

# Comparative Evaluation of Invariant and Multiplicative EKF for 3D Pose Estimation with IMU Sensors and GPS or Landmark Measurements

Niels van der Laan

Master of Science Thesis



# **Comparative Evaluation of Invariant and Multiplicative EKF for 3D Pose Estimation with IMU Sensors and GPS or Landmark Measurements**

MASTER OF SCIENCE THESIS

For the degree of Master of Science in Systems and Control at Delft  
University of Technology

Niels van der Laan

June 9, 2023

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of  
Technology



Copyright © Delft Center for Systems and Control (DCSC)  
All rights reserved.



---

# Abstract

This thesis investigates the performance of the invariant extended Kalman filter (IEKF) compared to the multiplicative extended Kalman filter (MEKF) in the context of nonlinear state estimation on matrix Lie groups. The IEKF, a relatively recent variant of the EKF, is particularly suitable for systems with group-affine process models and invariant measurement models. When applied to such systems, the IEKF exhibits guaranteed state-independent error dynamics, which proves advantageous in cases of poor or inaccurate system initialization.

While previous studies have highlighted the benefits of the IEKF in poorly initialized systems, it is unclear whether the IEKF and the multiplicative EKF exhibit significant differences in performance when the system is already accurately initialized. Therefore, this thesis aims to investigate whether the IEKF demonstrates improved performance over the MEKF in 3D pose estimation using inertial measurement units (IMUs).

Specifically, the main research question of this thesis is: How does the estimation accuracy of the invariant EKF compare to the multiplicative EKF in the context of pose estimation? In order to gain insight into this, the investigation focuses on three main questions. Firstly, what are the advantages of utilizing a left-invariant EKF (LIEKF) over an MEKF when dealing with a left-invariant measurement model, and similarly, what are the benefits of employing a right-invariant IEKF (RIEKF) over an MEKF when dealing with a right-invariant measurement model? Secondly, how does the IMU sensor noise magnitude affect the converging performance of the filters differently? Thirdly, How does the sensor noise magnitude of the external measurements affect the converging performance of the filters differently?

Additionally to the distinction between the left- and right-IEKF, a similar distinction is made for the MEKF. This thesis distinguishes between an MEKF with orientation deviation states resolved in the body frame (MEKF-b) and an MEKF with orientation deviation states resolved in navigation frame (MEKF-n). This distinction is made since it allows for a more natural comparison between the IEKF and MEKF.

To conduct the evaluation, extensive simulations are performed, allowing for controlled variations in these parameters. The simulation results provide insights into the comparative performance of the IEKF and multiplicative EKF under different conditions, shedding light on their strengths and limitations in 3D pose estimation with IMUs.

It was found that the IEKF and MEKF show very comparable results in a large amount of the applications. The state-independent error dynamics have been shown to be beneficial in situations where the initial information of the state of the system is uncertain. Furthermore, the IEKF has been shown to be beneficial in certain edge cases. Firstly, the IEKF shows to be less sensitive to small process noise covariance matrices  $\mathbf{Q}$ . Secondly, once the gyroscopic noise becomes very large, the RIEKF showed higher estimation accuracy over the MEKF-n. The LIEKF did also show a marginal improvement in estimation accuracy over the MEKF-b. Finally, it was found in this thesis there are two ways that the external measurement noise influenced the comparison of the estimation accuracy between the IEKF and MEKF. The MEKF-n showed to be sensitive to a low covariance measurement matrix  $\mathbf{R}$  and additionally, the MEKF-b and MEKF-n both seemed to be marginally more affected by higher external measurement noise than the LIEKF and RIEKF, respectively.

In conclusion, this thesis provides a comprehensive evaluation of the IEKF and MEKF in 3D pose estimation with IMUs. While the IEKF and MEKF exhibit comparable performance in many cases, the IEKF's state-independent error dynamics and its advantages in certain scenarios highlight its potential superiority over the MEKF. These findings contribute to the understanding of nonlinear state estimation on matrix Lie groups and offer valuable insights for selecting the appropriate filter for specific applications.

---

# Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1-1	Motivation . . . . .	3
1-2	Thesis Goal . . . . .	5
1-3	Thesis Outline . . . . .	6
<b>2</b>	<b>Background Information</b>	<b>7</b>
2-1	Matrix Lie Groups . . . . .	7
2-1-1	Overview . . . . .	7
2-1-2	Uncertainty Representation . . . . .	8
2-1-3	The Baker-Campbell-Hausdorff Formula . . . . .	8
2-1-4	Linearization . . . . .	8
2-2	Multiplicative Extended Kalman Filter . . . . .	9
2-3	Invariant Extended Kalman Filter . . . . .	10
2-4	Related Work . . . . .	12
<b>3</b>	<b>Models</b>	<b>15</b>
3-1	Gyroscope Measurement Models . . . . .	16
3-2	Accelerometer Measurement Models . . . . .	16
3-3	Dynamical models . . . . .	17
3-4	Position Measurements . . . . .	17
3-5	Landmark Position Measurements . . . . .	18
<b>4</b>	<b>Method</b>	<b>21</b>
4-1	Simulation Setup . . . . .	21
4-2	Left-Invariant Extended Kalman Filter Derivation . . . . .	23
4-3	Multiplicative EKF with Orientation Deviation States in Body Frame . . . . .	31
4-4	Right-Invariant Extended Kalman Filter Derivation . . . . .	34
4-5	Multiplicative EKF with Orientation Deviation States in Navigation Frame . . . . .	42

<b>5</b>	<b>Results</b>	<b>49</b>
5-1	Simulation Parameters . . . . .	49
5-2	Simulation Results . . . . .	50
5-2-1	Left-Invariant Measurements . . . . .	50
5-2-2	Right-invariant measurements . . . . .	54
<b>6</b>	<b>Conclusions and Future Work</b>	<b>63</b>
6-1	Conclusions . . . . .	63
6-2	Future Work . . . . .	66
<b>A</b>	<b>Parametrization of Orientation Representation</b>	<b>67</b>
A-1	Rotation Matrix . . . . .	67
<b>B</b>	<b>Matrix Lie Groups</b>	<b>69</b>
B-1	The Special Orthogonal Group $SO(3)$ . . . . .	69
B-2	The Group of Double Direct Isometries $SE_2(3)$ . . . . .	70
<b>C</b>	<b>Supplementary Results</b>	<b>73</b>
	<b>Glossary</b>	<b>79</b>
	List of Acronyms . . . . .	79
	List of Symbols . . . . .	79

---

# List of Figures

3-1	Problem setup. . . . .	15
3-2	Problem setup for pose estimation, red crossed represent landmarks. . . . .	19
4-1	A simulated trajectory. The ten landmarks are represented by the red crosses. . .	22
5-1	RMS of the norm of the error when increasing the initial error. . . . .	51
5-2	The norm of the estimation error for each state when increasing the initial orientation error to $\delta\xi^\phi = 1$ rad for two different single simulation runs. . . . .	52
5-3	RMS of the norm of the error when increasing the amplitude of the sensor noise. . . . .	53
5-4	The norm of the estimation error for each state with zero accelerometer noise. . . . .	54
5-5	RMS of the norm of the error when increasing the amplitude of the accelerometer sensor noise. . . . .	55
5-6	The norm of the estimation error over time for each state when increasing the noise on the gyroscope to $\sigma_\omega = 1$ rad/s (left) and on the GPS measurements to $\sigma_R = 1$ m (right). . . . .	55
5-7	Increasing the initial orientation error with 10 known landmarks. . . . .	56
5-8	RMS of the norm of the error when increasing the amplitude of the sensor noise with 10 known landmarks. . . . .	57
5-9	Increasing the initial orientation error with 3 known landmarks. . . . .	58
5-10	RMS of the norm of the error when increasing the amplitude of the sensor noise with 3 known landmarks. . . . .	59
5-11	The norm of the estimation error for each state with zero accelerometer noise. . . . .	60
5-12	The norm of the estimation error for each state with zero landmark measurement noise. . . . .	61
5-13	RMS of the norm of the error when increasing the amplitude of the sensor noise with 3 known landmarks. . . . .	62
C-1	RMS of the norm of the error when increasing the initial error in velocity and position with 10 known landmarks. . . . .	73
C-2	RMS of the norm of the error when increasing the initial error in velocity and position with 3 known landmarks. . . . .	74



---

# List of Tables



---

# Chapter 1

---

## Introduction

In recent years, accurate pose estimation has become increasingly important in a wide range of fields, including robotics, computer vision, and autonomous navigation [1, 2]. The ability to precisely estimate the position, orientation and sometimes velocity of objects in three-dimensional space is crucial for enabling tasks such as object tracking, localization, and mapping [2, 3]. To achieve reliable and robust pose estimation, to ensure accuracy, efficiency and effectiveness, filtering techniques are employed, among which the Extended Kalman Filter (EKF) has gained significant attention [4, 5].

Nonlinear pose estimation techniques play a crucial role in accurately estimating the position, orientation and sometimes velocity of objects or systems in various fields, including robotics, computer vision, and autonomous navigation [1, 2]. Traditional linear estimation methods, such as the Kalman Filter (KF), are not suitable for handling nonlinear systems. In such cases, nonlinear pose estimation techniques, like the Extended Kalman Filter (EKF), Particle Filter, and Unscented Kalman Filter (UKF), have emerged as powerful tools to tackle the complexities associated with nonlinear dynamics and measurements [4].

The EKF, Particle Filter, and UKF are all nonlinear pose estimation techniques that aim to estimate the pose of a system based on nonlinear measurements or dynamics. The EKF approximates the nonlinear system with a linearized model and uses a combination of prediction and update steps to iteratively estimate the pose [4]. The Particle Filter, on the other hand, employs a particle-based representation of the pose distribution, using a set of weighted particles to approximate the posterior distribution [4, 6]. The UKF takes a different approach by using a deterministic sampling technique called the unscented transformation to capture the statistical properties of the nonlinear system [4, 7].

While each technique has its own strengths and limitations, the focus of this discussion will be primarily on the EKF. The EKF is widely used in many applications due to its simplicity, efficiency, and effectiveness in handling nonlinear systems [4, 5]. It is an extension of the traditional Kalman Filter that incorporates linearization techniques to approximate the nonlinear system. By linearizing the system around the current estimate, the EKF propagates the state estimate using linear equations while updating it based on nonlinear measurements.

The EKF provides an iterative estimation process that yields a posterior distribution of the pose, taking into account both prediction and measurement update steps [4].

The EKF has found applications in various fields, including robotics, where it is commonly used for localization, mapping, and navigation tasks [2, 3]. It has also been applied in computer vision for object tracking and augmented reality applications. Despite its widespread usage, the EKF does have limitations, such as reliance on accurate linearization and the assumption of Gaussian distributions, which may not always hold true in practice [4, 8]. Nevertheless, with appropriate modelling, the EKF can provide reliable and accurate pose estimation in many scenarios.

Inertial Measurement Units (IMUs) have emerged as crucial sensor systems for estimating the state of dynamic systems. An IMU typically consists of a combination of sensors such as accelerometers, gyroscopes, and sometimes magnetometers. These sensors provide measurements of linear acceleration, angular velocity, and in some cases, the Earth's magnetic field [9]. The EKF combines sensor measurements with a dynamic model to estimate the state of a system. IMUs, with their ability to measure linear acceleration and angular velocity, play a fundamental role in EKF-based state estimation, particularly in applications involving motion tracking, navigation, robotics, and virtual reality. The EKF utilizes the measurements from the IMU sensors to update state estimate over time. The dynamic model describes the system's motion and how it evolves over time, while the IMU measurements provide valuable information about the system's acceleration and rotational velocity. One key advantage of using IMUs with the EKF is their ability to provide high-frequency measurements, allowing for real-time state estimation. These measurements, when integrated over time, provide estimates of position, velocity, and orientation. However, IMU measurements are not without limitations. They are prone to errors and biases, which can accumulate over time, leading to drift in the estimated state [4]. These errors arise from factors such as sensor noise, bias instability, and external disturbances. To mitigate these issues, the EKF incorporates a recursive estimation process that updates the state estimate while compensating for sensor noise and biases [4].

Furthermore, the EKF can also be combined with other sensors, such as GPS or visual sensors, to enhance the accuracy of the state estimation. By fusing the measurements from multiple sensors, including IMUs, the EKF can leverage complementary information and overcome the limitations of individual sensors [10].

There are specialized variants of the EKF that address specific challenges in pose estimation. Two notable variants are the Multiplicative EKF (MEKF) and the Invariant EKF (IEKF). These variants build upon the foundations of the EKF while introducing additional enhancements to improve the accuracy and robustness of pose estimation.

The MEKF extends the EKF by explicitly modelling the orientation in a multiplicative manner, often also referred to as quaternion-based EKF [11]. By representing the orientation using rotation matrices or quaternions, the MEKF avoids the singularities associated with Euler angles and provides a more stable and globally consistent representation [12, 13]. This variant is particularly useful in scenarios where robust orientation estimation is critical, such as robotics and 3D motion tracking.

The IEKF, on the other hand, is specifically designed to handle state estimation on nonlinear manifolds, specifically matrix Lie groups. It achieves this by performing updates directly

on the manifold, ensuring that the estimated state remains valid and consistent with the manifold's geometry [14]. This is in contrast to the EKF, which relies on linearizations and can introduce errors when applied to nonlinear systems. In the IEKF, the dynamic model and measurement model are defined on the manifold itself, accounting for the nonlinearities of the system. By directly operating on the manifold, the IEKF provides more accurate and reliable state estimates for nonlinear systems compared to the traditional EKF [14].

It is important to note that for the IEKF, there are two approaches, these are known in literature as left- and right-invariant Extended Kalman Filtering techniques have also been developed.

Left-invariant Extended Kalman Filtering (LIEKF) and right-invariant Extended Kalman Filtering (RIEKF) are two alternative approaches that address the challenges of state estimation on Lie groups or nonlinear manifolds. These techniques employ left or right group actions, respectively, to maintain consistency with the manifold's structure during the estimation process.

For the LIEKF, the dynamic model and measurements are transformed using the left group action. This approach ensures that the filter remains invariant under left group transformations, enabling accurate and consistent estimation on the manifold. Similarly, the RIEKF achieves invariance by utilizing the right group action. The choice of employing a left- or right-IEKF is typically dependent on the type of measurement model used in the filter. A distinction is made between a left- and right-invariant measurement model, where for the use of a left-invariant measurement model, the LIEKF is employed to ensure invariant error dynamics. Whereas for a right-invariant measurement model, the RIEKF is suitable. In this thesis, a similar distinction is for the MEKF. This study distinguishes between an MEKF with orientation deviation states resolved in the body frame (MEKF-b) and an MEKF with orientation deviation states resolved in navigation frame (MEKF-n). This distinction is made since it allows for a more natural comparison between the IEKF and MEKF.

## 1-1 Motivation

The IEKF in continuous time was introduced in [8] and in discrete time in [15]. Both these papers offer the theoretical foundation on which the structure of the IEKF is based. This foundation is built on the theory of symmetry preserving observers on matrix Lie groups [16, 17]. In [15], mostly the theoretical background with an application presented to simultaneous localization and mapping (SLAM). The IEKF has been gaining popularity for the application to SLAM [18, 19, 20, 21], due to its ability to overcome inconsistency issues. Inconsistency in this case refers to the inability of the EKF output covariance matrix to correctly reflect the error dispersion [15] and its inability to correctly reflect the unobservabilities of the SLAM problem [15, 22]. Although the topic of SLAM is a popular one, it is not within the scope of this research. In [8], two applications are briefly presented that are of interest for this research, since it discusses the performance of the IEKF compared to that of the EKF or MEKF.

The first application shows a 2D problem where the heading and position of a car driving in a circle, are to be estimated. The estimation is done through an EKF and IEKF with high rate odometry and low rate GPS measurements. They initialize the simulation in two

different ways, first with a small initial orientation error and subsequently, with a larger initial orientation error. The reported results indicate that both filters show comparable performance in terms of the state estimation error when the initial orientation error is small. However, this performance quickly differs once the simulation is initialized with a high error in initial orientation. The IEKF quickly converges to the true trajectory in this simulation, whereas the EKF does not and shows to slowly converge towards the true trajectory.

The second application involves a 3D problem where a plane flies in a circular motion and 3 landmarks are placed close to the trajectory of the plane. The goal is to perform aided inertial navigation based on high rate accelerometers' and gyroscope measurements and low rate observation of known landmarks. The estimation is done through an IEKF and an MEKF. This time the simulation is initialized in two different ways, the first simulation is initialized with very a small or tight process noise covariance matrix  $\mathbf{Q}$ , whereas the second simulation is initialized with a relatively bigger or inflated covariance matrix  $\mathbf{Q}$ . The results show that the use of a tight covariance matrix  $\mathbf{Q}$  poses issues for the MEKF where it does not for the IEKF. They explain this is due to the gains of the MEKF rapidly decreasing during the transitory phase of the simulation while the attitude error is not reduced enough because of non-linearities, since the position estimate is affected, the gains are too small to correct this error. The second simulation shows that artificially inflating the covariance matrix  $\mathbf{Q}$  in the filter, to use it as a type of tuning parameter, overcomes this issue for the MEKF and the estimation seems to be able to converge to the true trajectory, although it still converges slower to the true trajectory than the IEKF [15].

In [23], the performance of the IEKF is compared with the EKF in a very straightforward problem. A car with perfect odometry measurements is set on a 2D trajectory with noisy GPS measurements acting as corrective measurements. In situations where the dynamics are perfectly known, the linear Kalman filter can handle the situation effectively. In such cases, the filter's gains can be initially large if the initial information is inaccurate, gradually decreasing to zero. However, this ideal scenario poses significant challenges for the EKF. Due to the presence of nonlinearities, the gains in the EKF can approach zero while the estimation error has not reduced sufficiently. This can result in a static asymptotic non-zero error or even divergence, even when the measurements are free from noise [23]. The main result this paper offers is that the IEKF shows superior performance over the EKF even when the IEKF is initialized with a very high error in initial attitude and the error in initial attitude for the EKF is kept relative low. Showing similar findings as the ones presented in [8].

In [24], an invariant Rauch-Tung-Striebel (IRTS) smoother is introduced for pose estimation on the special Euclidian group,  $SE(3)$ . This is the matrix Lie group consisting of 3D rotation matrices and position vector. Although this thesis focuses on the IEKF and not on a smoother, the results in this paper are of interest since the IRTS smoother is very closely related to the IEKF, since the forward pass of the smoother is identical to the IEKF but with the addition of a backward pass. The main results from this paper is that when the initialization of the system is poor, then the IRTS outperforms its multiplicative counterpart. This is in agreement with the findings in [8, 23] and build on the fact that the error dynamics of the invariant smoother is not dependent on the state estimate and thus is not influenced when these state estimates are inaccurate.

In [25], 3D pose estimation is performed for underwater navigation using the Right-IEKF and compared it to a quaternion based EKF. They make use of IMUs and doppler velocity

logs (DVL). They ran a simulation consisting of a simple descent of the vehicle with thrusters slightly pushing forward. Then, each filter was run 100 times for the first 2 seconds of simulation with varying initial starting points. The RIEKF was reported to converge faster to the true trajectory than its quaternion based counterpart. They also reported that the quaternion based EKF showed issues with convergence when the initialization was done with high uncertainty, whereas the RIEKF did not seem affected at all by this, while having comparable results in computation time. Again, this agrees with the findings of [8, 23].

The existing literature primarily emphasizes the superiority of the IEKF over the (M)EKF by demonstrating improved performance through increased uncertainty in the initial estimate. While [8] briefly discusses the impact of a small process covariance matrix  $\mathbf{Q}$ , there appears to be a lack of focus in the literature on examining the influence of sensor and measurement noise in detail.

## 1-2 Thesis Goal

The main research question driving this thesis is: "How does the estimation accuracy of the invariant EKF compare to the multiplicative EKF in the context of 3D pose estimation?". This research question aims to investigate and compare the performance of the invariant EKF and the multiplicative EKF specifically in the context of pose estimation. By conducting an evaluation, we can gain insights into the strengths and weaknesses of each filtering technique and determine which one provides superior performance in terms of accuracy, stability, and convergence.

To address this research question, several sub-questions will be explored:

- What are the advantages of utilizing an LIEKF over an MEKF when dealing with a left-invariant measurement model, and similarly, what are the benefits of employing a RIEKF over an MEKF when dealing with a right-invariant measurement model?
- How does the IMU sensor noise magnitude affect the converging performance of the filters differently?
- How does the sensor noise magnitude of the external measurements affect the converging performance of the filters differently?

The contributions of this thesis are as follows. Firstly, we will investigate the benefits of utilizing specific variants of the invariant EKF for different measurement models. Specifically, the use of the Left-Invariant EKF (LIEKF) for left-invariant measurement models and the Right-Invariant EKF (RIEKF) for right-invariant measurement models will be analyzed and compared to their suitable multiplicative counterparts, the MEKF-b and MEKF-n. The filters have been derived in discrete-time in the matrix Lie group of double direct isometries. By evaluating the performance of these variants in their respective scenarios, we can determine the advantages and effectiveness of employing specific filters for different pose estimation applications.

Additionally, we will investigate how the presence of Inertial Measurement Unit (IMU) sensor noise affects the converging performance of the filters. It is hypothesized that the invariant

EKF may exhibit different behavior compared to the multiplicative EKF when subjected to IMU sensor noise. Understanding these differences will provide valuable insights into the filters' robustness and adaptability to noisy measurements.

Furthermore, we will analyze the impact of the quality of external measurements on the converging performance of the filters. External measurements, such as those obtained from GPS or vision sensors, play a crucial role in refining pose estimates. By examining how the filters handle these external measurements differently, we can gain a deeper understanding of their ability to incorporate additional data sources effectively.

## 1-3 Thesis Outline

The goal of this thesis is to deepen our understanding of the differences between the invariant EKF and the multiplicative EKF and how these differences are influenced by various parameters, consisting of IMU sensor noise and external measurement sensor noise. To achieve this objective, the thesis is structured as follows:

Chapter 2 provides the reader with essential background information on matrix Lie groups, the multiplicative EKF and the invariant EKF. This chapter serves as a foundation for comprehending the subsequent discussions and analyses. Furthermore, it explores the existing body of work on the invariant EKF, including its comparisons with traditional and multiplicative EKF approaches, specifically for pose estimation. This chapter highlights the strengths and limitations of each filtering technique, laying the groundwork for the subsequent methodology.

Chapter 3 describes the system models used to represent the sample problem under investigation. This includes the IMU models, the dynamical models and the measurement models.

Chapter 4 delves into the methodology employed in this thesis. First, it discusses the simulation setup. Additionally, it presents the derivations of the variations of the invariant and multiplicative EKFs in discrete-time for the matrix Lie group of double direct isometries. This chapter provides a detailed insight into the filtering algorithms employed for the pose estimation task and the specific choices in the simulation models with the goal of reflecting realistic scenarios made.

Chapter 5 presents the corresponding results obtained, aiming to address the research questions posed in this thesis. It provides a comprehensive analysis of the performance of the invariant EKF and the multiplicative EKF, considering the influence of IMU sensor noise and external measurement sensor noise. The results shed light on the comparative performance and robustness of the two filtering techniques.

Finally, Chapter 6 concludes the findings of this thesis, summarizing the key insights and implications. It also offers recommendations for future research directions, highlighting areas that warrant further investigation to advance the understanding and application of pose estimation using the invariant EKF and the multiplicative EKF.

Through this structured approach, this thesis aims to contribute to the existing knowledge by providing a comprehensive analysis of the invariant EKF and the multiplicative EKF, specifically focusing on their behavior under different noise conditions.

## Background Information

### 2-1 Matrix Lie Groups

#### 2-1-1 Overview

This overview of the theory on matrix Lie groups is based on [26]. A matrix Lie group  $\mathcal{G}$  is composed of  $n \times n$  invertible matrices that are closed under multiplication. This means that if two elements from  $\mathcal{G}$  are multiplied together, the resulting matrix is still an element of  $\mathcal{G}$  [26]. The matrix Lie algebra associated with  $\mathcal{G}$  is denoted by  $\mathfrak{g}$  and represents the space of differential transformations or the tangent space around the identity of  $\mathcal{G}$ , denoted by  $T_1\mathcal{G}$ . The matrix Lie algebra is a vector space and is closed under the operation of the matrix Lie bracket, defined as  $[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA}$ ,  $\forall \mathbf{A}, \mathbf{B} \in \mathfrak{g}$  [4]. Additionally, for any  $\mathbf{X} \in \mathcal{G}$  and  $\mathbf{A} \in \mathfrak{g}$ , we have  $\mathbf{XAX}^{-1} \in \mathfrak{g}$ . Moreover, any  $\mathbf{A} \in \mathfrak{g}$  can be written as  $\mathbf{A} = \boldsymbol{\xi}^\wedge = \sum_{i=1}^n \xi_i \mathbf{B}_i$ , where  $\mathbf{B}_1, \dots, \mathbf{B}_n$  is a basis for  $\mathfrak{g}$ , also referred to as the generators, and  $\boldsymbol{\xi} \in \mathbb{R}^d$ . Furthermore,  $\mathbf{A}^\vee = [\xi_1, \dots, \xi_n]^\top = \boldsymbol{\xi}$ .

The exponential map takes elements from the Lie algebra and maps them to the Lie group. For matrix Lie groups, this exponential map is the matrix exponential. The matrix logarithm performs the inverse operation of the matrix exponential and maps elements from the matrix Lie group to the matrix Lie algebra. This can be summarized as follows:  $\exp(\cdot) : \mathfrak{g} \rightarrow \mathcal{G}$  and  $\log(\cdot) : \mathcal{G} \rightarrow \mathfrak{g}$ , so  $\exp(\log(\mathbf{X})) = \mathbf{X}$ ,  $\forall \mathbf{X} \in \mathcal{G}$ , and  $\mathbf{X} = \exp(\boldsymbol{\xi}^\wedge)$ , where  $\boldsymbol{\xi} \in \mathbb{R}^d$  [26].

The matrix representation of the adjoint operator is particularly useful when working with matrix Lie groups, as it preserves the structure of the group. The adjoint representation of  $\mathbf{X}$  is denoted by  $\text{Ad}(\mathbf{X})$ . Then,  $(\text{Ad}(\mathbf{X}) \boldsymbol{\xi})^\wedge = \mathbf{X} \boldsymbol{\xi}^\wedge \mathbf{X}^{-1}$ , which leads to the equation:

$$\mathbf{X} \exp(\boldsymbol{\xi}^\wedge) \mathbf{X}^{-1} = \exp((\text{Ad}(\mathbf{X}) \boldsymbol{\xi})^\wedge). \quad (2-1)$$

The adjoint representation of an element of the matrix Lie algebra can be defined as follows [27]: given  $\boldsymbol{\xi}^\wedge, \boldsymbol{\zeta}^\wedge \in \mathfrak{g}$ , the adjoint matrix satisfies  $\text{ad}(\boldsymbol{\zeta}^\wedge) \boldsymbol{\xi} = -\text{ad}(\boldsymbol{\xi}^\wedge) \boldsymbol{\zeta}$ , and

$$\boldsymbol{\xi}^\wedge \boldsymbol{\zeta}^\wedge - \boldsymbol{\zeta}^\wedge \boldsymbol{\xi}^\wedge = (-\text{ad}(\boldsymbol{\zeta}^\wedge) \boldsymbol{\xi})^\wedge. \quad (2-2)$$

### 2-1-2 Uncertainty Representation

In vector spaces, uncertainty is additive, meaning that it can be added and the resulting vector is still an element of the same vector space, such that  $\mathbf{x} = \bar{\mathbf{x}} + \delta\mathbf{x}$ , where  $\delta\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ . However, as mentioned before, matrix Lie groups are not closed under addition but rather under multiplication. This means that a multiplicative uncertainty must be used, such that the resulting state is still an element of the same Lie group [28]. Two options then arise to represent the uncertainty,

$$\mathbf{X} = \bar{\mathbf{X}} \exp(\delta\xi^\wedge), \quad (2-3a)$$

$$\mathbf{X} = \exp(\delta\xi^\wedge) \bar{\mathbf{X}}, \quad (2-3b)$$

where  $\delta\xi \sim \mathcal{N}(\mathbf{0}, \Sigma)$ . Two additional definitions for the uncertainty can be defined. These two definitions are given by

$$\mathbf{X} = \bar{\mathbf{X}} \exp(-\delta\xi^\wedge), \quad (2-4a)$$

$$\mathbf{X} = \exp(-\delta\xi^\wedge) \bar{\mathbf{X}}. \quad (2-4b)$$

In this thesis, the definition as in (2-4) is used. These definitions are referred to as left-invariant and right-invariant uncertainty representations, respectively. These naming conventions are consistent with the left-invariant and right-invariant error definitions, which will be introduced in section 2-3.

### 2-1-3 The Baker-Campbell-Hausdorff Formula

It is widely known that two scalar exponential functions can be combined as follows,

$$\exp(a) \exp(b) = \exp(a + b)$$

where  $a, b \in \mathbb{R}$ . Unfortunately, this does not hold for matrix exponentials. To compound two matrix exponentials as in

$$\mathbf{l} = \log(\exp(\mathbf{m}^\wedge) \exp(\mathbf{n}^\wedge)),$$

the Baker-Campbell-Hausdorff (BCH) formula can be used [4, 15]. The BCH formula gives a series expansion, but in particular it ensures

$$\exp(\mathbf{m}^\wedge) \exp(\mathbf{n}^\wedge) = \exp(\mathbf{m}^\wedge + \mathbf{n}^\wedge + \mathcal{H}),$$

where  $\mathcal{H}$  is of the order  $\mathcal{O}(\|\mathbf{m}\|_2^2, \|\mathbf{n}\|_2^2, \|\mathbf{m}\|_2 \|\mathbf{n}\|_2)$ . Herein only a first-order approximation is considered, being

$$\log(\exp(\mathbf{m}^\wedge) \exp(\mathbf{n}^\wedge)) = \mathbf{m}^\wedge + \mathbf{n}^\wedge.$$

This is exact in the case that  $[\mathbf{m}^\wedge, \mathbf{n}^\wedge] = \mathbf{0}$  [4].

### 2-1-4 Linearization

As mentioned before, an element of a matrix Lie group can be expressed using the matrix exponential,

$$\mathbf{X} = \exp(\xi^\wedge).$$

The matrix exponential can be described by a power series and is defined as follows,

$$\begin{aligned}\exp(\boldsymbol{\xi}^\wedge) &= \sum_{k=0}^{\infty} \frac{1}{k!} (\boldsymbol{\xi}^\wedge)^k, \\ &= \mathbf{I} + \boldsymbol{\xi}^\wedge + \frac{(\boldsymbol{\xi}^\wedge)^2}{2} + \frac{(\boldsymbol{\xi}^\wedge)^3}{6} + \dots\end{aligned}$$

Considering the case where  $\boldsymbol{\xi}$  is small, the small element of  $\mathbb{R}^d$  is denoted by  $\delta\boldsymbol{\xi}$  and  $\delta\mathbf{X} = \exp(\delta\boldsymbol{\xi}^\wedge)$ . Seeing that  $\delta\boldsymbol{\xi}$  is already considered small, it is commonly assumed that high order terms like  $\mathcal{O}(\|\delta\boldsymbol{\xi}\|^2)$  can be neglected. This leads to the following approximation,

$$\delta\mathbf{X} \approx \mathbf{I} + \delta\boldsymbol{\xi}^\wedge. \quad (2-5)$$

Following this reasoning, the uncertainty representations (2-3a) and (2-3b) can be approximated as

$$\mathbf{X} = \bar{\mathbf{X}}(\mathbf{I} + \delta\boldsymbol{\xi}^\wedge), \quad (2-6a)$$

$$\mathbf{X} = (\mathbf{I} + \delta\boldsymbol{\xi}^\wedge)\bar{\mathbf{X}}, \quad (2-6b)$$

respectively. The same approximation can be done on the representations (2-4a) and (2-4b), which results in

$$\mathbf{X} = \bar{\mathbf{X}}(\mathbf{I} - \delta\boldsymbol{\xi}^\wedge), \quad (2-7a)$$

$$\mathbf{X} = (\mathbf{I} - \delta\boldsymbol{\xi}^\wedge)\bar{\mathbf{X}}, \quad (2-7b)$$

respectively.

## 2-2 Multiplicative Extended Kalman Filter

The multiplicative Extended Kalman Filter (MEKF) is a variant of the traditional EKF that is commonly used for state estimation in systems where the orientation is represented using rotation matrices or quaternions. The EKF is a recursive estimation algorithm that combines measurements from sensors with a dynamic model to estimate the state of a system.

In the context of orientation estimation, the MEKF is employed to estimate the attitude or orientation of an object in three-dimensional space [29, 30]. While there are different ways to represent orientation, rotation matrices and quaternions are popular choices due to their mathematical properties and computational efficiency.

When using rotation matrices, the MEKF operates in the space of rotation matrices rather than the space of Euler angles or other representations. This helps avoid the issues of singularities and nonlinearities that can arise with other representations. The state vector in the MEKF contains the elements of the rotation matrix, and the dynamic model describes the evolution of these elements over time. The measurement model relates the observed sensor measurements to the rotation matrix elements.

Alternatively, quaternions can be used to represent orientation in the MEKF [31]. Quaternions provide a compact representation and avoid some of the drawbacks associated with rotation

matrices. The state vector in the MEKF consists of the quaternion elements, and the dynamic model describes the quaternion propagation. The measurement model relates the sensor measurements to the quaternion elements [31, 32].

The multiplicative EKF is named so because it employs a multiplicative update step to estimate the state of the system [31, 9]. This update step is performed in the manifold of rotation matrices or quaternions, ensuring that the estimated orientation remains valid and consistent with the constraints imposed by the chosen representation.

This thesis chose to represent the orientation state using rotation matrices rather than quaternions. This was to ensure that the similarities between the MEKF and the IEKF are highlighted, since the IEKF, in this thesis, also represents the orientation state by a rotation matrix in its state matrix. The rotation matrices are updated using

$$\hat{\mathbf{C}}_{nb_k} = \check{\mathbf{C}}_{nb_k} \exp(\delta \hat{\boldsymbol{\xi}}_{b_k}^{\phi^\times}),$$

if the orientation deviation is defined in body frame  $\mathcal{F}_b$ , this MEKF will be referred to as MEKF-b, or

$$\hat{\mathbf{C}}_{nb_k} = \exp(\delta \hat{\boldsymbol{\xi}}_{n_k}^{\phi^\times}) \check{\mathbf{C}}_{nb_k},$$

if the orientation deviation is defined in navigation frame  $\mathcal{F}_n$ , this MEKF will be referred to as MEKF-n.

## 2-3 Invariant Extended Kalman Filter

In recent years, Barrau and Bonnabel have introduced the Invariant Extended Kalman Filter (IEKF) [8], building on the theory of symmetry preserving observers on matrix Lie groups [17, 16]. The IEKF exploits the fact that, in robotics, the estimated states are elements of a matrix Lie group. It has been shown that carefully defining the error leads to state-independent error dynamics [8], implying that the Jacobians are state-independent. Using this fact, it can be shown that the IEKF is a locally asymptotically stable observer, no matter the trajectory [8].

Let  $\mathcal{G} \in \mathbb{R}^{n \times n}$  be a matrix Lie group and its matrix Lie algebra is denoted by  $\mathfrak{g} \in \mathbb{R}^{d \times d}$ . Any element of a matrix Lie group can be expressed using the exponential map,

$$\mathbf{X} = \exp(\boldsymbol{\xi}^\wedge), \quad (2-8)$$

where  $\boldsymbol{\xi}^\wedge \in \mathfrak{g}$ . Suppose the dynamical system can be described as follows,

$$\mathbf{X}_k = \mathbf{F}_{k-1}(\mathbf{X}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \quad (2-9)$$

this represents the most general case, however, an approximation is often used to ease the derivations. In (2-9), the function is denoted by  $\mathbf{F}_{k-1}$  with a capital letter since it denotes a function taking matrices as arguments. Before this approximation can be made, it should restated that there are two types of invariant errors between the true state and its estimate in invariant filtering, similar to the uncertainty representation in (2-4). These errors are referred to as the left and right-invariant errors and they are

$$\delta \mathbf{X}^L = \mathbf{X}^{-1} \hat{\mathbf{X}}, \quad (2-10)$$

$$\delta \mathbf{X}^R = \hat{\mathbf{X}} \mathbf{X}^{-1}, \quad (2-11)$$

respectively, where  $\mathbf{X}$  is the true state and  $\hat{\mathbf{X}}$  is the estimated state. Now going back to the approximation of (2-9), letting  $\mathbf{F}_{k-1}(\mathbf{X}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0})$  be the true state, then the perturbed state can be written as

$$\mathbf{X}_k = \mathbf{F}_{k-1}(\mathbf{X}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}) \exp(\mathbf{w}_{k-1}^\wedge), \quad (2-12)$$

for the left-invariant error. For the right-invariant error this approximation is

$$\mathbf{X}_k = \exp(\mathbf{w}_{k-1}^\wedge) \mathbf{F}_{k-1}(\mathbf{X}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}), \quad (2-13)$$

For the properties of the IEKF to hold, the dynamic model must be group affine. The group-affine definition involves neglecting the noise, meaning that the choice of model does not have an influence. Choosing the model (2-9), group affine function satisfies,

$$\mathbf{F}(\mathbf{X}_1 \mathbf{X}_2, \mathbf{u}, \mathbf{0}) = \mathbf{F}(\mathbf{X}_1, \mathbf{u}, \mathbf{0}) \mathbf{F}(\mathbf{I}, \mathbf{u}, \mathbf{0})^{-1} \mathbf{F}(\mathbf{X}_2, \mathbf{u}, \mathbf{0}), \quad (2-14)$$

where the subscripts  $k$  have been omitted to ease the notation.

The prediction step of the IEKF is given by

$$\check{\mathbf{X}}_k = \mathbf{F}_{k-1}(\hat{\mathbf{X}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}), \quad (2-15)$$

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1} \hat{\mathbf{P}}_{k-1} \mathbf{F}_{k-1}^\top + \mathbf{L}_{k-1} \mathbf{Q}_{k-1} \mathbf{L}_{k-1}^\top. \quad (2-16)$$

It should be noted here that  $\mathbf{F}_{k-1}(\cdot)$  is a function taking matrices as arguments and  $\mathbf{F}_{k-1}$  is a Jacobian matrix, which should not be confused with each other. The update step of the IEKF is performed when measurements are available, just as for the EKF. The invariant error definitions have been introduced, however, it is still unclear when to use what error definition. This depends on the type of measurement model, there are two types of measurement models, the left and right-invariant measurement models and they are considered to be

$$\mathbf{y}_k^L = \mathbf{X}_k \mathbf{b}_k + \mathbf{e}_k, \quad (2-17)$$

$$\mathbf{y}_k^R = \mathbf{X}_k^{-1} \mathbf{b}_k + \mathbf{e}_k, \quad (2-18)$$

respectively, where  $\mathbf{b}_k$  is some known vector and  $\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ . When confronted with a left-invariant measurement model and a group affine process model, the left-invariant error should be used. The same goes for the right-invariant error, when confronted with a right-invariant measurement model. The Kalman gain is computed using

$$\mathbf{K}_k = \check{\mathbf{P}}_k \mathbf{H}_k^\top \left( \mathbf{H}_k \check{\mathbf{P}}_k \mathbf{H}_k^\top + \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top \right)^{-1}. \quad (2-19)$$

One might argue that since the measurement model for the MEKF and IEKF are the same, that therefore  $\mathbf{H}_k$  and  $\mathbf{M}_k$  will be the same for both filters. This is, however, not the case, since by algebra the measurement model for the IEKF gets altered to fit either the left or the right-invariant measurement model while ensuring that the state  $\mathbf{X}_k$  is an element of the matrix Lie group. The covariance is updated using

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \check{\mathbf{P}}_k (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^\top + \mathbf{K}_k \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top \mathbf{K}_k^\top. \quad (2-20)$$

Detailed applications of the IEKF are worked out and discussed in Chapter 4

## 2-4 Related Work

This chapter serves to familiarize the reader with the current existing literature on the topic of the IEKF and MEKF for pose estimation related to the work presented in this thesis. This thesis considers pose estimation employed by the IEKF and MEKF on a 3D sample problem using IMU sensors and GPS or landmark measurements.

In [33], it has been discussed how to employ the IEKF using an accelerometer, gyroscope and two position receivers to estimate position, velocity and attitude with the IEKF framework. Since the estimation encompasses attitude, position but also velocity, the relevant matrix Lie group of the study presented in [33] is the group of double direct isometries,  $SE_2(3)$ . This is the same Lie group that is considered in this thesis and an overview of this group is presented in Appendix B-2. To the best of the author's knowledge, there are limited published works available on the topic of state estimation using this particular Lie group, consisting of [34, 35, 36]. However, in [34] the focus does not lie on invariant filtering in this Lie group, but rather on solving the problem regarding linear quadratic regulator control on a quadrotor, which is not related to the work considered in this thesis. Additionally, in [35] the work is focused on the Strapdown Inertial Navigation System (SINS) for the initial alignment based on the group  $SE_2(3)$ . Although, the use of the invariant framework turned out to be helpful for their situation, especially if the initial misalignment of the attitude was large, which is in agreement with findings reported in [8, 24], this work is not closely related to the work considered in this thesis.

The study described in [33] is among the few research efforts that specifically investigate the group  $SE_2(3)$  and is closely related to the scope of this thesis. They consider two position measurements, this corresponds to having two position receivers fixed on a rigid body, where each of position measurement is of the form

$$\mathbf{y}_k = \mathbf{r}_{n_k} + \mathbf{e}_k,$$

where  $\mathbf{r}_{n_k}$  denotes the position of the body and  $\mathbf{e}_k$  is a noise term, this model is identical to the one considered in this thesis and is discussed in more detail in chapter 3. This measurement model is left-invariant as it corresponds to the structure (2-17). In [33] only a left-invariant EKF is considered and it is compared to the MEKF. The LIEKF is derived in continuous-time. The reported results show that the LIEKF shows improved performance, meaning that it shows a lower estimation error in the attitude, velocity and position. This is consistent with other literature [8].

Another study that considers the group  $SE_2(3)$  and being closely related to this thesis is presented in [36]. The presented study focuses on using the IEKF on the estimation sideslip angle in a vehicle and compare its performance to the performance of an EKF. The application of this study is not relevant for this thesis, however, the problem setup and conclusions drawn in the study are valuable and provide insight in the performance of the IEKF compared to the EKF. The study in [36] employs an LIEKF to an estimation problem where IMU measurements, consisting of an accelerometer and a gyroscope, and Global Navigation Satellite System (GNSS) measurements, which in their case provides measurement updates for the velocity and position. The reported results are that the IEKF shows little improvement over the EKF when all states for both methods are well estimated or calibrated. Nonetheless, the

authors in [36] highlighted that the IEKF offers the advantage of reduced initialization requirements and improved convergence efficiency. This is attributed to the fact that the state propagation in the IEKF is independent of the state estimate. This is again in agreement with the findings of [8].

In [37], a quaternion based IEKF is presented for spacecraft attitude estimation. In the other studies and also in this thesis the orientation was parametrized in the form of rotation matrices as in Appendix A. However, [37] chooses a parametrization in the form of quaternions. For attitude estimation in 3D, the corresponding matrix Lie group is  $SO(3)$  as presented in Appendix B-1. This is not the Lie group that is used in [37], rather a corresponding Lie group suited for quaternions described by  $\mathbb{S}^3 \times \mathbb{R}^3 \in \mathcal{G}$ , where  $\mathbb{S}^3$  is the set of unit quaternions [37, 38]. The available measurements are assumed to be the angular velocity from three-axis gyroscopes and vector measurements from attitude sensors, such as sun sensors, magnetometers and star trackers. This study has two interesting aspects, firstly, the gyroscopes are assumed to be biased and bias estimation is performed and secondly, both an LIEKF and an RIEKF are designed and compared to an MEKF. The first aspect ensures the fact that because bias estimation is included in the filters, the dynamical model is not group affine. As was discussed in section 2-3, since the dynamical is not group affine, it is not guaranteed that the process Jacobian  $\mathbf{F}$  is state-estimate independent. The second aspect is interesting due to the fact that both the left-invariant as the right-invariant EKF are employed on the same system. This means that at best only one of these two filters will have guaranteed measurement model Jacobians  $\mathbf{H}$  since for the other system the measurement model will not have the corresponding invariant structure. The available measurement are coming from attitude sensors which are inherently in body frame. The RIEKF consists of a transformation matrix taking the measurements from body frame to navigation frame. The LIEKF consists of a transformation matrix that takes the measurements from navigation frame to body frame. Due to these transformations, this system with the attitude sensor measurements, will be a more natural fit for the RIEKF, since the LIEKF has to transform the measurements to navigation frame in order for it to fit the LIEKF structure. This also explains the reported results in which it was found that the RIEKF showed less dependence on the estimated trajectory and because of this it also showed more accurate estimation and faster convergence to the true trajectory than the LIEKF and MEKF. Whereas the LIEKF and MEKF showed very comparable results.

This thesis has defined two different MEKFs, the difference between these two filters is the way the orientation deviation is defined, in one filter the orientation deviation is defined in the body frame and in the other filter this is defined in navigation frame, referred to as MEKF-b and MEKF-n, respectively. To the authors best knowledge, this distinction has not yet been made in the context of comparison of the IEKF and MEKF for pose estimation. In [39], a multiplicative extended Kalman filter (MEKF) is proposed for estimating the relative state of a multicopter vehicle when operating in an environment where GPS measurements are unavailable. The MEKF combines data from an inertial measurement unit and an altimeter with relative-pose updates obtained from either a keyframe-based visual odometry or a laser scan-matching algorithm. Since the global position and heading states of the vehicle cannot be directly observed without GPS measurements, the MEKF in this research estimates the state relative to a local frame that is aligned with the odometry keyframe. They derive the filter both for traditional dynamics defined with respect to an inertial frame, and for robocentric dynamics defined with respect to the vehicle's body frame. They reported that both filters

showed very similar results, but subtle differences were noticeable in the error dynamics.

---

# Chapter 3

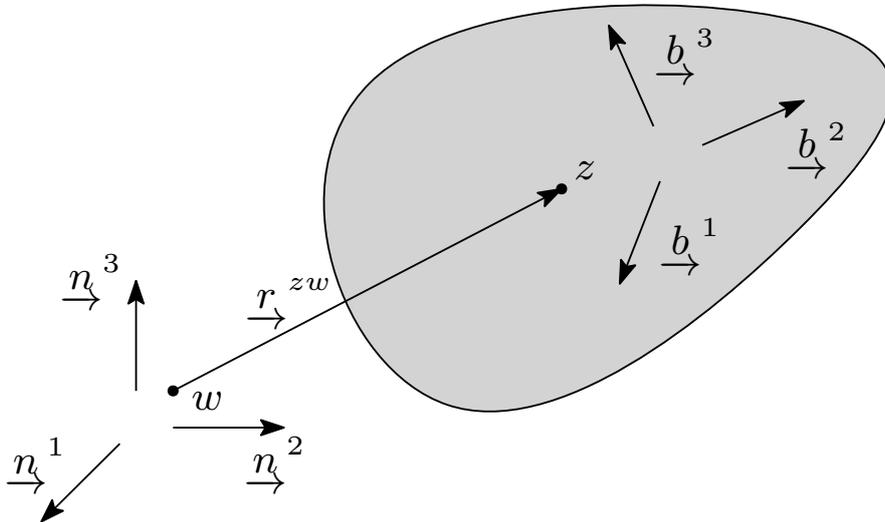
---

## Models

In this thesis, a sample problem is designed. The problem involves 3D pose estimation using IMUs, consisting of a gyroscope and an accelerometer, and considers both GPS measurements and relative position measurements.

A body is free to rotate and translate in 3D space. Let  $\mathcal{F}_n$  be the navigation frame and let frame  $\mathcal{F}_b$  be the frame that rotates with the body. Additionally, point  $w$  is a reference point and point  $z$  is a point attached to the body. The body is equipped with an IMU consisting of a gyroscope and an accelerometer.

A graphical representation of the situation is given in Figure 3-1.



**Figure 3-1:** Problem setup.

This research focuses on pose estimation, more specifically, the estimation of orientation, position and velocity. It is necessary to discuss the underlying models assumed on the dynamics and the incoming measurements in order to perform a sensible estimation on the before-mentioned states. This will be the focus of this section. The models that will be discussed

are the models for the IMU sensors, such as the gyroscope and accelerometer, the models describing the dynamics of the system and the measurement models, such as global position measurements and position measurements relative to some known landmark.

### 3-1 Gyroscope Measurement Models

The gyroscope measures angular velocity  $\boldsymbol{\omega}_b^{bn}$  of the body frame with respect to the navigation frame, resolved in the body frame, at each time instance  $k$ . It is assumed here that the travelled distances of the body are relatively small compared to the size of the earth and thus that the navigation frame can be assumed to be stationary. Additionally, the magnitude of the rotation of the earth is assumed to be fairly small compared to the magnitude of the actual measurements. The gyroscope measurements are assumed to be corrupted by noise  $\mathbf{w}_{b_k}^\omega$ , and are therefore modelled as [9]

$$\mathbf{u}_{b_k}^\omega = \boldsymbol{\omega}_{b_k}^{bn} - \mathbf{w}_{b_k}^\omega. \quad (3-1)$$

In [9] it was shown that the gyroscope measurement noise is relatively Gaussian and thus it is often assumed to be  $\mathbf{w}_{b_k}^\omega \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^\omega)$ . It is also assumed that the axes in the sensors are properly calibrated and that it thus can be assumed that [9]

$$\mathbf{Q}_k^\omega = \begin{bmatrix} \sigma_x^{\omega^2} & 0 & 0 \\ 0 & \sigma_y^{\omega^2} & 0 \\ 0 & 0 & \sigma_z^{\omega^2} \end{bmatrix}. \quad (3-2)$$

In this thesis it is assumed there is no bias on the gyroscope, as including bias estimation in the system would render the dynamical model no longer group affine [24].

### 3-2 Accelerometer Measurement Models

The accelerometer measures the specific force  $\mathbf{f}_{b_k}$  at each time instance  $k$ . These measurements corrupted by noise  $\mathbf{w}_{b_k}^a$  are modelled as [9]

$$\mathbf{u}_{b_k}^a = \mathbf{f}_{b_k} - \mathbf{w}_{b_k}^a. \quad (3-3)$$

Similar as for the gyroscope measurement noise, in [9] it was found that accelerometer measurement noise is relatively Gaussian and is assumed to be  $\mathbf{w}_{b_k}^a \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^a)$ , with  $\mathbf{Q}_k^a$  defined similar to (3-2), provided that the sensors are properly calibrated. It is assumed there is no bias on the accelerometer, as including bias estimation in the system would render the dynamical model no longer group affine [24]. It is further assumed that the magnitude of the Coriolis acceleration is small compared to the magnitude of the accelerometer measurements. The centrifugal acceleration is often absorbed into the local gravity vector. Incorporating this information into (3-3), yields [9]

$$\mathbf{u}_{b_k}^a = \mathbf{C}_{b_k n}(\mathbf{a}_{n_k} - \mathbf{g}_n) - \mathbf{w}_{b_k}^a, \quad (3-4)$$

where  $\mathbf{a}_{n_k}$  represents the linear acceleration in the navigation frame and  $\mathbf{g}_n$  is the gravity vector.

### 3-3 Dynamical models

The dynamical models for this problem are similar to ones considered in [40]. The continuous-time relations between position, velocity and acceleration are known as

$$\mathbf{v}_n^{zw/n} = \frac{d\mathbf{r}_n^{zw}}{dt}, \quad \mathbf{a}_n^{zw/n} = \frac{d\mathbf{v}_n^{zw/n}}{dt}, \quad (3-5)$$

the orientation and angular velocity are related as [9]

$$\frac{d\mathbf{C}_{nb}}{dt} = \mathbf{C}_{nb}\boldsymbol{\omega}_b^{bn^\times}, \quad (3-6)$$

where  $\mathbf{C}_{nb}$  represents the rotation matrix from the reference frame  $\mathcal{F}_b$  to  $\mathcal{F}_n$ , see also Appendix A. These relations can be discretized using an Euler discretization, which could be seen as a form of a Taylor expansion in which higher-order terms than quadratic are ignored, and assuming that the acceleration is constant between each time instance, the dynamics of the position and velocity can be expressed in terms of the acceleration,

$$\mathbf{r}_n^{z_k w} = \mathbf{r}_n^{z_{k-1} w} + T\mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2}\mathbf{a}_n^{z_{k-1} w/n}, \quad (3-7)$$

$$\mathbf{v}_n^{z_k w/n} = \mathbf{v}_n^{z_{k-1} w/n} + T\mathbf{a}_n^{z_{k-1} w/n}, \quad (3-8)$$

where  $T$  is the time step between two samples. For the dynamics of the orientation, the rotation matrices are updated as

$$\mathbf{C}_{nb_k} = \mathbf{C}_{nb_{k-1}} \exp_{SO(3)}(T\boldsymbol{\omega}_b^{bn^\times}), \quad (3-9)$$

where the subscript for the exponential map is used to clarify that it belongs to the matrix Lie group  $SO(3)$ , the special orthogonal group, an overview of this can be found in Appendix B-1. Dynamical models serve to describe the change a state of the system undergoes over time. The states to be estimated in this research are position, velocity and a parametrization of the orientation, in this case rotation matrices. The inertial measurements from the IMU can be used as inputs for the dynamic equation of the aforementioned states. Using (3-1) and (3-4) and use these as inputs for the dynamics of the system, results in the dynamical model being

$$\begin{bmatrix} \mathbf{r}_n^{z_k w} \\ \mathbf{v}_n^{z_k w/n} \\ \mathbf{C}_{nb_k} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_n^{z_{k-1} w} + T\mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2}(\mathbf{C}_{nb_{k-1}}(\mathbf{u}_{b_{k-1}}^a + \mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n) \\ \mathbf{v}_n^{z_{k-1} w/n} + T(\mathbf{C}_{nb_{k-1}}(\mathbf{u}_{b_{k-1}}^a + \mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n) \\ \mathbf{C}_{nb_{k-1}} \exp_{SO(3)}(T(\mathbf{u}_{b_{k-1}}^\omega + \mathbf{w}_{b_{k-1}}^\omega)^\times) \end{bmatrix}. \quad (3-10)$$

### 3-4 Position Measurements

Position measurements can be acquired from a Global Positioning System (GPS), or a Global Navigation Satellite System (GNSS), receiver for example. GPS refers specifically to the navigation system developed and operated by the United States government. It consists of a network of satellites orbiting the Earth that transmit signals to GPS receivers on the ground.

These signals are used to determine the receiver's position, velocity, and time information [41].

On the other hand, GNSS is a more general term that encompasses multiple satellite systems from different countries and organizations. In addition to GPS, GNSS includes other satellite systems such as GLONASS (Russia), Galileo (European Union), BeiDou (China), and NavIC (India). These systems operate independently but are compatible with each other, allowing receivers to utilize signals from multiple constellations [41].

The main difference, therefore, lies in the scope and coverage. GPS refers specifically to the American satellite system, while GNSS encompasses a broader range of satellite systems from different countries. By using multiple satellite constellations within the GNSS framework, receivers can access a larger number of satellites, leading to improved positioning accuracy, availability, and reliability [41]. In practice, many modern receivers support both GPS and GNSS, allowing them to utilize signals from multiple satellite systems.

A simplified model is assumed in this thesis, where the sensors directly measure the position, this model is given as [9]

$$\mathbf{y}_{n_k}^p = \mathbf{r}_n^{z_k w} + \mathbf{e}_{n_k}^p, \quad (3-11)$$

where  $\mathbf{r}_n^{z_k w}$  is the position of point  $z$  with respect to point  $w$  resolved in the navigation frame, this can graphically be seen in Figure 3-1. Additionally,  $\mathbf{e}_{n_k}^p \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ , where  $\mathbf{R}_k$  is a diagonal covariance matrix, similar to (3-2). These measurements will be referred to as position measurements or GPS measurements in the remainder of this thesis.

In order to perform Monte Carlo simulations, the initial error of the estimate is taken from a normal distribution with zero mean and a covariance matrix  $\mathbf{P}_0$ , such that  $\delta \mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_0)$  for the MEKF and  $\delta \boldsymbol{\xi}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_0)$  for the IEKF. These initial errors can then be used together with the true states in order to initialize the filters. For the MEKFs this means

$$\begin{aligned} \hat{\mathbf{r}}_n^{z_0 w} &= \mathbf{r}_n^{z_0 w} + \delta \mathbf{r}_0, \\ \hat{\mathbf{v}}_n^{z_0 w/n} &= \mathbf{v}_n^{z_0 w/n} + \delta \mathbf{v}_0, \\ \hat{\mathbf{C}}_{nb_0} &= \mathbf{C}_{nb_0} \exp_{SO(3)}(\delta \boldsymbol{\phi}_0). \end{aligned}$$

For the LIEKF and RIEKF this initialization is done by taking  $\delta \mathbf{T}_0 = \exp(\delta \hat{\boldsymbol{\xi}}_0)$  and then the initial state can be found using the appropriate error definition to get

$$\begin{aligned} \hat{\mathbf{T}}_0^L &= \mathbf{T}_0 \delta \mathbf{T}_0, \\ \hat{\mathbf{T}}_0^R &= \delta \mathbf{T}_0 \mathbf{T}_0. \end{aligned}$$

### 3-5 Landmark Position Measurements

Another form of position measurements are from Light Detection and Ranging (LIDAR), or stereo cameras, for example. The landmark position measurements give information on the distance and orientation with respect to some known landmarks. The difference with the position measurements from the previous subsection is that those measurements are in the navigation frame and the landmark position measurements give information of the position

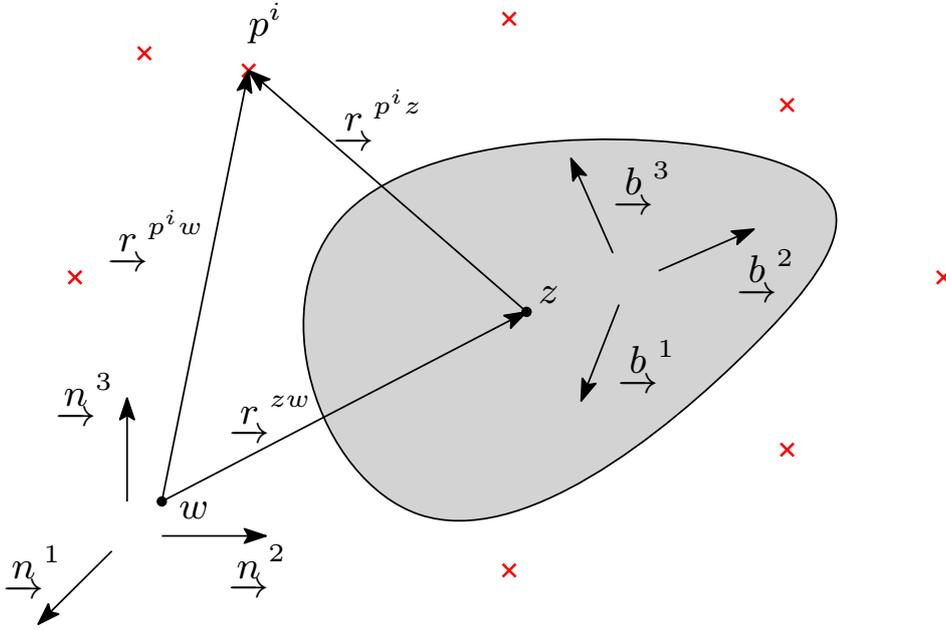
in body frame. The position of the  $i^{\text{th}}$  is denoted by  $p^i$ , the landmark position measurement model is then assumed to be [24]

$$\mathbf{y}_{b_k}^i = \mathbf{r}_b^{p^i z_k} + \mathbf{e}_{b_k}^i, \quad (3-12)$$

where  $\mathbf{r}_b^{p^i z}$  is the position of landmark  $i$  relative to point  $z$  resolved in the body frame and where  $\mathbf{e}_{b_k}^i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k^i)$  with  $\mathbf{R}_k^i$  a diagonal covariance matrix, similar to (3-2). To show the information these measurements have on directly on the orientation and position states, this model is written as

$$\mathbf{y}_{b_k}^i = \mathbf{C}_{nb_k}^T (\mathbf{r}_n^{p^i w} - \mathbf{r}_n^{z_k w}) + \mathbf{e}_{b_k}^i, \quad (3-13)$$

where  $\mathbf{r}_n^{p^i w}$  is the known position of the  $i^{\text{th}}$  landmark relative to point  $w$  in the navigation frame  $\mathcal{F}_n$ . A graphical representation is given in Figure 3-2.



**Figure 3-2:** Problem setup for pose estimation, red crossed represent landmarks.



---

# Chapter 4

---

## Method

The method chapter of this thesis aims to address the research questions posed at the beginning, specifically focusing on the comparison between the Invariant Extended Kalman Filter (IEKF) and the Multiplicative Extended Kalman Filter (MEKF). To investigate these questions, a simulated trajectory is used as the basis for the comparison.

The first question examines the benefits of using a left-Invariant Extended Kalman Filter (LIEKF) over an MEKF when dealing with a left-invariant measurement model, and similarly, whether employing a Right-Invariant Extended Kalman Filter (RIEKF) is more advantageous over an MEKF for a right-invariant measurement model. In order to do this, a distinction is made for the MEKF in the way the orientation deviation is defined. This can either be defined as the orientation deviation being resolved in the frame rotating with the body  $\mathcal{F}_b$ , further referred to as MEKF-b, or the orientation deviation being resolved in the navigation frame  $\mathcal{F}_n$ , further referred to as MEKF-n. This distinction is made since the MEKF-b is more closely related to the LIEKF, due to the nature of the left-invariant structure. The MEKF-n is more closely related to the RIEKF for similar reasons. In order to highlight the similarities between the IEKF and MEKF, the LIEKF will be compared to the MEKF-b and the RIEKF will be compared with the MEKF-n.

A sample problem is designed and simulated. The problem involves 3D pose estimation using IMUs, consisting of a gyroscope and an accelerometer, and considers both GPS measurements and relative position measurements, as described in chapter 3 and is graphically represented in Figure 3-1.

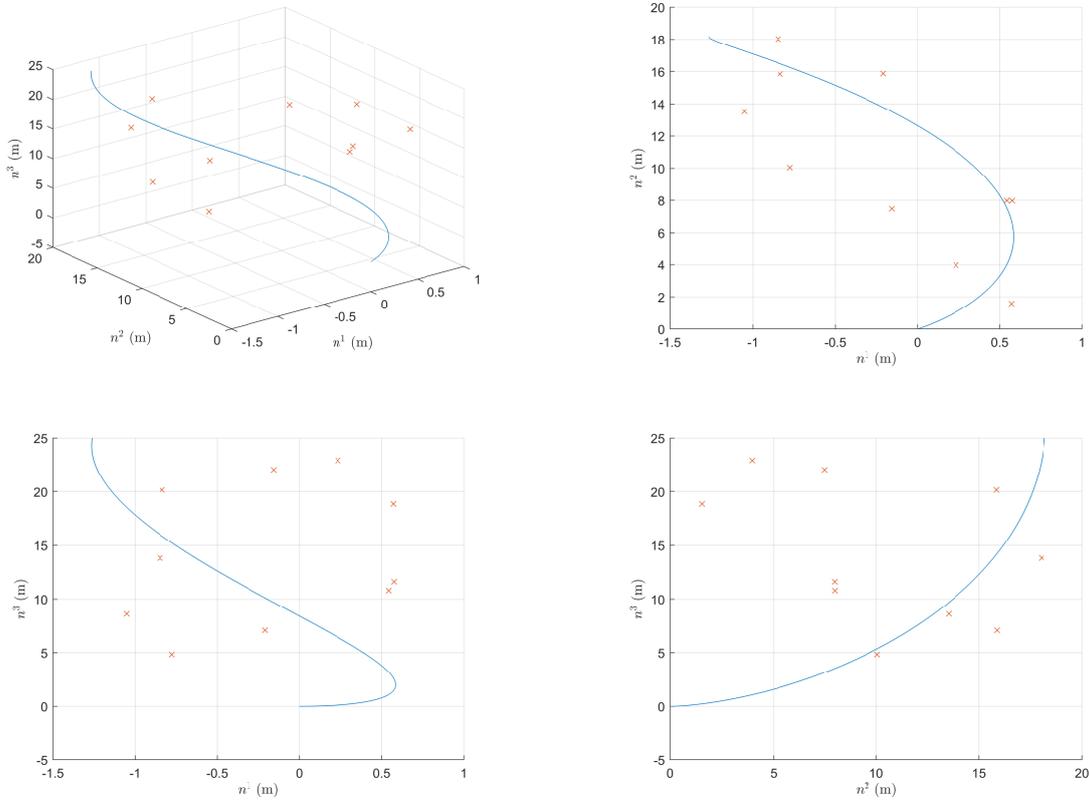
### 4-1 Simulation Setup

This section discusses how the simulation has been setup. A body is to fly around in 3D space, this could be similar to any unmanned aerial vehicle (UAV). To simulate the trajectory, an initial value has been set for the orientation, velocity and position. A constant value was chosen for the acceleration and angular velocity throughout the trajectory, thus the body was constrained to have a constant change in roll, pitch and yaw. Between each Monte

Carlo simulation this constant value would be drawn from a normal distribution with a small variance around a mean value  $\bar{\mathbf{a}}$  and  $\bar{\boldsymbol{\omega}}$ . The propagation of the trajectory was computed using the dynamical models (3-7), (3-8) and (3-9) for a time interval of 100 seconds and using chosen initial. The corresponding true orientation state was then computed at each time step from the rotation matrix through the logarithmic map by

$$\delta\boldsymbol{\xi}_k^{\phi^x} = \log_{SO(3)}(\mathbf{C}_{nb_k}).$$

Landmarks were placed along the trajectory. This was done by randomly placing them with the space enclosed by the minimum and maximum of the trajectory in each of the three axes. A typical trajectory with ten placed landmarks is shown in Figure 4-1.



**Figure 4-1:** A simulated trajectory. The ten landmarks are represented by the red crosses.

Due to the fact that the acceleration and the angular velocity are drawn from a normal distribution at each Monte Carlo simulation, this trajectory will not be identical for each Monte Carlo simulation.

The body is equipped with a gyroscope and an accelerometer sensor that both operate at 100 Hz. The IMU sensor data is simulated using the true angular velocity, acceleration and trajectory information with (3-1) and (3-4) and  $\mathbf{w}_{b_k}^\omega \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^\omega)$ , with  $\mathbf{Q}_k^\omega = \sigma_\omega^2 \mathbf{I}$  and  $\mathbf{w}_{b_k}^a \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^a)$  with  $\mathbf{Q}_k^a = \sigma_a^2 \mathbf{I}$ .

Corrective measurements from either the GPS or the relative position to known landmarks are received at 10 Hz. In order to simulate a visual limit for the body, a threshold value

of 35 meters for  $\|\mathbf{r}_b^{p^i z}\|$  has been set. This means that if the norm of the distance from the body to a landmark exceeds this threshold value, then this landmark will not be visible for the body to observe. This value was chosen in such a manner that the body will be able to observe multiple landmarks at each instance but not all of them at every instance. The GPS measurements are modelled using the true trajectory and (3-11) with  $\mathbf{e}_{n_k}^p \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$  where  $\mathbf{R}_k = \sigma_R^2 \mathbf{I}$ . The relative position measurements to known landmarks are modelled using the true trajectory and (3-13) with  $\mathbf{e}_{b_k}^i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$  where  $\mathbf{R}_k^i = \sigma_R^2 \mathbf{I}$ .

## 4-2 Left-Invariant Extended Kalman Filter Derivation

This section will present the implementation of the left-IEKF or LIEKF on pose estimation for the system models described in chapter 3. It will follow the reasoning presented in [15] in combination with [42].

The dynamical model (3-10) needs to be put into a form suitable for the IEKF, meaning it needs to be an element of a matrix Lie group. The matrix Lie group that encompasses position, velocity and rotation matrices is known as the matrix Lie group of double direct isometries,  $SE_2(3)$  [43]. The state matrix in  $SE_2(3)$  is denoted by  $\mathbf{T}$  and is given as

$$\mathbf{T}_k = \begin{bmatrix} \mathbf{C}_{nb_k} & \mathbf{v}_n^{z_k w/n} & \mathbf{r}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}. \quad (4-1)$$

For the derivation of the LIEKF, a left-invariant measurement model needs to be considered. Position measurements are suitable for this, for example from a GPS receiver, as described in (3-11). A measurement model yielding position measurements is given as

$$\mathbf{y}_{n_k}^p = \mathbf{r}_n^{z_k w} + \mathbf{e}_{n_k}^p, \quad (4-2)$$

$$= \mathbf{T}_k \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix}. \quad (4-3)$$

Comparing the structure of this measurement model to the one given in (2-17), this can be recognised as a left-invariant model, with  $\mathbf{X}_k = \mathbf{T}_k$ ,  $\mathbf{y}_k^L = \mathbf{y}_{n_k}^p$  and  $\mathbf{e}_{n_k} = \begin{bmatrix} \mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix}$ .

A noise matrix  $\mathbf{W}_{b_k} \in \mathfrak{se}_2(3)$  associated with the process noise  $\mathbf{w}_{b_k} = \begin{bmatrix} \mathbf{w}_{b_k}^{\omega \top} & \mathbf{w}_{b_k}^a \top & \mathbf{w}_{b_k}^a \top \end{bmatrix}^\top$  can be defined as

$$\begin{aligned} \mathbf{W}_{b_k} &= \begin{bmatrix} \mathbf{w}_{b_k}^{\omega \times} & \mathbf{w}_{b_k}^a & \mathbf{w}_{b_k}^a \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \\ &= \mathbf{w}_{b_k}^\wedge, \end{aligned}$$

where  $(\cdot)^\wedge$  is used as defined in Appendix B-2. As discussed in section 2-3, the dynamics can be approximated as (2-12), meaning the discrete time dynamical model can be written as the perturbed state

$$\mathbf{T}_k = \mathbf{F}(\mathbf{T}_{k-1}, \mathbf{u}_{k-1}) \exp(T \mathbf{w}_{b_{k-1}}^\wedge), \quad (4-4)$$

by letting  $\mathbf{F}(\mathbf{T}_{k-1}, \mathbf{u}_{k-1})$  be the true state. So the dynamics can be written as an element of  $SE_2(3)$ , by substituting in (3-10), as

$$\begin{aligned} \mathbf{T}_k &= \begin{bmatrix} \mathbf{C}_{nb_k} & \mathbf{v}_n^{z_k w/n} & \mathbf{r}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{C}_{nb_{k-1}} & \mathbf{v}_n^{z_{k-1} w/n} + T\mathbf{g}_n & \mathbf{r}_n^{z_{k-1} w} + T\mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2}\mathbf{g}_n \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \exp_{SO(3)}(T\mathbf{u}_{b_{k-1}}^{\omega^\times}) & T\mathbf{u}_{b_{k-1}}^a & \frac{T^2}{2}\mathbf{u}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \exp_{SO(3)}(T\mathbf{w}_{b_{k-1}}^{\omega^\times}) & T\mathbf{w}_{b_{k-1}}^a & \frac{T^2}{2}\mathbf{w}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \end{aligned}$$

Now, approximating the squared sampling time term times the accelerometer noise as just the sampling time term times the accelerometer noise, yields

$$\begin{aligned} \mathbf{T}_k &\approx \begin{bmatrix} \mathbf{C}_{nb_{k-1}} & \mathbf{v}_n^{z_{k-1} w/n} + T\mathbf{g}_n & \mathbf{r}_n^{z_{k-1} w} + T\mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2}\mathbf{g}_n \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \exp_{SO(3)}(T\mathbf{u}_{b_{k-1}}^{\omega^\times}) & T\mathbf{u}_{b_{k-1}}^a & \frac{T^2}{2}\mathbf{u}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \exp_{SO(3)}(T\mathbf{w}_{b_{k-1}}^{\omega^\times}) & T\mathbf{w}_{b_{k-1}}^a & T\mathbf{w}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \underbrace{\begin{bmatrix} \mathbf{C}_{nb_{k-1}} & \mathbf{v}_n^{z_{k-1} w/n} + T\mathbf{g}_n & \mathbf{r}_n^{z_{k-1} w} + T\mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2}\mathbf{g}_n \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}}_{\mathbf{F}(\mathbf{T}_{k-1})} \\ &= \underbrace{\begin{bmatrix} \exp_{SO(3)}(T\mathbf{u}_{b_{k-1}}^{\omega^\times}) & T\mathbf{u}_{b_{k-1}}^a & \frac{T^2}{2}\mathbf{u}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}}_{\mathbf{\Xi}_{k-1}} \exp(T\mathbf{w}_{b_{k-1}}^\wedge) \end{aligned} \quad (4-5)$$

$$= \mathbf{F}(\mathbf{T}_{k-1})\mathbf{\Xi}_{k-1} \exp(T\mathbf{w}_{b_{k-1}}^\wedge). \quad (4-6)$$

Here  $\mathbf{F}(\mathbf{T}_{k-1})$  and  $\mathbf{\Xi}_{k-1}$  are defined to indicate that one term is dependent on just the state  $\mathbf{T}_{k-1}$  and the other term is only dependent on the accelerometer and gyroscope inputs. From this derivation it can be concluded that, albeit not exact for these dynamical models, the dynamical models can be approximated to be equal to (4-4). As mentioned in section 2-3, for the IEKF to have error dynamics that are trajectory independent, the dynamical model must be group affine. Group affine systems satisfy (2-14) [15]. This equation (2-14) can be adapted to the matrix Lie group as

$$\mathbf{F}(\mathbf{T}_1\mathbf{T}_2, \mathbf{u}) = \mathbf{F}(\mathbf{T}_1, \mathbf{u})\mathbf{F}(\mathbf{T}_2, \mathbf{u}). \quad (4-7)$$

So first, it must be verified that this condition holds, in the following derivation notation

associated with the time step and the reference frames are omitted for clarity, let

$$\begin{aligned}\mathbf{T}_1\mathbf{T}_2 &= \begin{bmatrix} \mathbf{C}_1 & \mathbf{v}_1 & \mathbf{r}_1 \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{C}_2 & \mathbf{v}_2 & \mathbf{r}_2 \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{C}_1\mathbf{C}_2 & \mathbf{C}_1\mathbf{v}_2 + \mathbf{v}_1 & \mathbf{C}_1\mathbf{r}_2 + \mathbf{r}_1 \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}.\end{aligned}$$

Substituting this as the argument into  $\mathbf{F}(\mathbf{T}, \mathbf{u})$  as defined in (4-5), leads to

$$\begin{aligned}\mathbf{F}(\mathbf{T}_1\mathbf{T}_2, \mathbf{u}) &= \mathbf{F}(\mathbf{T}_1\mathbf{T}_2)\boldsymbol{\Xi}_{k-1} \\ &= \begin{bmatrix} \mathbf{C}_1\mathbf{C}_2 & \mathbf{C}_1\mathbf{v}_2 + \mathbf{v}_1 + T\mathbf{g} & \mathbf{C}_1\mathbf{r}_2 + \mathbf{r}_1 + T(\mathbf{C}_1\mathbf{v}_2 + \mathbf{v}_1) + \frac{T^2}{2}\mathbf{g} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \boldsymbol{\Xi}_{k-1}.\end{aligned}\quad (4-8)$$

Then, compute

$$\begin{aligned}\mathbf{F}(\mathbf{I}, \mathbf{u})^{-1} &= \left( \begin{bmatrix} \mathbf{I} & T\mathbf{g} & \frac{T^2}{2}\mathbf{g} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \boldsymbol{\Xi}_{k-1} \right)^{-1} \\ &= \boldsymbol{\Xi}_{k-1}^{-1} \begin{bmatrix} \mathbf{I} & -T\mathbf{g} & -\frac{T^2}{2}\mathbf{g} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix},\end{aligned}$$

substituting this into (4-7), yields

$$\begin{aligned}\mathbf{F}(\mathbf{T}_1\mathbf{T}_2, \mathbf{u}) &= \begin{bmatrix} \mathbf{C}_1 & \mathbf{v}_1 + T\mathbf{g} & \mathbf{r}_1 + T\mathbf{v}_1 + \frac{T^2}{2}\mathbf{g} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \boldsymbol{\Xi}_{k-1} \boldsymbol{\Xi}_{k-1}^{-1} \begin{bmatrix} \mathbf{I} & -T\mathbf{g} & -\frac{T^2}{2}\mathbf{g} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{C}_2 & \mathbf{v}_2 + T\mathbf{g} & \mathbf{r}_2 + T\mathbf{v}_2 + \frac{T^2}{2}\mathbf{g} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \boldsymbol{\Xi}_{k-1} \\ &= \begin{bmatrix} \mathbf{C}_1 & \mathbf{v}_1 + T\mathbf{g} & \mathbf{r}_1 + T\mathbf{v}_1 + \frac{T^2}{2}\mathbf{g} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{C}_2 & \mathbf{v}_2 & \mathbf{r}_2 + T\mathbf{v}_2 \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \boldsymbol{\Xi}_{k-1} \\ &= \begin{bmatrix} \mathbf{C}_1\mathbf{C}_2 & \mathbf{C}_1\mathbf{v}_2 + \mathbf{v}_1 + T\mathbf{g} & \mathbf{C}_1(\mathbf{r}_2 + T\mathbf{v}_2) + \mathbf{r}_1 + T\mathbf{v}_1 + \frac{T^2}{2}\mathbf{g} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \boldsymbol{\Xi}_{k-1},\end{aligned}$$

which is equal to (4-8), so (4-7) is satisfied and the dynamical model can be concluded to be group affine.

**Propagation** The time update is performed by using the left-invariant error definition, which for the case of the matrix Lie group  $SE_2(3)$  is equal to [42],

$$\delta\check{\mathbf{T}}_k = (\mathbf{T}_{nb_k})^{-1}\check{\mathbf{T}}_{nb_k} \quad (4-9)$$

$$\begin{aligned}&= (\mathbf{F}(\mathbf{T}_{k-1})\boldsymbol{\Xi}_{k-1} \exp(T\mathbf{w}_{b_{k-1}}^\wedge))^{-1}\mathbf{F}(\hat{\mathbf{T}}_{k-1})\boldsymbol{\Xi}_{k-1} \\ &= \exp(-T\mathbf{w}_{b_{k-1}}^\wedge)\boldsymbol{\Xi}_{k-1}^{-1}\mathbf{F}(\mathbf{T}_{k-1})^{-1}\mathbf{F}(\hat{\mathbf{T}}_{k-1})\boldsymbol{\Xi}_{k-1}.\end{aligned}\quad (4-10)$$

Taking the following left-invariant error definition

$$\begin{aligned}
\delta \mathbf{T}_k &= (\mathbf{T}_{nb_k})^{-1} \hat{\mathbf{T}}_{nb_k} \\
&= \begin{bmatrix} \mathbf{C}_{nb_k}^\top & -\mathbf{C}_{nb_k}^\top \mathbf{v}_n^{z_k w/n} & -\mathbf{C}_{nb_k}^\top \mathbf{r}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{C}}_{nb_k} & \hat{\mathbf{v}}_n^{z_k w/n} & \hat{\mathbf{r}}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{C}_{nb_k}^\top \hat{\mathbf{C}}_{nb_k} & \mathbf{C}_{nb_k}^\top (\hat{\mathbf{v}}_n^{z_k w/n} - \mathbf{v}_n^{z_k w/n}) & \mathbf{C}_{nb_k}^\top (\hat{\mathbf{r}}_n^{z_k w} - \mathbf{r}_n^{z_k w}) \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\
&= \exp(\delta \hat{\boldsymbol{\xi}}_k).
\end{aligned}$$

This way, the errors are defined to be

$$\begin{aligned}
\delta \mathbf{C}_k &= \mathbf{C}_{nb_k}^\top \hat{\mathbf{C}}_{nb_k} = \exp_{SO(3)}(\boldsymbol{\xi}^{\phi^\times}), \\
\delta \mathbf{v}_n^{z_k w/n} &= \mathbf{C}_{nb_k}^\top (\hat{\mathbf{v}}_n^{z_k w/n} - \mathbf{v}_n^{z_k w/n}) = \mathbf{J} \boldsymbol{\xi}^v, \\
\delta \mathbf{r}_n^{z_k w} &= \mathbf{C}_{nb_k}^\top (\hat{\mathbf{r}}_n^{z_k w} - \mathbf{r}_n^{z_k w}) = \mathbf{J} \boldsymbol{\xi}^r.
\end{aligned}$$

Then, to ease the notation, let

$$\mathbf{v}_{k-1} = \mathbf{v}_n^{z_{k-1} w/n} + T \mathbf{g}_n, \quad (4-11a)$$

$$\hat{\mathbf{v}}_{k-1} = \hat{\mathbf{v}}_n^{z_{k-1} w/n} + T \mathbf{g}_n, \quad (4-11b)$$

$$\mathbf{r}_{k-1} = \mathbf{r}_n^{z_{k-1} w} + T \mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2} \mathbf{g}_n, \quad (4-11c)$$

$$\hat{\mathbf{r}}_{k-1} = \hat{\mathbf{r}}_n^{z_{k-1} w} + T \hat{\mathbf{v}}_n^{z_{k-1} w/n} + \frac{T^2}{2} \mathbf{g}_n, \quad (4-11d)$$

such that the following yields,

$$\begin{aligned}
\mathbf{F}(\mathbf{T}_{k-1})^{-1} \mathbf{F}(\hat{\mathbf{T}}_{k-1}) &= \begin{bmatrix} \mathbf{C}_{nb_{k-1}}^\top & -\mathbf{C}_{nb_{k-1}}^\top \mathbf{v}_{k-1} & -\mathbf{C}_{nb_{k-1}}^\top \mathbf{r}_{k-1} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{C}}_{nb_{k-1}} & \hat{\mathbf{v}}_{k-1} & \hat{\mathbf{r}}_{k-1} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{C}_{nb_{k-1}}^\top \hat{\mathbf{C}}_{nb_{k-1}} & \delta \mathbf{F}_{12} & \delta \mathbf{F}_{13} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix},
\end{aligned}$$

where

$$\begin{aligned}
\delta \mathbf{F}_{12} &= \mathbf{C}_{nb_{k-1}}^\top (\hat{\mathbf{v}}_{k-1} - \mathbf{v}_{k-1}) \\
&= \mathbf{C}_{nb_{k-1}}^\top (\hat{\mathbf{v}}_n^{z_{k-1} w/n} - \mathbf{v}_n^{z_{k-1} w/n}) \\
&= \delta \mathbf{v}_n^{z_{k-1} w/n}, \\
\delta \mathbf{F}_{13} &= \mathbf{C}_{nb_{k-1}}^\top (\hat{\mathbf{r}}_{k-1} - \mathbf{r}_{k-1}) \\
&= \mathbf{C}_{nb_{k-1}}^\top (\hat{\mathbf{r}}_n^{z_{k-1} w} - \mathbf{r}_n^{z_{k-1} w} + T(\hat{\mathbf{v}}_n^{z_{k-1} w/n} - \mathbf{v}_n^{z_{k-1} w/n})) \\
&= \delta \mathbf{r}_n^{z_{k-1} w} + T \delta \mathbf{v}_n^{z_{k-1} w/n}.
\end{aligned}$$

This results in

$$\mathbf{F}(\mathbf{T}_{k-1})^{-1}\mathbf{F}(\hat{\mathbf{T}}_{k-1}) = \begin{bmatrix} \delta\mathbf{C}_{k-1} & \delta\mathbf{v}_n^{z_{k-1}w/n} & \delta\mathbf{r}_n^{z_{k-1}w} + T\delta\mathbf{v}_n^{z_{k-1}w/n} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}.$$

Substituting this expression into (4-10), defining  $\Psi_{k-1} = \exp_{SO(3)}(T\mathbf{u}_{b_{k-1}}^{\omega^\times})$  and dropping the notation on the reference frames in  $\delta\mathbf{v}_n^{z_{k-1}w/n}$  and  $\delta\mathbf{r}_n^{z_{k-1}w}$  such that these are denoted by  $\delta\mathbf{v}_{k-1}$  and  $\delta\mathbf{r}_{k-1}$ , respectively, for a more compact derivation, gives

$$\begin{aligned} \delta\check{\mathbf{T}}_k &= \exp(-T\mathbf{w}_{b_{k-1}}^\wedge)\Xi_{k-1}^{-1}\mathbf{F}(\mathbf{T}_{k-1})^{-1}\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1} \\ &= \exp(-T\mathbf{w}_{b_{k-1}}^\wedge) \begin{bmatrix} \Psi_{k-1}^\top & -T\Psi_{k-1}^\top\mathbf{u}_{b_{k-1}}^a & -\frac{T^2}{2}\Psi_{k-1}^\top\mathbf{u}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &\quad \begin{bmatrix} \delta\mathbf{C}_{k-1} & \delta\mathbf{v}_{k-1} & \delta\mathbf{r}_{k-1} + T\delta\mathbf{v}_{k-1} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \Psi_{k-1} & T\mathbf{u}_{b_{k-1}}^a & \frac{T^2}{2}\mathbf{u}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \exp(-T\mathbf{w}_{b_{k-1}}^\wedge) \begin{bmatrix} \Psi_{k-1}^\top & -T\Psi_{k-1}^\top\mathbf{u}_{b_{k-1}}^a & -\frac{T^2}{2}\Psi_{k-1}^\top\mathbf{u}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &\quad \begin{bmatrix} \delta\mathbf{C}_{k-1}\Psi_{k-1} & T\delta\mathbf{C}_{k-1}\mathbf{u}_{b_{k-1}}^a + \delta\mathbf{v}_{k-1} & \frac{T^2}{2}\delta\mathbf{C}_{k-1}\mathbf{u}_{b_{k-1}}^a + \delta\mathbf{r}_{k-1} + T\delta\mathbf{v}_{k-1} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \exp(-T\mathbf{w}_{b_{k-1}}^\wedge) \\ &\quad \begin{bmatrix} \Psi_{k-1}^\top\delta\mathbf{C}_{k-1}\Psi_{k-1} & T\Lambda + \Psi_{k-1}^\top\delta\mathbf{v}_{k-1} & \frac{T^2}{2}\Lambda + \Psi_{k-1}^\top(\delta\mathbf{r}_{k-1} + T\delta\mathbf{v}_{k-1}) \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}, \end{aligned}$$

where  $\Lambda = \Psi_{k-1}^\top(\delta\mathbf{C}_{k-1} - \mathbf{I})\mathbf{u}_{b_{k-1}}^a$ . This can be linearized by letting  $\delta\check{\mathbf{T}}_k \approx \mathbf{I} + \delta\check{\xi}_k^\wedge$ ,  $\delta\mathbf{C}_{k-1} \approx \mathbf{I} + \delta\check{\xi}_{k-1}^{\phi^\times}$ ,  $\delta\mathbf{v}_{k-1} = \mathbf{J}\delta\check{\xi}_{k-1}^v$ ,  $\delta\mathbf{r}_{k-1} = \mathbf{J}\delta\check{\xi}_{k-1}^r$ ,  $\mathbf{J} \approx \mathbf{I}$  and  $\exp(-T\mathbf{w}_{b_{k-1}}^\wedge) \approx \mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge$ , and ignoring terms of order  $\mathcal{O}(\|\delta\check{\xi}_{k-1}\|_2 \|\delta\mathbf{w}_{b_{k-1}}\|_2)$ . This results in

$$\begin{aligned} \mathbf{I} + \delta\check{\xi}_{k-1}^\wedge &= (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \begin{bmatrix} \Psi_{k-1}^\top(\mathbf{I} + \delta\check{\xi}_k^{\phi^\times})\Psi_{k-1} & \mathbf{Z}_{12} & \mathbf{Z}_{13} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \begin{bmatrix} \mathbf{I} + (\Psi_{k-1}^\top\delta\check{\xi}_{k-1}^{\phi^\times})^\times & \mathbf{Z}_{12} & \mathbf{Z}_{13} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}, \end{aligned}$$

where

$$\begin{aligned}
\mathbf{Z}_{12} &= T\Psi_{k-1}^\top \delta\check{\xi}_{k-1}^\phi \times \mathbf{u}_{b_{k-1}}^a + \Psi_{k-1}^\top \delta\check{\xi}_{k-1}^v \\
&= -T\Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times \delta\check{\xi}_{k-1}^\phi + \Psi_{k-1}^\top \delta\check{\xi}_{k-1}^v, \\
\mathbf{Z}_{13} &= \frac{T^2}{2} \Psi_{k-1}^\top \delta\check{\xi}_{k-1}^\phi \times \mathbf{u}_{b_{k-1}}^a + \Psi_{k-1}^\top (\delta\check{\xi}_{k-1}^r + T\delta\check{\xi}_{k-1}^v) \\
&= -\frac{T^2}{2} \Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times \delta\check{\xi}_{k-1}^\phi + \Psi_{k-1}^\top (\delta\check{\xi}_{k-1}^r + T\delta\check{\xi}_{k-1}^v).
\end{aligned}$$

This can then be continued by

$$\begin{aligned}
\mathbf{I} + \delta\check{\xi}_k^\wedge &= (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \left( \mathbf{I} - \begin{bmatrix} (\Psi_{k-1}^\top \delta\check{\xi}_{k-1}^\phi) \times & \mathbf{Z}_{12} & \mathbf{Z}_{13} \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \right) \\
&= (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \left( \mathbf{I} + \begin{bmatrix} \Psi_{k-1}^\top \delta\check{\xi}_{k-1}^\phi \\ \mathbf{Z}_{12} \\ \mathbf{Z}_{13} \end{bmatrix}^\wedge \right) \\
&= (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \left( \mathbf{I} + \begin{bmatrix} \Psi_{k-1}^\top \delta\check{\xi}_{k-1}^\phi \\ -T\Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times \delta\check{\xi}_{k-1}^\phi + \Psi_{k-1}^\top \delta\check{\xi}_{k-1}^v \\ -\frac{T^2}{2} \Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times \delta\check{\xi}_{k-1}^\phi + \Psi_{k-1}^\top (\delta\check{\xi}_{k-1}^r + T\delta\check{\xi}_{k-1}^v) \end{bmatrix}^\wedge \right) \\
&= (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \left( \mathbf{I} + \begin{bmatrix} \Psi_{k-1}^\top & \mathbf{0} & \mathbf{0} \\ -T\Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times & \Psi_{k-1}^\top & \mathbf{0} \\ -\frac{T^2}{2} \Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times & T\Psi_{k-1}^\top & \Psi_{k-1}^\top \end{bmatrix} \begin{bmatrix} \delta\check{\xi}_{k-1}^\phi \\ \delta\check{\xi}_{k-1}^v \\ \delta\check{\xi}_{k-1}^r \end{bmatrix} \right)^\wedge \\
&= (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) (\mathbf{I} + (\mathbf{F}_{k-1} \delta\check{\xi}_{k-1})^\wedge) \\
&\approx \mathbf{I} + (\mathbf{F}_{k-1} \delta\check{\xi}_{k-1})^\wedge - T\delta\mathbf{w}_{b_{k-1}}^\wedge
\end{aligned}$$

Then finally, the following dynamic equation can be found

$$\delta\check{\xi}_k = \mathbf{F}_{k-1} \delta\check{\xi}_{k-1} + \mathbf{L}_{k-1} \delta\mathbf{w}_{b_{k-1}}, \quad (4-12)$$

with

$$\mathbf{F}_k = \begin{bmatrix} \Psi_{k-1}^\top & \mathbf{0} & \mathbf{0} \\ -T\Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times & \Psi_{k-1}^\top & \mathbf{0} \\ -\frac{T^2}{2} \Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times & T\Psi_{k-1}^\top & \Psi_{k-1}^\top \end{bmatrix}, \quad (4-13)$$

and

$$\mathbf{L}_k = -T\mathbf{I} \quad (4-14)$$

**Measurement update** In order to update the state estimates, position measurements are

used and these are modelled as (4-3). The innovation is given by [15]

$$\begin{aligned} \mathbf{z}_k &= \check{\mathbf{T}}_k^{-1}(\mathbf{y}_{n_k}^p - \check{\mathbf{y}}_{n_k}^p) \\ &= \check{\mathbf{T}}_k^{-1}\left(\mathbf{T}_k \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix} - \check{\mathbf{T}}_k \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}\right) \\ &= \delta\check{\mathbf{T}}_k \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \check{\mathbf{T}}_k^{-1} \begin{bmatrix} \mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \end{aligned}$$

This can be linearized by letting  $\delta\check{\mathbf{T}}_k^{-1} \approx \mathbf{I} - \delta\check{\boldsymbol{\xi}}_k^\wedge$  and  $\mathbf{e}_{n_k}^p = \bar{\mathbf{e}}_{n_k}^p + \delta\mathbf{e}_{n_k}^p$  with  $\bar{\mathbf{e}}_{n_k}^p = \mathbf{0}$ , such that,

$$\begin{aligned} \mathbf{z}_k &\approx (\mathbf{I} - \delta\check{\boldsymbol{\xi}}_k^\wedge) \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \check{\mathbf{T}}_k^{-1} \begin{bmatrix} \delta\mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} \\ &= -\delta\check{\boldsymbol{\xi}}_k^\wedge \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \check{\mathbf{T}}_k^{-1} \begin{bmatrix} \delta\mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix} \\ &= -\begin{bmatrix} \delta\check{\boldsymbol{\xi}}_k^\phi & \delta\check{\boldsymbol{\xi}}_k^v & \delta\check{\boldsymbol{\xi}}_k^r \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \check{\mathbf{T}}_k^{-1} \begin{bmatrix} \delta\mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix} \\ &= \begin{bmatrix} -\delta\check{\boldsymbol{\xi}}_k^r \\ 0 \\ 0 \end{bmatrix} + \check{\mathbf{T}}_k^{-1} \begin{bmatrix} \delta\mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix} \\ &= \mathbf{H}_k \delta\check{\boldsymbol{\xi}}_k + \mathbf{M}_k \begin{bmatrix} \delta\mathbf{e}_{n_k}^p \\ \mathbf{0} \end{bmatrix}, \end{aligned} \tag{4-15}$$

where

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{4-16}$$

and

$$\mathbf{M}_k = \check{\mathbf{T}}_k^{-1}. \tag{4-17}$$

It can be noted that the bottom rows of the innovation (4-15) is always equal to 0, so it can be reduced to

$$\mathbf{z}_k = \mathbf{H}_k \delta\check{\boldsymbol{\xi}}_k + \mathbf{M}_k \delta\mathbf{e}_{n_k}^p, \tag{4-18}$$

where

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{I} \end{bmatrix}, \tag{4-19}$$

and

$$\mathbf{M}_k = \check{\mathbf{C}}_{nb_k}^\top. \tag{4-20}$$

A summary of the equations for the LIEKF when employed for pose estimation using position measurements, is given in Algorithm 1.

---

**Algorithm 1** Left-Invariant Extended Kalman Filter
 

---

**Inputs:** Measurement data  $\mathbf{y}_{n_k}^p$  and its covariance matrices.

**Outputs:** An estimate of the state matrix and the covariance matrix for  $k = 1, \dots, N$ .

---

1. Initialize with  $\mathbf{r}_n^{z_1 w}$ ,  $\mathbf{v}_n^{z_1 w/n}$ ,  $\hat{\mathbf{C}}_{nb_1}$  and  $\hat{\mathbf{P}}_1$
2. **for**  $k = 2, \dots, N$  **do**
  - (a) *Prediction.* Compute

$$\check{\mathbf{T}}_k = \mathbf{F}(\hat{\mathbf{T}}_{k-1})\check{\mathbf{\Xi}}_{k-1}, \quad (4-21a)$$

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{F}_{k-1}^\top + \mathbf{L}_{k-1}\hat{\mathbf{Q}}_{k-1}\mathbf{L}_{k-1}^\top, \quad (4-21b)$$

where  $\mathbf{F}(\mathbf{T}_{k-1})$  and  $\check{\mathbf{\Xi}}_{k-1}$  are defined as in (4-5),  $\mathbf{F}_{k-1}$  is defined as in (4-13) and  $\mathbf{L}_{k-1}$  is defined as in (4-14).

- (b) *Update.* Compute

$$\mathbf{K}_k = \check{\mathbf{P}}_k \mathbf{H}_k^\top \left( \mathbf{H}_k \check{\mathbf{P}}_k \mathbf{H}_k^\top + \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top \right)^{-1},$$

$$\mathbf{z}_k = \check{\mathbf{T}}_k^{-1} (\mathbf{y}_{n_k}^p - \check{\mathbf{y}}_{n_k}^p).$$

Then, to update the state matrix and the covariance estimate, compute

$$\hat{\mathbf{T}}_k = \check{\mathbf{T}}_k \exp(-(\mathbf{K}_k \mathbf{z}_k)^\wedge) \quad (4-22)$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \check{\mathbf{P}}_k (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^\top + \mathbf{K}_k \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top \mathbf{K}_k^\top, \quad (4-23)$$

where  $\mathbf{H}_k$  and  $\mathbf{M}_k$  are computed following (4-19) and (4-20), respectively.

---

### 4-3 Multiplicative EKF with Orientation Deviation States in Body Frame

This section will present the implementation of the MEKF for the use of pose estimation with the orientation deviation being defined as resolved in body frame, referred to as MEKF-b. This section will be based on the derivations and results shown in [40]. Instead of expressing the parametrization of the linearization point in quaternions, it will be expressed in terms of rotation matrices.

The orientation is modelled such that it is in terms of a linearization point parametrized as a rotation matrix  $\mathbf{C}_{nb_k}$  and an orientation deviation parametrized as a rotation vector  $\delta\phi_{b_k}$ . The orientation can be modelled as

$$\mathbf{C}_{nb_k} = \bar{\mathbf{C}}_{nb_k} \exp(\delta\phi_{b_k}^\times), \quad (4-24a)$$

and the linearization of the position and velocity states are modelled as

$$\mathbf{r}_n^{z_k w} = \bar{\mathbf{r}}_n^{z_k w} + \delta\mathbf{r}_n^{z_k w}, \quad (4-24b)$$

$$\mathbf{v}_n^{z_k w/n} = \bar{\mathbf{v}}_n^{z_k w/n} + \delta\mathbf{v}_n^{z_k w/n}. \quad (4-24c)$$

**Propagation** The time update is performed by using the dynamical model and updating the linearization point. Firstly, a linearized dynamic equation for the position will be derived, this is done by using the dynamics for the position state in (3-10) and substituting in (4-24), which gives

$$\begin{aligned} \bar{\mathbf{r}}_n^{z_k w} + \delta\mathbf{r}_n^{z_k w} &= \bar{\mathbf{r}}_n^{z_{k-1} w} + \delta\mathbf{r}_n^{z_{k-1} w} + T(\bar{\mathbf{v}}_n^{z_{k-1} w/n} + \delta\mathbf{v}_n^{z_{k-1} w/n}) \\ &\quad + \frac{T^2}{2}(\bar{\mathbf{C}}_{nb_{k-1}} \exp(\delta\phi_{b_{k-1}}^\times)(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a + \delta\mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n) \\ &\approx \bar{\mathbf{r}}_n^{z_{k-1} w} + \delta\mathbf{r}_n^{z_{k-1} w} + T\bar{\mathbf{v}}_n^{z_{k-1} w/n} + T\delta\mathbf{v}_n^{z_{k-1} w/n} \\ &\quad + \frac{T^2}{2}(\bar{\mathbf{C}}_{nb_{k-1}}(\mathbf{I} + \delta\phi_{b_{k-1}}^\times)(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a + \delta\mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n) \\ &= \bar{\mathbf{r}}_n^{z_{k-1} w} + \delta\mathbf{r}_n^{z_{k-1} w} + T\bar{\mathbf{v}}_n^{z_{k-1} w/n} + T\delta\mathbf{v}_n^{z_{k-1} w/n} \\ &\quad + \frac{T^2}{2}\mathbf{g}_n + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\delta\mathbf{w}_{b_{k-1}}^a \\ &\quad + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\delta\phi_{b_{k-1}}^\times(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\delta\phi_{b_{k-1}}^\times\delta\mathbf{w}_{b_{k-1}}^a, \end{aligned}$$

where  $\mathbf{w}_{b_k}^a = \bar{\mathbf{w}}_{b_k}^a + \delta\mathbf{w}_{b_k}^a$  with  $\bar{\mathbf{w}}_{b_k}^a = \mathbf{0}$ , which is consistent with  $\mathbf{w}_{b_k}^a \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^a)$ . Ignoring terms of order  $\mathcal{O}(\|\delta\phi_{b_{k-1}}\|_2 \|\delta\mathbf{w}_{b_{k-1}}^a\|_2)$  and subtracting the nominal dynamical equation

$$\bar{\mathbf{r}}_n^{z_k w} = \bar{\mathbf{r}}_n^{z_{k-1} w} + T\bar{\mathbf{v}}_n^{z_{k-1} w/n} + \frac{T^2}{2}(\bar{\mathbf{C}}_{nb_{k-1}}(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + \mathbf{g}_n), \quad (4-25)$$

then yields the following linearized dynamic equation for position,

$$\begin{aligned} \delta\mathbf{r}_n^{z_k w} &\approx \delta\mathbf{r}_n^{z_{k-1} w} + T\delta\mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\delta\phi_{b_{k-1}}^\times\mathbf{u}_{b_{k-1}}^a + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\delta\mathbf{w}_{b_{k-1}}^a \\ &= \delta\mathbf{r}_n^{z_{k-1} w} + T\delta\mathbf{v}_n^{z_{k-1} w/n} - \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^a \times \delta\phi_{b_{k-1}} + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\delta\mathbf{w}_{b_{k-1}}^a \end{aligned} \quad (4-26)$$

Following a similar procedure, a linearized dynamic equation for the velocity will be derived. To do so, substitute (4-24) into the dynamics of the velocity state in (3-10), this gives

$$\begin{aligned}
\bar{\mathbf{v}}_n^{z_k w/n} + \delta \mathbf{v}_n^{z_k w/n} &= \bar{\mathbf{v}}_n^{z_{k-1} w/n} + \delta \mathbf{v}_n^{z_{k-1} w/n} \\
&\quad + T(\bar{\mathbf{C}}_{nb_{k-1}} \exp(\delta \phi_{b_{k-1}}^\times))(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a + \delta \mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n \\
&\approx \bar{\mathbf{v}}_n^{z_{k-1} w/n} + \delta \mathbf{v}_n^{z_{k-1} w/n} \\
&\quad + T(\bar{\mathbf{C}}_{nb_{k-1}}(\mathbf{I} + \delta \phi_{b_{k-1}}^\times))(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a + \delta \mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n \\
&= \bar{\mathbf{v}}_n^{z_{k-1} w/n} + \delta \mathbf{v}_n^{z_{k-1} w/n} + T \mathbf{g}_n + T \bar{\mathbf{C}}_{nb_{k-1}}(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a \\
&\quad + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \phi_{b_{k-1}}^\times (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \phi_{b_{k-1}}^\times \delta \mathbf{w}_{b_{k-1}}^a.
\end{aligned}$$

Again ignoring terms of order  $\mathcal{O}(\|\delta \phi_{b_{k-1}}\|_2 \|\delta \mathbf{w}_{b_{k-1}}^a\|_2)$  and subtracting the nominal dynamical equation

$$\bar{\mathbf{v}}_n^{z_k w/n} = \bar{\mathbf{v}}_n^{z_{k-1} w/n} + T(\bar{\mathbf{C}}_{nb_{k-1}}(\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + \mathbf{g}_n), \quad (4-27)$$

the following linearized dynamic equation for the velocity state can be derived,

$$\begin{aligned}
\delta \mathbf{v}_n^{z_k w/n} &\approx \delta \mathbf{v}_n^{z_{k-1} w/n} + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \phi_{b_{k-1}}^\times \mathbf{u}_{b_{k-1}}^a \\
&= \delta \mathbf{v}_n^{z_{k-1} w/n} - T \bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a \times \delta \phi_{b_{k-1}} + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a.
\end{aligned} \quad (4-28)$$

In the MEKF the time update for the orientation deviation is performed by using the dynamical model to update the linearization point as

$$\check{\mathbf{C}}_{nb_k} = \hat{\mathbf{C}}_{nb_{k-1}} \exp(T \mathbf{u}_{b_{k-1}}^{\omega \times}). \quad (4-29)$$

To derive the linearized dynamic equation for the orientation deviation, take the dynamics for the rotation matrix in (3-10) and substitute in (4-24a) and (4-29), which gives

$$\begin{aligned}
\bar{\mathbf{C}}_{nb_k} \exp(\delta \phi_{b_k}^\times) &= \bar{\mathbf{C}}_{nb_{k-1}} \exp(\delta \phi_{b_{k-1}}^\times) \exp(T(\mathbf{u}_{b_{k-1}}^\omega + \mathbf{w}_{b_{k-1}}^\omega)^\times), \\
\exp(\delta \phi_{b_k}^\times) &= \bar{\mathbf{C}}_{nb_k}^{-1} \bar{\mathbf{C}}_{nb_{k-1}} \exp(\delta \phi_{b_{k-1}}^\times) \exp(T(\mathbf{u}_{b_{k-1}}^\omega + \mathbf{w}_{b_{k-1}}^\omega)^\times) \\
&= \exp(T \mathbf{u}_{b_k}^\times)^{-1} \exp(\delta \phi_{b_{k-1}}^\times) \exp(T(\mathbf{u}_{b_{k-1}}^\omega + \mathbf{w}_{b_{k-1}}^\omega)^\times) \\
&= \exp(T \mathbf{u}_{b_k}^\times)^{-1} \exp(\delta \phi_{b_{k-1}}^\times) \exp(T \mathbf{u}_{b_{k-1}}^{\omega \times}) + \exp(T \mathbf{w}_{b_{k-1}}^{\omega \times})
\end{aligned}$$

using (2-1) and also using from section B-1 that the adjoint representation of an element of  $SO(3)$  is equal to that same element of  $SO(3)$ , this can be further simplified to

$$\exp(\delta \phi_{b_k}^\times) = \exp((\exp(T \mathbf{u}_{b_{k-1}}^{\omega \times})^{-1} \delta \phi_{b_{k-1}}^\times)^\times) + \exp(T \mathbf{w}_{b_{k-1}}^{\omega \times}).$$

Then, using the BCH formula yields,

$$\begin{aligned}
\delta \phi_{b_k}^\times &= (\exp(T \mathbf{u}_{b_{k-1}}^{\omega \times})^{-1} \delta \phi_{b_{k-1}}^\times)^\times + T \mathbf{w}_{b_{k-1}}^{\omega \times}, \\
\delta \phi_{b_k} &= \exp(-T \mathbf{u}_{b_{k-1}}^{\omega \times}) \delta \phi_{b_{k-1}} + T \mathbf{w}_{b_{k-1}}^\omega
\end{aligned} \quad (4-30)$$

where  $\mathbf{w}_{b_k}^\omega = \bar{\mathbf{w}}_{b_k}^\omega + \delta\mathbf{w}_{b_k}^\omega$  with  $\bar{\mathbf{w}}_{b_k}^\omega = \mathbf{0}$ , which is consistent with  $\mathbf{w}_{b_k}^\omega \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^\omega)$ .

Combining the linearized dynamic equations, gives the following linearized dynamical model,

$$\begin{aligned} \begin{bmatrix} \delta\mathbf{r}_n^{z_k w} \\ \delta\mathbf{v}_n^{z_k w/n} \\ \delta\phi_{b_k} \end{bmatrix} &= \begin{bmatrix} \delta\mathbf{r}_n^{z_{k-1} w} + T\delta\mathbf{v}_n^{z_{k-1} w/n} - \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^{a^\times} \delta\phi_{b_{k-1}} + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\delta\mathbf{w}_{b_{k-1}}^a \\ \delta\mathbf{v}_n^{z_{k-1} w/n} - T\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^{a^\times} \delta\phi_{b_{k-1}} + T\bar{\mathbf{C}}_{nb_{k-1}}\delta\mathbf{w}_{b_{k-1}}^a \\ \exp(-T\mathbf{u}_{b_{k-1}}^{\omega^\times})\delta\phi_{b_{k-1}} + T\delta\mathbf{w}_{b_{k-1}}^\omega \end{bmatrix} \\ &= \mathbf{F}_k \begin{bmatrix} \delta\mathbf{r}_n^{z_{k-1} w} \\ \delta\mathbf{v}_n^{z_{k-1} w/n} \\ \delta\phi_{b_{k-1}} \end{bmatrix} + \mathbf{L}_k \begin{bmatrix} \delta\mathbf{w}_{b_{k-1}}^a \\ \delta\mathbf{w}_{b_{k-1}}^\omega \end{bmatrix}. \end{aligned}$$

An expression for the Jacobians can then be found, these are given by

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & T\mathbf{I} & -\frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^{a^\times} \\ \mathbf{0} & \mathbf{I} & -T\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^{a^\times} \\ \mathbf{0} & \mathbf{0} & \exp(-T\mathbf{u}_{b_{k-1}}^{\omega^\times}) \end{bmatrix}, \quad (4-31)$$

and

$$\mathbf{L}_k = \begin{bmatrix} \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}} & \mathbf{0} \\ T\bar{\mathbf{C}}_{nb_{k-1}} & \mathbf{0} \\ \mathbf{0} & T\mathbf{I} \end{bmatrix} \quad (4-32)$$

**Measurement update** To update the state estimates, position measurement are used. These measurements are given by (4-2). Then substituting in (4-24b) and using that  $\mathbf{e}_{n_k}^p = \bar{\mathbf{e}}_{n_k}^p + \delta\mathbf{e}_{n_k}^p$  with  $\bar{\mathbf{e}}_{n_k}^p = \mathbf{0}$ , which is consistent with  $\mathbf{e}_{n_k}^p \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ , yields

$$\bar{\mathbf{y}}_k^p + \delta\mathbf{y}_{n_k}^p = \bar{\mathbf{r}}_n^{z_k w} + \delta\mathbf{r}_n^{z_k w} + \bar{\mathbf{e}}_{n_k}^p + \delta\mathbf{e}_{n_k}^p,$$

and subtracting the nominal equation  $\bar{\mathbf{y}}_k^p = \bar{\mathbf{r}}_n^{z_k w}$  then gives

$$\begin{aligned} \delta\mathbf{y}_{n_k}^p &= \delta\mathbf{r}_n^{z_k w} + \delta\mathbf{e}_{n_k}^p \\ &= \mathbf{H}_k \begin{bmatrix} \delta\mathbf{r}_n^{z_{k-1} w} \\ \delta\mathbf{v}_n^{z_{k-1} w/n} \\ \delta\phi_{b_{k-1}} \end{bmatrix} + \mathbf{M}_k \delta\mathbf{e}_{n_k}^p. \end{aligned} \quad (4-33)$$

From this equation an expression for the Jacobians  $\mathbf{H}_k$  and  $\mathbf{M}_k$  can be found, which are given by

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (4-34)$$

and

$$\mathbf{M}_k = \mathbf{I}, \quad (4-35)$$

respectively.

After the measurement update, the linearization point will be updated. In [40], the relinearization is considered as the measurement update of the linearization point, so this updates  $\check{\mathbf{C}}_{nb_k}$  to  $\hat{\mathbf{C}}_{nb_k}$  as

$$\hat{\mathbf{C}}_{nb_k} = \check{\mathbf{C}}_{nb_k} \exp(\delta \hat{\boldsymbol{\phi}}_{b_k}^\times). \quad (4-36)$$

The update for the position and velocity states are performed in a manner that is identical as for the original EKF. A summary of the equations for the MEKF with orientation deviation states in body frame when employed for pose estimation, is given in Algorithm 2.

## 4-4 Right-Invariant Extended Kalman Filter Derivation

The implementation of the right-IEKF or RIEKF on pose estimation is discussed in this section. It follows the reasoning presented in [15] in combination with [42]. The dynamical model (3-10) will again be used. As was discussed in section 4-2, the matrix Lie group suitable for estimating position, velocity and rotation matrices, is denoted by  $SE_2(3)$ . The state matrix  $\mathbf{T}_k$  is given as (4-1).

For the derivation of the RIEKF, a right-invariant measurement model needs to be considered. Relative position measurements relative to the body, for example measurements from a LIDAR or camera measurements of known landmarks relative to the body are suitable for this. A measurement model yielding relative position measurements has been discussed in 3-5 and is given as

$$\mathbf{y}_{b_k}^i = \mathbf{C}_{nb_k}^\top (\mathbf{r}_n^{p^i w} - \mathbf{r}_n^{z_k w}) + \mathbf{e}_{b_k}^i. \quad (4-42)$$

Comparing the structure of this measurement model to the one given in (2-18), this can be recognised as a right-invariant model.

For the RIEKF, the dynamical model can be approximated as (2-13), doing so means that the dynamical model can be written as the perturbed state

$$\mathbf{T}_k = \exp(T \mathbf{w}_{b_{k-1}}^\wedge) \mathbf{F}(\mathbf{T}_{k-1}, \mathbf{u}_{k-1}), \quad (4-43)$$

by letting  $\mathbf{F}(\mathbf{T}_{k-1}, \mathbf{u}_{k-1})$  be the true state. So the dynamics (3-10) are written as

$$\begin{aligned} \mathbf{T}_k &= \begin{bmatrix} \mathbf{C}_{nb_k} & \mathbf{v}_n^{z_k w/n} & \mathbf{r}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \exp(T \mathbf{w}_{b_{k-1}}^\wedge) \underbrace{\begin{bmatrix} \mathbf{C}_{nb_{k-1}} & \mathbf{v}_n^{z_{k-1} w/n} + T \mathbf{g}_n & \mathbf{r}_n^{z_{k-1} w} + T \mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2} \mathbf{g}_n \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}}_{\mathbf{F}(\mathbf{T}_{k-1})} \\ &\quad \underbrace{\begin{bmatrix} \exp_{SO(3)}(T \mathbf{u}_{b_{k-1}}^{\omega \times}) & T \mathbf{u}_{b_{k-1}}^a & \frac{T^2}{2} \mathbf{u}_{b_{k-1}}^a \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}}_{\boldsymbol{\Xi}_{k-1}} \end{aligned} \quad (4-44)$$

$$= \exp(T \mathbf{w}_{b_{k-1}}^\wedge) \mathbf{F}(\mathbf{T}_{k-1}) \boldsymbol{\Xi}_{k-1}. \quad (4-45)$$

---

**Algorithm 2** Multiplicative EKF with orientation deviation states in body frame
 

---

**Inputs:** Measurement data  $\mathbf{y}_k$  and its covariance matrices.

**Outputs:** An estimate of position, velocity, the rotation matrix and the covariance matrix for  $k = 1, \dots, N$ .

---

1. Initialize with  $\mathbf{r}_n^{z_1 w}$ ,  $\mathbf{v}_n^{z_1 w/n}$ ,  $\hat{\mathbf{C}}_{nb_1}$  and  $\hat{\mathbf{P}}_1$
2. **for**  $k = 2, \dots, N$  **do**
  - (a) *Prediction.* Compute

$$\check{\mathbf{r}}_n^{z_k w} = \hat{\mathbf{r}}_n^{z_{k-1} w} + T \hat{\mathbf{v}}_n^{z_{k-1} w/n} + \frac{T^2}{2} (\hat{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a + \mathbf{g}_n), \quad (4-37a)$$

$$\check{\mathbf{v}}_n^{z_k w/n} = \hat{\mathbf{v}}_n^{z_{k-1} w/n} + T (\hat{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a + \mathbf{g}_n), \quad (4-37b)$$

$$\check{\mathbf{C}}_{nb_k} = \hat{\mathbf{C}}_{nb_{k-1}} \exp(T \mathbf{u}_{b_{k-1}}^\times), \quad (4-37c)$$

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1} \hat{\mathbf{P}}_{k-1} \mathbf{F}_{k-1}^\top + \mathbf{L}_{k-1} \hat{\mathbf{Q}}_{k-1} \mathbf{L}_{k-1}^\top, \quad (4-37d)$$

where  $\mathbf{F}_{k-1}$  is defined as in (4-31) and  $\mathbf{L}_{k-1}$  is defined as in (4-32).

- (b) *Update.* Compute

$$\mathbf{K}_k = \check{\mathbf{P}}_k \mathbf{H}_k^\top (\mathbf{H}_k \check{\mathbf{P}}_k \mathbf{H}_k^\top + \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top)^{-1},$$

$$\mathbf{z}_k = \mathbf{y}_k - \mathbf{h}(\check{\mathbf{C}}_{nb_k}).$$

Then, to update the state and covariance estimate, compute

$$\delta \hat{\boldsymbol{\xi}}_k = \begin{bmatrix} \delta \hat{\boldsymbol{\xi}}_k^r \\ \delta \hat{\boldsymbol{\xi}}_k^v \\ \delta \hat{\boldsymbol{\xi}}_k^\phi \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{r}}_n^{z_k w} - \check{\mathbf{r}}_n^{z_k w} \\ \hat{\mathbf{v}}_n^{z_k w/n} - \check{\mathbf{v}}_n^{z_k w/n} \\ \delta \hat{\phi}_{b_k} \end{bmatrix} \quad (4-38)$$

$$= \mathbf{K}_k \mathbf{z}_k, \quad (4-39)$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \check{\mathbf{P}}_k (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^\top + \mathbf{K}_k \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top \mathbf{K}_k^\top, \quad (4-40)$$

where  $\mathbf{H}_k$  and  $\mathbf{M}_k$  are computed following (4-34) and (4-35), respectively.

- (c) *Relinearization.* Compute

$$\hat{\mathbf{r}}_n^{z_k w} = \check{\mathbf{r}}_n^{z_k w} + \delta \hat{\boldsymbol{\xi}}_k^r, \quad (4-41a)$$

$$\hat{\mathbf{v}}_n^{z_k w/n} = \check{\mathbf{v}}_n^{z_k w/n} + \delta \hat{\boldsymbol{\xi}}_k^v, \quad (4-41b)$$

$$\hat{\mathbf{C}}_{nb_k} = \check{\mathbf{C}}_{nb_k} \exp(\delta \hat{\boldsymbol{\xi}}_k^\times). \quad (4-41c)$$


---

The measurement model (4-42) can be expressed in terms of the state matrix  $\mathbf{T}_k$  as

$$\begin{aligned} \begin{bmatrix} \mathbf{y}_{b_k}^i \\ 0 \\ 1 \end{bmatrix} &= \begin{bmatrix} \mathbf{C}_{nb_k}^\top & -\mathbf{C}_{nb_k}^\top \mathbf{v}_n^{z_k w/n} & -\mathbf{C}_{nb_k}^\top \mathbf{r}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} \\ &= \mathbf{T}_k^{-1} \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix}. \end{aligned} \quad (4-46)$$

This measurement model is of the form (2-18), which shows that this model is indeed right-invariant and it is therefore suitable for the RIEKF.

**Propagation** Equivalent to its left-invariant counterpart, the time update is performed by using the right-invariant error definition. Before doing this, the following right-invariant error definition,

$$\begin{aligned} \delta \mathbf{T}_k &= \hat{\mathbf{T}}_{nb_k} (\mathbf{T}_{nb_k})^{-1} \\ &= \begin{bmatrix} \hat{\mathbf{C}}_{nb_k} & \hat{\mathbf{v}}_n^{z_k w/n} & \hat{\mathbf{r}}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{C}_{nb_k}^\top & -\mathbf{C}_{nb_k}^\top \mathbf{v}_n^{z_k w/n} & -\mathbf{C}_{nb_k}^\top \mathbf{r}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \hat{\mathbf{C}}_{nb_k} \mathbf{C}_{nb_k}^\top & \hat{\mathbf{v}}_n^{z_k w/n} - \hat{\mathbf{C}}_{nb_k} \mathbf{C}_{nb_k}^\top \mathbf{v}_n^{z_k w/n} & \hat{\mathbf{r}}_n^{z_k w} - \hat{\mathbf{C}}_{nb_k} \mathbf{C}_{nb_k}^\top \mathbf{r}_n^{z_k w} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \exp(\delta \hat{\boldsymbol{\xi}}_k^\wedge), \end{aligned}$$

ensures that the following errors can be defined

$$\begin{aligned} \delta \mathbf{C}_k &= \hat{\mathbf{C}}_{nb_k} \mathbf{C}_{nb_k}^\top = \exp_{SO(3)}(\boldsymbol{\xi}^{\phi^\times}), \\ \delta \hat{\mathbf{v}}_n^{z_k w/n} &= \hat{\mathbf{v}}_n^{z_k w/n} - \hat{\mathbf{C}}_{nb_k} \mathbf{C}_{nb_k}^\top \mathbf{v}_n^{z_k w/n} = \mathbf{J} \boldsymbol{\xi}^v, \\ \delta \hat{\mathbf{r}}_n^{z_k w} &= \hat{\mathbf{r}}_n^{z_k w} - \hat{\mathbf{C}}_{nb_k} \mathbf{C}_{nb_k}^\top \mathbf{r}_n^{z_k w} = \mathbf{J} \boldsymbol{\xi}^r. \end{aligned}$$

Now, for the time update, the right-invariant error definition in the case of the matrix Lie group  $SE_2(3)$  is used to find an expression for the state update. This error definition is given as

$$\delta \check{\mathbf{T}}_k = \check{\mathbf{T}}_{nb_k} (\mathbf{T}_{nb_k})^{-1} \quad (4-47)$$

$$\begin{aligned} &= \mathbf{F}(\hat{\mathbf{T}}_{k-1}) \boldsymbol{\Xi}_{k-1} (\exp(T \mathbf{w}_{b_{k-1}}^\wedge) \mathbf{F}(\mathbf{T}_{k-1}) \boldsymbol{\Xi}_{k-1})^{-1} \\ &= \mathbf{F}(\hat{\mathbf{T}}_{k-1}) \mathbf{F}(\mathbf{T}_{k-1})^{-1} \exp(-T \mathbf{w}_{b_{k-1}}^\wedge), \end{aligned} \quad (4-48)$$

where  $\mathbf{F}(\hat{\mathbf{T}}_{k-1})$  is defined as in (4-44). Using the definitions in (4-11) to ease the notation, the following expression can be derived,

$$\begin{aligned} \mathbf{F}(\hat{\mathbf{T}}_{k-1}) \mathbf{F}(\mathbf{T}_{k-1})^{-1} &= \begin{bmatrix} \hat{\mathbf{C}}_{nb_{k-1}} & \hat{\mathbf{v}}_{k-1} & \hat{\mathbf{r}}_{k-1} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{C}_{nb_{k-1}}^\top & -\mathbf{C}_{nb_{k-1}}^\top \mathbf{v}_{k-1} & -\mathbf{C}_{nb_{k-1}}^\top \mathbf{r}_{k-1} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \delta \mathbf{C}_k & \delta \mathbf{F}_{12} & \delta \mathbf{F}_{13} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}, \end{aligned} \quad (4-49)$$

where

$$\begin{aligned}
\delta \mathbf{C}_k &= \hat{\mathbf{C}}_{nb_{k-1}} \mathbf{C}_{nb_{k-1}}^\top, \\
\delta \mathbf{F}_{12} &= \hat{\mathbf{v}}_{k-1} - \mathbf{C}_{nb_{k-1}}^\top \mathbf{C}_{nb_{k-1}}^\top \mathbf{v}_{k-1} \\
&= \hat{\mathbf{v}}_n^{z_{k-1}w/n} + T \mathbf{g}_n - \mathbf{C}_{nb_{k-1}}^\top \mathbf{C}_{nb_{k-1}}^\top (\mathbf{v}_n^{z_{k-1}w/n} + T \mathbf{g}_n) \\
&= \delta \mathbf{v}_n^{z_{k-1}w/n} + T \mathbf{g}_n - T \mathbf{C}_{nb_{k-1}}^\top \mathbf{C}_{nb_{k-1}}^\top \mathbf{g}_n \\
&= \delta \mathbf{v}_n^{z_{k-1}w/n} + T(\mathbf{I} - \delta \mathbf{C}_k) \mathbf{g}_n, \\
\delta \mathbf{F}_{13} &= \hat{\mathbf{r}}_{k-1} - \mathbf{C}_{nb_{k-1}}^\top \mathbf{C}_{nb_{k-1}}^\top \mathbf{r}_{k-1} \\
&= \hat{\mathbf{r}}_n^{z_{k-1}w} + T \hat{\mathbf{v}}_n^{z_{k-1}w/n} + \frac{T^2}{2} \mathbf{g}_n - \mathbf{C}_{nb_{k-1}}^\top \mathbf{C}_{nb_{k-1}}^\top (\mathbf{r}_n^{z_{k-1}w} + T \mathbf{v}_n^{z_{k-1}w/n} + \frac{T^2}{2} \mathbf{g}_n) \\
&= \delta \mathbf{r}_n^{z_{k-1}w} + T \delta \mathbf{v}_n^{z_{k-1}w/n} + \frac{T^2}{2} (\mathbf{I} - \delta \mathbf{C}_k) \mathbf{g}_n.
\end{aligned}$$

Substituting these definitions into (4-48), yields

$$\delta \check{\mathbf{T}}_k = \begin{bmatrix} \delta \mathbf{C}_k & \delta \mathbf{v}_n^{z_{k-1}w/n} + T(\mathbf{I} - \delta \mathbf{C}_k) \mathbf{g}_n & \delta \mathbf{r}_n^{z_{k-1}w} + T \delta \mathbf{v}_n^{z_{k-1}w/n} + \frac{T^2}{2} (\mathbf{I} - \delta \mathbf{C}_k) \mathbf{g}_n \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \exp(-T \mathbf{w}_{b_{k-1}}^\wedge)$$

This can be linearized by letting  $\delta \check{\mathbf{T}}_k \approx \mathbf{I} + \delta \check{\boldsymbol{\xi}}_k^\wedge$ ,  $\delta \mathbf{C}_{k-1} \approx \mathbf{I} + \delta \boldsymbol{\xi}_{k-1}^{\phi^\times}$ ,  $\delta \mathbf{v}_n^{z_{k-1}w/n} = \mathbf{J} \delta \boldsymbol{\xi}_{k-1}^v$ ,  $\delta \mathbf{r}_n^{z_{k-1}w} = \mathbf{J} \delta \boldsymbol{\xi}_{k-1}^r$ ,  $\mathbf{J} \approx \mathbf{I}$  and  $\exp(-T \mathbf{w}_{b_{k-1}}^\wedge) \approx \mathbf{I} - T \delta \mathbf{w}_{b_{k-1}}^\wedge$ , and ignoring terms of order

$\mathcal{O}(\|\delta\xi_{k-1}\|_2 \|\delta\mathbf{w}_{b_{k-1}}\|_2)$ . This results in

$$\begin{aligned}
\mathbf{I} + \delta\check{\xi}_k^\wedge &\approx \begin{bmatrix} \mathbf{I} + \delta\xi_{k-1}^{\phi^\times} & \delta\xi_{k-1}^v + T(-\delta\xi_{k-1}^{\phi^\times})\mathbf{g}_n & \delta\xi_{k-1}^r + T\delta\xi_{k-1}^v + \frac{T^2}{2}(-\delta\xi_{k-1}^{\phi^\times})\mathbf{g}_n \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\
&(\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \\
&= \left( \mathbf{I} + \begin{bmatrix} \delta\xi_{k-1}^{\phi^\times} & \delta\xi_{k-1}^v - T\delta\xi_{k-1}^{\phi^\times}\mathbf{g}_n & \delta\xi_{k-1}^r + T\delta\xi_{k-1}^v - \frac{T^2}{2}\delta\xi_{k-1}^{\phi^\times}\mathbf{g}_n \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \right) \\
&(\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \\
&= \left( \mathbf{I} + \begin{bmatrix} \delta\xi_{k-1}^{\phi^\times} & \delta\xi_{k-1}^v + T\mathbf{g}_n^\times\delta\xi_{k-1}^\phi & \delta\xi_{k-1}^r + T\delta\xi_{k-1}^v + \frac{T^2}{2}\mathbf{g}_n^\times\delta\xi_{k-1}^\phi \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \right) \\
&(\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \\
&= \left( \mathbf{I} + \begin{bmatrix} \delta\xi_{k-1}^\phi & & \\ \delta\xi_{k-1}^v + T\mathbf{g}_n^\times\delta\xi_{k-1}^\phi & & \\ \delta\xi_{k-1}^r + T\delta\xi_{k-1}^v + \frac{T^2}{2}\mathbf{g}_n^\times\delta\xi_{k-1}^\phi & & \end{bmatrix} \right)^\wedge (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \\
&= \left( \mathbf{I} + \left( \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ T\mathbf{g}_n^\times & \mathbf{I} & \mathbf{0} \\ \frac{T^2}{2}\mathbf{g}_n^\times & T\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \delta\xi_{k-1}^\phi \\ \delta\xi_{k-1}^v \\ \delta\xi_{k-1}^r \end{bmatrix} \right) \right)^\wedge (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \\
&= (\mathbf{I} + (\mathbf{F}_{k-1}\delta\check{\xi}_{k-1})^\wedge) (\mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge) \\
&\approx \mathbf{I} + (\mathbf{F}_{k-1}\delta\xi_{k-1})^\wedge - T\delta\mathbf{w}_{b_{k-1}}^\wedge
\end{aligned}$$

Then finally, the following dynamic equation can be found

$$\delta\check{\xi}_k = \mathbf{F}_{k-1}\delta\xi_{k-1} + \mathbf{L}_{k-1}\delta\mathbf{w}_{b_{k-1}}, \quad (4-50)$$

with

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ T\mathbf{g}_n^\times & \mathbf{I} & \mathbf{0} \\ \frac{T^2}{2}\mathbf{g}_n^\times & T\mathbf{I} & \mathbf{I} \end{bmatrix}, \quad (4-51)$$

and

$$\mathbf{L}_k = -T\mathbf{I} \quad (4-52)$$

Although (4-43) is mathematically consistent for a right-invariant error definition, this does not accurately describe the way noise enters the system. The dynamical system considered here has sensor noise coming in through the accelerometer and gyroscope, which are both resolved in the body frame  $\mathcal{F}_b$ . The right matrix multiplication in (4-43) between the noise term and the state, which has the rotation matrix  $\mathbf{C}_{nb_k}$  incorporated in it, is mathematically incorrect. As has been derived in section 4-2, the noise enters the system in a way more consistent with the left-invariant definition. So instead of using the right-invariant assumption

on the dynamical model, the left-invariant dynamical model assumption (4-4) will be used. The derivation using the left-invariant model assumption will be covered next.

Substituting (4-4) into the right-invariant error definition yields

$$\begin{aligned}
\delta\check{\mathbf{T}}_k &= \check{\mathbf{T}}_{nb_k}(\mathbf{T}_{nb_k})^{-1} & (4-53) \\
&= \mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1}(\mathbf{F}(\mathbf{T}_{k-1})\Xi_{k-1}\exp(T\mathbf{w}_{b_{k-1}}^\wedge))^{-1} \\
&= \mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1}\exp(-T\mathbf{w}_{b_{k-1}}^\wedge)(\mathbf{F}(\mathbf{T}_{k-1})\Xi_{k-1})^{-1} \\
&= \mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1}\exp(-T\mathbf{w}_{b_{k-1}}^\wedge)(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})^{-1}\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1}(\mathbf{F}(\mathbf{T}_{k-1})\Xi_{k-1})^{-1} \\
&= \exp(-T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\mathbf{w}_{b_{k-1}}^\wedge))\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1}(\mathbf{F}(\mathbf{T}_{k-1})\Xi_{k-1})^{-1} \\
&= \exp(-T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\mathbf{w}_{b_{k-1}}^\wedge))\mathbf{F}(\hat{\mathbf{T}}_{k-1})\mathbf{F}(\mathbf{T}_{k-1})^{-1}, & (4-54)
\end{aligned}$$

where the adjoint representation (2-1) was used. Substituting in (4-49) into (4-54)

$$\begin{aligned}
\delta\check{\mathbf{T}}_k &= \exp(-T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\mathbf{w}_{b_{k-1}}^\wedge)) \\
&\begin{bmatrix} \delta\mathbf{C}_k & \delta\mathbf{v}_n^{z_{k-1}w/n} + T(\mathbf{I} - \delta\mathbf{C}_k)\mathbf{g}_n & \delta\mathbf{r}_n^{z_{k-1}w} + T\delta\mathbf{v}_n^{z_{k-1}w/n} + \frac{T^2}{2}(\mathbf{I} - \delta\mathbf{C}_k)\mathbf{g}_n \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} & (4-55)
\end{aligned}$$

Again, this can be linearized by letting  $\delta\check{\mathbf{T}}_k \approx \mathbf{I} + \delta\check{\xi}_k^\wedge$ ,  $\delta\mathbf{C}_{k-1} \approx \mathbf{I} + \delta\xi_{k-1}^{\phi^\times}$ ,  $\delta\mathbf{v}_n^{z_{k-1}w/n} = \mathbf{J}\delta\xi_{k-1}^v$ ,  $\delta\mathbf{r}_n^{z_{k-1}w} = \mathbf{J}\delta\xi_{k-1}^r$ ,  $\mathbf{J} \approx \mathbf{I}$  and  $\exp(-T\mathbf{w}_{b_{k-1}}^\wedge) \approx \mathbf{I} - T\delta\mathbf{w}_{b_{k-1}}^\wedge$ , and ignoring terms of order

$\mathcal{O}(\|\delta\xi_{k-1}\|_2 \|\delta\mathbf{w}_{b_{k-1}}\|_2)$ . This results in

$$\begin{aligned}
\mathbf{I} + \delta\check{\xi}_k^\wedge &\approx (\mathbf{I} - T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\delta\mathbf{w}_{b_{k-1}})^\wedge) \\
&\quad \begin{bmatrix} \mathbf{I} + \delta\xi_{k-1}^{\phi^\times} & \delta\xi_{k-1}^v + T(-\delta\xi_{k-1}^{\phi^\times})\mathbf{g}_n & \delta\xi_{k-1}^r + T\delta\xi_{k-1}^v + \frac{T^2}{2}(-\delta\xi_{k-1}^{\phi^\times})\mathbf{g}_n \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \\
&= (\mathbf{I} - T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\delta\mathbf{w}_{b_{k-1}})^\wedge) \\
&\quad \left( \mathbf{I} + \begin{bmatrix} \delta\xi_{k-1}^{\phi^\times} & \delta\xi_{k-1}^v - T\delta\xi_{k-1}^{\phi^\times}\mathbf{g}_n & \delta\xi_{k-1}^r + T\delta\xi_{k-1}^v - \frac{T^2}{2}\delta\xi_{k-1}^{\phi^\times}\mathbf{g}_n \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \right) \\
&= (\mathbf{I} - T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\delta\mathbf{w}_{b_{k-1}})^\wedge) \\
&\quad \left( \mathbf{I} + \begin{bmatrix} \delta\xi_{k-1}^{\phi^\times} & \delta\xi_{k-1}^v + T\mathbf{g}_n^\times\delta\xi_{k-1}^\phi & \delta\xi_{k-1}^r + T\delta\xi_{k-1}^v + \frac{T^2}{2}\mathbf{g}_n^\times\delta\xi_{k-1}^\phi \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \right) \\
&= (\mathbf{I} - T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\delta\mathbf{w}_{b_{k-1}})^\wedge) \left( \mathbf{I} + \begin{bmatrix} \delta\xi_{k-1}^\phi \\ \delta\xi_{k-1}^v + T\mathbf{g}_n^\times\delta\xi_{k-1}^\phi \\ \delta\xi_{k-1}^r + T\delta\xi_{k-1}^v + \frac{T^2}{2}\mathbf{g}_n^\times\delta\xi_{k-1}^\phi \end{bmatrix}^\wedge \right) \\
&= (\mathbf{I} - T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\delta\mathbf{w}_{b_{k-1}})^\wedge) \left( \mathbf{I} + \left( \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ T\mathbf{g}_n^\times & \mathbf{I} & \mathbf{0} \\ \frac{T^2}{2}\mathbf{g}_n^\times & T\mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \delta\xi_{k-1}^\phi \\ \delta\xi_{k-1}^v \\ \delta\xi_{k-1}^r \end{bmatrix} \right)^\wedge \right) \\
&= (\mathbf{I} - T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\delta\mathbf{w}_{b_{k-1}})^\wedge) (\mathbf{I} + (\mathbf{F}_{k-1}\delta\check{\xi}_{k-1})^\wedge) \\
&\approx \mathbf{I} + (\mathbf{F}_{k-1}\delta\check{\xi}_{k-1})^\wedge - T(\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1})\delta\mathbf{w}_{b_{k-1}})^\wedge
\end{aligned}$$

Then finally, the following dynamic equation can be found

$$\delta\check{\xi}_k = \mathbf{F}_{k-1}\delta\check{\xi}_{k-1} + \mathbf{L}_{k-1}\delta\mathbf{w}_{b_{k-1}}, \quad (4-56)$$

with

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ T\mathbf{g}_n^\times & \mathbf{I} & \mathbf{0} \\ \frac{T^2}{2}\mathbf{g}_n^\times & T\mathbf{I} & \mathbf{I} \end{bmatrix}, \quad (4-57)$$

and

$$\mathbf{L}_k = -T\text{Ad}(\mathbf{F}(\hat{\mathbf{T}}_{k-1})\Xi_{k-1}) \quad (4-58)$$

**Measurement update** To update the state estimates, position measurements relative to known landmarks are used. These measurements are modelled as (4-46). Whenever such a measurement is available, the state estimate is corrected following

$$\hat{\mathbf{T}}_k = \exp(-(\mathbf{K}_k\mathbf{z}_k)^\wedge)\check{\mathbf{T}}_k.$$

The innovation for a right-invariant system, for the  $i^{\text{th}}$  landmark, is given by

$$\begin{aligned}
\begin{bmatrix} \mathbf{z}_k^i \\ 0 \\ 0 \end{bmatrix} &= \check{\mathbf{T}}_k \left( \begin{bmatrix} \mathbf{y}_{b_k}^i \\ 0 \\ 1 \end{bmatrix} - \begin{bmatrix} \check{\mathbf{y}}_{b_k}^i \\ 0 \\ 1 \end{bmatrix} \right) \\
&= \check{\mathbf{T}}_k \left( \mathbf{T}_k^{-1} \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} - \mathbf{T}_k^{-1} \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} \right) \\
&= \delta \mathbf{T}_k \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} + \check{\mathbf{T}}_k \begin{bmatrix} \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix}.
\end{aligned}$$

This can then be linearized by letting  $\delta \check{\mathbf{T}}_k \approx \mathbf{I} + \delta \check{\boldsymbol{\xi}}_k^\wedge$  and  $\mathbf{e}_{n_k}^i = \bar{\mathbf{e}}_{n_k}^i + \delta \mathbf{e}_{n_k}^i$  with  $\bar{\mathbf{e}}_{n_k}^i = 0$ , such that,

$$\begin{aligned}
\begin{bmatrix} \mathbf{z}_k^i \\ 0 \\ 0 \end{bmatrix} &\approx (\mathbf{I} + \delta \check{\boldsymbol{\xi}}_k^\wedge) \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} + \check{\mathbf{T}}_k \begin{bmatrix} \delta \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} \\
&= \delta \check{\boldsymbol{\xi}}_k^\wedge \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} + \check{\mathbf{T}}_k \begin{bmatrix} \delta \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} \delta \check{\boldsymbol{\xi}}_k^{\phi^\times} & \delta \check{\boldsymbol{\xi}}_k^v & \delta \check{\boldsymbol{\xi}}_k^r \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{r}_n^{p_i w} \\ 0 \\ 1 \end{bmatrix} + \check{\mathbf{T}}_k \begin{bmatrix} \delta \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{r}_n^{p_i w} \delta \check{\boldsymbol{\xi}}_k^{\phi^\times} + \delta \check{\boldsymbol{\xi}}_k^r \\ 0 \\ 0 \end{bmatrix} + \check{\mathbf{T}}_k \begin{bmatrix} \delta \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} -\mathbf{r}_n^{p_i w \times} \delta \check{\boldsymbol{\xi}}_k^\phi + \delta \check{\boldsymbol{\xi}}_k^r \\ 0 \\ 0 \end{bmatrix} + \check{\mathbf{T}}_k \begin{bmatrix} \delta \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} -\mathbf{r}_n^{p_i w \times} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \check{\boldsymbol{\xi}}_k + \check{\mathbf{T}}_k \begin{bmatrix} \delta \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix} \\
&= \mathbf{H}_k^i \check{\boldsymbol{\xi}}_k + \mathbf{M}_k^i \begin{bmatrix} \delta \mathbf{e}_{b_k}^i \\ 0 \\ 0 \end{bmatrix}, \tag{4-59}
\end{aligned}$$

where

$$\mathbf{H}_k^i = \begin{bmatrix} -\mathbf{r}_n^{p_i w \times} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{4-60}$$

and

$$\mathbf{M}_k^i = \check{\mathbf{T}}_k. \tag{4-61}$$

It can be noted that the bottom rows of the innovation (4-59) are always equal to 0, so it can be reduced to

$$\mathbf{z}_k^i = \mathbf{H}_k^i \delta \check{\boldsymbol{\xi}}_k + \mathbf{M}_k^i \delta \mathbf{e}_{n_k}^i, \quad (4-62)$$

where

$$\mathbf{H}_k^i = \begin{bmatrix} -\mathbf{r}_n^{p_i w^\times} & \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad (4-63)$$

and

$$\mathbf{M}_k^i = \check{\mathbf{C}}_{nb_k}. \quad (4-64)$$

For all  $m$  landmarks, these Jacobians then become,

$$\mathbf{H}_k = \underset{i=1, \dots, m}{\text{row}} \begin{bmatrix} -\mathbf{r}_n^{p_i w^\times} & \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad (4-65)$$

and

$$\mathbf{M}_k = \text{diag}(\check{\mathbf{C}}_{nb_k}, \dots, \check{\mathbf{C}}_{nb_k}). \quad (4-66)$$

A summary of the equations for the RIEKF when employed for pose estimation using position measurements relative to known landmark positions, is given in Algorithm 3.

## 4-5 Multiplicative EKF with Orientation Deviation States in Navigation Frame

This section will present the implementation of the MEKF for the use of pose estimation with the orientation deviation being defined as resolved in the navigation frame, referred to as MEKF-n. This section will be based on the derivations and results shown in [40, 9]. Instead of expressing the parametrization of the linearization point in quaternions, it will be expressed in terms of rotation matrices. Instead of using GPS measurements, for the derivation of this MEKF the relative position measurement model (4-42) will also be considered, in order to compare this MEKF to the RIEKF.

The orientation is modelled such that it is in terms of a linearization point parametrized as a rotation matrix  $\mathbf{C}_{nb_k}$  and an orientation deviation parametrized as a rotation vector  $\boldsymbol{\phi}_{n_k}$ . The orientation can be modelled as

$$\mathbf{C}_{nb_k} = \exp(\delta \boldsymbol{\phi}_{n_k}^\times) \bar{\mathbf{C}}_{nb_k}, \quad (4-70)$$

and the linearization of the position and velocity states are modelled as (4-24b) and (4-24c), respectively.

**Propagation** The time update is done by using the dynamical model and updating the linearization point. The linearized dynamic equation for the position will be derived first.

---

**Algorithm 3** Right-Invariant Extended Kalman Filter
 

---

**Inputs:** Measurement data  $\mathbf{y}_k^i$ , for  $i = 1, \dots, m$ , and its covariance matrices.

**Outputs:** An estimate of the state matrix and the covariance matrix for  $k = 1, \dots, N$ .

---

1. Initialize with  $\mathbf{r}_n^{z_1 w}$ ,  $\mathbf{v}_n^{z_1 w/n}$ ,  $\hat{\mathbf{C}}_{nb_1}$  and  $\hat{\mathbf{P}}_1$
2. **for**  $k = 2, \dots, N$  **do**
  - (a) *Prediction.* Compute

$$\check{\mathbf{T}}_k = \mathbf{F}(\hat{\mathbf{T}}_{k-1})\boldsymbol{\Xi}_{k-1}, \quad (4-67a)$$

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{F}_{k-1}^\top + \mathbf{L}_{k-1}\hat{\mathbf{Q}}_{k-1}\mathbf{L}_{k-1}^\top, \quad (4-67b)$$

where  $\mathbf{F}(\mathbf{T}_{k-1})$  and  $\boldsymbol{\Xi}_{k-1}$  are defined as in (4-44),  $\mathbf{F}_{k-1}$  is defined as in (4-51) and  $\mathbf{L}_{k-1}$  is defined as in (4-52).

- (b) *Update.* Compute

$$\mathbf{K}_k = \check{\mathbf{P}}_k \mathbf{H}_k^\top \left( \mathbf{H}_k \check{\mathbf{P}}_k \mathbf{H}_k^\top + \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top \right)^{-1},$$

$$\mathbf{z}_k = \check{\mathbf{T}}_k (\mathbf{y}_{n_k}^p - \check{\mathbf{y}}_{n_k}^p).$$

Then, to update the state matrix and the covariance estimate, compute

$$\hat{\mathbf{T}}_k = \exp(-(\mathbf{K}_k \mathbf{z}_k)^\wedge) \check{\mathbf{T}}_k \quad (4-68)$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \check{\mathbf{P}}_k (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^\top + \mathbf{K}_k \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top \mathbf{K}_k^\top, \quad (4-69)$$

where  $\mathbf{H}_k$  and  $\mathbf{M}_k$  are computed following (4-65) and (4-66), respectively.

---

This is done by substituting (4-70), (4-24b) and (4-24c) into the dynamics for the position in (3-10), which yields

$$\begin{aligned}
\bar{\mathbf{r}}_n^{z_k w} + \delta \mathbf{r}_n^{z_k w} &= \bar{\mathbf{r}}_n^{z_{k-1} w} + \delta \mathbf{r}_n^{z_{k-1} w} + T(\bar{\mathbf{v}}_n^{z_{k-1} w/n} + \delta \mathbf{v}_n^{z_{k-1} w/n}) \\
&\quad + \frac{T^2}{2} (\exp(\delta \phi_{n_{k-1}}^\times) \bar{\mathbf{C}}_{nb_{k-1}} (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a + \delta \mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n) \\
&\approx \bar{\mathbf{r}}_n^{z_{k-1} w} + \delta \mathbf{r}_n^{z_{k-1} w} + T \bar{\mathbf{v}}_n^{z_{k-1} w/n} + T \delta \mathbf{v}_n^{z_{k-1} w/n} \\
&\quad + \frac{T^2}{2} ((\mathbf{I} + \delta \phi_{n_{k-1}}^\times) \bar{\mathbf{C}}_{nb_{k-1}} (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a + \delta \mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n) \\
&= \bar{\mathbf{r}}_n^{z_{k-1} w} + \delta \mathbf{r}_n^{z_{k-1} w} + T \bar{\mathbf{v}}_n^{z_{k-1} w/n} + T \delta \mathbf{v}_n^{z_{k-1} w/n} \\
&\quad + \frac{T^2}{2} \mathbf{g}_n + \frac{T^2}{2} \bar{\mathbf{C}}_{nb_{k-1}} (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + \frac{T^2}{2} \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a \\
&\quad + \frac{T^2}{2} \delta \phi_{n_{k-1}}^\times \bar{\mathbf{C}}_{nb_{k-1}} (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + \frac{T^2}{2} \delta \phi_{n_{k-1}}^\times \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a,
\end{aligned}$$

where  $\mathbf{w}_{b_k}^a = \bar{\mathbf{w}}_{b_k}^a + \delta \mathbf{w}_{b_k}^a$  with  $\bar{\mathbf{w}}_{b_k}^a = \mathbf{0}$ , which is consistent with  $\mathbf{w}_{b_k}^a \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^a)$ . Ignoring terms of order  $\mathcal{O}(\|\delta \phi_{n_{k-1}}\|_2 \|\delta \mathbf{w}_{b_{k-1}}^a\|_2)$  and subtracting the nominal dynamical equation (4-25), the following linearized dynamic equation for position can be derived,

$$\begin{aligned}
\delta \mathbf{r}_n^{z_k w} &\approx \delta \mathbf{r}_n^{z_{k-1} w} + T \delta \mathbf{v}_n^{z_{k-1} w/n} + \frac{T^2}{2} \delta \phi_{n_{k-1}}^\times \bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a + \frac{T^2}{2} \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a \\
&= \delta \mathbf{r}_n^{z_{k-1} w} + T \delta \mathbf{v}_n^{z_{k-1} w/n} - \frac{T^2}{2} (\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a)^\times \delta \phi_{n_{k-1}} + \frac{T^2}{2} \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a \quad (4-71)
\end{aligned}$$

Following a similar procedure, a linearized dynamic equation for the velocity will be derived. To do so, substitute (4-70), (4-24b) and (4-24c) into the dynamics of the velocity state in (3-10), this gives

$$\begin{aligned}
\bar{\mathbf{v}}_n^{z_k w/n} + \delta \mathbf{v}_n^{z_k w/n} &= \bar{\mathbf{v}}_n^{z_{k-1} w/n} + \delta \mathbf{v}_n^{z_{k-1} w/n} \\
&\quad + T(\exp(\delta \phi_{n_{k-1}}^\times) \bar{\mathbf{C}}_{nb_{k-1}} (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a + \delta \mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n) \\
&\approx \bar{\mathbf{v}}_n^{z_{k-1} w/n} + \delta \mathbf{v}_n^{z_{k-1} w/n} \\
&\quad + T((\mathbf{I} + \delta \phi_{n_{k-1}}^\times) \bar{\mathbf{C}}_{nb_{k-1}} (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a + \delta \mathbf{w}_{b_{k-1}}^a) + \mathbf{g}_n) \\
&= \bar{\mathbf{v}}_n^{z_{k-1} w/n} + \delta \mathbf{v}_n^{z_{k-1} w/n} + T \mathbf{g}_n + T \bar{\mathbf{C}}_{nb_{k-1}} (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a \\
&\quad + T \delta \phi_{n_{k-1}}^\times \bar{\mathbf{C}}_{nb_{k-1}} (\mathbf{u}_{b_{k-1}}^a + \bar{\mathbf{w}}_{b_{k-1}}^a) + T \delta \phi_{n_{k-1}}^\times \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a.
\end{aligned}$$

Again ignoring terms of order  $\mathcal{O}(\|\delta \phi_{n_{k-1}}\|_2 \|\delta \mathbf{w}_{b_{k-1}}^a\|_2)$  and subtracting the nominal dynamical equation (4-27), the following linearized dynamic equation for the velocity state can be derived,

$$\begin{aligned}
\delta \mathbf{v}_n^{z_k w/n} &\approx \delta \mathbf{v}_n^{z_{k-1} w/n} + T \delta \phi_{n_{k-1}}^\times \bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a \\
&= \delta \mathbf{v}_n^{z_{k-1} w/n} - T (\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a)^\times \delta \phi_{n_{k-1}} + T \bar{\mathbf{C}}_{nb_{k-1}} \delta \mathbf{w}_{b_{k-1}}^a. \quad (4-72)
\end{aligned}$$

In the MEKF the time update for the orientation deviation is performed by using the dynamical model to update the linearization point as

$$\check{\mathbf{C}}_{nb_k} = \hat{\mathbf{C}}_{nb_{k-1}} \exp(T\mathbf{u}_{b_k}^\times). \quad (4-73)$$

To derive the linearized dynamic equation for the orientation deviation, take the dynamics for the rotation matrix in (3-10) and substitute in (4-70) and (4-73), which gives

$$\begin{aligned} \exp_{SO(3)}(\delta\phi_{n_k}^\times)\bar{\mathbf{C}}_{nb_k} &= \exp_{SO(3)}(\delta\phi_{n_k}^\times)\bar{\mathbf{C}}_{nb_{k-1}} \exp_{SO(3)}(T(\mathbf{u}_{b_{k-1}}^\omega + \mathbf{w}_{b_{k-1}}^\omega)^\times), \\ \exp_{SO(3)}(\delta\phi_{n_k}^\times) &= \exp_{SO(3)}(\delta\phi_{n_k}^\times)\bar{\mathbf{C}}_{nb_{k-1}} \exp_{SO(3)}(T(\mathbf{u}_{b_{k-1}}^\omega + \mathbf{w}_{b_{k-1}}^\omega)^\times)\bar{\mathbf{C}}_{nb_k}^{-1} \\ &= \exp_{SO(3)}(\delta\phi_{n_k}^\times)\bar{\mathbf{C}}_{nb_{k-1}} \exp_{SO(3)}(T(\mathbf{u}_{b_{k-1}}^\omega + \mathbf{w}_{b_{k-1}}^\omega)^\times) \exp_{SO(3)}(-T\mathbf{u}_{b_{k-1}}^\omega) \bar{\mathbf{C}}_{nb_{k-1}}^{-1} \\ &= \exp_{SO(3)}(\delta\phi_{n_k}^\times)\bar{\mathbf{C}}_{nb_{k-1}} \exp_{SO(3)}(T\mathbf{w}_{b_{k-1}}^\omega)^\times \bar{\mathbf{C}}_{nb_{k-1}}^{-1} \end{aligned}$$

using (2-1) and also using from section B-1 that the adjoint representation of an element of  $SO(3)$  is equal to that same element of  $SO(3)$ , this can be further simplified to

$$\exp(\delta\phi_{n_k}^\times) = \exp_{SO(3)}(\delta\phi_{n_k}^\times) \exp_{SO(3)}(T(\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{w}_{b_{k-1}}^\omega)^\times).$$

Then, using the BCH formula yields,

$$\begin{aligned} \delta\phi_{n_k}^\times &= \delta\phi_{n_k}^\times + T(\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{w}_{b_{k-1}}^\omega)^\times, \\ \delta\phi_{n_k} &= \delta\phi_{n_k} + T\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{w}_{b_{k-1}}^\omega \end{aligned} \quad (4-74)$$

where  $\mathbf{w}_{b_k}^\omega = \bar{\mathbf{w}}_{b_k}^\omega + \delta\mathbf{w}_{b_k}^\omega$  with  $\bar{\mathbf{w}}_{b_k}^\omega = \mathbf{0}$ , which is consistent with  $\mathbf{w}_{b_k}^\omega \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k^\omega)$ .

Combining the linearized dynamic equations, gives the following linearized dynamical model,

$$\begin{aligned} \begin{bmatrix} \delta\mathbf{r}_n^{z_k w} \\ \delta\mathbf{v}_n^{z_k w/n} \\ \delta\phi_{n_k} \end{bmatrix} &= \begin{bmatrix} \delta\mathbf{r}_n^{z_{k-1} w} + T\delta\mathbf{v}_n^{z_{k-1} w/n} - \frac{T^2}{2}(\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a)^\times \delta\phi_{n_{k-1}} + \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}} \delta\mathbf{w}_{b_{k-1}}^a \\ \delta\mathbf{v}_n^{z_{k-1} w/n} - T(\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a)^\times \delta\phi_{n_{k-1}} + T\bar{\mathbf{C}}_{nb_{k-1}} \delta\mathbf{w}_{b_{k-1}}^a \\ \delta\phi_{n_{k-1}} + T\bar{\mathbf{C}}_{nb_{k-1}} \delta\mathbf{w}_{b_{k-1}}^\omega \end{bmatrix} \\ &= \mathbf{F}_k \begin{bmatrix} \delta\mathbf{r}_n^{z_{k-1} w} \\ \delta\mathbf{v}_n^{z_{k-1} w/n} \\ \delta\phi_{n_{k-1}} \end{bmatrix} + \mathbf{L}_k \begin{bmatrix} \delta\mathbf{w}_{b_{k-1}}^a \\ \delta\mathbf{w}_{b_{k-1}}^\omega \end{bmatrix}. \end{aligned}$$

An expression for the Jacobians can then be found, these are given by

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & T\mathbf{I} & -\frac{T^2}{2}(\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a)^\times \\ \mathbf{0} & \mathbf{I} & -T(\bar{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a)^\times \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad (4-75)$$

and

$$\mathbf{L}_k = \begin{bmatrix} \frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}} & \mathbf{0} \\ T\bar{\mathbf{C}}_{nb_{k-1}} & \mathbf{0} \\ \mathbf{0} & T\bar{\mathbf{C}}_{nb_{k-1}} \end{bmatrix} \quad (4-76)$$

**Measurement update** In order to update the state estimates, relative position measurements to known landmark positions are used, these measurements are modelled as (4-42). To get a linearized measurement model, substituting in (4-24b) and (4-70), and using that  $\mathbf{e}_{b_k}^i = \bar{\mathbf{e}}_{n_k}^i + \delta\mathbf{e}_{n_k}^i$  with  $\bar{\mathbf{e}}_{n_k}^i = \mathbf{0}$ , which is consistent with  $\mathbf{e}_{n_k}^i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{n_k}^i)$ , yields

$$\begin{aligned}\bar{\mathbf{y}}_k^i + \delta\mathbf{y}_k^i &= (\exp(\delta\phi_{n_k}^\times) \bar{\mathbf{C}}_{nb_k})^\top (\mathbf{r}_n^{p^i w} - \bar{\mathbf{r}}_n^{z_k w} - \delta\mathbf{r}_n^{z_k w}) + \bar{\mathbf{e}}_{n_k}^i + \delta\mathbf{e}_{n_k}^i \\ &\approx \bar{\mathbf{C}}_{nb_k}^\top (\mathbf{I} - \delta\phi_{n_k}^\times) (\mathbf{r}_n^{p^i w} - \bar{\mathbf{r}}_n^{z_k w} - \delta\mathbf{r}_n^{z_k w}) + \bar{\mathbf{e}}_{n_k}^i + \delta\mathbf{e}_{n_k}^i.\end{aligned}$$

Subtracting the nominal equation  $\bar{\mathbf{y}}_k^i = \bar{\mathbf{C}}_{nb_k}^\top (\mathbf{r}_n^{p^i w} - \bar{\mathbf{r}}_n^{z_k w})$  and again ignoring terms of order  $\mathcal{O}(\|\delta\phi_{n_{k-1}}\|_2 \|\delta\mathbf{r}_n^{z_k w}\|_2)$ , then gives

$$\begin{aligned}\delta\mathbf{y}_k^i &= -\bar{\mathbf{C}}_{nb_k}^\top \delta\mathbf{r}_n^{z_k w} - \bar{\mathbf{C}}_{nb_k}^\top \delta\phi_{n_k}^\times (\mathbf{r}_n^{p^i w} - \bar{\mathbf{r}}_n^{z_k w}) + \bar{\mathbf{C}}_{nb_k}^\top \delta\phi_{n_k}^\times \delta\mathbf{r}_n^{z_k w} + \delta\mathbf{e}_{n_k}^i \\ &\approx -\bar{\mathbf{C}}_{nb_k}^\top \delta\mathbf{r}_n^{z_k w} + \bar{\mathbf{C}}_{nb_k}^\top (\mathbf{r}_n^{p^i w} - \bar{\mathbf{r}}_n^{z_k w})^\times \delta\phi_{n_k} + \delta\mathbf{e}_{n_k}^i \\ &= \mathbf{H}_k^i \begin{bmatrix} \delta\mathbf{r}_n^{z_{k-1} w} \\ \delta\mathbf{v}_n^{z_{k-1} w/n} \\ \delta\phi_{n_{k-1}} \end{bmatrix} + \mathbf{M}_k \delta\mathbf{e}_{n_k}^i.\end{aligned}$$

From this equation an expression for the Jacobians  $\mathbf{H}_k$  and  $\mathbf{M}_k$  can be found, which are given by

$$\mathbf{H}_k^i = \begin{bmatrix} -\bar{\mathbf{C}}_{nb_k}^\top & \mathbf{0} & \bar{\mathbf{C}}_{nb_k}^\top (\mathbf{r}_n^{p^i w} - \bar{\mathbf{r}}_n^{z_k w})^\times \end{bmatrix}$$

and

$$\mathbf{M}_k = \mathbf{I},$$

respectively. For all  $m$  landmarks these then become

$$\mathbf{H}_k = \underset{i=1, \dots, m}{\text{row}} \left( \begin{bmatrix} -\bar{\mathbf{C}}_{nb_k}^\top & \mathbf{0} & \bar{\mathbf{C}}_{nb_k}^\top (\mathbf{r}_n^{p^i w} - \bar{\mathbf{r}}_n^{z_k w})^\times \end{bmatrix} \right), \quad \mathbf{M}_k = \mathbf{I}. \quad (4-77)$$

After the measurement update, the linearization point will be updated. In [9], the relinearization is considered as the measurement update of the linearization point, so this updates  $\check{\mathbf{C}}_{nb_k}$  to  $\hat{\mathbf{C}}_{nb_k}$  as

$$\hat{\mathbf{C}}_{nb_k} = \exp(\delta\hat{\phi}_{n_k}^\times) \check{\mathbf{C}}_{nb_k}. \quad (4-78)$$

The update for the position and velocity states are performed in a manner that is identical as for the original EKF. A summary of the equations for the MEKF with orientation deviation states in navigation frame when employed for pose estimation, is given in Algorithm 4.

---

**Algorithm 4** Multiplicative EKF with orientation deviation states in navigation frame

---

**Inputs:** Measurement data  $\mathbf{y}_k^i$ , for  $i = 1, \dots, m$  and its covariance matrices.

**Outputs:** An estimate of position, velocity, the rotation matrix and the covariance matrix for  $k = 1, \dots, N$ .

---

1. Initialize with  $\mathbf{r}_n^{z_1 w}$ ,  $\mathbf{v}_n^{z_1 w/n}$ ,  $\hat{\mathbf{C}}_{nb_1}$  and  $\hat{\mathbf{P}}_1$
2. **for**  $k = 2, \dots, N$  **do**
  - (a) *Prediction.* Compute

$$\check{\mathbf{r}}_n^{z_k w} = \hat{\mathbf{r}}_n^{z_{k-1} w} + T\hat{\mathbf{v}}_n^{z_{k-1} w/n} + \frac{T^2}{2}(\hat{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a + \mathbf{g}_n), \quad (4-79a)$$

$$\check{\mathbf{v}}_n^{z_k w/n} = \hat{\mathbf{v}}_n^{z_{k-1} w/n} + T(\hat{\mathbf{C}}_{nb_{k-1}} \mathbf{u}_{b_{k-1}}^a + \mathbf{g}_n), \quad (4-79b)$$

$$\check{\mathbf{C}}_{nb_k} = \hat{\mathbf{C}}_{nb_{k-1}} \exp(T\mathbf{u}_{b_{k-1}}^\times), \quad (4-79c)$$

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1} \hat{\mathbf{P}}_{k-1} \mathbf{F}_{k-1}^\top + \mathbf{L}_{k-1} \hat{\mathbf{Q}}_{k-1} \mathbf{L}_{k-1}^\top, \quad (4-79d)$$

where  $\mathbf{F}_{k-1}$  is defined as in (4-75) and  $\mathbf{L}_{k-1}$  is defined as in (4-76).

- (b) *Update.* Compute

$$\mathbf{K}_k = \check{\mathbf{P}}_k \mathbf{H}_k^\top (\mathbf{H}_k \check{\mathbf{P}}_k \mathbf{H}_k^\top + \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top)^{-1},$$

$$\mathbf{z}_k = \mathbf{y}_k - \mathbf{h}(\check{\mathbf{C}}_{nb_k}).$$

Then, to update the state and covariance estimate, compute

$$\delta \hat{\boldsymbol{\xi}}_k = \begin{bmatrix} \delta \hat{\boldsymbol{\xi}}_k^r \\ \delta \hat{\boldsymbol{\xi}}_k^v \\ \delta \hat{\boldsymbol{\xi}}_k^\phi \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{r}}_n^{z_k w} - \check{\mathbf{r}}_n^{z_k w} \\ \hat{\mathbf{v}}_n^{z_k w/n} - \check{\mathbf{v}}_n^{z_k w/n} \\ \delta \hat{\phi}_{n_k} \end{bmatrix} \quad (4-80)$$

$$= \mathbf{K}_k \mathbf{z}_k, \quad (4-81)$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \check{\mathbf{P}}_k (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^\top + \mathbf{K}_k \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top \mathbf{K}_k^\top, \quad (4-82)$$

where  $\mathbf{H}_k$  and  $\mathbf{M}_k$  are computed following (4-77).

- (c) *Relinearization.* Compute

$$\hat{\mathbf{r}}_n^{z_k w} = \check{\mathbf{r}}_n^{z_k w} + \delta \hat{\boldsymbol{\xi}}_k^r, \quad (4-83a)$$

$$\hat{\mathbf{v}}_n^{z_k w/n} = \check{\mathbf{v}}_n^{z_k w/n} + \delta \hat{\boldsymbol{\xi}}_k^v, \quad (4-83b)$$

$$\hat{\mathbf{C}}_{nb_k} = \exp(\delta \hat{\boldsymbol{\xi}}_k^{\phi \times}) \check{\mathbf{C}}_{nb_k}. \quad (4-83c)$$


---



### 5-1 Simulation Parameters

Two sets of simulations have been set up. In the first set of simulations, the LIEKF and the MEKF-b are compared. In the second set of simulations, the RIEKF and the MEKF-n are compared.

In order to perform Monte Carlo simulations, the initial error of the estimate is taken from a normal distribution with zero mean and a covariance matrix  $\mathbf{P}_0$ , such that  $\delta\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_0)$  for the MEKF and  $\delta\boldsymbol{\xi}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_0)$  for the IEKF. These initial errors can then be used together with the true states in order to initialize the filters. For the MEKFs this means

$$\begin{aligned}\hat{\mathbf{r}}_n^{z_0w} &= \mathbf{r}_n^{z_0w} + \delta\mathbf{r}_0, \\ \hat{\mathbf{v}}_n^{z_0w/n} &= \mathbf{v}_n^{z_0w/n} + \delta\mathbf{v}_0, \\ \hat{\mathbf{C}}_{nb_0} &= \mathbf{C}_{nb_0} \exp_{SO(3)}(\delta\boldsymbol{\phi}_0).\end{aligned}$$

For the LIEKF and RIEKF this initialization is done by taking  $\delta\mathbf{T}_0 = \exp(\delta\boldsymbol{\xi}_0^\wedge)$  and then the initial state can be found using the appropriate error definition to get

$$\begin{aligned}\hat{\mathbf{T}}_0^L &= \mathbf{T}_0\delta\mathbf{T}_0, \\ \hat{\mathbf{T}}_0^R &= \delta\mathbf{T}_0\mathbf{T}_0.\end{aligned}$$

The IMU sensor measurement sensor noise standard deviations were set to  $\sigma_a = 0.05$  m/s<sup>2</sup>,  $\sigma_\omega = \frac{\sqrt{\pi}}{6}$  rad/s. This corresponds to a variance of 5 degrees per second for the gyroscope. The GPS and landmark measurement sensor noise standard deviations were set to  $\sigma_R = \sqrt{0.05}$  m. Typical standard deviations of GPS signals in smartphones have been reported to be roughly around 3m [44], the higher end of quality receivers such as GNSS have been reported to have standard deviations as low as approximately 10cm [45], so the chosen value for  $\sigma_R$  is on the lower side. The initial errors are taken from a normal distribution with covariance  $\mathbf{P}_0 = \text{diag}(\sigma_\phi^2\mathbf{I}, \sigma_v^2\mathbf{I}, \sigma_r^2\mathbf{I})$ , where  $\sigma_\phi = \frac{\pi}{36}$  rad/s,  $\sigma_{v_0} = 0.05$  m/s, and  $\sigma_{r_0} = 0.05$  m.

## 5-2 Simulation Results

As mentioned in the previous section, the initial state estimates and sensor noises were taken from normal distributions with set values for the standard deviation. Two sets of simulations have been set up in which the IEKF and the MEKF are compared in terms of the root mean square (RMS) of the norm of the estimation error. The first set encompasses the filters corresponding to the left-invariant measurements, the GPS measurements. The second set uses the right-invariant measurements, the relative position measurements to known landmarks. The set-up of both sets of simulations is identical, all of the standard deviations for the initial estimates and sensor noises for the filters will remain constant except at each trial one of them will be incrementally increased from 0 to 1. At each increment a set number of Monte Carlo simulations will be performed. The figures show the mean of the RMS of the norm error of each state estimate, next to this each figure shows the 95% of the RMS of the norm of the errors of all Monte Carlo simulations at each increment.

### 5-2-1 Left-Invariant Measurements

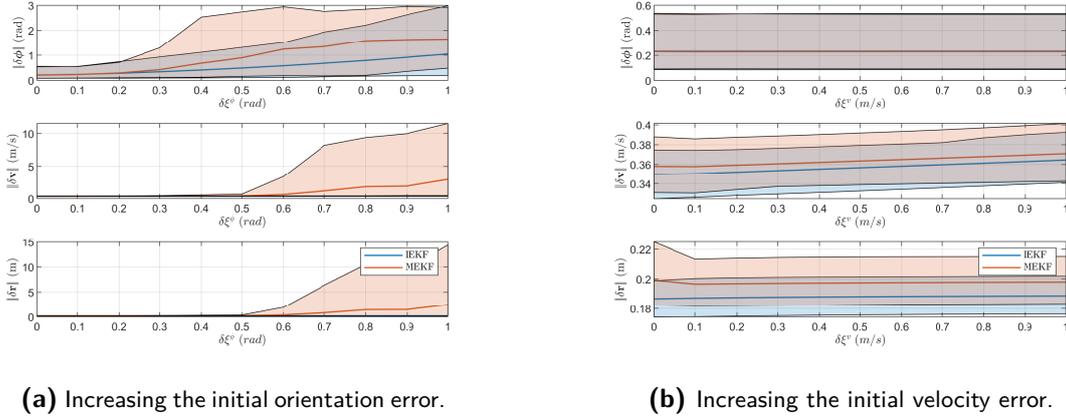
As mentioned before, for the left-invariant measurements GPS measurements are used. To check the influence of the error on the initial estimate of each of the three states, this error is incrementally increased from 0 to 1 with increments of 0.1, for each state and at each increment 100 Monte Carlo simulations have been performed.

#### 5-2-1-1 Increasing Initial Error

These results clearly show a reduced estimation accuracy of the MEKF when the error on the initial orientation estimate gets increased, whereas the performance of the IEKF for the estimation of the velocity and position, stays reasonably constant. In Figure 5-2 the norm of the estimation error for each state is shown plotted over time for two different runs when the error in the initial orientation estimate is was increased to its highest value on the range of Figure 5-1.

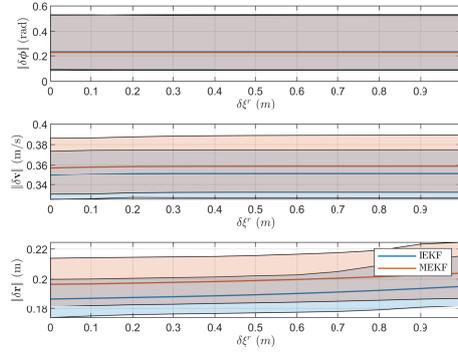
These results agree with the findings reported in current literature, such as [24, 8]. This is not surprising since in [15, 8] it was claimed that when the dynamical model is group affine and the measurement model fits either the left- or the right-invariant structure, then the error dynamics are independent of the state of the system. Translating this to the situation at hand, with a group affine dynamical model and a left-invariant measurement model, means that it is guaranteed for the IEKF to not have a dependence of the state estimate, while this does not necessarily hold for the MEKF, and thus a poor initial estimate will potentially influence the MEKF more than the IEKF. Looking at the Jacobians corresponding to the propagation and the update of the system also confirms this claim. The Jacobian corresponding to the propagation of the LIEKF are as follows

$$\mathbf{F}_k = \begin{bmatrix} \Psi_{k-1}^\top & \mathbf{0} & \mathbf{0} \\ -T\Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times & \Psi_{k-1}^\top & \mathbf{0} \\ -\frac{T^2}{2}\Psi_{k-1}^\top \mathbf{u}_{b_{k-1}}^a \times & T\Psi_{k-1}^\top & \Psi_{k-1}^\top \end{bmatrix}, \quad (5-1)$$



(a) Increasing the initial orientation error.

(b) Increasing the initial velocity error.



(c) Increasing the initial position error.

**Figure 5-1:** RMS of the norm of the error when increasing the initial error.

where  $\Psi_{k-1} = \exp_{SO(3)}(T\mathbf{u}_{b_{k-1}}^{\omega \times})$ , and the Jacobian corresponding to the update is

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{I} \end{bmatrix}, \quad (5-2)$$

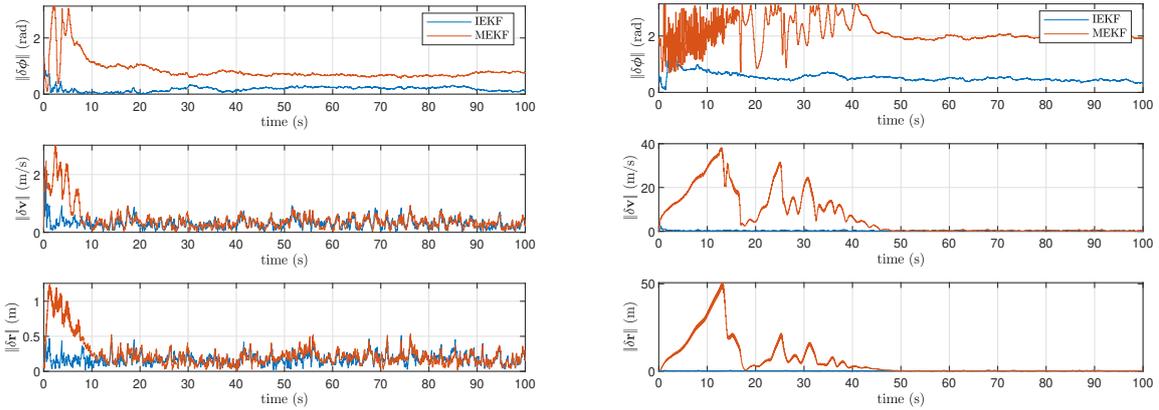
whereas the Jacobian corresponding to the propagation of the MEKF-b is given as

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & T\mathbf{I} & -\frac{T^2}{2}\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^a \times \\ \mathbf{0} & \mathbf{I} & -T\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^a \times \\ \mathbf{0} & \mathbf{0} & \exp(-T\mathbf{u}_{b_{k-1}}^{\omega \times}) \end{bmatrix}, \quad (5-3)$$

and the Jacobian for the update is

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \end{bmatrix}. \quad (5-4)$$

From (5-1) it can be observed that the  $\mathbf{F}_k$  for the LIEKF does not depend on any state estimate of the system, but the propagation of the MEKF-b does depend on the orientation of the system. Although that both filters use the same dynamical models to propagate the state, the Jacobians associated with the linearization of the dynamics are used to propagate the state covariance matrix, which in turn is used in the computation of the Kalman gain.



**Figure 5-2:** The norm of the estimation error for each state when increasing the initial orientation error to  $\delta\xi^{\phi} = 1$  rad for two different single simulation runs.

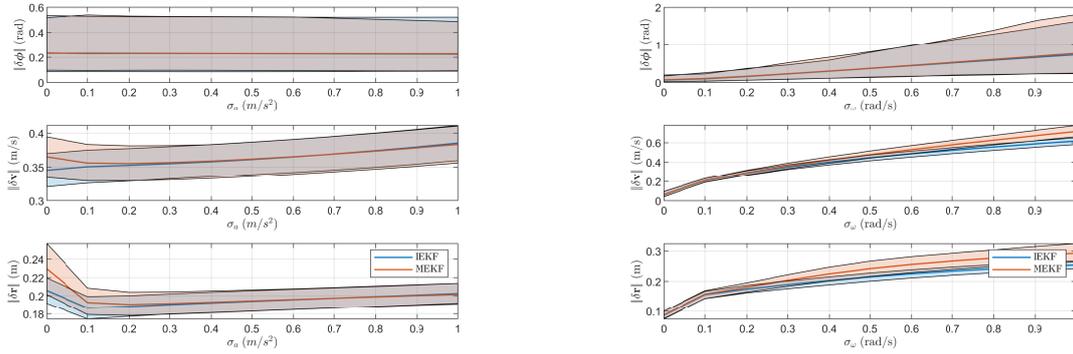
Since that the propagation Jacobian (5-3) has multiple occurrences of the orientation estimate embedded in the rotation matrices, it can be explained that the MEKF-b is not guaranteed to converge to the true trajectory. The shaded area of the MEKF in Figure 5-1a shows a large error in the estimation of the states whilst the mean of these estimates remains significant but relatively lower. This indicates that the estimation of the states does not converge to the true state in a large portion of the simulations but not for all simulations. The increase of the error in the initial estimates of the velocity and position does not show a significant change in the performance of either filter. This is also not expected since there is no clear indicator why either filter would diverge as a result from this increase, unlike for the orientation estimate.

### 5-2-1-2 Increasing Sensor Noise

Next, the influence of the amplitude of the sensor noises on the performance of both filters are considered. The results of incrementally increasing each of the different sensor noises are shown in Figure 5-3.

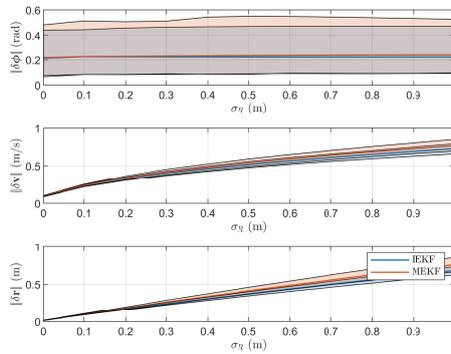
The errors on the state estimate when increasing the noise on the accelerometer measurements seem to be almost identical. There is a clear distinction, however, the error in the estimation of the velocity and position for the MEKF-b is larger when there is no accelerometer sensor noise. Similar behaviour has been reported in [8], they reported that the MEKF suffers from having small or tight  $\mathbf{Q}$  matrices. The terminology tight reflects that this concerns covariance matrices and a tight process noise covariance matrix means that the filter assumes higher accuracy of the sensors. This is the result of the Kalman gains of the MEKF rapidly decreasing during the transitory phase of the simulation while the attitude error is not reduced enough because of non-linearities, since the position estimate is affected, the gains are too small to correct this error [8]. In Figure 5-4, the norm of the estimation error is shown plotted over time. From this it can be seen that the attitude error is indeed not reduced, however, although a slight difference in position estimation error can be seen between the two filters, it seems that the GPS measurements are able to correct it for the MEKF.

In this case the accelerometer noise is zero and the  $\mathbf{Q}^a$  is set to a very small value, namely  $10^{-6}\mathbf{I}$ , in order to prevent having an empty covariance matrix. Figure 5-5 shows the results



(a) Increasing the amplitude of the noise on the accelerometer.

(b) Increasing the amplitude of the noise on the gyroscope.

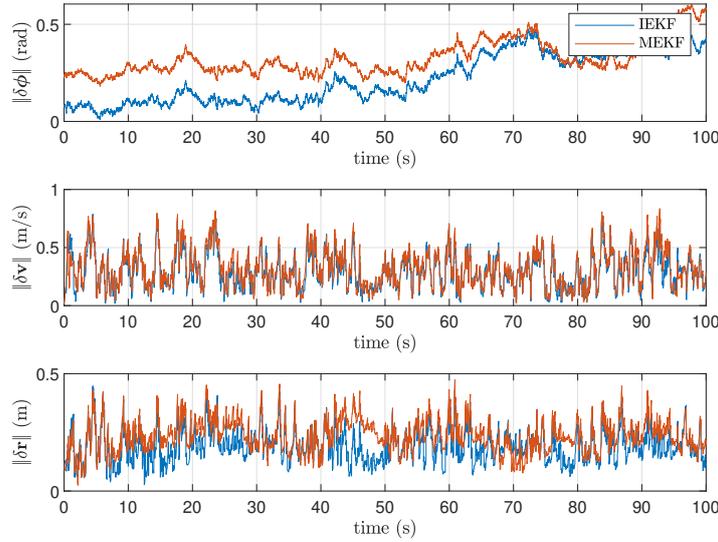


(c) Increasing the amplitude of the noise on the GPS measurements.

**Figure 5-3:** RMS of the norm of the error when increasing the amplitude of the sensor noise.

of incrementally increasing the accelerometer noise on a smaller interval, from 0 to  $0.2 \text{ m/s}^2$  with increments of 0.02, to find where the effect of having a small covariance matrix  $\mathbf{Q}$  starts to decrease. It also shows that increasing this covariance matrix at zero noise does indeed improve the performance for the MEKF-b.

The increase of the gyroscope sensor noise shows a small but increasing difference in the error of the velocity and position estimate between the LIEKF and the MEKF-b. There is no clear indication as to why this should be expected to be observed. Both filters experience an increase in the error of the orientation deviation estimate with an increasing gyroscope sensor noise. A possible explanation could be that since the MEKF-b is state estimate dependent, the rotation matrix in (5-3), it might experience more influence of the increasing error on the orientation deviation estimate on top of the fact that the body experiences a constant rotation throughout the trajectory. This increase in the error of the orientation estimate is difficult to correct for since the only corrective update is done through position measurements from a GPS, which does not offer information on the true orientation. In Figure 5-6 the estimation error is plotted over time for a single simulation when the noise on the gyroscope is increased. It can be seen that indeed both filters are experiencing issues regarding the correction of the



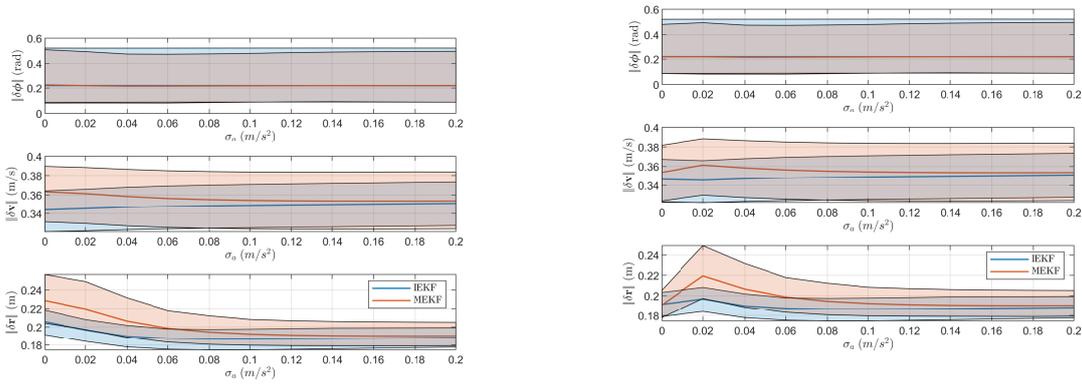
**Figure 5-4:** The norm of the estimation error for each state with zero accelerometer noise.

orientation. Both filters also show that the estimation errors of the velocity and position do tend to increase within GPS measurement updates in a very similar manner. The MEKF-b seems to have the harshest increases in error between the measurement updates, albeit very comparable to the LIEKF.

The increasing GPS measurement noise shows a small difference in the estimation done by the LIEKF and MEKF-b, slightly favoring the LIEKF at higher sensor noise. There is no clear indication why the LIEKF should be performing better in this situation. It might be that during the transient phase of the simulation, the LIEKF shows a better performance. However, at steady state, the estimate should be close enough to the true trajectory that the state-dependence of the MEKF-b should no longer be of an influence. In Figure 5-6 the estimation error is plotted over time for a single simulation when the noise on the GPS measurements is increased. This does indeed show that the state-dependence of the MEKF-b does not seem to heavily influence the state estimation. There is a small improvement noticeable between the estimation accuracy of the LIEKF over the MEKF-b.

## 5-2-2 Right-invariant measurements

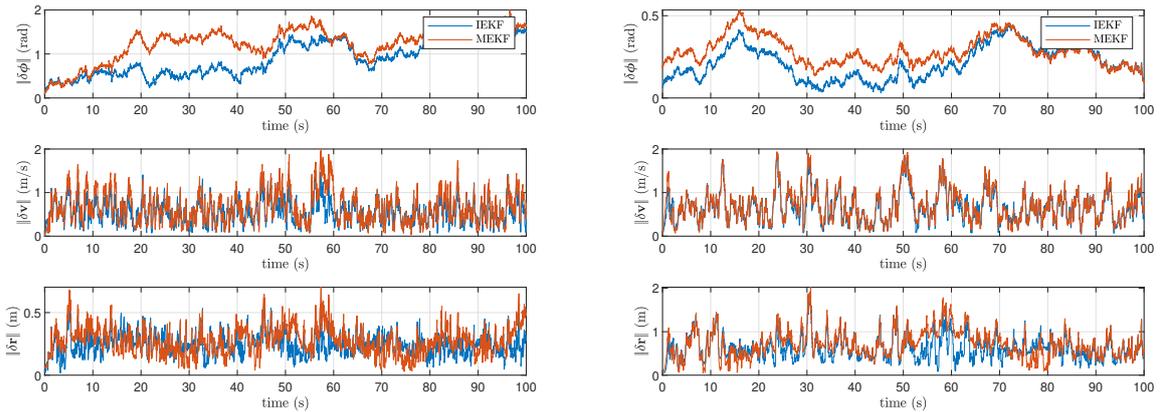
The same set of simulations are performed using right-invariant measurements, the relative position measurements to known landmarks. Similar as for the left-invariant measurements, 100 Monte Carlo simulations have been performed at each increment. Firstly, these simulations are performed using ten landmarks with known position randomly placed across the trajectory. After this, the same set of simulations is performed using only three known landmarks.



(a) Increasing the amplitude of the noise on the accelerometer on smaller interval.

(b) Increasing the amplitude of the noise on the accelerometer on smaller interval with increased  $Q^a$  at zero noise.

**Figure 5-5:** RMS of the norm of the error when increasing the amplitude of the accelerometer sensor noise.

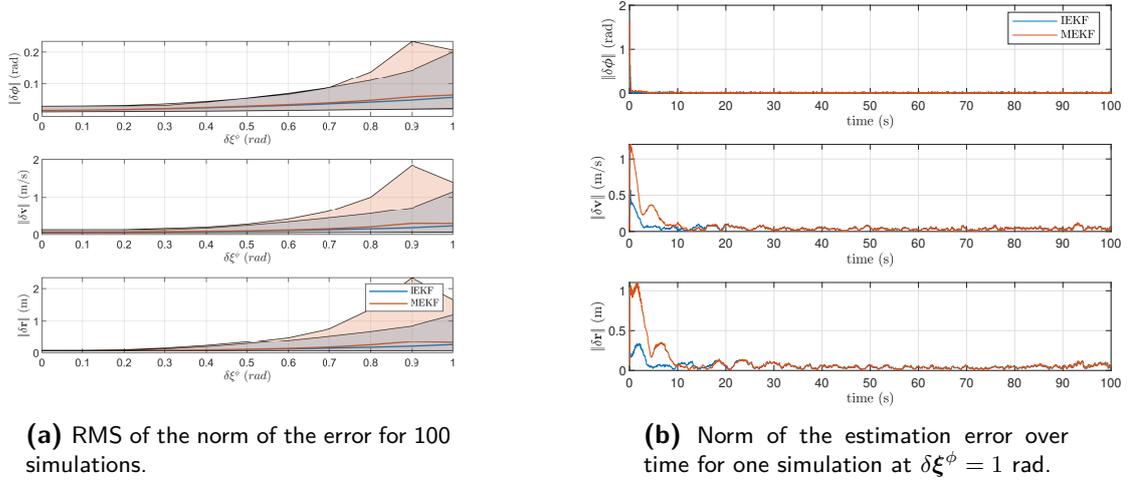


**Figure 5-6:** The norm of the estimation error over time for each state when increasing the noise on the gyroscope to  $\sigma_\omega = 1$  rad/s (left) and on the GPS measurements to  $\sigma_R = 1$  m (right).

### 5-2-2-1 Ten Landmarks with Known Position

#### 5-2-2-1-1 Increasing Initial Error

First the influence of incrementally increasing the initial error on the states is checked, the results of these simulations regarding the increase of the initial orientation error are shown in Figure 5-7, the results regarding increasing the initial position and velocity errors are shown in Appendix C since these do not offer new insights or clear differences between the RIEKF and MEKF-n.



**Figure 5-7:** Increasing the initial orientation error with 10 known landmarks.

From these figures it can be seen that, although not as prevalent, similar to the left-invariant measurements an increase in the estimation error can be observed for the MEKF-n with an increasing error in the initial orientation deviation. There is also an increase noticeable for the RIEKF with the increasing error in the initial orientation deviation. This comes from the fact that with an increased initial estimation error, the time to converge and settle takes slightly longer. The same holds for the error of MEKF-n, unlike for the MEKF-b, it does not diverge from the true trajectory but instead it needs a longer time to converge to the true trajectory. Just as for the left-invariant measurements, an influence of the initial orientation deviation was to be expected for the MEKF-n. Looking at the propagation (5-5) and update (5-6) Jacobians of the MEKF-n shows multiple occurrences of the orientation deviation state estimate,

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & T\mathbf{I} & -\frac{T^2}{2}(\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^a)^\times \\ \mathbf{0} & \mathbf{I} & -T(\bar{\mathbf{C}}_{nb_{k-1}}\mathbf{u}_{b_{k-1}}^a)^\times \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad (5-5)$$

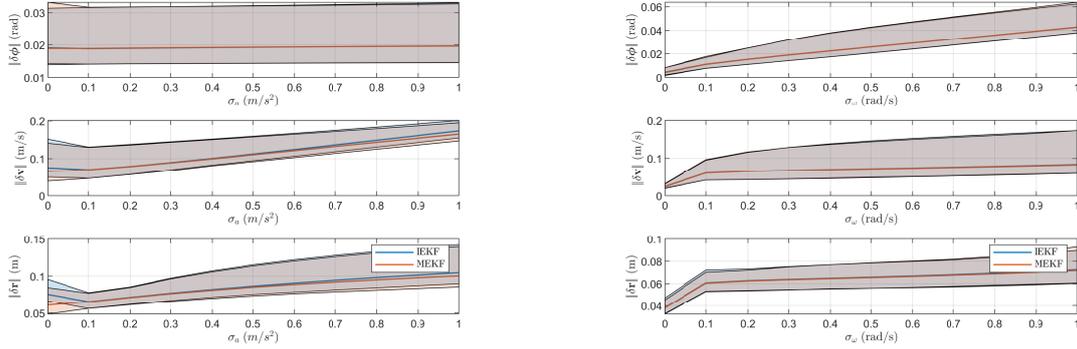
$$\mathbf{H}_k = \underset{i=1,\dots,m}{\text{row}} \left( \begin{bmatrix} -\bar{\mathbf{C}}_{nb_k}^\top & \mathbf{0} & \bar{\mathbf{C}}_{nb_k}^\top (\mathbf{r}_n^{p^{iw}} - \bar{\mathbf{r}}_n^{z_k^w})^\times \end{bmatrix} \right). \quad (5-6)$$

Similar to the left-invariant measurement simulations, increasing the initial velocity or position estimates does not reveal a significant difference between the RIEKF and MEKF-n.

### 5-2-2-1-2 Increasing Sensor Noise

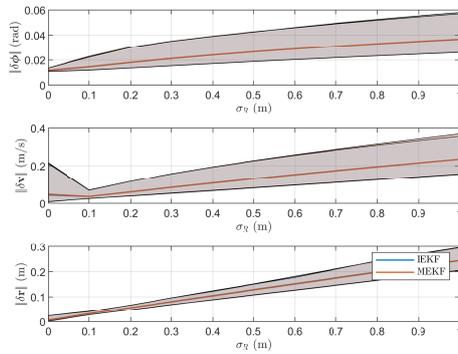
Next the influence of increasing the sensor noises are considered. The results of incrementally increasing the standard deviation of each sensors noise are shown in Figure 5-8.

As can be observed Figure 5-8, both filters show almost identical performance results when increasing each of the sensors noises. Unlike for the MEKF-b the MEKF-n does not seem to



(a) Increasing the amplitude of the noise on the accelerometer.

(b) Increasing the amplitude of the noise on the gyroscope.



(c) Increasing the amplitude of the noise on the landmark position measurements.

**Figure 5-8:** RMS of the norm of the error when increasing the amplitude of the sensor noise with 10 known landmarks.

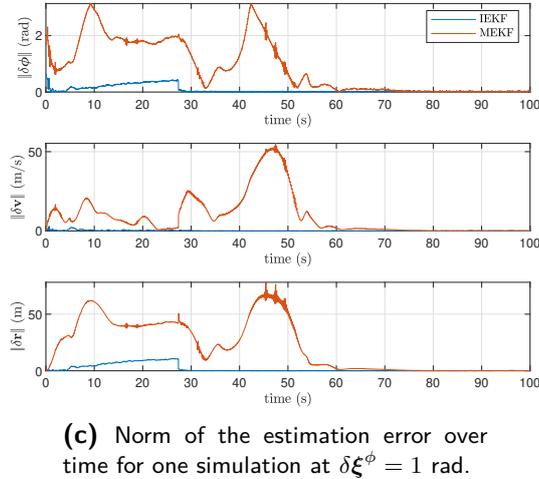
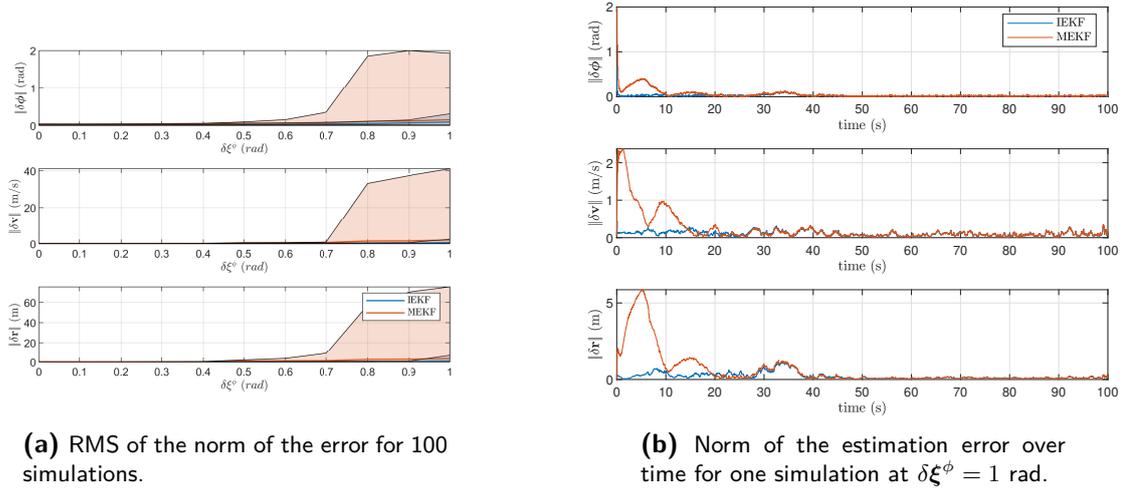
be suffering from small a  $\mathbf{Q}$  due to the accelerometer noise being close to zero. Increasing the gyroscopic noise also does not reveal a difference in performance between the RIEKF and MEKF-n, unlike for the LIEKF and the MEKF-b. This is likely due to the nature of the right-invariant measurements, these measurements are position measurements relative to known landmarks and therefore have information on the orientation explicitly embedded in them, as can be seen in (3-13). This raises the question whether having less relative position measurements at each time instance, and thus decreasing the information for the update step, could influence the performance of the filters.

### 5-2-2-2 3 Landmarks with Known Position

The simulations performed here are done to check if decreasing the information at each update step, could potentially influence the performance of the RIEKF and MEKF-n. The total amount of known landmarks along the trajectory is reduced to 3 landmarks instead of the former 10 landmarks.

### 5-2-2-2-1 Increasing Initial Error

Similar to before, first the influence of increasing the error in the initial estimate of the states is checked. The results of these simulations regarding the increase of the initial orientation error are shown in Figure 5-9, the results regarding increasing the initial position and velocity errors are shown in Appendix C since these do not offer new insights or clear differences between the RIEKF and MEKF-n.

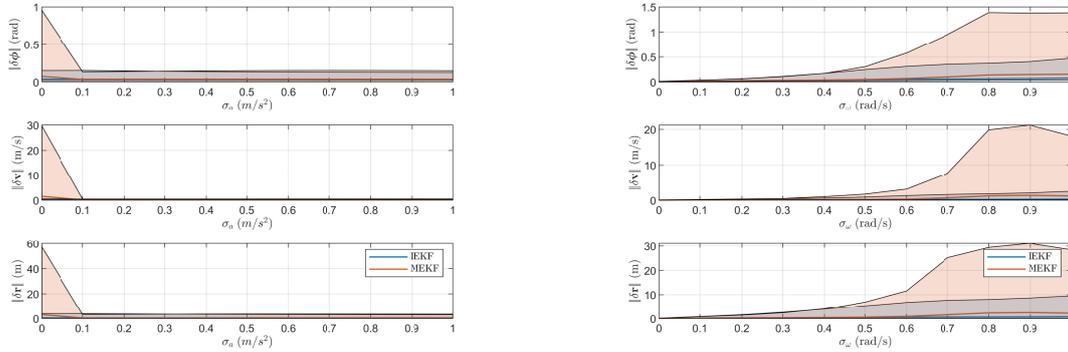


**Figure 5-9:** Increasing the initial orientation error with 3 known landmarks.

A clear difference is visible in the performance of the MEKF-n now when compared to having access to a total of 10 known landmarks. The MEKF-n shows to have more issues with converging to the true trajectory, which can be seen from the two different simulations in Figure 5-9. This happens more frequently now that it has less information during the update steps while incrementally increasing the initial orientation deviation. This agrees with the findings for the similar trial for the MEKF-b and can be traced back to the occurrences of the orientation deviation in the filters linearization (5-5) and (5-6). Incrementally increasing the initial position and velocity errors did not show any significant difference between both filters.

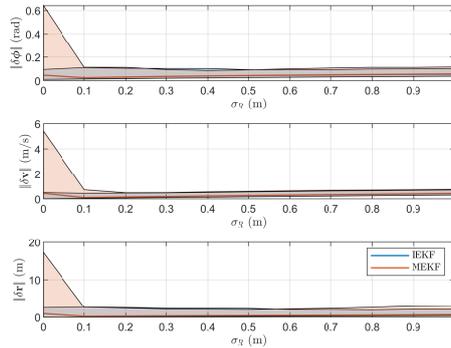
**5-2-2-2 Increasing Sensor Noise**

Again, the influence of incrementally increasing each of the sensor noise is checked and shown in Figure 5-10.



**(a)** Increasing the amplitude of the noise on the accelerometer.

**(b)** Increasing the amplitude of the noise on the gyroscope.

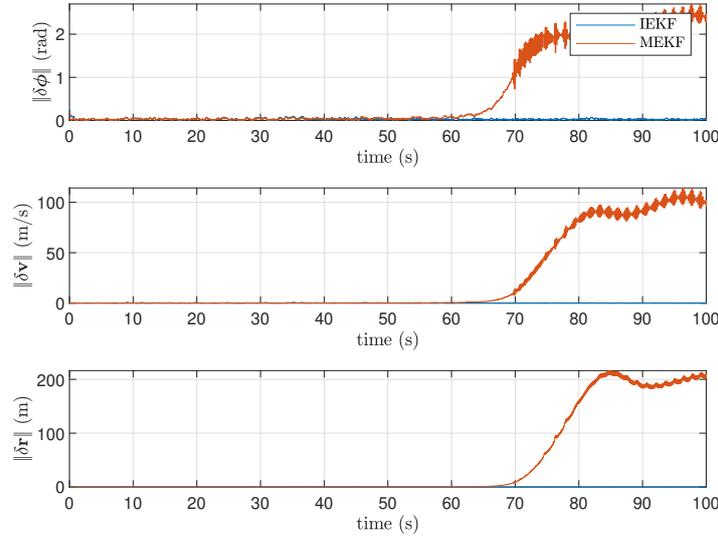


**(c)** Increasing the amplitude of the noise on the landmark position measurements.

**Figure 5-10:** RMS of the norm of the error when increasing the amplitude of the sensor noise with 3 known landmarks.

Contrary to what was seen when the total number of landmarks was 10, there is a clear difference between the performance of the RIEKF compared to the MEKF-n when incrementally increasing each of the sensor noise. The change in the accelerometer noise shows a similar trend as it did for the MEKF-b. However, instead of taking longer to converge to the true trajectory due to a small  $\mathbf{Q}$  as it did for the MEKF-b, the MEKF-n does not always seem to converge at all. This can be concluded from the mean line of the MEKF-n being relatively low but its shaded area, containing 95% of the simulation data, shows quite a disparity between the estimation errors. Figure 5-11 shows that the MEKF-n does indeed not converge to the true trajectory when the accelerometer noise is zero.

The increase in the standard deviation for the gyroscopic noise also starts to show a worse performance for the MEKF-n compared to the RIEKF. For the MEKF-b this was also true,



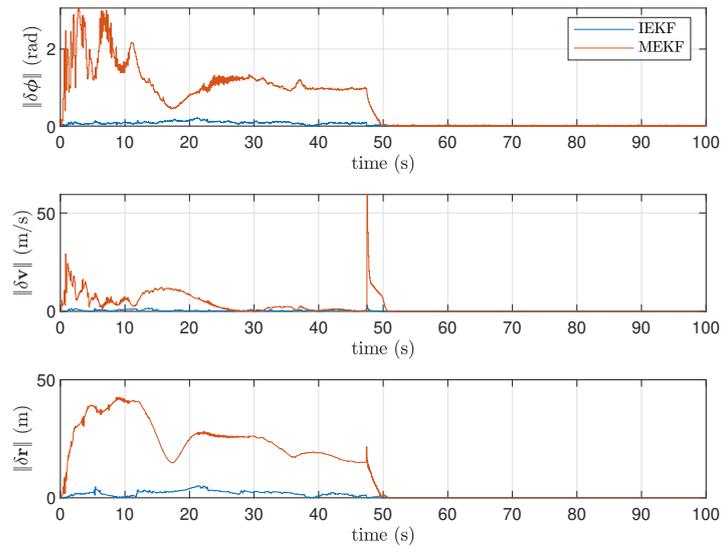
**Figure 5-11:** The norm of the estimation error for each state with zero accelerometer noise.

albeit more subtle as it did the error increased for the MEKF-b but it did not completely diverge from the trajectory. There is no clear indication why this behaviour should be expected to be observed. However, this could come from the fact that in the Jacobians for the propagation (5-5) and the measurement update (5-6) for the MEKF-n, multiple instances of the orientation deviation in terms of the rotation matrix are observed. This could mean that once the orientation estimation starts to be influenced by an input of the gyroscopic, which contains higher levels of noise, the linearization of the system starts to become less accurate. On top of this, (5-6) is also dependent on the position estimate, but since the position estimate is influenced by having less accurate propagation and update Jacobians, a poor position estimate could also influence this linearization. However, this influence of the position estimate in itself should not be too significant, since that would have been more apparent from the trial where just the error in the initial position estimate was increased.

The trial of incrementally increasing the standard deviation for measurement noise shows highly erroneous behaviour for the MEKF-n whilst the RIEKF does not suffer from this. This behaviour was not shown by the MEKF-b. This seems to be the effect of having a very tight  $\mathbf{R} = 10^{-6}\mathbf{I}$  when there is no measurement noise. This tight  $\mathbf{R}$  in combination with the body only seeing 1 landmark due to the set threshold for the maximum range for the first part of some of the trajectories. This can be seen in Figure 5-12. Once the body registers a second landmark, the error in the orientation seems to get correctly updated, but this induced a large error in the velocity estimation.

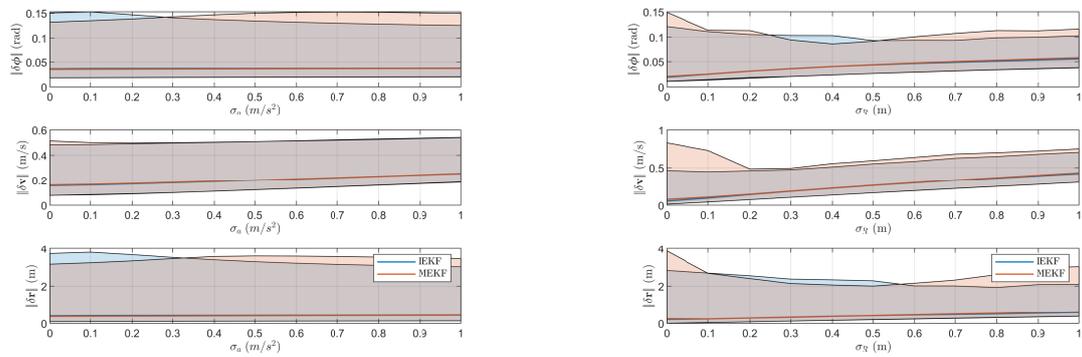
To test to see whether inflating or increasing the  $\mathbf{Q}$  in the trial for the increase of the accelerometer noise and inflating or increasing the  $\mathbf{R}$  in the trial for the increase of the measurement noise, new simulations have been performed using these increased covariance matrices. The simulation are shown in Figure 5-13

These results show that the issues bothering the MEKF-n can be overcome by inflating the  $\mathbf{Q}$  when the accelerometer noise is set to zero. Additionally, the issues of diverging results when the measurement noise is zero are also less prevalent when the covariance matrix  $\mathbf{R}$  is increased. Although, it also shows that increasing the noise on the measurements does



**Figure 5-12:** The norm of the estimation error for each state with zero landmark measurement noise.

have a negative impact on the performance of the MEKF-n. This could be the results of the MEKF-n having dependencies on the rotation matrix and position estimate in the linearized measurement update (5-6).



**(a)** Increasing the amplitude of the noise on the accelerometer, with an increased covariance matrix  $\mathbf{Q}$ .

**(b)** Increasing the amplitude of the noise on the landmark position measurements, with an increased covariance matrix  $\mathbf{R}$ .

**Figure 5-13:** RMS of the norm of the error when increasing the amplitude of the sensor noise with 3 known landmarks.

# Conclusions and Future Work

## 6-1 Conclusions

In this thesis an in-depth evaluation of pose estimation is done for the invariant EKF and the multiplicative EKF. In order to do this, the goal of this thesis was to answer the main research question:

*How does the estimation accuracy of the invariant EKF compare to the multiplicative EKF in the context of pose estimation?*

To answer this question, three sub-questions were posed. The results discussed in this thesis, led to the following answers.

*What are the advantages of utilizing an LIEKF over an MEKF when dealing with a left-invariant measurement model, and similarly, what are the benefits of employing a RIEKF over an MEKF when dealing with a right-invariant measurement model?*

As has been covered in other work such as [8, 15], in the case of a high error in the initial orientation, both the LIEKF and the RIEKF show superior accuracy over the MEKF-b and the MEKF-n, respectively. This is due to the convergent properties of the IEKF theory and the state dependence in the linearization of the MEKF. These are not novel ideas and serve as confirmation of what was already known, in the case of low accuracy of the initialization of the filters, the IEKF is still guaranteed to converge to the true trajectory, whereas the MEKF is not and has been reported to occasionally diverge from the true trajectory. Consequently, if the situation allows for the utilization of either a left- or a right-invariant measurement model, it would be beneficial to employ the LIEKF or RIEKF over the MEKF-b and MEKF-n, respectively. However, these findings do not encompass the full scope of possibilities as IMU and measurement sensor noise also play a role, which leads us to the next sub-question.

*How does the IMU sensor noise magnitude affect the converging performance of the filters differently?*

The LIEKF and MEKF-b and the RIEKF and MEKF-n were compared to each other in terms of estimation error over a range of standard deviations for the accelerometer noise and the gyroscopic noise. The MEKF-b showed to converge slower than the LIEKF when confronted with very small accelerometer noise, with the standard deviation being in the range of  $\sigma_a = 0 - 0.12 \text{ m/s}^2$ . This issue can be mitigated by inflating the associated covariance matrix  $\mathbf{Q}^a$  to a bigger value. In [8], it was also discussed that the MEKF suffers from small or tight process covariance matrices and was shown that inflating this matrix overcomes this issue. For the MEKF-n similar results were found when confronted with very little accelerometer noise. In this case it was more severe than just converging slower than the RIEKF, since the MEKF-n also showed instances where it would fail to converge to the true trajectory. To get a feeling for this range of the standard deviation, a high-end IMU sensor from Bosch such as the BMI088 has reportedly a noise density of  $175 \mu\text{g}/\sqrt{\text{Hz}}$ , which at a sampling rate of 100 Hz is equivalent to a standard deviation of about  $\sim 0.017 \text{ m/s}^2$ . So the range at which the MEKF-b experiences issues due to a small process covariance matrix  $\mathbf{Q}$  is a very feasible range.

The behaviour for the LIEKF and MEKF-b when confronted with an increasing standard deviation for the gyroscopic noise showed to be very similar at the lower ranges. However, a slight change in performance was noticeable between the LIEKF and MEKF-b starting from  $\sigma_\omega = 0.2 \text{ rad/s}$  going up, but became more apparent from  $\sigma_\omega = 0.4 \text{ rad/s}$  onwards. The mean RMS error of the velocity and position estimates was consistently higher for the MEKF-b than the LIEKF. This difference is likely due to the left-invariant measurements being absolute position measurements and offering no correction for the orientation deviation. Since in the linearized dynamics of the MEKF-b there is a dependency on this orientation deviation in terms of the rotation matrix, the MEKF-b seems to suffer more from this lack of correction in the orientation estimate. This might be confirmed by the results of increasing the gyroscopic noise for the RIEKF and MEKF-n. In the first set of simulations, both filters have corrective position measurements relative to up to ten landmarks. Since these measurements are relative to a known position, this also includes orientation information. This results in no clear drop in performance for both the RIEKF and MEKF-n even though the gyroscopic noise is increased. However, looking at the second set of simulations, where both filters only have position measurements relative to up to three known landmarks, the performance of the MEKF-n seems to be heavily influenced by the increase in gyroscopic noise. Just as for the MEKF-b the linearized dynamics are dependent on the orientation estimate. Since the corrective measurements offer less information, compared to having ten known landmarks, the MEKF-n starts to show that it is not always able to converge to the true trajectory starting from around  $\sigma_\omega = 0.5 - 0.6 \text{ rad/s}$  upwards. To get a feeling of this range for the standard deviation, the same high-end Bosch IMU sensor, reports a noise density of  $0.014 \text{ }^\circ/\text{s}/\sqrt{\text{Hz}}$  for the gyroscope. At a sampling rate of 100 Hz, this gives a standard deviation of  $\sim 2.4 * 10^{-4} \text{ rad/s}$ . This indicates that for typical IMU sensors the non-converging issues of the MEKF-n and also the lower performance of the MEKF-b compared to the LIEKF, are unlikely to be noticeable. However, it does suggest that the MEKF is more susceptible to issues related to robustness compared to the IEKF.

From these results it can be concluded that IMU sensor noise does have more impact on the MEKF than on the IEKF. To the best knowledge of the author, this has not yet been reported in current literature and seem to a novel result.

*How does the sensor noise magnitude of the external measurements affect the*

*converging performance of the filters differently?*

It was briefly touched upon in the answer on the previous sub-question, for the right-invariant measurements, the relative position measurements, the number of landmarks made quite a difference. The MEKF-n showed to have almost identical performance to the RIEKF when both filters had access to the relative position information of up to ten landmarks. However, reducing this information to having a maximum of three landmarks, showed that in several scenarios of different combinations of sensor and measurement noise, the MEKF-n does not always have the ability to converge to the true trajectory.

Apart from this, the MEKF-n shows to not always converge to the true trajectory when the measurement noise is zero and the covariance matrix  $\mathbf{R}$  is very small. Inflating this covariance matrix seems to solve this issue. This is likely due to the fact that there is a threshold range at which the body can see a landmark, because of this at certain periods of time during the trajectory, the body will only see one landmark. At points where it suddenly sees more landmarks it seems to over-correct itself, this might be due to large gains since these are proportional with the inverse term of the covariance matrix  $\mathbf{R}$ . The results of increasing the measurements noise for the MEKF-n and RIEKF when the covariance matrix  $\mathbf{R}$  is increased at zero noise, shows that increasing the noise on the measurements does have a negative impact on the performance of the MEKF-n. This could be the results of the MEKF-n having dependencies on the rotation matrix and position estimate in the linearized measurement update.

The LIEKF and MEKF-b very comparable results with the increase of the noise on the GPS measurements. The LIEKF seems to have a slightly better performance, but this difference with the MEKF-b is very minimal and is too small to be able to conclude a real difference.

To conclude the answer to this question, the quality of the measurements seem to have an impact on the converging capabilities of the MEKF. The MEKF-n shows to be sensitive to small a small measurement covariance matrix  $\mathbf{R}$  when the information it gets from the measurements is limited. Additionally, due to the state-dependence of the linearization of the MEKF-n, it seems to be affected more by higher measurement noise than the RIEKF. To the best knowledge of the author, this result has not yet been reported in current literature and seems to be a novel idea.

Overall, the following can be concluded from this thesis and the main research question can be answered as follows, the IEKF and MEKF show very comparable results in a large amount of the applications. It should be noted that these results were drawn from one sample problem using two different types of measurement models in a simulated environment, so it cannot be stated that the IEKF will always show superior estimation accuracy in these scenarios. Nonetheless, the state-independent error dynamics have been shown to be beneficial in situations where the initial information of the state of the system is uncertain. Apart from this, the IEKF has been shown to be beneficial in certain edge cases. Firstly, the IEKF shows to be less sensitive to small process noise covariance matrices  $\mathbf{Q}$ . Secondly, once the gyroscopic noise becomes very large, the RIEKF showed higher estimation accuracy over the MEKF-n. The LIEKF did also show a marginal improvement in estimation accuracy over the MEKF-b. However, it is important to note that such high levels of noise are typically not encountered in most real-world applications, making the significance of this observation limited in practical scenarios. Finally, there seem to be two ways that the external measurement noise influenced the comparison of the estimation accuracy between the IEKF and MEKF. The MEKF-n

showed to be sensitive to a low covariance measurement matrix  $\mathbf{R}$  and additionally, the MEKF-b and MEKF-n both seemed to be marginally more affected by higher external measurement noise than the LIEKF and RIEKF, respectively.

## 6-2 Future Work

This thesis focuses on comparing the performance of the invariant extended Kalman filter (IEKF) and the multiplicative extended Kalman filter (MEKF) under different measurement scenarios, either GPS measurements or relative position measurements. While the MEKF can handle both GPS and relative position measurements simultaneously, the IEKF possesses the capability to utilize both measurement models by performing a switch between left- and right-invariant errors [24, 46]. It could prove of interest to compare the performance of the LIEKF with the MEKF-b, as well as RIEKF with the MEKF-n when both GPS and relative position measurements are employed.

The evaluated systems in this thesis satisfy the necessary conditions for the IEKF to guarantee state-estimate independent error dynamics. However, it is worth exploring scenarios where the dynamical model is not group affine and the measurement models are not invariant, such as incorporating sensor bias in the state. Assessing the performance of the IEKF compared to the MEKF in such cases would help identify potential benefits of using an invariant framework even when the error dynamics are not guaranteed to be state-independent.

Furthermore, considering that the Jacobians of the IEKF are no longer exact in non-group affine scenarios with non-invariant measurement models, incorporating an iterative step into the IEKF and comparing it with an iterative MEKF variant could prove beneficial. This iterative approach may enhance the estimation accuracy and convergence properties of both filters.

## Parametrization of Orientation Representation

A critical aspect of the ability to estimate how objects are moving around the world, is the ability to parametrize its orientation, or rotation. This section will serve as an overview of the different parametrizations of rotations.

### A-1 Rotation Matrix

There are two different ways of interpreting rotations, the so-called *active* and *passive* rotations [47]. The active rotations describe the operator that rotates a vector  $\mathbf{u}_a$  resolved in reference frame  $\mathcal{F}_a$ , to a vector resolved in the same frame, for example,

$$\mathbf{v}_a = \mathbf{R}\mathbf{u}_a,$$

where the subscript denotes that this concerns the components of the vector resolved in  $\mathcal{F}_a$ . Active rotations are not very relevant in this research and will not be elaborately discussed.

Passive rotations, also referred to as rotation transformations, correspond to the orientation of one reference frame to another. In other words, for example, it corresponds to the mapping of a vector  $\mathbf{u}_a$  resolved in reference frame  $\mathcal{F}_a$  to frame  $\mathcal{F}_b$ , so

$$\mathbf{u}_b = \mathbf{C}_{ba}\mathbf{u}_a,$$

where [4]

$$\mathbf{C}_{ba} = \underline{\mathcal{F}}_{\rightarrow b} \cdot \underline{\mathcal{F}}_{\rightarrow a}^T.$$

Rotation matrices have the following properties,

$$\mathbf{C} \in \mathbb{R}^{3 \times 3}, \quad \det(\mathbf{C}) = 1 \text{ and } \mathbf{C}^T \mathbf{C} = \mathbf{I}. \quad (\text{A-1})$$

All rotation matrices that satisfy these properties belong to the special orthogonal group  $SO(3)$ , the properties governing matrices belonging to this group are further covered in section B-1.



## Matrix Lie Groups

### B-1 The Special Orthogonal Group $SO(3)$

This overview of the properties of  $SO(3)$  is based on [4, Ch. 7]. The special orthogonal group  $SO(3)$  represents three dimensional rotations and is the set of valid rotation matrices:

$$SO(3) = \{ \mathbf{C} \in \mathbb{R}^{3 \times 3} \mid \mathbf{C}^T \mathbf{C} = \mathbf{I}, \det \mathbf{C} = 1 \}.$$

$SO(3)$  has three degrees of freedom in rotation. The associated Lie algebra for  $SO(3)$  is

$$\mathfrak{so}(3) = \{ \boldsymbol{\phi}^\times \in \mathbb{R}^{3 \times 3} \mid \boldsymbol{\phi} \in \mathbb{R}^3 \},$$

where  $\boldsymbol{\phi}^\times$  is the skew-symmetric representation of  $\boldsymbol{\phi}$  and is given by

$$\boldsymbol{\phi}^\times = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}^\times = \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix}.$$

The adjoint representation of an element of  $SO(3)$  is equal to that element,  $\text{Ad}(\mathbf{C}) = \mathbf{C}$ . The same holds for the adjoint representation of an element of  $\mathfrak{so}(3)$ ,  $\text{ad}(\boldsymbol{\phi}^\times) = \boldsymbol{\phi}^\times$ . The exponential map from  $\mathfrak{so}(3)$  to  $SO(3)$  is given by the Rodrigues formula,

$$\exp(\boldsymbol{\phi}^\times) = \cos \phi \mathbf{I} + (1 - \cos \phi) \mathbf{a} \mathbf{a}^T + \sin \phi \boldsymbol{\phi}^\times,$$

where  $\phi = \|\boldsymbol{\phi}\|$  and  $\mathbf{a} = \boldsymbol{\phi}/\phi$ . The logarithmic map from  $SO(3)$  to  $\mathfrak{so}(3)$  is given by

$$\log(\mathbf{C}) = (\mathbf{a}\phi)^\times,$$

where the angle  $\phi$  is given by

$$\phi = \cos^{-1} \left( \frac{\text{tr}(\mathbf{C} - \mathbf{I})}{2} \right) + 2\pi m,$$

and the axis  $\mathbf{a}$  is given by

$$\mathbf{a} = \frac{1}{2 \sin \phi} \begin{bmatrix} \mathbf{C}_{2,3} - \mathbf{C}_{3,2} \\ \mathbf{C}_{3,1} - \mathbf{C}_{1,3} \\ \mathbf{C}_{1,2} - \mathbf{C}_{2,1} \end{bmatrix}.$$

## B-2 The Group of Double Direct Isometries $SE_2(3)$

The group of double direct isometries was introduced in [8, 43]. This overview of the properties of  $SE_2(3)$  is based on [43]. The group  $SE_2(3)$  represents poses with the addition of a velocity state, as

$$SE_2(3) = \left\{ \mathbf{T} = \begin{bmatrix} \mathbf{C} & \mathbf{v} & \mathbf{r} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \in \mathbb{R}^{5 \times 5} \mid \mathbf{C} \in SO(3), \mathbf{v}, \mathbf{r} \in \mathbb{R}^3 \right\}.$$

The inverse of  $\mathbf{T}$  is defined as [48]

$$\mathbf{T}^{-1} = \begin{bmatrix} \mathbf{C}^\top & -\mathbf{C}^\top \mathbf{v} & -\mathbf{C}^\top \mathbf{r} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}.$$

The associated matrix Lie algebra of  $SE_2(3)$  is

$$\mathfrak{se}_2(3) = \left\{ \Xi = \xi^\wedge \in \mathbb{R}^{5 \times 5} \mid \xi \in \mathbb{R}^9 \right\},$$

where

$$\xi^\wedge = \begin{bmatrix} \xi^\phi \\ \xi^v \\ \xi^r \end{bmatrix}^\wedge = \begin{bmatrix} \xi^{\phi^\times} & \xi^v & \xi^r \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix} \in \mathbb{R}^{5 \times 5}, \xi^\phi, \xi^v, \xi^r \in \mathbb{R}^3.$$

The exponential map from  $\mathfrak{se}_2(3)$  to  $SE_2(3)$  is given by

$$\exp(\xi^\wedge) = \begin{bmatrix} \exp_{SO(3)}(\xi^{\phi^\times}) & \mathbf{J}\xi^v & \mathbf{J}\xi^r \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix},$$

where  $\mathbf{J}$  is

$$\mathbf{J} = \frac{\sin \phi}{\phi} \mathbf{I} + \left(1 - \frac{\sin \phi}{\phi}\right) \mathbf{a}\mathbf{a}^\top + \frac{1 - \cos \phi}{\phi} \mathbf{a}^\times, \quad (\text{B-1})$$

where  $\phi = \|\xi^\phi\|$  and  $\mathbf{a} = \xi^\phi / \phi$ . The logarithmic map from  $SE_2(3)$  to  $\mathfrak{se}_2(3)$  is

$$\log(\mathbf{T}) = \begin{bmatrix} \log_{SO(3)}(\mathbf{C}) & \mathbf{J}^{-1}\mathbf{v} & \mathbf{J}^{-1}\mathbf{r} \\ \mathbf{0} & 0 & 0 \\ \mathbf{0} & 0 & 0 \end{bmatrix},$$

where  $\mathbf{J}^{-1}$  is given by

$$\mathbf{J}^{-1} = \frac{\phi}{2} \cot \frac{\phi}{2} \mathbf{I} + \left(1 - \frac{\phi}{2} \cot \frac{\phi}{2}\right) \mathbf{a}\mathbf{a}^\top - \frac{\phi}{2} \mathbf{a}^\times. \quad (\text{B-2})$$

The adjoint operator for  $SE_2(3)$  is given by [48]

$$\text{Ad}(\mathbf{T}) = \begin{bmatrix} \mathbf{C} & \mathbf{0} & \mathbf{0} \\ \mathbf{v}^\times \mathbf{C} & \mathbf{C} & \mathbf{0} \\ \mathbf{r}^\times \mathbf{C} & \mathbf{0} & \mathbf{C} \end{bmatrix} \in \mathbb{R}^{9 \times 9}.$$

Its inverse is given by [48]

$$\text{Ad}(\mathbf{T})^{-1} = \text{Ad}(\mathbf{T}^{-1}) = \begin{bmatrix} \mathbf{C}^\top & \mathbf{0} & \mathbf{0} \\ -\mathbf{C}^\top \mathbf{v}^\times & \mathbf{C}^\top & \mathbf{0} \\ -\mathbf{C}^\top \mathbf{r}^\times & \mathbf{0} & \mathbf{C}^\top \end{bmatrix} \in \mathbb{R}^{9 \times 9}.$$

The adjoint operator for  $\mathfrak{se}(3)$  is given by

$$\text{ad}(\boldsymbol{\xi}) = \begin{bmatrix} \boldsymbol{\xi}^{\phi^\times} & \mathbf{0} & \mathbf{0} \\ \boldsymbol{\xi}^{v^\times} & \boldsymbol{\xi}^{\phi^\times} & \mathbf{0} \\ \boldsymbol{\xi}^{r^\times} & \mathbf{0} & \boldsymbol{\xi}^{\phi^\times} \end{bmatrix} \in \mathbb{R}^{9 \times 9}.$$



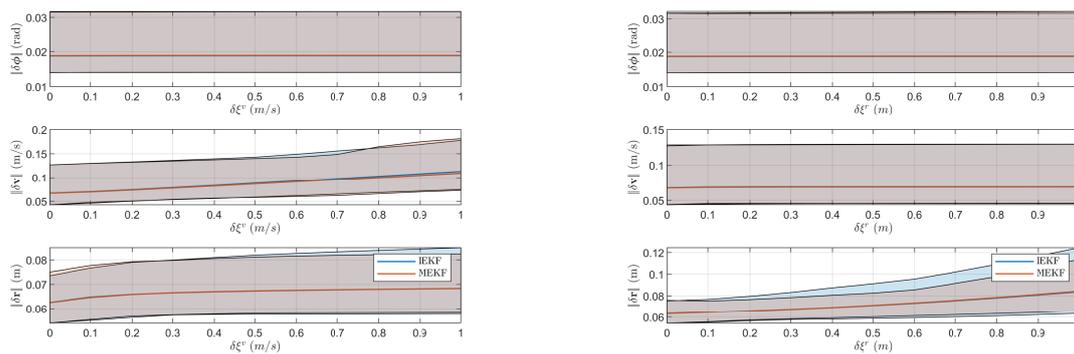
---

# Appendix C

---

## Supplementary Results

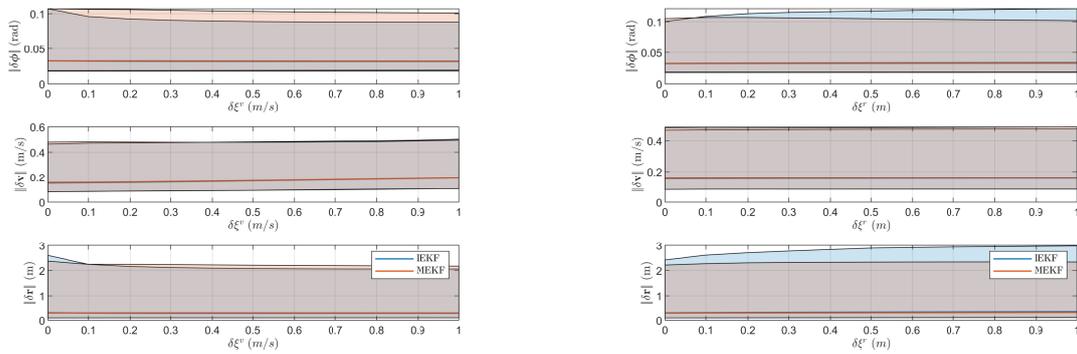
In this appendix the results omitted in chapter 5 are shown.



(a) Increasing the initial velocity error.

(b) Increasing the initial position error.

**Figure C-1:** RMS of the norm of the error when increasing the initial error in velocity and position with 10 known landmarks.



(a) Increasing the initial velocity error.

(b) Increasing the initial position error.

**Figure C-2:** RMS of the norm of the error when increasing the initial error in velocity and position with 3 known landmarks.

---

# Bibliography

- [1] Mingyang Li, Byung Hyung Kim, and Anastasios I. Mourikis. “Real-time motion tracking on a cellphone using inertial sensing and a rolling-shutter camera”. In: *2013 IEEE International Conference on Robotics and Automation*. 2013, pp. 4712–4719. DOI: [10.1109/ICRA.2013.6631248](https://doi.org/10.1109/ICRA.2013.6631248).
- [2] Nathaniel Merrill et al. “Symmetry and Uncertainty-Aware Object SLAM for 6DoF Object Pose Estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2022, pp. 14901–14910.
- [3] Shunsuke KAMIJO, Yanlei Gu, and Li-Ta Hsu. “Autonomous Vehicle Technologies :Localization and Mapping”. In: *IEICE ESS Fundamentals Review* 9 (Oct. 2015), pp. 131–141. DOI: [10.1587/essfr.9.2\\_131](https://doi.org/10.1587/essfr.9.2_131).
- [4] Timothy D. Barfoot. *State Estimation for Robotics*. Cambridge University Press, 2017. DOI: [10.1017/9781316671528](https://doi.org/10.1017/9781316671528).
- [5] M. St-Pierre and D. Gingras. “Comparison between the unscented Kalman filter and the extended Kalman filter for the position estimation module of an integrated navigation information system”. In: *IEEE Intelligent Vehicles Symposium, 2004*. 2004, pp. 831–835. DOI: [10.1109/IVS.2004.1336492](https://doi.org/10.1109/IVS.2004.1336492).
- [6] P.M. Djuric et al. “Particle filtering”. In: *IEEE Signal Processing Magazine* 20.5 (2003), pp. 19–38. DOI: [10.1109/MSP.2003.1236770](https://doi.org/10.1109/MSP.2003.1236770).
- [7] E.A. Wan and R. Van Der Merwe. “The unscented Kalman filter for nonlinear estimation”. In: *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*. 2000, pp. 153–158. DOI: [10.1109/ASSPCC.2000.882463](https://doi.org/10.1109/ASSPCC.2000.882463).
- [8] Axel Barrau and Silvère Bonnabel. “The Invariant Extended Kalman Filter as a Stable Observer”. In: *IEEE Transactions on Automatic Control* 62.4 (2017), pp. 1797–1812. DOI: [10.1109/TAC.2016.2594085](https://doi.org/10.1109/TAC.2016.2594085).
- [9] Manon Kok, Jeroen D. Hol, and Thomas B. Schön. “Using Inertial Sensors for Position and Orientation Estimation”. In: *Foundations and Trends® in Signal Processing* 11.1-2 (2017), pp. 1–153. ISSN: 1932-8354. DOI: [10.1561/20000000094](https://doi.org/10.1561/20000000094).

- [10] Dan Simon. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. USA: Wiley-Interscience, 2006. ISBN: 0471708585.
- [11] Landis Markley and John Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Jan. 2014. ISBN: ISBN: 978-1-4939-0802-8. DOI: [10.1007/978-1-4939-0802-8](https://doi.org/10.1007/978-1-4939-0802-8).
- [12] Philippe Martin and Erwan Salaün. “Generalized Multiplicative Extended Kalman Filter for Aided Attitude and Heading Reference System”. In: *2010 AIAA Guidance, Navigation, and Control Conference*. Toronto, Canada, Aug. 2010, AIAA 2010–8300. URL: <https://hal-mines-paristech.archives-ouvertes.fr/hal-00555580>.
- [13] Landis Markley. “Attitude Error Representations for Kalman Filtering”. In: *Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM* 26 (Mar. 2003), pp. 311–317. DOI: [10.2514/2.5048](https://doi.org/10.2514/2.5048).
- [14] Karmvir Singh Phogat and Dong Eui Chang. “Invariant extended Kalman filter on matrix Lie groups”. In: *Automatica* 114 (2020), p. 108812. ISSN: 0005-1098. DOI: <https://doi.org/10.1016/j.automatica.2020.108812>.
- [15] Axel Barrau and Silvere Bonnabel. “Invariant Kalman Filtering”. In: *Annual Review of Control, Robotics, and Autonomous Systems* 1 (May 2018). DOI: [10.1146/annurev-control-060117-105010](https://doi.org/10.1146/annurev-control-060117-105010).
- [16] Silvere Bonnabel, Philippe Martin, and Pierre Rouchon. “Symmetry-Preserving Observers”. In: *IEEE Transactions on Automatic Control* 53.11 (2008), pp. 2514–2526. DOI: [10.1109/TAC.2008.2006929](https://doi.org/10.1109/TAC.2008.2006929).
- [17] Silvere Bonnabel, Philippe Martin, and Pierre Rouchon. “Non-Linear Symmetry-Preserving Observers on Lie Groups”. In: *IEEE Transactions on Automatic Control* 54.7 (2009), pp. 1709–1713. DOI: [10.1109/TAC.2009.2020646](https://doi.org/10.1109/TAC.2009.2020646).
- [18] Axel Barrau and Silvere Bonnabel. “An EKF-SLAM algorithm with consistency properties”. In: (2016). arXiv: [1510.06263](https://arxiv.org/abs/1510.06263) [cs.R0].
- [19] Silvere Bonnabel. *Symmetries in observer design: review of some recent results and applications to EKF-based SLAM*. 2011. arXiv: [1105.2254](https://arxiv.org/abs/1105.2254) [math.OA].
- [20] Teng Zhang et al. *Convergence and Consistency Analysis for A 3D Invariant-EKF SLAM*. 2017. arXiv: [1702.06680](https://arxiv.org/abs/1702.06680) [cs.R0].
- [21] Teng Zhang et al. *An Invariant-EKF VINS Algorithm for Improving Consistency*. 2017. arXiv: [1702.07920](https://arxiv.org/abs/1702.07920) [cs.R0].
- [22] Guoquan P. Huang, Anastasios I. Mourikis, and Stergios I. Roumeliotis. “Observability-based Rules for Designing Consistent EKF SLAM Estimators”. In: *The International Journal of Robotics Research* 29.5 (2010), pp. 502–528. DOI: [10.1177/0278364909353640](https://doi.org/10.1177/0278364909353640).
- [23] A. Barrau and S. Bonnabel. “Navigating with highly precise odometry and noisy GPS: a case study\*\*This work is supported by the company Safran.” In: *IFAC-PapersOnLine* 49.18 (2016). 10th IFAC Symposium on Nonlinear Control Systems NOLCOS 2016, pp. 618–623. ISSN: 2405-8963. DOI: <https://doi.org/10.1016/j.ifacol.2016.10.234>.

- [24] Niels van der Laan et al. “The Invariant Rauch-Tung-Striebel Smoother”. In: *IEEE Robotics and Automation Letters* 5.4 (2020), pp. 5067–5074. DOI: [10.1109/LRA.2020.3005132](https://doi.org/10.1109/LRA.2020.3005132).
- [25] Easton R. Potokar, Kalin Norman, and Joshua G. Mangelson. “Invariant Extended Kalman Filtering for Underwater Navigation”. In: *IEEE Robotics and Automation Letters* 6.3 (2021), pp. 5792–5799. DOI: [10.1109/LRA.2021.3085167](https://doi.org/10.1109/LRA.2021.3085167).
- [26] Ethan Eade. *Lie Groups for Computer Vision*. 2014.
- [27] Brian C. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*. Vol. 222. Graduate Texts in Mathematics. New York: Springer, 2010, 351 S. ISBN: 9781441923134.
- [28] Timothy D. Barfoot and Paul T. Furgale. “Associating Uncertainty With Three-Dimensional Poses for Use in Estimation Problems”. In: *IEEE Transactions on Robotics* 30.3 (2014), pp. 679–693. DOI: [10.1109/TR0.2014.2298059](https://doi.org/10.1109/TR0.2014.2298059).
- [29] E. LEFFERTS, Landis Markley, and M. SHUSTER. “Kalman filtering for spacecraft attitude estimation”. In: Jan. 1982. DOI: [10.2514/6.1982-70](https://doi.org/10.2514/6.1982-70).
- [30] Landis Markley. “Attitude Error Representations for Kalman Filtering”. In: *Journal of Guidance Control and Dynamics* 26 (Mar. 2003), pp. 311–317. DOI: [10.2514/2.5048](https://doi.org/10.2514/2.5048).
- [31] John Crassidis, Landis Markley, and Yang Cheng. “Survey of Nonlinear Attitude Estimation Methods”. In: *Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM* 30 (Jan. 2007), pp. 12–28. DOI: [10.2514/1.22452](https://doi.org/10.2514/1.22452).
- [32] Chunshi Fan, Ziyang Meng, and Xiaoyun Liu. “Multiplicative quaternion extended consensus Kalman filter for attitude and augmented state estimation”. In: *2016 35th Chinese Control Conference (CCC)*. 2016, pp. 8043–8048. DOI: [10.1109/ChiCC.2016.7554634](https://doi.org/10.1109/ChiCC.2016.7554634).
- [33] Natalia Pavlasek, Alex Walsh, and James Richard Forbes. “Invariant Extended Kalman Filtering Using Two Position Receivers for Extended Pose Estimation”. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. 2021, pp. 5582–5588. DOI: [10.1109/ICRA48506.2021.9561150](https://doi.org/10.1109/ICRA48506.2021.9561150).
- [34] Mitchell R. Cohen, Khairi Abdulrahim, and James Richard Forbes. “Finite-Horizon LQR Control of Quadrotors on  $SE_2(3)$ ”. In: *IEEE Robotics and Automation Letters* 5.4 (2020), pp. 5748–5755. DOI: [10.1109/LRA.2020.3010214](https://doi.org/10.1109/LRA.2020.3010214).
- [35] Lubin Chang, Fangjun Qin, and Jiangning Xu. “Strapdown Inertial Navigation System Initial Alignment Based on Group of Double Direct Spatial Isometries”. In: *IEEE Sensors Journal* 22.1 (2022), pp. 803–818. DOI: [10.1109/JSEN.2021.3108497](https://doi.org/10.1109/JSEN.2021.3108497).
- [36] Zhihuang Zhang et al. “Precise and robust sideslip angle estimation based on INS/GNSS integration using invariant extended Kalman filter”. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 0.0 (0), p. 09544070221102662. DOI: [10.1177/09544070221102662](https://doi.org/10.1177/09544070221102662).
- [37] Haichao Gui and Anton Ruiter. “Quaternion Invariant Extended Kalman Filtering for Spacecraft Attitude Estimation”. In: *Journal of Guidance, Control, and Dynamics* 41 (Dec. 2017), pp. 1–16. DOI: [10.2514/1.G003177](https://doi.org/10.2514/1.G003177).

- [38] Jean Gallier. “The Quaternions and the Spaces  $S^3$ ,  $SU(2)$ ,  $SO(3)$ , and  $RP^3$ ”. In: vol. 38. June 2011, pp. 281–300. ISBN: 978-1-4419-9960-3. DOI: [10.1007/978-1-4419-9961-0\\_9](https://doi.org/10.1007/978-1-4419-9961-0_9).
- [39] Daniel P Koch et al. “Relative multiplicative extended Kalman filter for observable GPS-denied navigation”. In: *The International Journal of Robotics Research* 39.9 (2020), pp. 1085–1121. DOI: [10.1177/0278364920903094](https://doi.org/10.1177/0278364920903094).
- [40] Manon Kok. “Probabilistic modeling for sensor fusion with inertial measurements.” PhD thesis. Linköping studies in science and technology, 2016.
- [41] Elliott Kaplan and Christopher Hegarty. 2017.
- [42] Axel Barrau and Silvère Bonnabel. “Invariant particle filtering with application to localization”. In: *53rd IEEE Conference on Decision and Control*. 2014, pp. 5599–5605. DOI: [10.1109/CDC.2014.7040265](https://doi.org/10.1109/CDC.2014.7040265).
- [43] Axel Barrau. “Non-linear state error based extended Kalman filters with applications to navigation”. PhD thesis. Sept. 2015.
- [44] Jacek Paziewski et al. “An analysis of multi-GNSS observations tracked by recent Android smartphones and smartphone-only relative positioning results”. In: *Measurement* 175 (2021), p. 109162. ISSN: 0263-2241. DOI: <https://doi.org/10.1016/j.measurement.2021.109162>. URL: <https://www.sciencedirect.com/science/article/pii/S0263224121001858>.
- [45] Dominik Prochniewicz, Jacek Kudrys, and Kamil Maciuk. “Noises in Double-Differenced GNSS Observations”. In: *Energies* 15.5 (2022). ISSN: 1996-1073. URL: <https://www.mdpi.com/1996-1073/15/5/1668>.
- [46] Matthew Hartley. “Contact-Aided State Estimation on Lie Groups for Legged Robot Mapping and Control”. In: 2019.
- [47] “Robot Dynamics Lecture Notes”. In: *Robotic Systems Lab, ETH Zurich* (2017).
- [48] Yarong Luo, Mengyuan Wang, and Chi Guo. *The Geometry and Kinematics of the Matrix Lie Group  $SE_K(3)$* . 2021. arXiv: [2012.00950](https://arxiv.org/abs/2012.00950) [cs.R0].

---

# Glossary

## List of Acronyms

<b>KF</b>	Kalman Filter
<b>EKF</b>	Extended Kalman Filter
<b>UKF</b>	Unscented Kalman Filter
<b>MEKF</b>	Multiplicative Exxtended Kalman Filter
<b>IEKF</b>	Invariant Extended Kalman Filter
<b>LIEKF</b>	Left-Invariant Extended Kalman Filter
<b>RIEKF</b>	Right-Invariant Extended Kalman Filter
<b>MEKF-b</b>	MEKF with orientation deviation resolved in body frame
<b>MEKF-n</b>	MEKF with orientation deviation resolved in navigation frame
<b>IMU</b>	Inertial Measurment Unit
<b>GPS</b>	Global Positioning System
<b>RMS</b>	Root Mean Square
<b>SLAM</b>	simultaneous localization and mapping
<b>IRTS</b>	Invariant Rauch-Tung-Striebel
<b>DVL</b>	doppler velocity logs
<b>BCH</b>	Baker-Campbell-Hausdorff
<b>LIDAR</b>	Light Detection and Ranging
<b>GNSS</b>	Global Navigation Satellite System
<b>UAV</b>	Unmanned Aerial Vehicle

## List of Symbols

$(\cdot)^\times$  Cross operator for  $\mathfrak{so}(3)$

$(\cdot)^T$	Transpose
$(\cdot)^\vee$	Operator mapping an element of $\mathfrak{g}$ to $\mathbb{R}^n$
$(\cdot)^\wedge$	Operator mapping an element of $\mathbb{R}^n$ to $\mathfrak{g}$
$\mathbb{R}^n$	The vector space of real n-dimensional vectors
$\mathbb{R}^{n \times m}$	The vector space of real n-dimensional vectors
$\mathbb{S}^n$	The set of unit quaternions
$\mathcal{F}_b$	Body reference frame
$\mathcal{F}_n$	Navigation reference frame
$\mathbf{0}$	Zero matrix
$\mathbf{C}_{nb}$	Rotation matrix of $\mathcal{F}_n$ relative to $\mathcal{F}_b$
$\mathbf{I}$	Identity matrix
$\mathbf{P}$	Covariance matrix
$\mathbf{Q}$	Process noise covariance matrix
$\mathbf{R}$	Measurement noise covariance matrix
$\mathbf{r}^{zw}$	The position of point z relative to point w
$\mathbf{r}_n$	The position resolved in $\mathcal{F}_n$
$\mathbf{v}^{zw/n}$	The velocity of point z relative to point w with respect to $\mathcal{F}_n$
$\underline{\mathbf{r}}$	The position vector
$\omega^{bn}$	Angular velocity of $\mathcal{F}_b$ relative to $\mathcal{F}_n$
Ad	Adjoint operator for $\mathcal{G}$
ad	Adjoint operator for $\mathfrak{g}$