z_t^L), \tag{3.30}$$

$$V_t^R = \hat{X}_t(\hat{z}_t^R - z_t^R), \tag{3.31}$$

where $\hat{z}_t^L$ and $\hat{z}_t^R$ are the estimated measurements calculated from the current state

estimate. Using first-order approximation $\eta_t = \exp(\xi_t) \approx I + \xi_t^{\wedge}$, the innovations are linearized as follows [10,23]

$$\begin{aligned} V_t^L &= \hat{X}_t^{-1}(\hat{z}_t^L - z_t^L) \\ &= \hat{X}_t^{-1}\left(\hat{X}_t d - X_t d - n_t \right) \\ &= d - (\eta_t^L)^{-1} d - \hat{X}_t^{-1} n_t \\ &\approx d - \left(I - \xi_t^{L\wedge} \right) d - \hat{X}_t^{-1} n_t \\ &= H^L \ \xi_t^L - \hat{X}_t^{-1} n_t. \end{aligned} \tag{3.32}$$

$$\begin{aligned} V_t^R &= \hat{X}_t(\hat{z}_t^R - z_t^R) \\ &= \hat{X}_t\left(\hat{X}_t^{-1} d - X_t^{-1} d - n_t \right) \\ &= d - \eta_t^R d - \hat{X}_t n_t \\ &\approx d - \left(I + \xi_t^{R\wedge} \right) d - \hat{X}_t n_t \\ &= H^R \ \xi_t^R - \hat{X}_t n_t. \end{aligned} \tag{3.33}$$

With proper error selection for either the left-/right-invariant measurement, the measurement matrix H^L or H^R would be *trajectory independent*. Ignoring the estimates included in the noise-related term, Kalman gain calculation is *trajectory independent* in the deterministic sense.

3.2.4 Adjoint

Two invariant errors can be transformed to each other by an adjoint matrix as follows [17,29]

$$\eta^R = \hat{X} X^{-1} = \hat{X} X^{-1} \hat{X} \ \hat{X}^{-1} = \hat{X} \eta^L \ \hat{X}^{-1},$$

$$\exp\left(\xi^{R\wedge}\right) = \hat{X}\exp\left(\xi^{L\wedge}\right)\hat{X}^{-1} = \exp(Ad_{\hat{X}}\xi^{L\wedge}),$$

$$\xi^{R} = Ad_{\hat{X}}\xi^{L}. \tag{3.34}$$

Using (3.34), linearized left- and right-invariant errors can be transformed to each other. Accordingly, error covariance can also be transformed as

$$P^{R} = E\left[\xi^{R}\xi^{R^{T}}\right] = Ad_{\hat{X}}E\left[\xi^{R}\xi^{R^{T}}\right]Ad_{\hat{X}}^{T} = Ad_{\hat{X}}P^{L}Ad_{\hat{X}}^{T}. \tag{3.35}$$

The adjoint matrix satisfies the following properties.

$$Ad_{X^{-1}} = Ad_{X}^{-1}, \tag{3.36}$$

$$Ad_{XY} = Ad_{X}Ad_{Y}. \tag{3.37}$$

3.3 IEKF for inertial navigation

In Section 4, the application methods of IEKF for systems using multiple measurements will be discussed. Moreover, the methods are applicable for any systems to which IEKF is applicable. Since this dissertation considers application in navigation systems, inertial navigation is considered. So, for the expressions described earlier in Section 3.2, the detailed equations are derived using the IMU state variables.

3.3.1 IMU states and error states

The overall structure of a strapdown inertial navigation system is depicted in [32]. From an angular rate ω_{ib}^{b} measured by the gyroscopes, the attitude of the vehicle is calculated. Then using the updated attitude, the specific force f^{b}

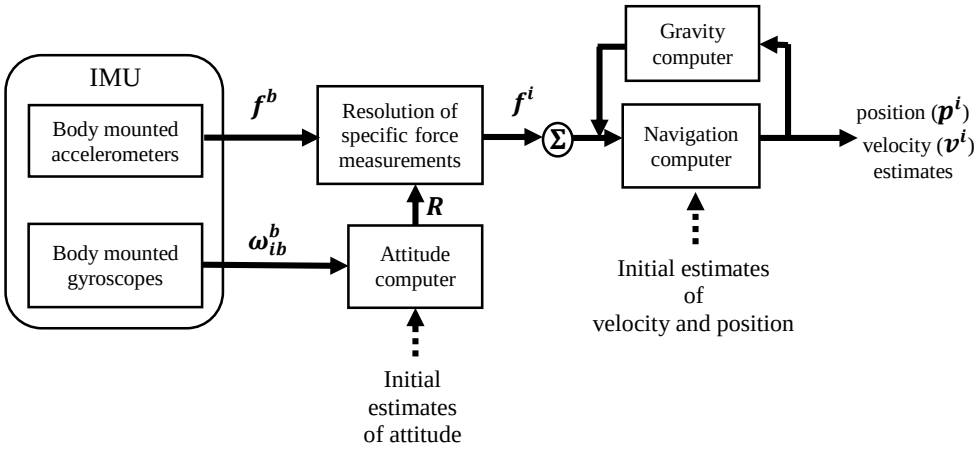

Figure 3.1 Strapdown inertial navigation system

measured by accelerometer is decomposed to the specific force in the navigation frame f^i – superscript i is used for general representation – and then integrated to calculate the velocity and position of the vehicle, compensating the gravity. So, to use IMU, the state is composed of attitude, velocity, and position.

$$X_t = \begin{bmatrix} R_t & v_t & p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}. \tag{3.38}$$

where R_t is a direction cosine matrix, v_t is a velocity in navigation frame, p_t is a position in the navigation frame, and t is time. The superscript i is removed for convenience.

Suppose there is a vector $\boldsymbol{\xi} = \begin{bmatrix} \xi_R^T & \xi_v^T & \xi_p^T \end{bmatrix}^T \in \mathbb{R}^9$ where $\xi_R, \xi_v, \xi_p \in \mathbb{R}^3$. Then the "hat" operator, which maps the vector to the element in the Lie algebra, is defined as

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix} (\xi_R)_\times & \xi_\mathrm{v} & \xi_p \\ 0_{1\times3} & 0 & 0 \\ 0_{1\times3} & 0 & 0 \end{bmatrix}, \tag{3.39}$$

where $(\xi_R)_\times$ denotes the skew-symmetric matrix of ξ_R. The exponential of $\mathbb{SE}_2(3)$ are defined as

$$\exp_{\mathbb{SE}_2(3)}(\boldsymbol{\xi}) = \begin{bmatrix} \exp_{\mathbb{SO}(3)}(\xi_R) & \mathcal{J}_{\xi_R}\xi_{\mathrm{v}} & \mathcal{J}_{\xi_R}\xi_p \\ 0_{1\times 3} & 0 & 0 \\ 0_{1\times 3} & 0 & 0 \end{bmatrix}, \tag{3.40}$$

where

$$\exp_{\mathbb{SO}(3)}(\xi_R) = I_{3\times 3} + \frac{\sin(\|\xi_R\|)}{\|\xi_R\|}(\xi_R)_\times + \frac{1-\cos(\|\xi_R\|)}{\|\xi_R\|}(\xi_R)_\times{}^2, \tag{3.41}$$

$$\mathcal{J}_{\xi_R} = I + \frac{1-\cos(\|\xi_R\|)}{\|\xi_R\|^2}(\xi_R)_\times + \frac{\|\xi_R\|-\sin(\|\xi_R\|)}{\|\xi_R\|^3}(\xi_R)_\times^2. \tag{3.42}$$

Here, (3.41) is the closed-form equation of exponential mapping for $\mathbb{SO}(3)$ and (3.42) is the left Jacobian for $\mathbb{SO}(3)$. The Jacobian for $\mathbb{SE}_2(3)$ and its inverse is given by [33]

$$\mathcal{J}_{\boldsymbol{\xi}} \coloneqq \begin{bmatrix} \mathcal{J}_{\xi_R} & 0_{3\times 3} & 0_{3\times 3} \\ Q_{\xi_R,\xi_v} & \mathcal{J}_{\xi_R} & 0_{3\times 3} \\ Q_{\xi_R,\xi_p} & 0_{3\times 3} & \mathcal{J}_{\xi_R} \end{bmatrix}, \tag{3.43}$$

$$\mathcal{J}_{\boldsymbol{\xi}}^{-1} \coloneqq \begin{bmatrix} \mathcal{J}_{\xi_R}^{-1} & 0_{3\times 3} & 0_{3\times 3} \\ -\mathcal{J}_{\xi_R}^{-1}Q_{\xi_R,\xi_v}\mathcal{J}_{\xi_R}^{-1} & \mathcal{J}_{\xi_R}^{-1} & 0_{3\times 3} \\ -\mathcal{J}_{\xi_R}^{-1}Q_{\xi_R,\xi_p}\mathcal{J}_{\xi_R}^{-1} & 0_{3\times 3} & \mathcal{J}_{\xi_R}^{-1} \end{bmatrix}, \tag{3.44}$$

where Q_{ξ_R,ξ_v} is computed as

$$\begin{aligned} Q_{\xi_R,\xi_{\mathrm{v}}} \coloneqq{}& \frac{1}{2}\xi_{v_\times} + c_1\left(\xi_{R_\times}\xi_{\mathrm{v}_\times} + \xi_{v_\times}\xi_{R_\times} + \xi_{R_\times}\xi_{\mathrm{v}_\times}\xi_{R_\times}\right) \\ &- \mathrm{c}_2\left(\xi_{R_\times}\xi_{R_\times}\xi_{\mathrm{v}_\times} + \xi_{\mathrm{v}_\times}\xi_{R_\times}\xi_{\mathrm{v}_\times} - 3\xi_{R_\times}\xi_{\mathrm{v}_\times}\xi_{R_\times}\right) \\ &+ c_3\left(\xi_{R_\times}\xi_{\mathrm{v}_\times}\xi_{R_\times}\xi_{R_\times} + \xi_{R_\times}\xi_{R_\times}\xi_{\mathrm{v}_\times}\xi_{R_\times}\right). \end{aligned} \tag{3.45}$$

The coefficients are calculated by $\mathrm{c}_1 = \frac{\|\xi_R\|-sin\|\xi_R\|}{\|\xi_R\|^3}$, $\mathrm{c}_2 = \frac{2-\|\xi_R\|^2-2cos\|\xi_R\|}{\|\xi_R\|^4}$, and

$c_3 = \frac{2\|\xi_R\| - 3sin\|\xi_R\| + \|\xi_R\| cos\|\xi_R\|}{2\|\xi_R\|^5}$. Q_{ξ_R,ξ_p} is calculated by replacing ξ_v by ξ_p.

3.3.2 Process model

The IMU dynamic model for flat Earth navigation is as follows

$$\dot{R}_t = R_t(\tilde{\omega} + w^g)_\times, \tag{3.46}$$

$$\dot{v}_t = R_t\ (\tilde{a} + w^a) + g, \tag{3.47}$$

$$\dot{p}_t = v_t. \tag{3.48}$$

Here, $\tilde{\omega}$ is the gyroscope output, $\tilde{a}$ is the accelerometer output, w^g is the white Gaussian noise of gyroscope output, w^a is the white Gaussian noise of accelerometer output, and g is the gravity vector. Assuming the flat Earth navigation, the gravity vector is a constant vector that has only an element in the local tangent vertical axis. For the state defined in the double direct isometries (3.38), the dynamic model is formulated by

$$\frac{\mathrm{d}}{\mathrm{dt}} X_t = \begin{bmatrix} R_t(\tilde{\omega})_\times & R_t\tilde{a} + g & v_t \\ 0_{1\times3} & 0 & 0 \\ 0_{1\times3} & 0 & 0 \end{bmatrix} + \begin{bmatrix} R_t & v_t & p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix} \begin{bmatrix} (w^g)_\times & w^a & 0_{3\times1} \\ 0_{1\times3} & 0 & 0 \\ 0_{1\times3} & 0 & 0 \end{bmatrix}. \tag{3.49}$$

Comparing (3.23) and (3.49),

$$f_{u_t}(X_t) = \begin{bmatrix} R_t(\tilde{\omega})_\times & R_t\tilde{a} + g & v_t \\ 0_{1\times3} & 0 & 0 \\ 0_{1\times3} & 0 & 0 \end{bmatrix},$$

$$w = \begin{bmatrix} w^g \\ w^a \\ 0_{3\times1} \end{bmatrix},$$

$$w^{\wedge} = \begin{bmatrix} (w^g)_{\times} & w^a & 0_{3\times1} \\ 0_{1\times3} & 0 & 0 \\ 0_{1\times3} & 0 & 0 \end{bmatrix}$$

are evident. To exploit the advantage of IEKF, the process model should be group affine, which means f_{u_t} satisfies (3.17).

The invariant error states of (3.38) are calculated by (3.15) and (3.16) as

$$\begin{aligned} \eta_t^L &= X_t^{-1}\hat{X}_t \\ &= \begin{bmatrix} R_t & v_t & p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \hat{R}_t & \hat{v}_t & \hat{p}_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} R_t^{-1}\hat{R}_t & R_t^{-1}(\hat{v}_t - v_t) & R_t^{-1}(\hat{p}_t - p_t) \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}, \end{aligned} \tag{3.50}$$

$$\begin{aligned} \eta_t^R &= \hat{X}_t X_t^{-1} \\ &= \begin{bmatrix} \hat{R}_t & \hat{v}_t & \hat{p}_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix} \begin{bmatrix} R_t & v_t & p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \hat{R}_t R_t^{-1} & \hat{v}_t - \hat{R}_t R_t^{-1} v_t & \hat{p}_t - \hat{R}_t R_t^{-1} p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}, \end{aligned} \tag{3.51}$$

where $\hat{X}_t$ is a state estimate of X_t and $\hat{R}_t$, $\hat{v}_t$, $\hat{p}_t$ are estimate of R_t, v_t, p_t respectively. Then, the linearized error differential equations for $\xi_t^L = \left[{\xi_{R_t}^L}^T, {\xi_{v_t}^L}^T, {\xi_{p_t}^L}^T\right]^T$ and $\xi_t^R = \left[{\xi_{R_t}^R}^T, {\xi_{v_t}^R}^T, {\xi_{p_t}^R}^T\right]^T$ are calculated as

$$\begin{aligned} \frac{d}{dt}\xi_t^L &= F^L\xi_t^L + G^L w \\ &= \begin{bmatrix} -(\tilde{\omega})_{\times} & 0_{3\times3} & 0_{3\times3} \\ -(\tilde{a})_{\times} & -(\tilde{\omega})_{\times} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & -(\tilde{\omega})_{\times} \end{bmatrix} \xi_t^L + \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \begin{bmatrix} w_a \\ w_g \end{bmatrix} \end{aligned} \tag{3.52}$$

$$\frac{d}{dt}\xi_t^R = F^R\xi_t^R + G^R w$$

$$= \begin{bmatrix} 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ (g)_\times & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} \end{bmatrix} \xi_t^R + \begin{bmatrix} 0_{3\times3} & \hat{R}_t \\ \hat{R}_t & (\hat{v}_t)_\times \hat{R}_t \\ 0_{3\times3} & (\hat{p}_t)_\times \hat{R}_t \end{bmatrix} \begin{bmatrix} w_a \\ w_g \end{bmatrix}. \quad (3.53)$$

Since the process model is group affine, the linearized error propagation matrices F^L and F^R are *state-trajectory independent*.

3.3.3 Measurement model

3.3.3.1 Left-invariant measurement

Most of the sensor outputs that measure the navigation/global frame values belong to the left-invariant measurement. Typical left-invariant measurement is the global navigation satellite system (GNSS) outputs. Usually, two types of integration are considered: loosely-coupled [34,35] and tightly-coupled [36]. Usually, the GNSS sensor outputs both position and velocity measurements for loosely-coupled systems and raw signals for the tightly-coupled system. For the position measurement, the measurement model is

$$z_t^{pos} = p_t + n_t^{pos}, \quad (3.54)$$

where $n_t^{pos} \sim N(0, R^{pos})$ is the measurement noise and R^{pos} is the noise covariance of position measurement. For $\mathbb{SE}_2(3)$ state, (3.54) becomes

$$z_t^{pos} = \Gamma\left(\begin{bmatrix} R_t & v_t & p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}\begin{bmatrix} 0_{3\times1} \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} n_t^{pos} \\ 0 \\ 0 \end{bmatrix}\right). \quad (3.55)$$

$\Gamma = [I_{3\times3} \quad 0_{3\times2}]$ is pre-multiplied to remove the unnecessary terms. Compared to

(3.28), the GNSS position is classified to a left-invariant measurement. Similarly, a position measurement in the global frame is also a left-invariant measurement. Since it is the left-invariant measurement, a proper invariant error should be a left-invariant error. Using the approximation $\eta_t^L \approx I + (\xi_t^L)^\wedge$, the innovation is linearized as follows.

$$
\begin{aligned}
\Gamma V_t^{pos} &= \Gamma \hat{X}_t^{-1}\left(\hat{z}_t^{pos} - z_t^{pos}\right) \\
&= \Gamma \hat{X}_t^{-1}\left(\hat{X}_t \begin{bmatrix} 0_{3\times1} \\ 0 \\ 1 \end{bmatrix} - X_t \begin{bmatrix} 0_{3\times1} \\ 0 \\ 1 \end{bmatrix} - \begin{bmatrix} n_t^{pos} \\ 0 \\ 0 \end{bmatrix} \right) \\
&\approx \Gamma \left(\begin{bmatrix} 0_{3\times1} \\ 0 \\ 1 \end{bmatrix} - (I - (\xi_t^L)^\wedge) \begin{bmatrix} 0_{3\times1} \\ 0 \\ 1 \end{bmatrix} - \hat{X}_t^{-1} \begin{bmatrix} n_t^{pos} \\ 0 \\ 0 \end{bmatrix} \right) \\
&= \Gamma \left((\xi_t^L)^\wedge \begin{bmatrix} 0_{3\times1} \\ 0 \\ 1 \end{bmatrix} - \hat{X}_t^{-1} \begin{bmatrix} n_t^{pos} \\ 0 \\ 0 \end{bmatrix} \right) \\
&= \Gamma \left(\begin{bmatrix} 0_{3\times3} & 0_{3\times3} & \mathrm{I}_{3\times3} \\ 0_{2\times3} & 0_{2\times3} & 0_{2\times3} \end{bmatrix} \xi_t^L - \begin{bmatrix} \hat{R}_t^T & -\hat{R}_t^T \hat{v}_t & -\hat{R}_t^T \hat{p}_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix} \begin{bmatrix} n_t^{pos} \\ 0 \\ 0 \end{bmatrix} \right) \\
&= [0_{3\times3} \quad 0_{3\times3} \quad I_{3\times3}]\xi_t^L - \hat{R}_t^T n_t^{pos}.
\end{aligned} \tag{3.56}
$$

Since the proper error is used, a left-invariant error for a left-invariant measurement, the measurement matrix $H^{pos} = [0_{3\times3} \quad 0_{3\times3} \quad I_{3\times3}]$ is *trajectory independent*. Although the attitude estimate affects the measurement noise, it would not affect the whole convergence property since it is bounded.

GNSS velocity measurement model is

$$z_t^{vel} = v_t + n_t^{vel}. \tag{3.57}$$

where $n_t^{vel} \sim N(0, R^{vel})$ is measurement noise and R^{vel} is noise covariance of velocity measurement. Similar to the position measurement, the measurement model

is a left-invariant model.

$$z^{vel} = \Gamma\left(\begin{bmatrix} R_t & v_t & p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}\begin{bmatrix} 0_{3\times1} \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} n_t^{vel} \\ 0 \\ 0 \end{bmatrix}\right)$$

In the same way, as in the case of position measurement, the innovation is calculated as

$$\begin{aligned} \Gamma V_t^{vel} &= \Gamma \hat{X}_t^{-1}\left(\hat{z}_t^{vel} - z_t^{vel}\right) \\ &\approx [0_{3\times3} \quad I_{3\times3} \quad 0_{3\times3}]\xi_t^L - \hat{R}_t^T n_t^{vel}, \end{aligned} \tag{3.58}$$

where the measurement matrix $H^{vel} = [0_{3\times3} \quad I_{3\times3} \quad 0_{3\times3}]$ is *trajectory independent*.

3.3.3.2 Right-invariant measurement

If the aiding sensor measures the state variables in sensor/body frame coordinate, it is likely a right-invariant measurement. For example, a wheel encoder attached to the wheel of a robot, outputs the velocity in the body frame [37,38]. For maritime applications, the doppler velocity log (DVL) measures the velocity of ships in DVL/body frame [39,40]. The number of axes for which the velocity is measured varies depending on the situation, but three axes velocity measurement is considered.

$$z_t^{odo} = R_t^T v_t + n_t^{odo}. \tag{3.59}$$

Here, "odo" means both wheel encoder and DVL. $n_t^{odo} \sim N(0, R^{odo})$ is measurement noise and R^{odo} is the noise covariance of velocity measurement. For $\mathbb{SE}_2(3)$ state, (3.59) becomes

$$z_t^{odo} = \Gamma\left(\begin{bmatrix} R_t & v_t & p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}^{-1}\begin{bmatrix} 0_{3\times1} \\ -1 \\ 0 \end{bmatrix} + \begin{bmatrix} n_t^{odo} \\ 0 \\ 0 \end{bmatrix}\right), \tag{3.60}$$

$\Gamma = [I_{3\times3} \quad 0_{3\times2}]$ is same with the left-invariant measurement model case. Compared to (3.29), (3.60) is a right-invariant measurement. The innovation is linearized as follows

$$\begin{aligned}
\Gamma V_t^{odo} &= \Gamma\hat{X}_t\left(\hat{z}_t^{odo} - z_t^{odo}\right) \\
&= \Gamma\hat{X}_t\left(\hat{X}_t^{-1}\begin{bmatrix} 0_{3\times1} \\ -1 \\ 0 \end{bmatrix} - X_t^{-1}\begin{bmatrix} 0_{3\times1} \\ -1 \\ 0 \end{bmatrix} - \begin{bmatrix} n_t^{odo} \\ 0 \\ 0 \end{bmatrix}\right) \\
&\approx \Gamma\left(\begin{bmatrix} 0_{3\times1} \\ -1 \\ 0 \end{bmatrix} - (I + (\xi_t^R)^\wedge)\begin{bmatrix} 0_{3\times1} \\ -1 \\ 0 \end{bmatrix} - \hat{X}_t\begin{bmatrix} n_t^{odo} \\ 0 \\ 0 \end{bmatrix}\right) \\
&= \Gamma\left((\xi_t^R)^\wedge\begin{bmatrix} 0_{3\times1} \\ 1 \\ 0 \end{bmatrix} - \hat{X}_t\begin{bmatrix} n_t^{odo} \\ 0 \\ 0 \end{bmatrix}\right) \\
&= \Gamma\left(\begin{bmatrix} 0_{3\times3} & \mathrm{I}_{3\times3} & 0_{3\times3} \\ 0_{2\times3} & 0_{2\times3} & 0_{2\times3} \end{bmatrix}\xi_t^R - \begin{bmatrix} \hat{R}_t & \hat{v}_t & \hat{p}_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}\begin{bmatrix} n_t^{odo} \\ 0 \\ 0 \end{bmatrix}\right) \\
&= [0_{3\times3} \quad I_{3\times3} \quad 0_{3\times3}]\xi_t^R - \hat{R}_t n_t^{odo}.
\end{aligned} \tag{3.61}$$

The measurement matrix $H^{odo} = [0_{3\times3} \quad I_{3\times3} \quad 0_{3\times3}]$ is *trajectory independent*.

Another example of the right-invariant measurement is the relative position of known features measured by a depth camera [11].

$$z_t^{known,i} = R_t^T\left(l^i - p_t\right) + n_t^{known}, \tag{3.62}$$

where $i = 1, \cdots, K$, K is number of known landmarks, $l^i \in \mathbb{R}^{3\times1}$ is the ith position of a known landmark, and $n_t^{known} \sim N(0, R^{known,i})$ and $R^{land,i}$ is the noise covariance of the position of the ith landmark. In this case, the measurement

model becomes

$$z_t^{known,i} = \Gamma\left(\begin{bmatrix} R_t & v_t & p_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} l^i \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} n_t^{known,i} \\ 0 \\ 0 \end{bmatrix}\right), \tag{3.63}$$

which is a right-invariant measurement. The innovation is linearized as

$$\begin{aligned} \Gamma V_t^{known,i} &= \Gamma \hat{X}_t \left(\hat{z}_t^{known,i} - z_t^{known,i}\right) \\ &= \Gamma \hat{X}_t \left(\hat{X}_t^{-1} \begin{bmatrix} l^i \\ 0 \\ 0 \end{bmatrix} - X_t^{-1} \begin{bmatrix} l^i \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} n_t^{known,i} \\ 0 \\ 0 \end{bmatrix}\right) \\ &= \Gamma \left(\begin{bmatrix} l^i \\ 0 \\ 0 \end{bmatrix} - (I + (\xi_t^R)^\wedge) \begin{bmatrix} l^i \\ 0 \\ 0 \end{bmatrix} - \hat{X}_t \begin{bmatrix} n_t^{known,i} \\ 0 \\ 0 \end{bmatrix}\right) \\ &= \Gamma \left(-(\xi_t^R)^\wedge \begin{bmatrix} l^i \\ 0 \\ 0 \end{bmatrix} - \hat{X}_t \begin{bmatrix} n_t^{known,i} \\ 0 \\ 0 \end{bmatrix}\right) \\ &= \Gamma \left(\begin{bmatrix} (l^i)_\times & 0_{3\times3} & 0_{3\times3} \\ 0_{2\times3} & 0_{2\times3} & 0_{2\times3} \end{bmatrix} \xi_t^R - \begin{bmatrix} \hat{R}_t & \hat{v}_t & \hat{p}_t \\ 0_{1\times3} & 1 & 0 \\ 0_{1\times3} & 0 & 1 \end{bmatrix} \begin{bmatrix} n_t^{known,i} \\ 0 \\ 0 \end{bmatrix}\right) \\ &= \begin{bmatrix} (l^i)_\times & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \xi_t^R - \hat{R}_t n_t^{known,i}, \end{aligned} \tag{3.64}$$

Augmenting (3.64) for all the landmark measurements, the linearized model becomes

$$\Gamma V_t^{known} = \begin{bmatrix} (l^1)_\times & 0_{3\times3} & 0_{3\times3} \\ \vdots & \vdots & \vdots \\ (l^K)_\times & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \xi_t^R - \begin{bmatrix} \hat{R}_t n_t^{known,1} \\ \vdots \\ \hat{R}_t n_t^{known,K} \end{bmatrix}. \tag{3.65}$$

Still, the measurement matrix $H^{known} = \begin{bmatrix} (l^1)_\times & 0_{3\times3} & 0_{3\times3} \\ \vdots & \vdots & \vdots \\ (l^K)_\times & 0_{3\times3} & 0_{3\times3} \end{bmatrix}$ is *trajectory*

independent. Often, the landmark's position is unknown. In this case, the position of landmarks is augmented to the state estimates and then estimated together. Using the special Euclidean group $\mathbb{SE}_{2+\mathrm{n}}(3)$ of (3.13), (3.65) is changed to

$$\Gamma V_t^{unknown} = \begin{bmatrix} 0_{3\times9} & \mathrm{I}_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ \vdots & \vdots & \vdots & \vdots \\ 0_{3\times9} & 0_{3\times3} & 0_{3\times3} & \mathrm{I}_{3\times3} \end{bmatrix} \xi_t^R - \begin{bmatrix} \hat{R}_t n_t^{unknown,1} \\ \vdots \\ \hat{R}_t n_t^{unknown,K} \end{bmatrix}, \tag{3.66}$$

where $\xi_t^R = \begin{bmatrix} \xi_{R_t}^T & \xi_{v_t}^T & \xi_{p_t}^T & \xi_{l_1}^T & \cdots & \xi_{l_K}^T \end{bmatrix}^{\mathrm{T}}$. The measurement matrix $H^{unknown}$is *trajectory independent*.

3.3.3.3 Other measurement models

Various measurements may not belong to either left-invariant or right-invariant measurement models. If it does not fit the model exactly, use the appropriate invariant error for the closest invariant model [37]. However, both LIEKF and RIEKF would have *state-trajectory dependent* measurement matrices since the model does not perfectly fit.

3.3.4 Adjoint transformation

For IMU states defined by (3.38), the adjoint transformation matrix is calculated as

$$Ad_{\hat{X}_t} = \begin{bmatrix} \hat{R}_t & 0_{3\times3} & 0_{3\times3} \\ (\hat{v}_t)_\times \hat{R}_t & \hat{R}_t & 0_{3\times3} \\ (\hat{p}_t)_\times \hat{R}_t & 0_{3\times3} & \hat{R}_t \end{bmatrix}. \tag{3.67}$$

Chapter 4

IEKF Using Multiple Measurements

In chapter 3, the overall framework of IEKF has been introduced. To fully exploit the advantage of the IEKF, two conditions are essential: 1) process model should be *group affine* system, 2) measurement should have the form of left- or right-invariant measurement. The IEKF will always be considered for the group affine process model. Therefore, the second condition is the key. Also, proper invariant error selection is required when the second condition is met. It is not a problem when the system uses only one measurement to correct the state estimate. However, navigation systems often use multiple aiding sensors to improve the estimation accuracy and ensure the robustness to the sensor outliers and the failures [41,42,43,44]. There is no problem if all the measurements belong to either the left- or right-invariant measurements. However, if both left- and right-invariant measurements are used, a problem occurs: what is a proper invariant error?

In this chapter, the centralized filter structure and the decentralized filter structure are introduced for the multi-measurement system. Especially, a federated filter structure is exploited to process the left- and right-invariant measurements separately. To compare the equations of both structure, two measurements are considered: 3D navigation frame position measurement and 3D body frame velocity measurement. 3D navigation frame position measurement of (3.55) is a left-invariant measurement. 3D body frame velocity measurement of (3.60) is a right-invariant

measurement. Then the measurement model becomes

$$z_t = \begin{bmatrix} z_t^{pos} \\ z_t^{odo} \end{bmatrix} = \begin{bmatrix} p_t \\ R_t^T v_t \end{bmatrix} + \begin{bmatrix} n_t^{pos} \\ n_t^{odo} \end{bmatrix} \tag{4.1}$$

4.1 Centralized filter implementation

The centralized filter structure processes all the aiding sensor measurements in the same filter (Figure 4.1 General centralized filter structurein [24]). The filter structure does not change when using only one aiding sensor and several. Since the update period also varies according to the sensor's sampling rate, even if multiple aiding sensors are used, only one measurement value may be used for each update. However, in this thesis, only the case where multiple sensor measurements are used for filter update simultaneously. In this case, the measurement matrix of multiple measurements is augmented into one and used for the update.

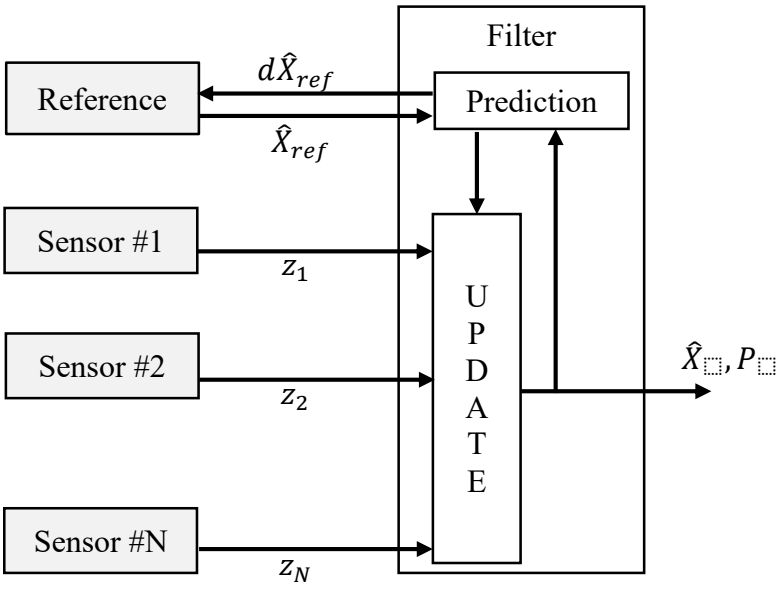

Figure 4.1 General centralized filter structure

Since there are both left- and right-invariant measurements, there is no best choice between the left- and right-invariant error. So, both LIEKF and RIEKF should be considered and compared. In the following sections, the linearized innovation models are derived for both cases.

4.1.1 Centralized LIEKF

The state (3.38), the left-invariant error (3.50), the linearized error, and the linearized process model (3.52) are revisited. Using the equations, the centralized LIEKF predicts the state estimate. To use two aiding sensor measurements, the linearized innovations of both measurements are augmented. The innovation model of the 3D body frame velocity measurement should be changed to augment the innovations since it is derived only for the right-invariant error. Using (3.34) and (3.67), the innovation model of 3D body frame velocity measurement for right-invariant error is derived as

$$\begin{aligned}
\Gamma V_t^{odo} &= [0_{3\times3} \quad I_{3\times3} \quad 0_{3\times3}]\xi_t^R - \hat{R}_t n_t^{odo} \\
&= [0_{3\times3} \quad I_{3\times3} \quad 0_{3\times3}]\ Ad_{\hat{X}_t}\xi_t^L - \hat{R}_t n_t^{odo} \\
&= [0_{3\times3} \quad I_{3\times3} \quad 0_{3\times3}]\begin{bmatrix} \hat{R}_t & 0_{3\times3} & 0_{3\times3} \\ (\hat{v}_t)_\times \hat{R}_t & \hat{R}_t & 0_{3\times3} \\ (\hat{p}_t)_\times \hat{R}_t & 0_{3\times3} & \hat{R}_t \end{bmatrix}\xi_t^L - \hat{R}_t n_t^{odo} \\
&= [(\hat{v}_t)_\times \hat{R}_t \quad \hat{R}_t \quad 0_{3\times3}]\xi_t^L - \hat{R}_t n_t^{odo}, \qquad (4.2)
\end{aligned}$$

where the measurement matrix of 3D body frame velocity measurement derived for the left-invariant error is defined as

$$H^{odo,L} = [(\hat{v}_t)_\times \hat{R}_t \quad \hat{R}_t \quad 0_{3\times3}]. \qquad \text{f(4.3)}$$

Here, the superscript odo means that it is a measurement matrix of 3D body frame

velocity measurement and L means that the measurement matrix is derived for left-invariant error. Superscript L is added since it is derived with inappropriate invariant error. If there is no L or R in the measurement matrix, it means that the measurement matrix is derived with proper invariant error. It can be seen that the measurement matrix includes the current state estimate because, unlike when derived for the right-invariant error, the left-invariant error is not an error proper for the right-invariant measurement model.

Using (3.56) and (4.2), the augmented linearized innovation for the 3D navigation frame position and the 3D body frame velocity measurement derived for left-invariant error is as follows

$$\begin{aligned} \Gamma V_t^L &= \begin{bmatrix} H^{pos} & \xi_t^L - \hat{R}_t^T n_t^{pos}, \\ H^{odo,L} & \xi_t^L - \hat{R}_t n_t^{odo} \end{bmatrix} \\ &= \begin{bmatrix} 0_{3\times3} & 0_{3\times3} & I_{3\times3} \\ (\hat{v}_t)_\times \hat{R}_t & \hat{R}_t & 0_{3\times3} \end{bmatrix} \xi_t^L - \begin{bmatrix} \hat{R}_t^T n_t^{pos} \\ \hat{R}_t n_t^{odo} \end{bmatrix}. \end{aligned} \tag{4.4}$$

The measurement matrix $H^{LIEKF} = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} & I_{3\times3} \\ (\hat{v}_t)_\times \hat{R}_t & \hat{R}_t & 0_{3\times3} \end{bmatrix}$ clearly contains current state estimates, which means it is not *trajectory independent*. It is a contradictory situation because the most significant advantage of IEKF is *trajectory independent* Kalman gain calculation. However, it is not *trajectory independent* when implemented with the centralized filter structure.

It is necessary to compare it with the conventional EKF, whose original measurement matrix is not *trajectory independent*. For the conventional EKF, the state and the error state are as follows

$$x_t = [\phi_t^T, v_t^T, p_t^T]^T \tag{4.5}$$

$$e_t = \hat{x}_t - x_t = [\delta\phi_t^T, \delta v_t^T, \delta p_t^T]^T \tag{4.6}$$

where $\phi_t \in \mathbb{R}^3$ is the attitude, $v_t \in \mathbb{R}^3$ is the velocity and $p_t \in \mathbb{R}^3$ is the position of the vehicle. The error is defined in the vector space by the vector addition, where $\delta\phi_t = \log\left(\hat{R}_t R_t^T\right)^\vee \in \mathbb{R}^3$ is the attitude error, $\delta v_t = \hat{v}_t - v_t$ is the velocity error, and $\delta p_t = \hat{p}_t - p_t$ is the position error. Then, the linearized innovation equation of the 3D navigation frame position and the 3D body frame velocity measurement is derived as

$$\begin{aligned}\delta z_t &= \begin{bmatrix} \hat{z}_t^{pos} - z_t^{pos} \\ \hat{z}_t^{odo} - z_t^{odo} \end{bmatrix} = \begin{bmatrix} \hat{p}_t - p_t - n_t^{pos} \\ \hat{R}_t^T \hat{v}_t - R_t^T v_t - n_t^{odo} \end{bmatrix} \\ &\approx \begin{bmatrix} \delta p_t - n_t^{pos} \\ \hat{R}_t^T (\hat{v}_t)_\times \delta\phi_t + \hat{R}_t^T \delta v_t - n_t^{odo} \end{bmatrix} \\ &= \begin{bmatrix} 0_{3\times3} & 0_{3\times3} & I_{3\times3} \\ \hat{R}_t^T (\hat{v}_t)_\times & \hat{R}_t^T & 0_{3\times3} \end{bmatrix} e_t - \begin{bmatrix} n_t^{pos} \\ n_t^{odo} \end{bmatrix}. \end{aligned} \tag{4.7}$$

Here, a first-order approximation of $\hat{R}_t R_t^T \approx I + (\delta\phi_t)_\times$ is used. The measurement matrix $H^{EKF} = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} & I_{3\times3} \\ \hat{R}_t^T (\hat{v}_t)_\times & \hat{R}_t^T & 0_{3\times3} \end{bmatrix}$ is *trajectory dependent*. Since (4.4) and (4.7) have similar forms and contain the current state estimates, the state estimate errors would degrade the estimation performance.

4.1.2 Centralized RIEKF

The state (3.38), the left-invariant error (3.51), the linearized error, and the linearized process model (3.53) are revisited. Using the equations, the centralized RIEKF predicts the state estimate. In this case, the innovation equation of the 3D navigation position measurement should be changed using the same method in Section 4.1.1, as follows,

$$
\begin{aligned}
\Gamma V_t^{pos} &= [0_{3\times3} \quad 0_{3\times3} \quad I_{3\times3}]\xi_t^L - \hat{R}_t^T n_t^{pos} \\
&= [0_{3\times3} \quad 0_{3\times3} \quad I_{3\times3}] \; Ad_{\hat{X}_t}^{-1}\xi_t^R - \hat{R}_t^T n_t^{pos} \\
&= [0_{3\times3} \quad 0_{3\times3} \quad I_{3\times3}]\begin{bmatrix} \hat{R}_t^T & 0_{3\times3} & 0_{3\times3} \\ -\hat{R}_t^T(\hat{v}_t)_\times & \hat{R}_t^T & 0_{3\times3} \\ -\hat{R}_t^T(\hat{p}_t)_\times & 0_{3\times3} & \hat{R}_t^T \end{bmatrix}\xi_t^R - \hat{R}_t^T n_t^{pos} \\
&= [-\hat{R}_t^T(\hat{p}_t)_\times \quad 0_{3\times3} \quad \hat{R}_t^T]\xi_t^R - \hat{R}_t^T n_t^{pos} \qquad (4.8)
\end{aligned}
$$

where the measurement matrix of 3D navigation frame position measurement derived for right-invariant error is defined as

$$
H^{pos,R} = [-\hat{R}_t^T(\hat{p}_t)_\times \quad 0_{3\times3} \quad \hat{R}_t^T]. \qquad (4.9)
$$

Using (3.61) and (4.9), the augmented linearized innovation for 3D navigation frame position and 3D body frame velocity measurement derived for right-invariant error is as follows

$$
\begin{aligned}
\Gamma V_t^R &= \begin{bmatrix} H^{pos,R} & \xi_t^R - \hat{R}_t^T n_t^{pos}, \\ H^{odo} & \xi_t^R - \hat{R}_t n_t^{odo} \end{bmatrix} \\
&= \begin{bmatrix} -\hat{R}_t^T(\hat{p}_t)_\times & 0_{3\times3} & \hat{R}_t^T \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} \end{bmatrix}\xi_t^L - \begin{bmatrix} \hat{R}_t^T n_t^{pos} \\ \hat{R}_t n_t^{odo} \end{bmatrix}. \qquad (4.10)
\end{aligned}
$$

The measurement matrix $H^{RIEKF} = \begin{bmatrix} -\hat{R}_t^T(\hat{p}_t)_\times & 0_{3\times3} & \hat{R}_t^T \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} \end{bmatrix}$ clearly contains current state estimates terms related to the 3D position measurement. From (4.4) and (4.10), it can be said that using a centralized filter structure to implement the IEKF for a multi-measurement system causes IEKF to lose its advantage, *trajectory independent* estimation. The centralized filter structure is depicted in Figure 4.2.

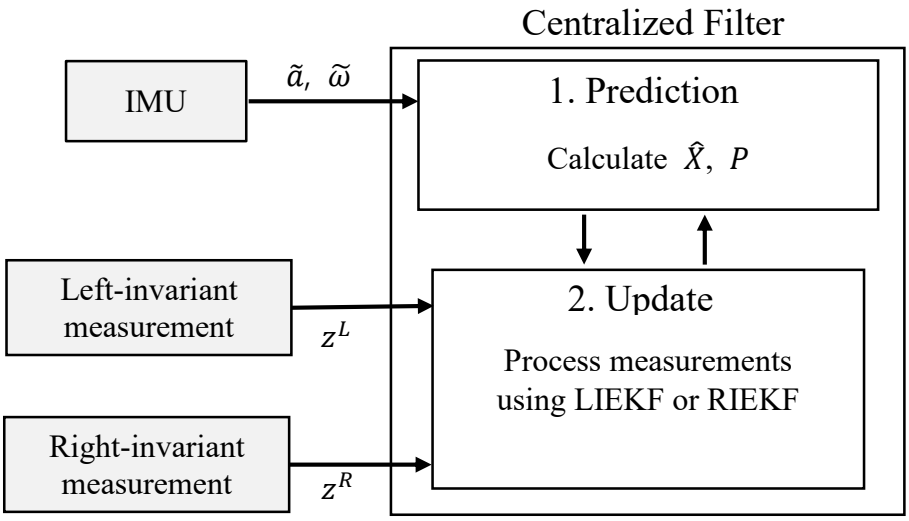

Figure 4.2 Centralized filter structure

4.2 Federated filter implementation

This section exploits the decentralized filter structure to design IEKF without losing the *trajectory independent* property. Since previous studies on IEKF usually assumed a single aiding sensor case, it is the first time introducing a decentralized filter structure to IEKF. Mainly, this thesis proposes a structure based on the federated filter structure. It is because federated filter requires information sharing stage which ensures the optimality of the estimation [24,25].

4.2.1 Overall structure

The main difference between the federated filter and other decentralized or centralized filters is that information sharing exists. The information sharing is required to preserve the total information of IMU. Consider the centralized filter structure. The IMU data is used to predict the state and error covariance in the filter.

It is not used repeatedly. On the other hand, in the federated filtering, the IMU data is used repeatedly in local filters. Then, the outputs of the local filters will be fused in the master filter. As a result, the information sharing is needed to conserve the total IMU information.

This information sharing is usually done by applying $1/\beta_i$ to the predicted *a priori* error covariances P^-. To reset the estimates and the error covariance of the local filters before the measurement update process, ith local filter is reset with *a priori* state estimate $\hat{x}^-$ and *a priori* inflated error covariance $(1/\beta_i)P^-$. Here, β_i satisfies the following equation

$$\sum_{i=1}^{L} \beta_i = 1 \tag{4.11}$$

where L is the number of the local filter. The information sharing and the reset of the local filter are depicted in Figure 2.2.

Suppose there are two local filters; local filter 1 (LF1), which handles the left-invariant measurement, and local filter 2 (LF2), which handles the right-invariant measurement. To use the proper invariant error, LF1 is designed with the LIEKF, and LF2 is designed with the RIEKF. Both invariant measurements will be processed through this separate handling of the left- and right-invariant measurements without losing the *trajectory independent* property. The overall structure of the modified federated filter structure to fuse the LIEKF and the RIEKF is depicted in Figure 4.3. The overall process is as follows:

1) In the master filter, *a priori* state $\hat{X}_m$ and covariance P_m^R are predicted using IMU data. In general, since the sampling rate of the IMU is higher than the measurement update rate,the prediction using only the IMU data continues until the measurement update. Here,

prediction is performed using the RIEKF's error and linearized process model. However, the LIEKF can be used if necessary, and the calculation related to the RIEKF should be changed to that of the LIEKF for the entire process (1. Prediction in Figure 4.3).

2) For the aiding sensor measurement update, the local filter is reset using the state estimate and covariance predicted by the master filter. Since LF1 uses the LIEKF, the covariance for the right-invariant error P_m^R is converted into the covariance for the left-invariant error P_m^L using the adjoint transform $P_m^L = Ad_{\hat{X}_m}^{-1} P_m^R Ad_{\hat{X}_m}$ (2. Transformation in Figure 4.3). The state estimate is independent of the invariant error, so no transformation is required. Then, after dividing the covariance by the information sharing factor β_i, the state and covariance of each local filter are initialized. Here, $\beta_1 = 0.5$ and $\beta_2 = 0.5$ are used.

3) The update is performed using the left-invariant measurement in LF1. In the case of the 3D navigation frame position measurement, (3.56) is used. If there are two or more left-invariant measurements, the measurement matrix is augmented and updated at once like the centralized filter. Since all the measurements are left-invariant, the augmented measurement matrix is *trajectory independent* (3. LIEKF update in Figure 4.3). *A posteriori* estimate of the state and the error covariance from LF1 is denoted as $\hat{X}^L$ and P^L.

4) In the same way in 3), LF2 processes all the right-invariant measurements (4. RIEKF update in Figure 4.3). *A posteriori* estimate of the state and the error covariance from LF2 are denoted as $\hat{X}^R$ and

P^R.

5) Before step 4), the state estimates of the two local filters are the same, but after step 4), the state estimates of the two local filters, $\hat{X}^L$ and $\hat{X}^R$, are different. To fuse the outputs of the local filters, P^L should be converted back to the covariance for the right-invariant error P^{R*} using $P^{R*} = Ad_{\hat{X}^L} P^L Ad_{\hat{X}^L}^{-1}$ (5. Transformation in Figure 4.3).
6) In the master filter, $\hat{X}^L$ with P^{R*} and $\hat{X}^R$ with P^R are fused (6. Fusion in Figure 4.3). The fused estimate $\hat{X}_{fus}$ and P_{fus}^R are used for the prediction of the next step.

The adjoint transformation requires a state estimate. Nevertheless, since each measurement is processed with *trajectory independent* Kalman gain, it can be said that the proposed structure exploits the full advantage of IEKF.

Conventional federated or decentralized filters use fusion equations like

$$P_{fus}^{-1} = \sum_{i=1}^{L} P_i^{-1}, \tag{4.12}$$

$$\hat{x}_{fus} = P_{fus} \sum_{i=1}^{L} P_i^{-1} \hat{x}_i, \tag{4.13}$$

where $\hat{x}_i$, P_i are outputs of ith local filter and $\hat{x}_{fus}$, P_{fus} are fused estimates of state and covariance. This conventional fusion method cannot be used for IEKF. Details about the fusion method of Lie group poses are introduced in the next section.

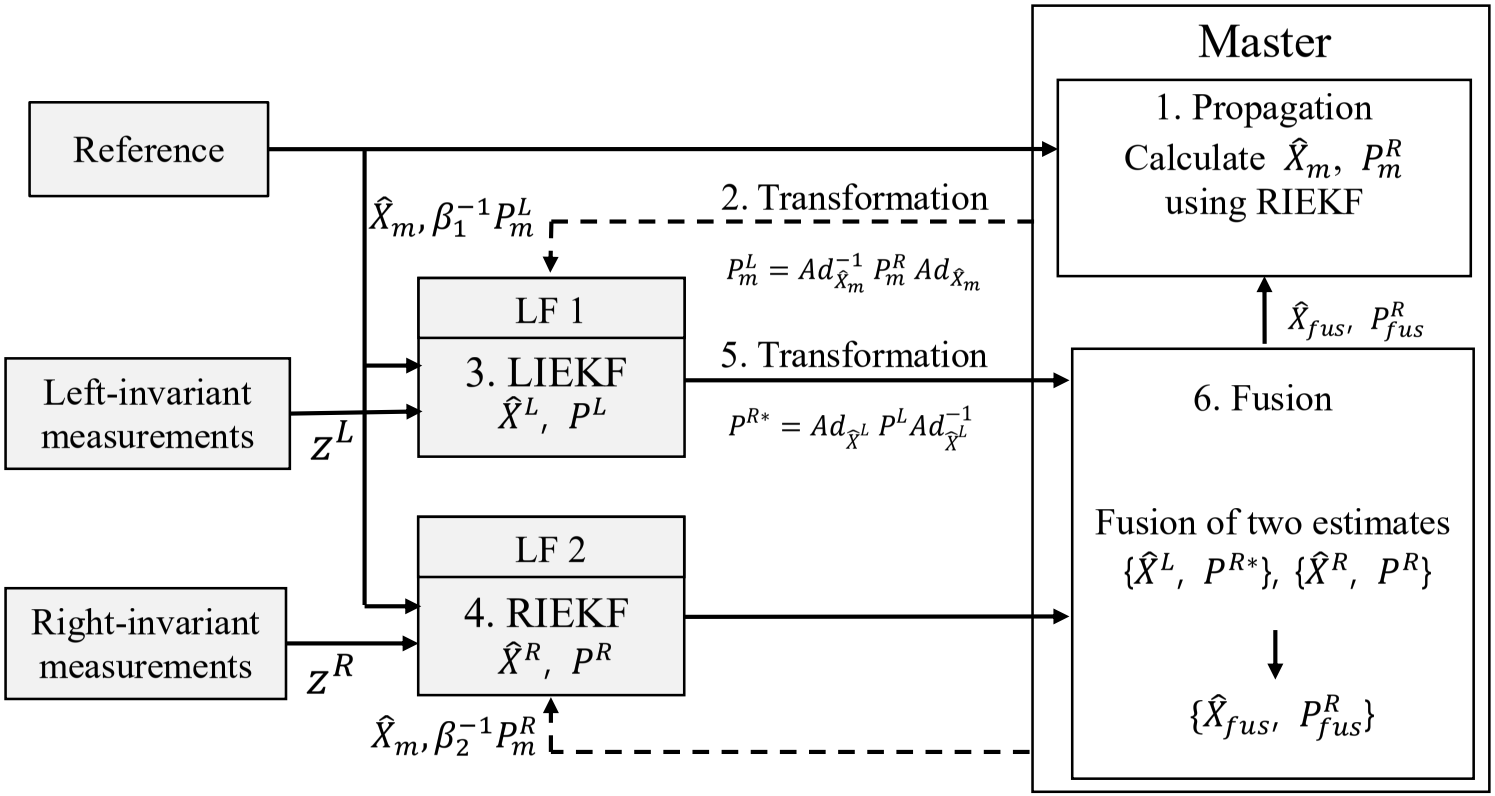

Figure 4.3 Proposed federated IEKF structure

4.2.2 Fusion process

As described in [16], poses defined on matrix Lie group and associated covariance cannot be fused using (4.12) and (4.13). It is because the error definition is changed. Unlike the traditional cases, the IEKF defined the error as left- or right-invariant. And the linearized error ξ in vector space is assumed to have white Gaussian distribution, $\xi \sim N(0, P_\xi)$. So, instead of (4.12) and (4.13), the iteration method fusing K poses on $\mathbb{SE}(3)$ and associated uncertainties is proposed in [16]. The method can be extended to $\mathbb{SE}_2(3)$ cases.

Define the errors of two local filters as follows

$$\begin{aligned}\epsilon^i &\coloneqq \log\left(\hat{X}_{opt}\ \hat{X}^{i^{-1}}\right)^\vee = \log\left(\exp(\xi^\wedge)\hat{X}_{fus}\hat{X}^{i^{-1}}\right)^\vee \\ &= \log\left(\exp(\xi^\wedge)\exp\left(\xi^{i^\wedge}\right)\right)^\vee \\ &\approx \xi^i + \mathcal{J}_i^{-1}\xi \end{aligned} \tag{4.14}$$

where $\{\hat{X}^L, P^{R*}\}$ and $\{\hat{X}^R, P^R\}$ are denoted as $\{\hat{X}^i, P^i\}$ for $i \in \{L\ , R\}$, the error is defined as $\epsilon^i \sim N(0, P^i)$, $\hat{X}_{opt}$ is the optimal state estimate, $\hat{X}_{fus}$ is the current guess, ξ is the perturbation between $\hat{X}_{opt}$ and $\hat{X}_{fus}$, and ξ^i is the difference between $\hat{X}_{fus}$ and $\hat{X}^i$. $\mathcal{J}_i^{-1}$ is the inverse Jacobian for ξ^i, calculated by (3.44).

The cost function is defined as

$$\mathrm{V} = \frac{1}{2}(\epsilon^{L^T} P^{L^{-1}} \epsilon^L + \epsilon^{R^T} P^{R^{-1}} \epsilon^R), \tag{4.15}$$

which is quadratic in ξ. Taking the derivative for ξ and setting to zero results in following

$$\left(\sum_{i=L,R} \mathcal{J}_i^{-T} P^{i^{-1}} \mathcal{J}_i^{-1}\right)\xi = -\sum_{i=L,R} \mathcal{J}_i^{-T} P^{i^{-1}} \xi^i. \tag{4.16}$$

The optimal perturbation ξ calculated from (4.16) is then applied to $\hat{X}_{fus}$,

$$\hat{X}_{fus} = \exp(\xi^{\wedge})\hat{X}_{fus}. \tag{4.17}$$

This calculation is repeated until the convergence. At the last iteration, the fused state $\hat{X}_{fus}$, and fused error covariance is calculated by

$$P_{fus}^{R} = \left(\sum_{i=L,R} \mathcal{J}_i^{-T} P^{i^{-1}} \mathcal{J}_i^{-1}\right)^{-1}. \tag{4.18}$$

The method described through (4.14)-(4.18) is the same as that of [16]. But, in this section, it is extended from $\mathbb{SE}(3)$ to $\mathbb{SE}_2(3)$ based on [33]. The original method for poses and uncertainties defined in the right-invariant form can be changed to fuse the poses defined in the left-invariant form if the master filter uses the LIEKF.

4.3 Numerical simulations

This section applies the proposed fusion method to a 3D inertial navigation example. As mentioned earlier, INS is compensated by two aiding measurements; the 3D navigation frame position and the 3D body frame velocity measurement. To verify the estimation performance of the proposed method, Monte-Carlo simulations are performed. The trajectory and the sensor measurements are generated according to the noise statistics from Table 4.1. The sampling rate of IMU is 100Hz, while the aiding sensors are 10Hz. The vehicle moves at the forward speed of 5m/s for 60 seconds along the 3D spiral trajectory described in Figure 4.4.

The information about the initial error and its distribution is generally given in the navigation frame. Since the initial error is defined in the navigation frame, the

proper transformation is required to set the initial error covariance. In addition, the performance comparison between LIEKF and RIEKF can be made by adding the error defined in the navigation frame. So, the initial errors are added to the true values using Table 4.2. The initial attitude error ξ_R^0 is added by $\hat{R} = \exp\left(\xi_R^{0\wedge}\right) R$, whereas the velocity and the position error δX are added by $\hat{X} = X + \delta X$.

The transformation from the navigation frame to the invariant-error space is required to set a proper initial covariance. It can be performed by first-order approximation of the invariant error. For example, the navigation frame error can be calculated from the left-invariant error using

$$\begin{bmatrix} \delta\phi_t \\ \delta v_t \\ \delta p_t \end{bmatrix} \approx \begin{bmatrix} \hat{R}_t & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \hat{R}_t J_{R_t} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & \hat{R}_t J_{R_t} \end{bmatrix} \begin{bmatrix} \xi_{R_t}^L \\ \xi_{v_t}^L \\ \xi_{p_t}^L \end{bmatrix}, \tag{4.19}$$

where J_{R_t} is a left Jacobian of $\mathbb{SO}(3)$. Through (4.19), the initial error covariance represented in the navigation frame can be transformed to the error covariance in the left-invariant error frame. Similarly, the navigation frame error can be calculated from the right-invariant error using

$$\begin{bmatrix} \delta\phi_t \\ \delta v_t \\ \delta p_t \end{bmatrix} \approx \begin{bmatrix} I_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ -(\hat{v}_t)_\times & J_{R_t} & 0_{3\times3} \\ -(\hat{p}_t)_\times & 0_{3\times3} & J_{R_t} \end{bmatrix} \begin{bmatrix} \xi_{R_t}^R \\ \xi_{v_t}^R \\ \xi_{p_t}^R \end{bmatrix}. \tag{4.20}$$

With (4.20), the initial error covariance represented in the navigation frame can be transformed to the one in the right-invariant error frame. So, the initial error covariance setting for the IEKF can be adequately done [37].

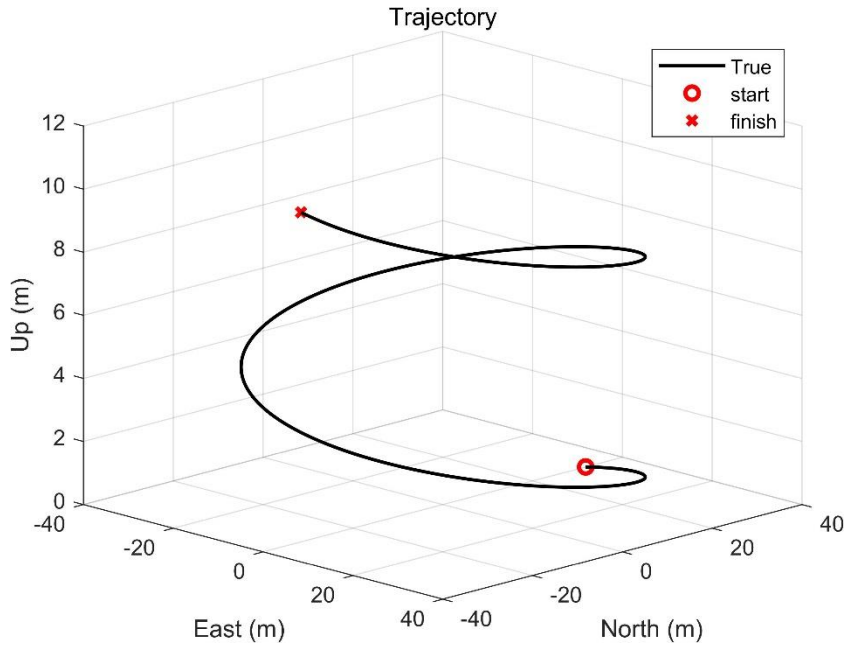

Figure 4.4 Simulation trajectory

Table 4.1 Simulation noise statistics

Measurement	Noise 1σ
Gyro ARW	$3 \times 10^{-4} rad/\sqrt{Hz}$
Accelerometer VRW	$3 \times 10^{-4} m/\sec^2 \ /\sqrt{Hz}$
3D position	5m
3D velocity	0.2m/s

Table 4.2 Initial error standard deviation

	Case	Position	Velocity	Attitude
Initial error 1σ	A	2.5m	0.1m/s	15°
	B	5.0m	0.2m/s	30°
	C	7.5m	0.3m/s	45°
	D	10.0m	0.4m/s	60°

The errors are calculated and compared in the navigation frame. The attitude error is calculated by $\xi = \log(\hat{X}X^{-1})^{\vee}$. The velocity and position error is calculated by vector subtraction $\hat{X} - X$. An error is more intuitive and interpretable in the navigation frame than in the Lie algebra space. Then, root-mean-square error (RMSE) will be calculated.

4.3.1 Convergence test

Since IEKF has its strength in error convergence property, many studies evaluated the error convergence using Monte Carlo simulation [17,23,37]. In this thesis, 100 Monte Carlo simulations are performed for the convergence comparison. The initial error covariance and initial state estimate are initialized using the values of case 3 in Table 4.2. Every ensemble uses the same sensor measurements set, but the initial state estimates are different. The convergence test results are plotted in Figures 4.5, 4.6, 4.7, and 4.8.

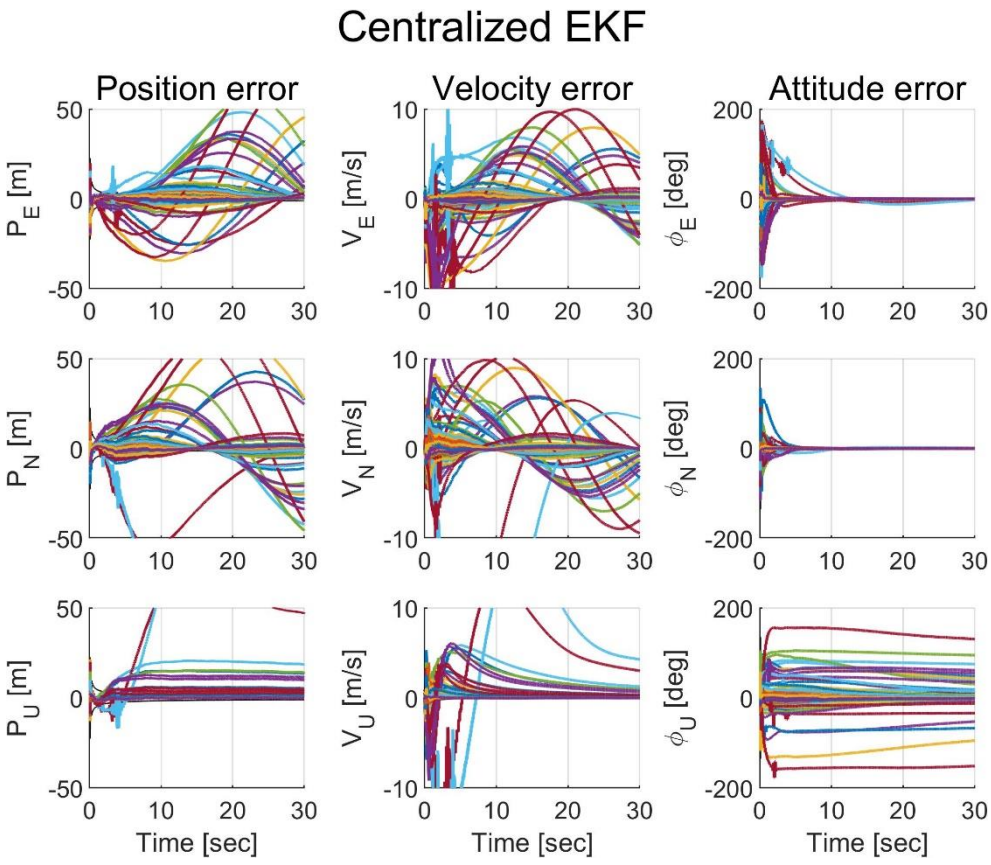

Figure 4.5 Error ensembles of centralized EKF for case C

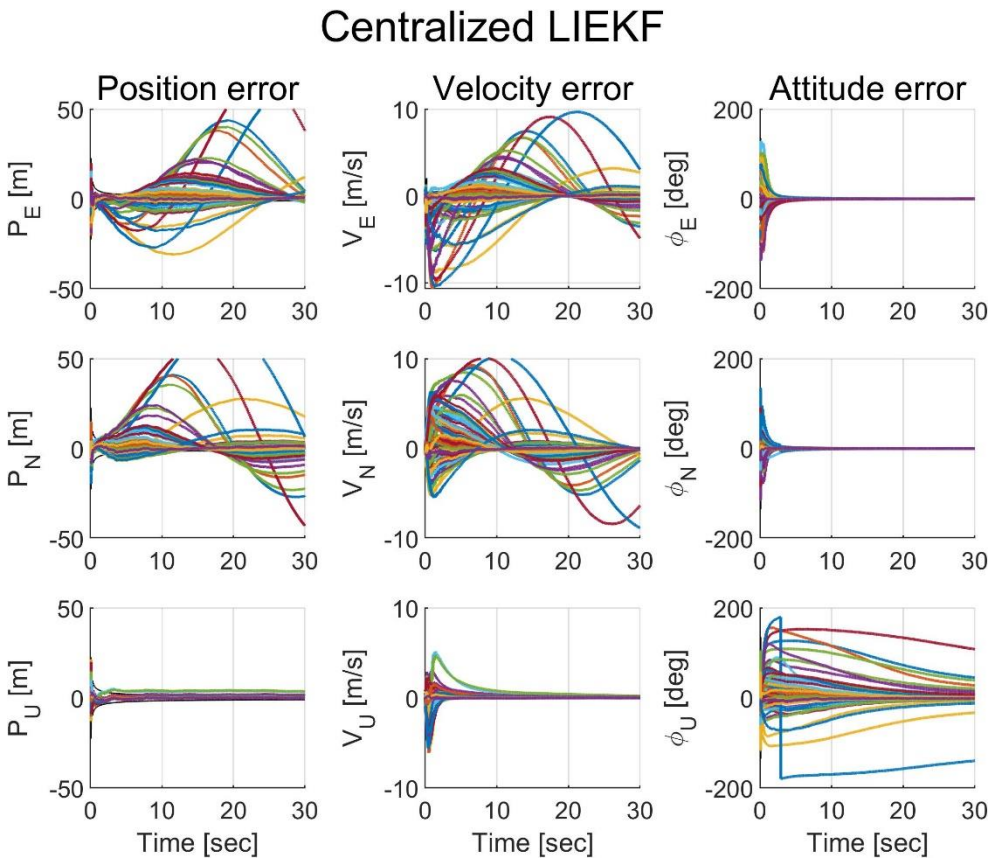

Figure 4.6 Error ensembles of centralized LIEKF for case C

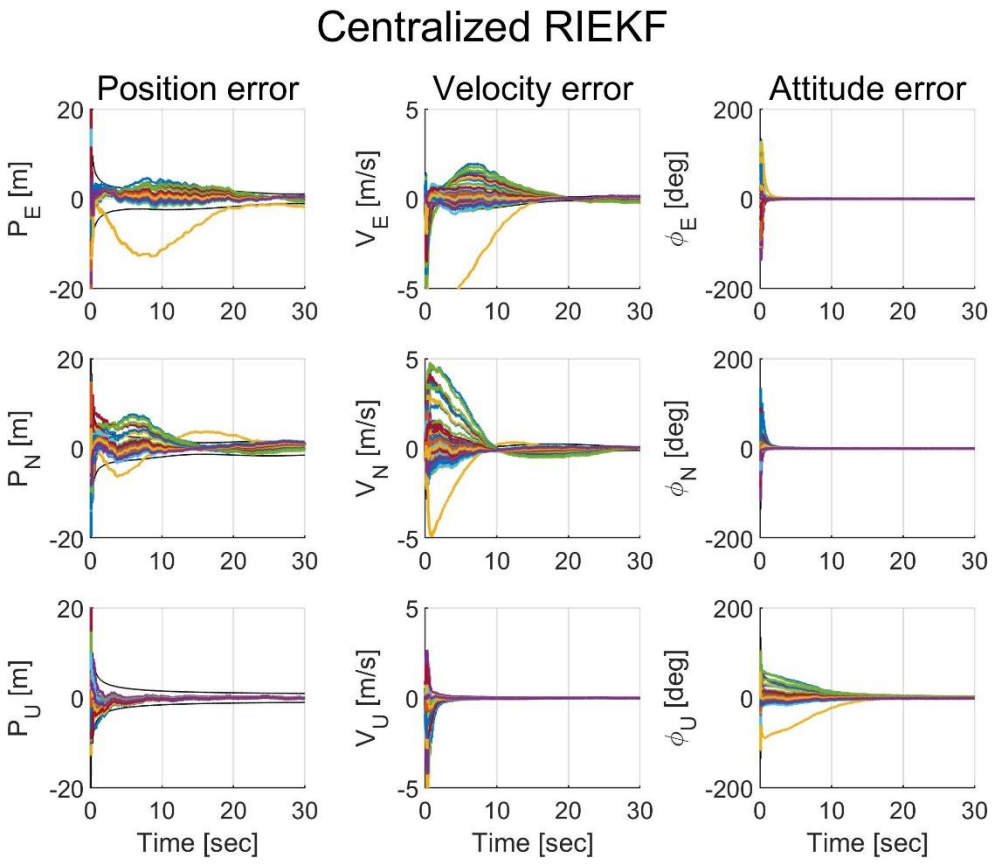

Figure 4.7 Error ensembles of centralized RIEKF for case C

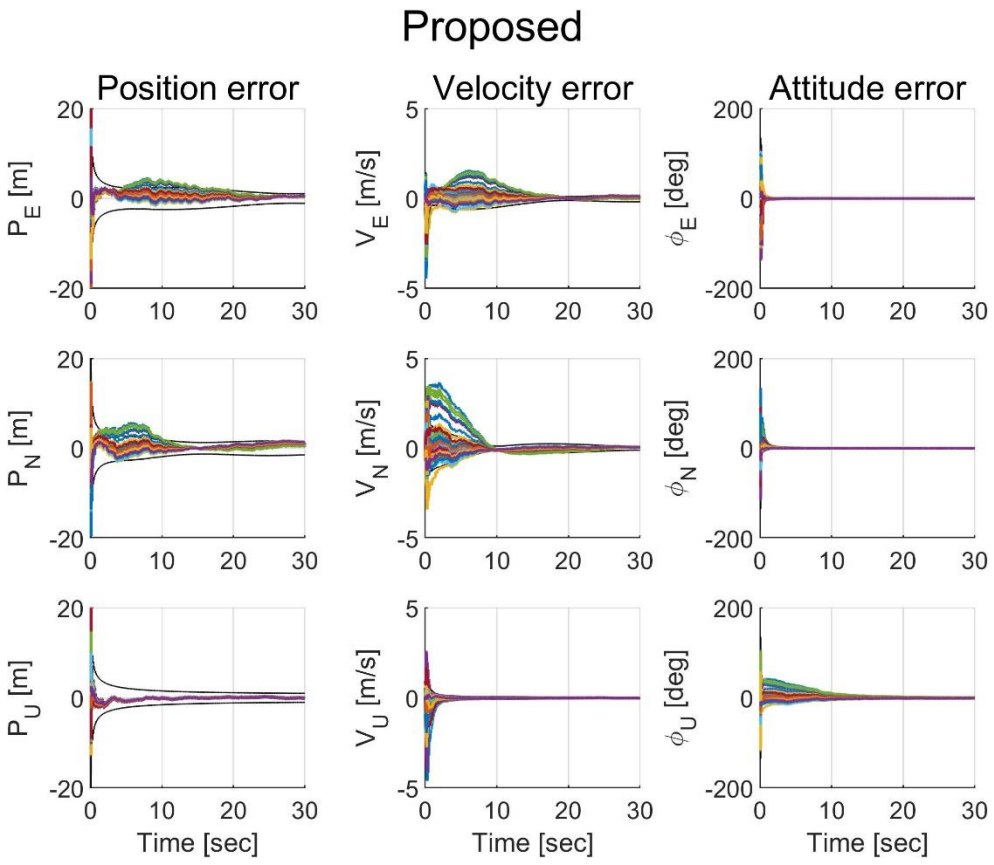

Figure 4.8 Error ensembles of proposed method for case C

The centralized EKF and LIEKF have troubles in error convergence. In many ensembles, error diverges or very slowly converges. On the other hand, the centralized RIEKF and the proposed method show perfect error convergence. Especially, it can be seen that the ensemble that did not converge quickly in the centralized RIEKF converges quickly in the proposed method. Also, from the fact that the distribution between each ensemble is denser in the proposed method, it can be seen that the error convergence of the proposed method is the best.

4.3.2 Comparison of centralized IEKF and EKF

Since the centralized IEKF uses *trajectory dependent* matrix similar to that of EKF, it is necessary to check whether there is a performance difference between EKF and IEKF. In invariant Kalman filtering, although the measurement matrix is *trajectory dependent*, the propagation of linearized error by the process model is made without linearization error. Since the propagation of the error is also affected by the linearization error in the conventional EKF, it can be predicted that there will be a performance difference between the IEKF and the EKF only by the difference in the prediction step.

To compare EKF and IEKF, 1,000 Monte Carlo simulations are performed, and RMSE is calculated for position, velocity, and attitude. Results of initial error cases A and B from Table 4.2 are compared.

4.3.2.1 Initial error case A

Table 4.3 compares the average RMSE of EKF, LIEKF, and RIEKF. It can be seen that both LIEKF and RIEKF have improved errors compared to EKF. Also, it can be seen that RIEKF has a larger error improvement rate than LIEKF among

IEKFs.

Table 4.3 Average RMSE comparison for case A

	Position [m]	Velocity [m/s]	Attitude [deg]
EKF (improvement)	0.4838 (0%)	0.0490 (0%)	0.4977 (0%)
LIEKF (improvement)	0.4703 (2.79%)	0.0477 (2.65%)	0.4888 (1.79%)
RIEKF (improvement)	0.4690 (3.06%)	0.0452 (8.41%)	0.4591 (7.76%)

Table 4.4 ANEES comparison for case A

	Position	Velocity	Attitude	Total
EKF	1.0653	1.2399	1.1731	1.1216
LIEKF	1.0046	1.0666	1.0719	1.0431
RIEKF	0.9996	1.0115	1.0180	1.0139

Table 4.4 compares the three methods' time-averaged average normalized estimation error squared (ANEES). ANEES is calculated by

$$\text{ANEES}(t_k) = \frac{1}{MN}\sum_i \xi_i(t_k)^T P_i(t_k)^{-1}\xi_i(t_k),$$

where M is the number of ensembles, N is the dimension of states, $\xi_i(t_k)$ is the error at the time t_k, and $P_i(t_k)$ is the error covariance of the filter at the time t_k. The closer the value of ANEES to 1, the better the filter consistency. From Table 4.4, it is evident that LIEKF and RIEKF have better consistency than EKF.

4.3.2.2 Initial error case B

From Figure 4.9 to Figure 4.11, RMSE for more extensive initial error conditions, case B, are compared. Since the small error assumption of the EKF is broken, the performance of the EKF severely deteriorates. In some ensembles, errors cannot converge to zero. In the case of IEKF, although the prediction step is not affected by the estimation error, *trajectory dependent* measurement matrices affect the estimation accuracy. In Table 4.5, compared to the degradation of the EKF estimation performance due to the large initial error, the degradation of the IEKF is insignificant, and the improvement of the estimation performance of the IEKF compared to the EKF is confirmed. The error improvement of 60~90% confirms that the performance of the IEKF is still superior to that of the EKF even when the *trajectory dependent* measurement matrix was used.

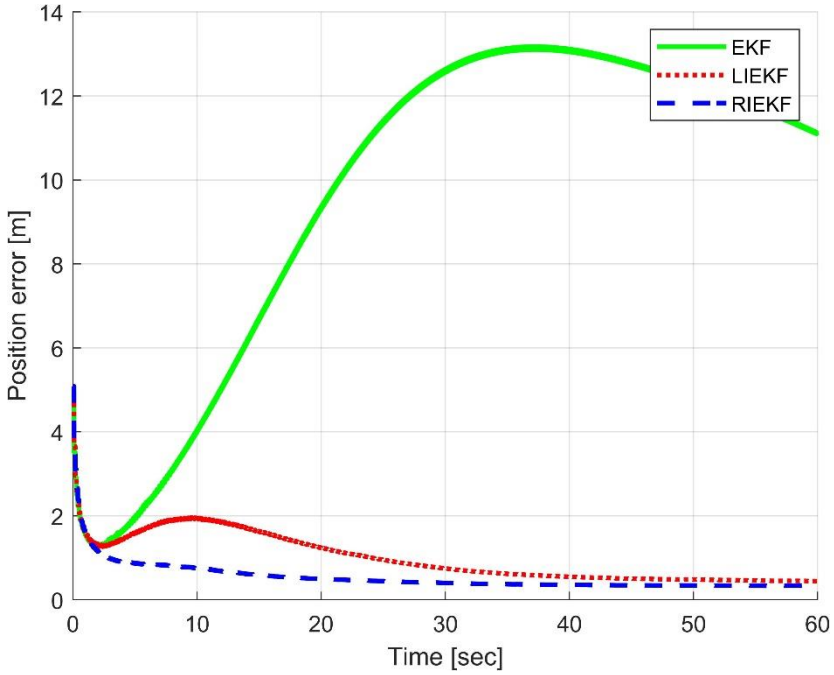

Figure 4.9 Position RMSE comparison of EKF, LIEKF, and RIEKF for case B

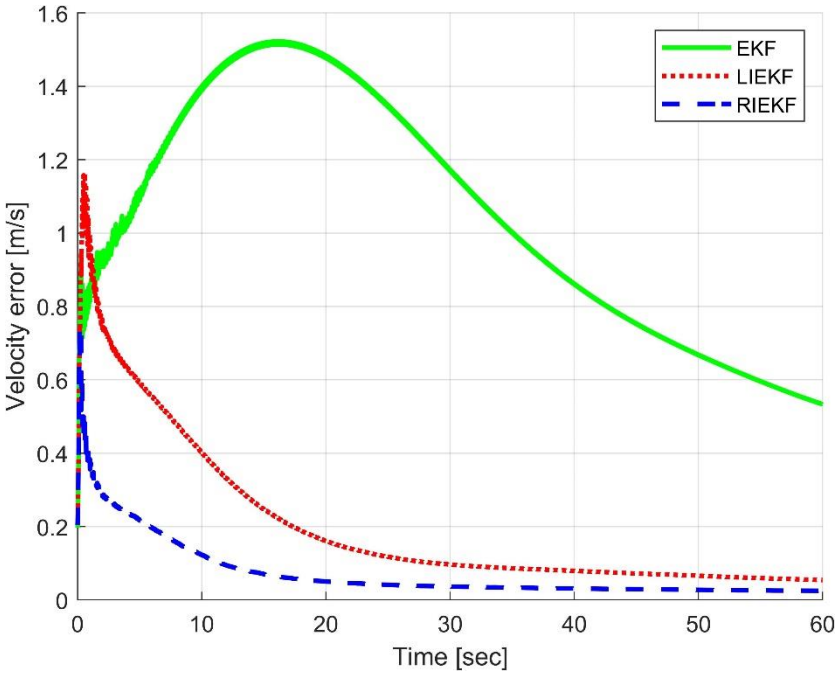

Figure 4.10 Velocity RMSE comparison of EKF, LIEKF, and RIEKF for case B

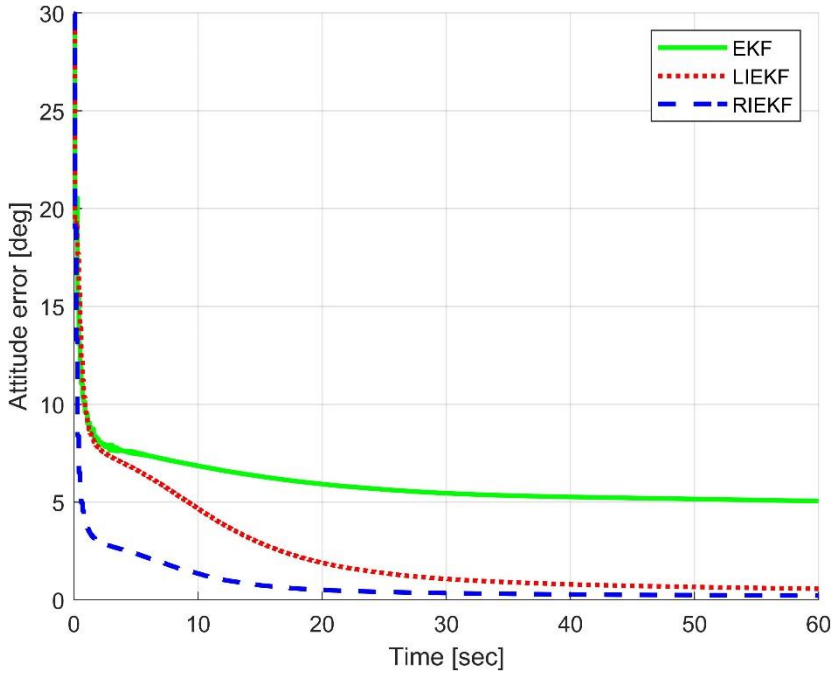

Figure 4.11 Attitude RMSE comparison of EKF, LIEKF, and RIEKF for case B

Table 4.5 Average RMSE comparison for case B

	Position [m]	Velocity [m/s]	Attitude [deg]
EKF (improvement)	9.6173 (0%)	1.0475 (0%)	5.9889 (0%)
LIEKF (improvement)	0.9862 (89.75%)	0.2032 (80.60%)	2.3545 (60.69%)
RIEKF (improvement)	0.5338 (94.45%)	0.0741 (92.93%)	0.8467 (85.86%)

Table 4.6 ANEES comparison for case B

	Position	Velocity	Attitude	Total
EKF	1,019	6,149	2,665	4,537
LIEKF	3.5010	11.5184	10.7975	9.3956
RIEKF	1.0747	1.3134	1.4376	1.2174

Interestingly, the difference in estimation performance between the LIEKF and the RIEKF is clear. From the average RMSE in Table 4.5 and ANEES in Table 4.6, it can be said that the RIEKF has a more consistent and accurate estimation performance than the LIEKF.

4.3.3 Comparison of IEKF and the proposed method

To evaluate the performance of the proposed method, RMSE of 1,000 Monte Carlo runs are compared for four initial error cases. The main point is to validate whether the proposed method can improve estimation performance compared to RIEKF.

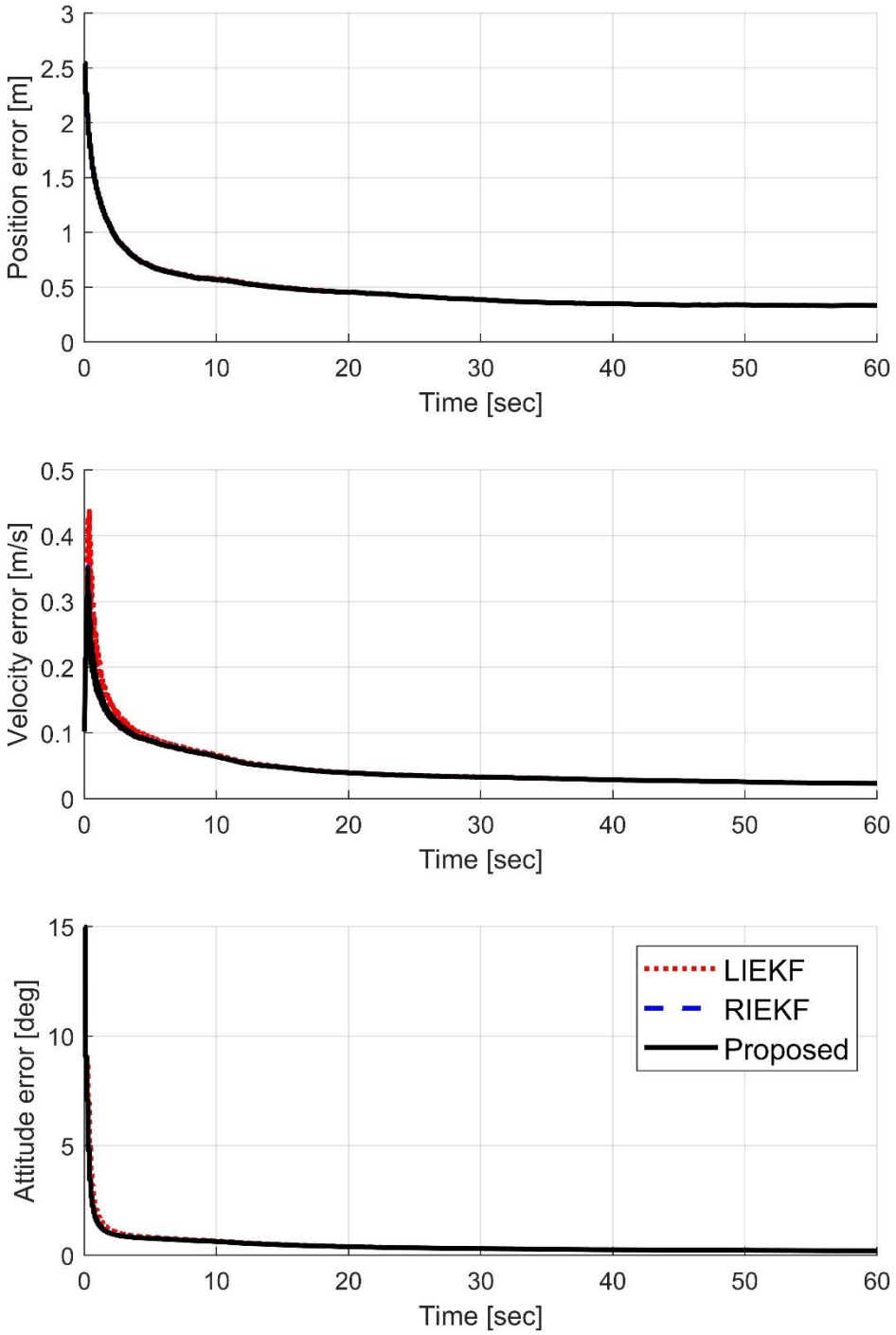

Figure 4.12 Comparison of IEKF and the proposed method for case A

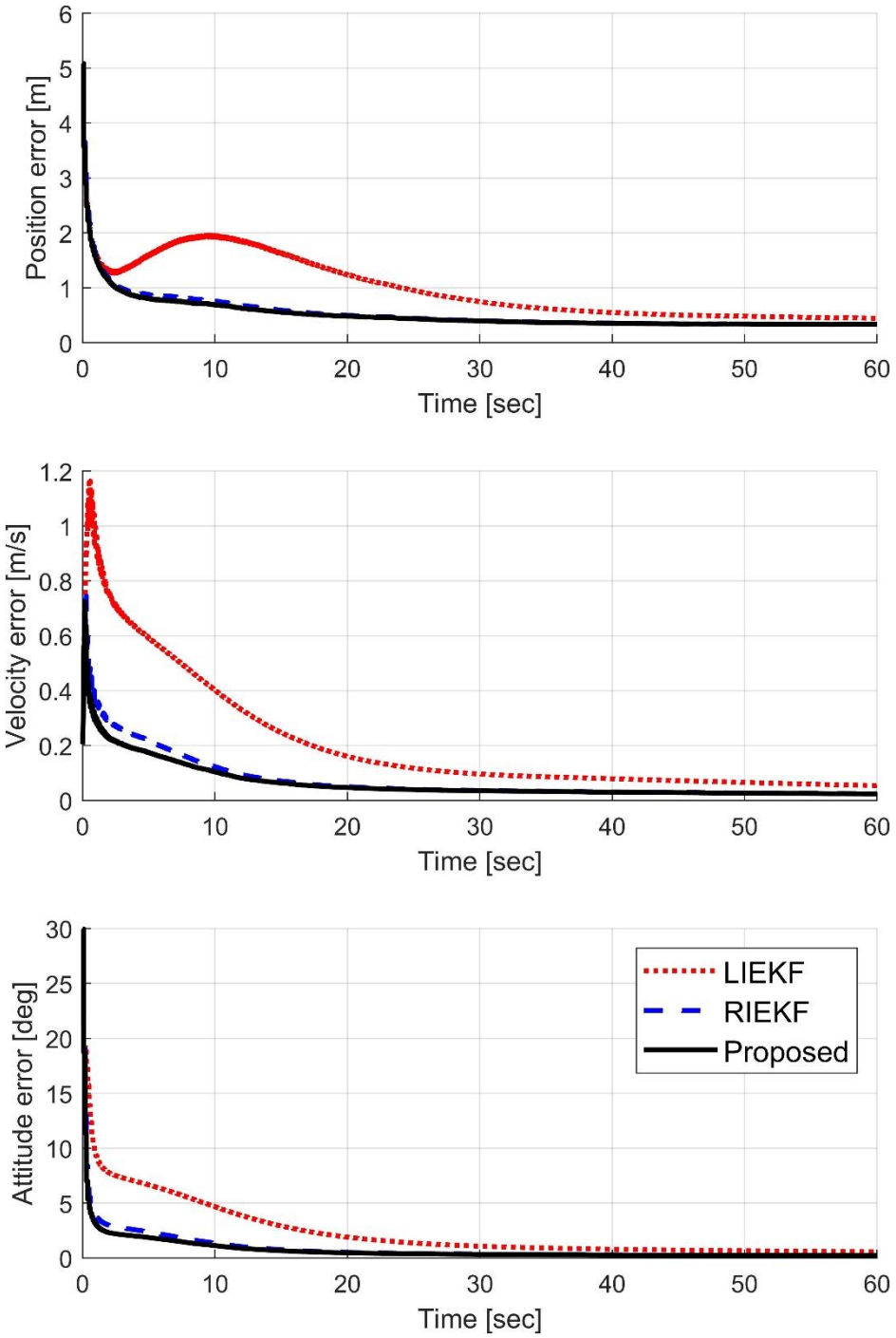

Figure 4.13 Comparison of IEKF and the proposed method for case B

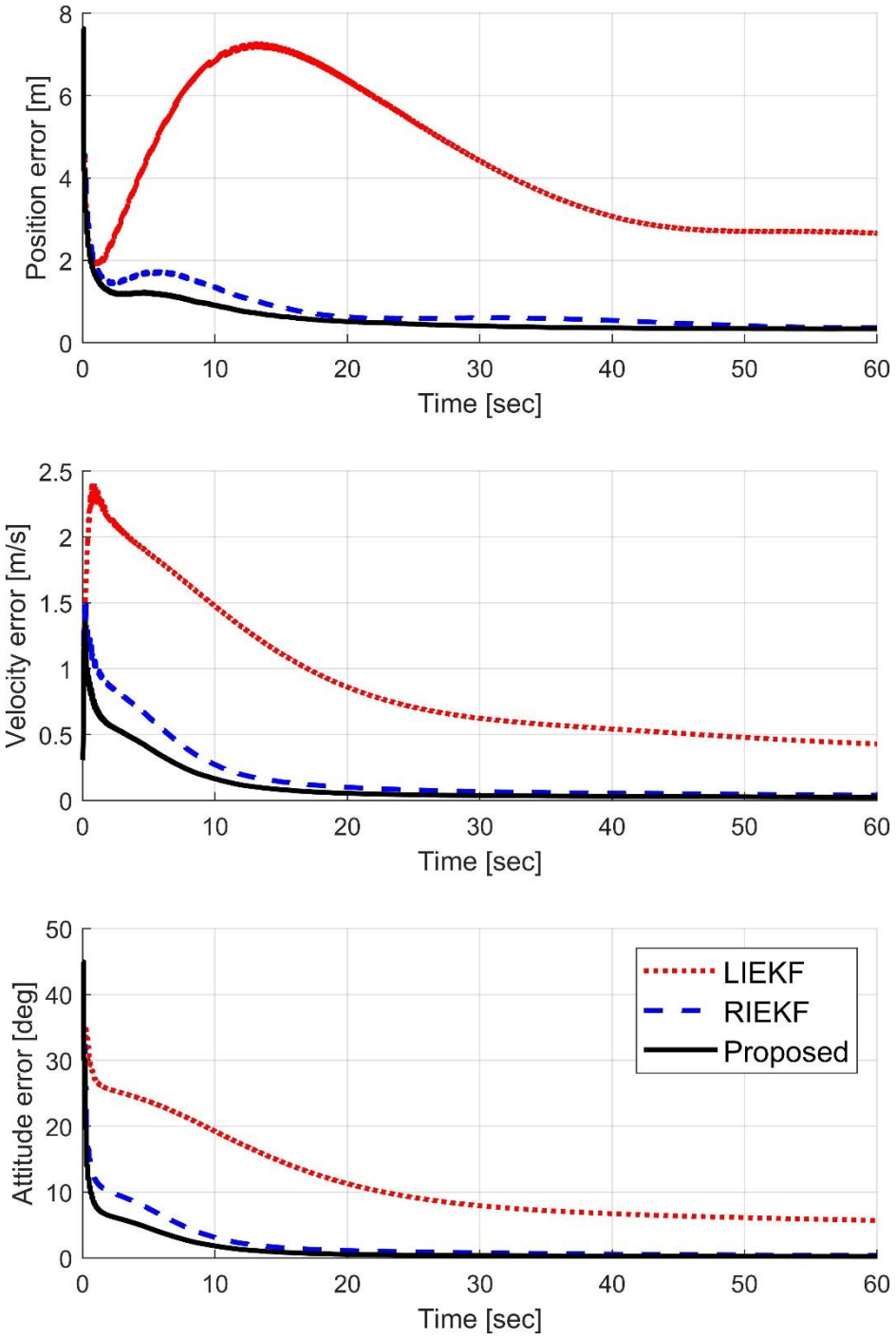

Figure 4.14 Comparison of IEKF and the proposed method for case C

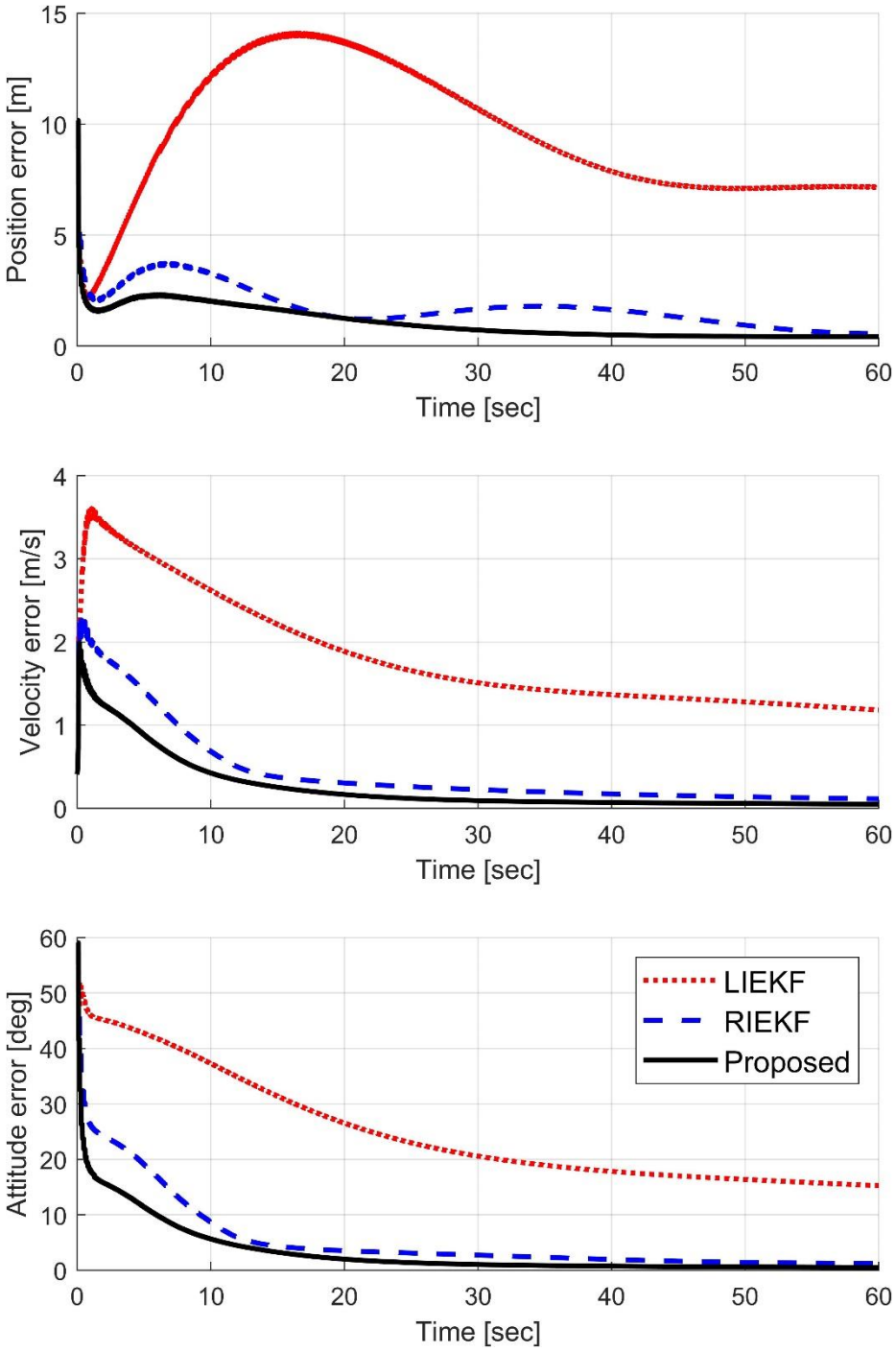

Figure 4.15 Comparison of IEKF and the proposed method for case D

Table 4.7 Average RMSE comparison for case A

	Position [m]	Velocity [m/s]	Attitude [deg]
LIEKF	0.4703	0.0477	0.4888
RIEKF	0.4690	0.0452	0.4591
Proposed	**0.4681**	**0.0451**	**0.4569**

Table 4.8 Average RMSE comparison for case B

	Position [m]	Velocity [m/s]	Attitude [deg]
LIEKF	0.9862	0.2032	2.3545
RIEKF	0.5338	0.0741	0.8467
Proposed	**0.5136**	**0.0647**	**0.7333**

Table 4.9 Average RMSE comparison for case C

	Position [m]	Velocity [m/s]	Attitude [deg]
LIEKF	4.3348	0.8743	11.2918
RIEKF	0.7803	0.1736	2.1065
Proposed	**0.5849**	**0.1089**	**1.2924**

Table 4.10 Average RMSE comparison for case D

	Position [m]	Velocity [m/s]	Attitude [deg]
LIEKF	9.4751	1.8161	24.9064
RIEKF	1.7507	0.4230	5.4453
Proposed	**1.0433**	**0.2500**	**3.2651**

Table 4.11 ANEES comparison for case A

	Position	Velocity	Attitude	Total
LIEKF	1.0046	1.0666	1.0719	1.0431
RIEKF	0.9996	1.0115	1.0180	1.0139
Proposed	**0.9975**	**1.0103**	**1.0165**	**1.0129**

Table 4.12 ANEES comparison for case B

	Position	Velocity	Attitude	Total
LIEKF	3.5010	11.5184	10.7975	9.3956
RIEKF	1.0747	1.3134	1.4376	1.2174
Proposed	**1.0270**	**1.1743**	**1.3070**	**1.1520**

Table 4.13 ANEES comparison for case C

	Position	Velocity	Attitude	Total
LIEKF	97	479	305	1,102
RIEKF	2.0366	7.2848	4.1601	8.3059
Proposed	**1.1542**	**2.2749**	**2.0720**	**1.9499**

Table 4.14 ANEES comparison for case D

	Position	Velocity	Attitude	Total
LIEKF	552	3,042	1,724	7,914
RIEKF	12.9273	69.0961	21.9635	126.5863
Proposed	**3.9400**	**25.7468**	**6.6385**	**47.0181**

From Figure 4.12 to Figure 4.15 and Table 4.7 to Table 4.10, the proposed method shows improved estimation accuracy compared to the RIEKF. In case A, the difference between LIEKF, RIEKF, and the proposed method is insignificant since the initial error is small. However, there is a slight improvement for all state variables. As the initial error gets larger, the difference becomes apparent. For case B, the proposed method shows improved accuracy compared to the RIEKF by 3.78% for the position, 12.69% for the velocity, and 13.39% for the attitude. In case C, the proposed method improved by 25%, 37%, 38% for position, velocity, and attitude, respectively. Finally, average RMSEs are improved by 40% for every state variable in case D.

In terms of the consistency, the proposed method always shows improved consistency compared to the RIEKF. The ANEES of LIEKF, RIEKF, and the proposed method are compared from Table 4.11 to Table 4.14. In each table, the consistency of the proposed method always has the smallest value. As the initial error conditions case C and case D go, the ANEES of the proposed method also increases. In addition, there would be errors due to the use of iteration-based fusion rather than closed-form solution in the fusion process of the master filter.

As a result, it can be said that the proposed method shows improved estimation performance compared to the centralized IEKF in aspects of the estimation accuracy and the consistency. Although the effect of increasing the initial error could not be eliminated, the most robust method to the initial error is the proposed method.

Chapter 5

Conclusion

5.1.1 Conclusion and summary

In this thesis, a federated IEKF structure is proposed for the multi-sensor systems using both left- and right-invariant measurements. The framework of the IEKF is arranged, and the equations of the IEKF are derived for the state variables defined in $\mathbb{SE}_2(3)$ to apply to the inertial navigation. For various left-/right-invariant measurement models, it was shown that the measurement matrix is *trajectory independent* when an appropriate invariant error is used.

A multi-sensor system using 3D navigation frame position measurement, which is a left-invariant measurement, and 3D body frame velocity measurement, a right-invariant measurement, is considered. Implementing the LIEKF and the RIEKF as the centralized filters would lose *trajectory independent* property. A federated filter structure is introduced to process each invariant measurement in each local filter. In the proposed structure, the pose fusion method proposed in previous studies is extended to the $\mathbb{SE}_2(3)$ case. Monte Carlo simulation is performed to perform numerical simulations for various initial error conditions, and the estimated performance and consistency of EKF, LIEKF, RIEKF, and the proposed method are compared. It was shown that the estimation performance of the centralized IEKF can be improved through the proposed method in a situation where the estimation

performance has deteriorated.

5.1.2 Future works

The bias of the accelerometer and the gyroscope must be estimated for the inertial navigation applications. So, the proposed method should be extended to estimate IMU biases. It can be done using the mixed state with biases augmented to $\mathbb{SE}_2(3)$. In addition, performance verification based on simulation and experimental data that considers the actual system rather than numerical simulation is required.

Bibliography

[1] D. Simon, *Optimal state estimation: Kalman, H∞, and nonlinear approaches*. John Wiley & Sons, USA: 2006.

[2] B. Robert Grover and P. Y. C. Hwang, *Introduction to random signals and applied Kalman filtering: with MATLAB exercises*, 2nd ed. Hoboken, NJ : John Wiley, USA: 1992.

[3] P. D. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*. Boston, MA, USA:Artech House, 2008.

[4] L. Perea, J. How, L. Breger, and P. Elosegui, "Nonlinearity in sensor fusion: Divergence issues in EKF, modified truncated SOF, and UKF," in *2007 AIAA Guid. Navig. Contr. Conf.,* Hilton Head, SC, USA, Aug. 20-23, 2007, pp.1863–1878.

[5] S. J. Julier and J. K. Uhlmann, "Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations," in *Proc. American Contr. Conf.*, Anchorage, AK, USA, May 8-10, 2002, pp. 887–892.

[6] S. J. Julier, "The Spherical Simplex Unscented Transformation," in *Proc. American Contr. Conf.*, Denver, CO, USA, Jun. 4-6, 2003, pp. 2430–2434.

[7] J. Wang et al., "A rapid and adaptive alignment under mooring condition using adaptive EKF and CNN-based learning," *Sensors*, vol. 20, no. 4, pp. 1–19, Aug. 2018.

[8] F. Daum, "Nonlinear filters: beyond the Kalman filter," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 20, no. 8, pp. 57-69, Aug. 2005.

[9] F. Daum and J. Huang, "Curse of dimensionality and particle filters," in *Proc. IEEE Aerosp. Conf.*, Big Sky, MT, USA, vol. 4, Mar. 8-15, 2003, pp. 1979–1993.

[10] A. Barrau and S. Bonnabel, "Invariant Kalman Filtering," *Annu. Rev. Contr., Robot., Auton. Syst.*, vol. 1, no. 1, pp. 237–257, May 2018.

[11] A. Barrau and S. Bonnabel, "The Invariant Extended Kalman Filter as a Stable Observer," *IEEE Trans. Automat. Contr.*, vol. 62, no. 4, pp. 1797–1812, Apr. 2017.

[12] S. Bonnabel, "Left-invariant extended Kalman filter and attitude estimation," in *Proc. 2007 46th IEEE Conf. on Decis. Contr.*, New Orleans, LA, USA, Dec. 12-14, 2007, pp. 1027-1032.

[13] S. Bonnable, P. Martin and E. Salaün, "Invariant Extended Kalman Filter: theory and application to a velocity-aided attitude estimation problem," in *Proc. the 48h IEEE Conf. Decis. Contr. held jointly with 2009 28th Chinese Contr. Conf.*, Shanghai, East China, China, Dec. 16-18, 2009, pp. 1297-1304.

[14] M. Barczyk and A. F. Lynch, "Invariant Observer Design for a Helicopter UAV Aided Inertial Navigation System," *IEEE Trans. Contr. Syst. Tech.*, vol. 21, no. 3, pp. 791-806, May 2013.

[15] G. Bourmaud, R. Mégret, M. Arnaudon, and A. Giremus, "Continuous-Discrete Extended Kalman Filter on Matrix Lie Groups Using Concentrated Gaussian Distributions," *J. Math. Imaging Vision*, vol. 51, no. 1, pp. 209–228, Jul. 2014.

[16] T. D. Barfoot and P. T. Furgale, "Associating Uncertainty With Three-Dimensional Poses for Use in Estimation Problems," *IEEE Trans. Robot.*, vol. 30, no. 3, pp. 679-693, Jun. 2014.

[17] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided

invariant extended Kalman filtering for robot state estimation," *Int. J. Robot. Research*, vol. 39, no. 4, pp. 402–430, Jan. 2020.

[18] T. Zhang, K. Wu, J. Song, S. Huang and G. Dissanayake, "Convergence and Consistency Analysis for a 3-D Invariant-EKF SLAM," *IEEE Robot. Automat. Lett.*, vol. 2, no. 2, pp. 733-740, Apr. 2017.

[19] K. Wu, T. Zhang, D. Su, S. Huang and G. Dissanayake, "An invariant-EKF VINS algorithm for improving consistency," in *Proc. 2017 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vancouver, BC, Canada, Sep. 24-28, 2017, pp. 1578-1585.

[20] S. Heo, J. H. Jung and C. G. Park, "Consistent EKF-Based Visual-Inertial Navigation Using Points and Lines," *IEEE Sensors Journal*, vol. 18, no. 18, pp. 7638-7649, Sep. 2018.

[21] M. Brossard, A. Barrau and S. Bonnabel, "Exploiting Symmetries to Design EKFs With Consistency Properties for Navigation and SLAM," *IEEE Sensors Journal*, vol. 19, no. 4, pp. 1572-1579, Feb. 2019.

[22] J. Cui, M. Wang, W. Wu, and X. He, "Lie group based nonlinear state errors for MEMS-IMU/GNSS/magnetometer integrated navigation," *J. Navig.*, vol. 74, no. 4, pp. 887–900, Mar. 2021.

[23] E. R. Potokar, K. Norman and J. G. Mangelson, "Invariant Extended Kalman Filtering for Underwater Navigation," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 5792-5799, Jul. 2021.

[24] Y. Gao, E. J. Krakiswsky, M. A. Abousalem, and J. F. McLellan, "Comparison and Analysis of Centralized, Decentralized, and Federated Filters," *Navigation*, vol. 40, no. 1, pp. 69–86, 1993.

[25] N. A. Carlson, "Federated square root filter for decentralized parallel processors," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 26, no. 3, pp. 517-525,

May 1990.

[26] A. Barrau, "Nonlinear state error based extended Kalman filters with applications to navigation," Ph.D. dissertation, Centre de Robotique, MINES ParisTech, Automatic. Mines Paristech, Paris, France, 2015.

[27] M. Brossard, S. Bonnabel, and A. Barrau, "Invariant Kalman Filtering for Visual Inertial SLAM," in *Proc. 2018 21st Int. Conf. Inf. Fusion*, Cambridge, UK, Jul. 10-13, 2018, pp. 2021–2028.

[28] M. Alam, G. Moreno, M. Sipos, and J. Rohac, "INS / GNSS Localization Using 15 State Extended Kalman Filter," in *Proc. Int. Conf. Aerosp. for Young Scientists*, Beijing, China, Nov. 12-13, 2016, pp. 425–435.

[29] M. Brossard, A. Barrau and S. Bonnabel, "AI-IMU Dead-Reckoning," *IEEE Trans. Intell. Vehic.*, vol. 5, no. 4, pp. 585-595, Dec. 2020.

[30] A. Barrau and S. Bonnabel, "Stochastic observers on Lie groups: a tutorial," in *Proc. 2018 IEEE Conf. Decis. and Contr.,* Miami Beach, FL, USA, Dec. 17-19, 2018, pp. 1264-1269.

[31] J. Solà, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," 2018. [Online]. Available: arXiv:1812.01537.

[32] D. H. Titterton and J. L. Weston, *Strapdown inertial navigation technology,* 2nd ed., vol. 17. London: IET, 2004.

[33] M. Brossard, A. Barrau, P. Chauchat and S. Bonnabel, "Associating Uncertainty to Extended Poses for on Lie Group IMU Preintegration With Rotating Earth," *IEEE Trans. Robot.*, Early Access, 2021.

[34] L. N. Hieu and V. H. Nguyen, "Loosely coupled GPS/INS integration with Kalman filtering for land vehicle applications," in *Proc. 2012 Int. Conf. Contr., Automat. Inf. Sci.,* Saigon, Southeast, Vietnam, Nov. 26-29*,* 2012, pp. 90-95.

[35] K. H. Kim, J. G. Lee and C. G. Park, "Adaptive Two-Stage Extended Kalman Filter for a Fault-Tolerant INS-GPS Loosely Coupled System," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 1, pp. 125-137, Jan. 2009.

[36] J. Wendel and G. F. Trommer, "Tightly coupled GPS/INS integration for missile applications," *Aerosp. Sci. Technol.*, vol. 8, no. 7, pp. 627–634, Oct. 2004.

[37] L. Chang, J. Di and F. Qin, "Inertial based Integration with Transformed INS Mechanization in Earth Frame," *IEEE/ASME Trans. Mechatron.*, Early Access, 2021.

[38] J. Borenstein and Liqiang Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Trans. Robot. Automat.,* vol. 12, no. 6, pp. 869-880, Dec. 1996.

[39] A. Tal, I. Klein, and R. Katz, "Inertial navigation system/doppler velocity log (INS/DVL) fusion with partial dvl measurements," *Sensors*, vol. 17, no. 2, pp. 1–20, Feb. 2017.

[40] D. Rudolph and T. A. Wilson, "Doppler Velocity Log theory and preliminary considerations for design and construction," in *Proc. 2012 IEEE Southeastcon*, Orlando, FL, USA, Mar. 15-18, 2012, pp. 1-7.

[41] K. Shunsuke, G. Yanlei and L. Hsu, "GNSS/INS/On-board Camera Integration for Vehicle Self-Localization in Urban Canyon," in *Proc. 2015 IEEE 18th Int. Conf. Intell. Transp. Syst.*, Gran Canaria, Marcaronesia, Spain, Sep. 15-18, 2015, pp. 2533-2538.

[42] A. Schutz, D. E. Sanchez-Morales, and T. Pany, "Precise positioning through a loosely-coupled sensor fusion of GNSS-RTK, INS and LiDAR for autonomous driving," in *2020 IEEE/ION Position, Locat. Navig. Symp. PLANS 2020,* Portland, OR, USA, Apr. 20-23, 2020*,* pp. 219–225.

[43] H. Deilamsalehy and T. C. Havens, "Sensor fused three-dimensional localization using IMU, camera and LiDAR," in *Proc. IEEE Sensors*, Orlando, FL, USA, Oct. 30- Nov. 3, 2016, pp. 1-3.

[44] S. Srinara, C. -M. Lee, S. Tsai, G. -J. Tsai and K. -W. Chiang, "Performance Analysis of 3D NDT Scan Matching for Autonomous Vehicles Using INS/GNSS/3D LiDAR-SLAM Integration Scheme," in *2021 IEEE Int. Symp. Inert. Sens. Syst. Proc.*, Kailua-Kona, HI, USA, Mar. 22-25, 2021, pp. 1-4.

국문초록

본 논문에서는 다수의 보정 센서를 사용하는 항법 시스템을 위한 연합형 불변 확장 칼만필터의 구현을 제안한다. 불변 확장 칼만필터는 일반적인 확장 칼만필터의 프레임워크는 그대로 사용하면서 상태변수를 행렬 리 그룹 상에서 정의하여 확장 칼만필터 대비 우수한 추정 성능을 가진다. 좌불변 혹은 우불변 측정치를 사용할 때 이에 적합한 불변 오차 정의를 선택하여 구현한다면 궤적 독립적인 추정이 가능하다. 대부분의 불변 확장 칼만필터에 대한 연구들은 단일 보정 센서의 사용을 가정한다. 그런데 실제 적용에 있어, 항법 시스템은 추정 성능을 향상하기 위해 다수의 보정 센서를 사용하는 경우가 많다. 좌불변 측정치와 우불변 측정치가 모두 사용되는 상황이라면, 중앙집중형 좌불변 확장 칼만필터와 우불변 확장 칼만필터는 모두 추정치에 영향을 받는 측정치 행렬을 사용하게 된다. 이로 인해 불변 확장칼만필터가 갖는 가장 큰 장점인 궤적 독립 특성을 잃는다. 반면에 연합형 필터 구조를 사용하면 각 측정치에 할당된 국소 필터에서 적절한 필터로 각 측정치를 처리할 수 있다. 따라서 이 논문에서는 불변 확장 칼만필터의 연합형 구조 구현을 제안한다. 리 그룹의 성질을 고려하는 적절한 융합 방식을 사용한 구조를 제안하며, 그 성능을 시뮬레이션을 통해 확인한다. 제안한 방식과 중앙집중형 불변 확장 칼만필터를 수렴성과 추정 정확도의 관점에서 비교하였다.

주요어: 불변 확장 칼만필터, 비선형 필터링, 다중 센서 항법, 연합형 필터
학번: 2020-24160