

# **State Estimation for Systems on Lie Groups with Nonideal Measurements**

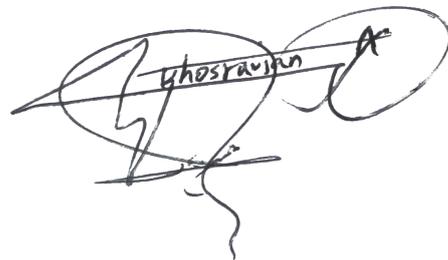
**Alireza Khosravian Hemami**

A thesis submitted for the degree of  
Doctor of Philosophy of  
The Australian National University

September 2016

© Alireza Khosravian Hemami 2016

Except where otherwise indicated, this thesis is my own original work.

A handwritten signature in black ink. The signature is stylized and appears to be 'Alireza Khosravian Hemami'. The name 'Khosravian' is written in a cursive script across the middle of the signature. There are several loops and flourishes around the text.

Alireza Khosravian Hemami

29 September 2016

I conducted this PhD thesis under the supervision of Dr. Jochen Trumpf and Prof. Robert Mahony. Most of the results in this thesis have been published at top-tier international conferences and journals. These publications are listed below and some of them have been achieved in collaboration with other researchers.

### **Journal Papers**

- A Khosravian, J Trumpf, R Mahony, C Lageman, "Observers for invariant systems on Lie groups with biased input measurements and homogeneous outputs", *Automatica*, vol. 55, pages 19-26, 2015 (cited as [86] and its arXiv version as [87]).
- A Khosravian, J Trumpf, R Mahony, T Hamel, "State estimation for invariant systems on Lie groups with delayed output measurements", vol. 68, pages 254-265, *Automatica* (cited as [84]).

### **Conference Papers**

- A Khosravian, J Trumpf, R Mahony, C Lageman, "Bias estimation for invariant systems on Lie groups with homogeneous outputs", *IEEE 52nd Annual Conference on Decision and Control (CDC)*, pages 4454-4460, 2013 (cited as [85]).
- A Khosravian, J Trumpf, R Mahony, T Hamel, "Velocity aided attitude estimation on  $SO(3)$  with sensor delay", *IEEE 53rd Annual Conference on Decision and Control (CDC)*, pages 114-120, 2014 (cited as [82]).
- A Khosravian, J Trumpf, R Mahony, T Hamel, "Recursive attitude estimation in the presence of multi-rate and multi-delay vector measurements", *American Control Conference (ACC)*, 2015 (cited as [83]).

Apart from the above publications, I published the following papers during my PhD education, the results of which are not presented in this thesis.

- A Khosravian, "Stability analysis and near optimal gain tuning of an attitude estimator on the special orthogonal group", *IEEE 52nd Annual Conference on Decision and Control (CDC)*, pages 5060-5065, 2013 (cited as [79]).
- A Khosravian, J Trumpf, R Mahony, "State estimation for nonlinear systems with delayed output measurements", *IEEE 54th Annual Conference on Decision and Control (CDC)*, pages 6630-6635, 2015 (cited as [81]).

To my dear parents, Badri and Mohammad



---

# Acknowledgments

---

It is my pleasure to thank those who made this thesis possible. I am indebted to my supervisors, Dr Jochen Trumpf and Professor Robert Mahony for their countless valuable advices, for being great teachers and mentors, and for believing on me at all stages of my PhD. I am incapable of thanking them enough for their immense patience, strong support, and professional supervision. I would like to thank my panel advisor Adrian Bishop as well as my collaborators Tarek Hamel, Christian Lageman, and Mohsen Zamani for their helpful discussions and contributions. I acknowledge the anonymous thesis examiners for their valuable comments and feedbacks in the review process. I owe my deepest gratitude to a group of colleges and friends who helped with conducting the experimental results of the thesis; Sean O'Brien, Andrew Tridgell, Grant Morphett, Paul Riseborough, Jack Pittar, Evan Slatyer, Benjamin Nizette, Salim Masoumi, Arash Khodaparastsichani, and Juan David Adarve.

I would like to thank the staff and residents of Graduate House and University House for making ANU campus feel like home to me. Particularly, Peter, Tony, Gina, Lyn, and Kaori in the management and all of my very good friends at the student leadership team; Anna, Belinda, Channa, Dan, Eleonora, Guanhua, Juan, Kim-long, Lara, Lauren, Louisa, Maria, Mark, Mohsen, Mojtaba, Ronald, Salim, Wakako, Zheng, and Zhison. I am also grateful to my amazing group of friends in Canberra who were always there for me; Abbas, Alireza K, Alireza M, Anita, Anna, Arash, Behrooz, Behzad, Ehsan A, Ehsan N, Fatemeh E, Fatemeh R, Fatemeh S, Hajar, Hamid, Ladan, Mahin, Mahmoud, Marzieh, Masoume, Mehdi, Mohammad D, Mohammad E, Mohammad N, Mohammadreza, Mohsen, Morteza, Mousa, Nojan, Pegah, Sadegh, Sahba, Salim, Sara M, Sara T, and Zahra. Last but not least, I would like to thank my parents, Badri and Mohammad, and my siblings, Atefeh and Mohammad Hossein, who have supported me during all these years with their continuous encouragement, incredible patience, immense kindness, and unconditional love.

This work is supported by the Australian National University and the Australian Research Council through the ARC Discovery Project DP120100316.



---

# Abstract

---

This thesis considers the state estimation problem for invariant systems on Lie groups with inputs in its associated Lie algebra and outputs in homogeneous spaces of the Lie group. A particular focus of this thesis is the development of state estimation methodologies for systems with nonideal measurements, especially systems with additive input measurement bias, output measurement delay, and sampled outputs. The main contribution of the thesis is to effectively employ the symmetries of the system dynamics and to benefit from the Lie group structure of the underlying state space in order to design robust state estimators that are computationally simple and are ideal for embedded applications in robotic systems.

We address the input measurement bias problem by proposing a novel nonlinear observer to adaptively eliminate the input measurement bias. Despite the nonlinear and non-autonomous nature of the resulting error dynamics and the complexity of the underlying state space, the proposed observer exhibits asymptotic/exponential convergence of the state and bias estimation errors to zero.

To tackle the output measurement delay problem, we propose novel dynamic predictors used in an observer-predictor arrangement. The observer provides estimates of the delayed state using the delayed output measurements and the predictor takes those estimates, compensates for the delay, and provides predictions of the current state. Separately, we propose output predictors employed in a predictor-observer arrangement to address the problem of sampled output measurements. The output predictors take the sampled measurements and provide continuous predictions of the current outputs. Feeding the predicted outputs into the observer yields estimates of the current state. Both methods rely on the invariance of the underlying system dynamics to recursively provide predictions with low computation requirements.

We demonstrate applications of the theory with examples of attitude, velocity, and position estimation on  $SO(3)$  and  $SE(3)$ . A key contribution of this thesis is the development of C++ libraries in an embedded implementation as well as experimental verification of the developed theory with real flight tests using model UAVs.



---

# Contents

---

<b>Acknowledgments</b>	<b>vii</b>
<b>Abstract</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 A first literature review . . . . .	4
1.3 Problems considered . . . . .	5
1.3.1 Unknown input measurement bias . . . . .	5
1.3.2 Output measurement delay . . . . .	6
1.3.3 Output measurement sampling . . . . .	6
1.4 Thesis contributions . . . . .	7
1.5 Notations and definitions . . . . .	9
<b>2 Input Bias Estimation for Invariant Systems on Lie Groups ...</b>	<b>11</b>
2.1 Related work . . . . .	12
2.2 Problem Formulation . . . . .	13
2.3 Error Definition and Autonomy of Error Dynamics . . . . .	17
2.4 Observer Design and Analysis . . . . .	20
2.5 Constructing Invariant Cost Functions on Lie Groups . . . . .	29
2.6 Example: Attitude Estimation Using Biased Angular Velocity Measurement . . . . .	31
2.7 Example: Pose Estimation Using Biased Velocity Measurements . . . . .	33
2.8 Summary . . . . .	39
<b>3 State Estimation for Systems with Delayed Output Measurements</b>	<b>41</b>
3.1 Related work . . . . .	42
3.2 Problem formulation . . . . .	44
3.3 Observer-predictor approach . . . . .	47

---

3.4	Recursive implementation of non-recursive predictors . . . . .	61
3.5	Simulations . . . . .	63
3.5.1	Pure prediction error . . . . .	63
3.5.2	Total observer-predictor error . . . . .	64
3.6	Summary . . . . .	67
<b>4</b>	<b>State Estimation for Systems with Sampled Output Measurements</b>	<b>69</b>
4.1	Related work . . . . .	69
4.2	Sensor modeling with samples and delays . . . . .	73
4.2.1	Physically inspired modeling of sampling and delays . . . . .	73
4.2.2	An input-to-output equivalent model . . . . .	75
4.3	Problem formulation . . . . .	76
4.4	Predictor-observer Approach . . . . .	79
4.5	Simulation results . . . . .	83
4.6	Embedded Software Development and Experimental Verification . . . . .	88
4.6.1	Autopilot software and experimental platform . . . . .	90
4.6.2	Employed predictor . . . . .	91
4.6.3	Experimental results . . . . .	94
4.6.4	Predictor-observer approach versus original ArduPilot EKF: ro- bust gain tuning . . . . .	99
4.7	Summary . . . . .	103
<b>5</b>	<b>Conclusions and Future Work</b>	<b>105</b>
5.1	Reverse predictor theory . . . . .	106
5.2	Stochastic propagation properties of predictors . . . . .	110
5.3	Predictor-observer with input measurement bias . . . . .	114
5.4	Application examples to other Lie groups . . . . .	116
<b>6</b>	<b>Appendix</b>	<b>117</b>
6.1	Lemma 6.1.1 . . . . .	117
6.2	Lemma 6.2.1 . . . . .	118
6.3	Lemma 6.3.1 . . . . .	118
6.4	Lemma 6.4.1 . . . . .	119
6.5	Predictor C++ library for ArduPilot . . . . .	119

---

6.5.1	AP_Predictors . . . . .	120
6.5.2	Combining AP_Predictors with AP_NavEKF2 . . . . .	122



---

# List of Figures

---

3.1	Proposed observer-predictor methodology. . . . .	48
3.2	Pure attitude and velocity prediction errors. . . . .	65
3.3	Pure observer error and total observer-predictor error. . . . .	66
4.1	Modelling the effect of sampling and delays in attitude sensors . . . . .	75
4.2	Simplified input-to-output equivalent model of Fig. 4.1 . . . . .	75
4.3	Illustration of the proposed predictor-observer approach. . . . .	79
4.4	Attitude estimation error of the combined predictor-observer. . . . .	86
4.5	Attitude estimation error of the ad-hoc observer. . . . .	86
4.6	Attitude estimation error of the combined predictor-observer (large sensor delay). . . . .	87
4.7	Attitude estimation error of the ad-hoc observer (large sensor delay). . .	87
4.8	Attitude estimation error of the combined predictor-observer (with de- lay uncertainty). . . . .	87
4.9	Photo taken at Canberra Model Aircraft Club. . . . .	89
4.10	Photo taken at Canberra Model Aircraft Club. . . . .	89
4.11	Flight path of the plane according to GPS measurements. . . . .	95
4.12	Estimates of the Euler angles of the plane . . . . .	96
4.13	North position of the plane provided by GPS versus its prediction . . . .	97
4.14	East position of the plane provided by GPS versus its prediction . . . .	98
4.15	North position estimates . . . (default EKF gains are used). . . . .	99
4.16	East position estimates . . . (default EKF gains are used). . . . .	101
4.17	North and East position estimates . . . (re-tuned EKF gains are used) . .	101
4.18	North position estimates . . . (high EKF gains are used). . . . .	102
4.19	East position estimates . . . (high EKF gains are used). . . . .	102



---

# Introduction

---

This chapter provides an overview of this thesis by presenting motivations for the subject of this thesis as well as a brief literature review, explaining the problems considered, and listing the main contributions of the thesis. In depth literature reviews and detailed explanations of the contributions are given at the beginning of each chapter.

## 1.1 Motivation

Many physical systems require the knowledge of their internal states such as their position, orientation, and velocity for various reasons such as effective control, navigations, fault detection, path planning, etc. In most practical situations, obtaining a reliable measurement of the internal states of those physical systems directly is not possible and it is necessary to use state estimators. In many applications, particularly in the field of robotics, those internal states naturally lives on Lie groups [17, 32, 40, 71]. For instance, take the attitude estimation problem which is to determine the orientation of a rigid body with respect to a known frame of reference [34, 55, 69, 80, 89, 101, 105, 120, 124, 130, 136, 144]. In this case, the orientation of a rigid body is modeled by a  $3 \times 3$  orthogonal matrix belonging to the Lie group  $SO(3)$ . Apart from the attitude estimation problem, design of state estimator on Lie groups is also motivated by real world applications such as; attitude estimation on pose estimation on  $SE(3)$  [25, 63, 65, 116, 121, 135], homography estimation on  $SL(3)$  [56], and motion estimation of chained systems on nilpotent Lie groups [97] (e.g. front-wheel drive cars or kinematic cars with  $k$  trailers). The Special Linear group  $SL(2)$  which arises in some compute vision applications [44, 73, 96], and complex-valued Lie groups and unitary groups arising in multiantenna transceiver techniques, in

sensor array applications in biomedicine, and in machine learning [51, Section 3] are other motivating examples.

Work on state estimation on Lie groups dates back to 1970s, see e.g. the seminal works of Brockett [38, 39] and Willsky [140, 141]. Stochastic filtering methods and deterministic observers are two competing approaches for designing state estimators. Stochastic methods rely on stochastic modeling of the sensor noise and system model uncertainties to design state estimators that are optimal with respect to some metric. Deterministic observers do not require modeling of stochastic noise. Instead, their objective is to guarantee the stability of the estimation error in the noise free condition. There is a rich literature on stochastic state estimation methods on Lie groups (see e.g. [48, 49, 139] as well as [42, Chapters 14 and 15] and the references therein) and their applications to real world problems [44, 46, 66, 98, 104, 105, 114, 126]. Nevertheless, this thesis chooses the deterministic setup for modeling and design of state estimators. Hence, in the context of this thesis, the word "state estimator" refers to the "state observer" and we use these two words interchangeably.

Systematic observer design methodologies for deterministic state estimation of invariant systems on *general* Lie groups have been proposed that lead to strong stability and robustness properties [28, 34, 94]. All of these observer design methodologies require fusing the measurements of both inputs and outputs of systems. In practice, those input and output measurements are nonideal and may be corrupted by sensor biases, sensor sampling, and sensor delays. These effects, if not compensated for properly, might lead to poor performance of the observers or even cause instability [22, 53, 74, 76, 100, 106, 109, 117, 125].

The effects of nonideal measurements are more significant in robotics applications involving low cost sensor suites. For instance, low cost MEMS<sup>1</sup> gyros usually exhibit a significant measurement bias which should be compensated to obtain reliable estimations of attitude [101, 116, 136]. Commercial GPS units usually exhibit a significant measurement delay that can be as large as hundreds of milliseconds. Those GPS units normally provide low sampling rates (less than 5 Hz) [58, 88]. The effects of these delays and sampling are very significant in commercial UAVs where onboard navigation algorithms normally run as fast as 50-200 Hz [4]. Similar delay and sampling effects are present in indoor flight environments where the pose

---

<sup>1</sup>Micro Electro-Mechanical Systems

---

data provided by devices such as VICON or OptiTrack are available to the onboard navigation system of UAVs with variable delays and sampling rates due to the communication channel from those sensors to the UAV. In satellite attitude estimation applications, sampling effects are present where high accuracy output sensors such as star trackers or earth sensors provide measurements at low sampling rates (0.5 to 10 Hz) [105].

For the special problem of attitude estimation on  $SO(3)$ , observers are available in the literature that adaptively compensate for the gyro bias [101, 116, 136]<sup>2</sup>. A similar observer is designed for adaptively estimating both the linear and angular velocity biases in the pose estimation problem on  $SE(3)$  [135]. This naturally raises the problem of generalizing the concurrent state and input bias estimation to invariant systems on *general* Lie groups. This problem is investigated in Chapter 2 where a systematic observer design methodology is proposed that unifies the  $SO(3)$  and  $SE(3)$  examples into a single framework that applies to any invariant kinematic system on a Lie group.

A particular focus of this thesis is state estimation with output measurement delays and sampling. This problem attracted our attention when a group of autopilot developers at Canberra UAV [7] reported occasional oscillations leading to instability of the popular attitude observer of [101]. In his talk at the Australian National University, Andrew Tridgell<sup>3</sup> [131] presented flight tests showing that the geometric observer of [101] is prone to instability if accelerometer measurements are aided by GPS velocity measurements. Reconstructing the flight test using a software-in-the-loop system, he concluded that the GPS delay and sampling effects are the main causes of the instability when the UAVs perform high acceleration maneuvers. This is the main motivation for consideration of the output measurement delay and sampling problems in Chapters 3 and 4, respectively. We propose predictors to compensate for the delay and sampling effects for invariant systems on general Lie groups. The resulting theory is not only applicable to the GPS delay problem discussed above, it is also applicable to lots of similar practical problems such as the time varying sampling and delay problem of VICON or OptiTrack in indoor flight environments, the sampling and delay problem of visual navigation systems, and attitude estima-

---

<sup>2</sup>This observer is commonly known as DCM (Direction Cosine Matrix) amongst UAV enthusiasts and the robotics community [8, 12].

<sup>3</sup>The main developer of the ArduPilot autopilot system [5].

tion using star sensors with low sampling rates. Through a close collaboration with Andrew Tridgell's team at Canberra UAV and 3DRobotics [1], we implemented the proposed predictors into the embedded autopilot system of model UAVs and verified their performance in real flight tests. The main strength of this thesis is to provide a balance between a very high level of abstraction of the developed theoretical results (by providing theories applicable to systems on general Lie groups) and justifying those theories by presenting real world applications and experimental verifications with real UAVs.

## 1.2 A first literature review

Systematic observer design methodologies for invariant systems on general Lie groups have been proposed that lead to strong stability and robustness properties. Particularly, Bonnabel *et al.* [33–35] consider observers which consist of a copy of the system and a correction term, along with a constructive method to find suitable symmetry-preserving correction terms. The construction utilizes the invariance of the system and the moving frame method, leading to local convergence properties of the observers. Also, [28] extends those constructive methods in order to apply them to a wider class of systems on Lie groups. This leads to development of a so-called Invariant Extended Kalman Filtering approach with provable local stability properties. Methods proposed in [92–94] to achieve almost globally convergent observers. A key aspect of the design approach proposed in [92–94] is the use of the invariance properties of the system to ensure that the error dynamics are globally defined and are autonomous. This leads to a straight forward stability analysis and excellent performance in practice. More recent extensions to early work in this area was the consideration of output measurements where a partial state measurement is generated by an action of the Lie group on a homogeneous output space [33–35, 92, 93, 102]. Also, [123, 143] develop a rigorous theory for designing minimum energy observers on Lie groups with near optimal performance. Recently, [69, 70] proposed state estimation methods on the specific Lie groups  $SO(3)$  and  $SE(3)$  based on the Lagrange–d'Alembert principle.

Compensation of input measurement bias is studied in the specific cases of attitude estimation on  $SO(3)$  and pose estimation on  $SE(3)$  [101, 135, 136]. These

---

methods strongly depend on particular properties of the specific Lie groups  $SO(3)$  or  $SE(3)$  and do not directly generalize to general Lie groups.

Deterministic state estimation for systems on  $\mathbb{R}^n$  with sampled and delayed measurements is a classical problem that has been extensively studied and many results with strong stability proofs are currently available [16, 18, 19, 30, 41, 41, 53, 74–76, 134]. Nevertheless, no such results are available (prior to this thesis) for systems on general Lie groups. The only available publications in this are [24] and [58] that consider the special case of attitude estimation.

In this thesis, we consider the state estimation for invariant systems on general Lie groups where input measurement bias, output measurement delay and sampling are present. In the following, we further describe these problems. In depth literature reviews and detailed explanations of each problem are given at the beginning of the associated chapter.

### 1.3 Problems considered

We consider three classes of nonideal measurements; input measurement corrupted by unknown bias, output measurement with delay, and output measurement with sampling.

#### 1.3.1 Unknown input measurement bias

In many applications, measurements of system input are often corrupted by an unknown additive bias that must be estimated and compensated to achieve good observer error performance. In practice, it is usually possible to approximately model the sensor bias with unknown additive constants. This way, the underlying observer design problem is to estimate the state and the unknown input bias concurrently. In Chapter 2, we provide observer design methodologies for adaptively estimating the state and input bias for invariant systems on general Lie groups. Moreover, we provide a detailed example of attitude and position estimation on  $SE(3)$  where measurements of the angular and linear velocity are biased to illustrate the applications of the developed observer design methodology.

### 1.3.2 Output measurement delay

In many practical scenarios, measurements of the outputs of the system are usually available to the user with some delay (due to various reasons including physical properties of sensors or the environment, slow transients, internal signal processing of sensors, extensive filtering of sensor measurements for noise reduction, and communication delays from sensors to processing units) while the inputs are measured without significant delays. For example, in the velocity aided attitude estimation problem, measurements of the linear velocity (output) provided by commercial GPS units are usually delayed with respect to the actual velocity of the vehicle. In contrast, measurements of the vehicle's angular velocity and linear acceleration provided by an onboard IMU are almost instantaneous. In Chapter 3, we provide an *observer-predictor* methodology to cope with the output measurement delay for invariant systems on general Lie groups. In this method, the observer takes the delayed measurements and provides estimates of the delayed state. We propose predictors that take the estimates of the delayed state, compensate for the delay, and provide predictions of the current state.

### 1.3.3 Output measurement sampling

In applications involving state estimation of mechanical systems, measurements of the input are usually obtained at a very high sampling rate either through odometry or via inertial sensors. In many applications, however, measurements of the system outputs are obtained at much lower sampling rate compared to the sampling rate of system inputs. For instance, in satellite attitude estimation applications, high accuracy output sensors such as star trackers or earth sensors provide measurements at low sampling rates (0.5 to 10 Hz) while onboard gyros can easily provide high bandwidth measurements at kHz rates. In outdoor UAV navigation scenarios, low cost GPS units provide measurements of position and velocity at around 5 Hz which is much lower than the rate of input measurements provide by IMUs (gyros and accelerometers). As explained in the previous Section, these output measurements are also usually delayed compared to the input measurements. Sensor sampling and delays can negatively affect the stability and robustness of any observer or filter and degrade their performance if they are not compensated for properly. In Chapter 4,

---

we investigate the problem of state estimation for invariant systems on general Lie groups where the measurements of outputs are sampled and delayed. We propose a cascade *predictor-observer* approach in which the predictor takes the sampled and delayed output measurements and provides predictions of the current outputs. The predicted outputs are then fed into an observer or filter to estimate the system states. Compared to the method proposed in Chapter 3, the method of Chapter 4 provides stronger tools for state estimation by eliminating both output sampling and delay effects. Nevertheless, it extensively relies on symmetries of the output maps of the system and hence is less flexible than the method of Chapter 3 in dealing with diverse classes of outputs.

## 1.4 Thesis contributions

The main theoretical contribution of the thesis is to effectively employ the underlying symmetries of the system dynamics and output maps in order to propose methodologies to cope with the problems discussed in the previous section. In the following, we briefly discuss the main contributions of the thesis. More detailed discussions about the contributions of each chapter is given in the introductory part and the conclusion of each chapter.

- We study the problem of concurrent state and input bias estimation for invariant systems on general Lie groups with outputs living in (possibly different) homogeneous spaces. This problem is investigated in Chapter 2. We prove that any candidate observer that is *implementable* based on the measurements of the inputs and outputs of the system produces non-invariant error dynamics unless the Lie group is Abelian (Theorem 2.3.1). Despite the nonlinear and non-autonomous nature of the error dynamics, we propose a novel observer and we prove the exponential stability of its error dynamics (Theorems 2.4.1 and 2.4.3). The approach taken employs a general gain mapping applied to the differential of a cost function to generate the innovation term of the observer. We establish a systematic method for construction of invariant cost functions ensuring that the resulting observer is implementable and yields desired stability of the error dynamics (Proposition 2.5.1). The results of Chapter 2 are published in [85–87].

- We tackle the problem of state estimation for systems on general matrix Lie groups when measurements of system outputs are delayed. This problem is investigated in Chapter 3. We propose an observer-predictor methodology for invariant systems on Lie groups. Given an observer or filter that has desired stability properties when the system outputs are delay-free, we propose an observer-predictor methodology that preserves those stability properties when the system outputs are delayed. We design dynamic predictors that use the delayed estimates from the observers together with the current inputs in order to predict the current state of the system (Theorems 3.3.3 and 3.3.4). The proposed predictors are computationally very cheap and demonstrate excellent robustness properties, making them ideal for embedded implementation on low cost robotics applications. The results of Chapter 3 are presented in [82, 84].
- We consider the state estimation problem for invariant systems on Lie groups where sampled and delayed output measurements are available. This problem is investigated in Chapter 4. We propose a cascade predictor-observer approach in which the predictor takes the sampled and delayed output measurements and provides predictions of the current outputs. The predicted outputs are then fed into an observer or filter to estimate the system states. We prove that in noise free conditions, the current prediction of the output is indeed equal to the actual ideal output, independent of the sampling rate and delay associated with the output measurement (Theorem 4.4.1). This is a very strong result which enables application of the developed predictors in a wide range of real world applications involving multi rate sensors with possibly different and even time-varying delays. A preliminary version of the results of Chapter 4 are presented in [83].
- Along with presenting deep technical results, we provide real world applications and experimental verification of the developed theory to provide the reader with a deeper insight. In Chapter 4, we provide an example of output predictor design for attitude and velocity estimation to cope with the measurement sampling and delay of GPS and magnetometer. A key contribution of this thesis is embedded implementation and verification of the performance of the

proposed predictors by presenting real flight tests with model UAVs (Section 4.6.1). We further illustrate the advantages of the proposed method over available state-of-the-art estimation approaches by offline processing of sensor data using post-processing tools of the ArduPilot system (Section 4.6.4).

## 1.5 Notations and definitions

We use the following notations throughout this thesis. Let  $G$  be a finite-dimensional real connected Lie group with associated Lie algebra  $\mathfrak{g}$ . Denote the identity element of  $G$  by  $I$ . Left (resp. right) multiplication of  $X \in G$  by  $S \in G$  is denoted by  $L_S X = SX$  (resp.  $R_S X = XS$ ). The Lie algebra  $\mathfrak{g}$  can be identified with the tangent space at the identity element of the Lie group, i.e.  $\mathfrak{g} \cong T_I G$ . For any  $u \in \mathfrak{g}$ , one can obtain a tangent vector at  $S \in G$  by left (resp. right) translation of  $u$  denoted by  $S[u] := T_I L_S[u] \in T_S G$  (resp.  $[u]S := T_I R_S[u] \in T_S G$ ). The element inside the brackets  $[\cdot]$  denotes the vector on which a linear mapping (here the tangent map  $T_I L_S: \mathfrak{g} \rightarrow T_S G$  or  $T_I R_S: \mathfrak{g} \rightarrow T_S G$ ) acts. For convenience, we omit the notation  $[\cdot]$  if there is no risk of confusion. The adjoint map at the point  $S \in G$  is denoted by  $\text{Ad}_S: \mathfrak{g} \rightarrow \mathfrak{g}$  and is defined by  $\text{Ad}_S[u] := S[u]S^{-1} = T_S R_{S^{-1}}[T_I L_S[u]] = T_S R_{S^{-1}} \circ T_I L_S[u]$  where  $\circ$  denotes the composition of two maps. Choose an inner product  $\langle\langle \cdot, \cdot \rangle\rangle$  on  $\mathfrak{g}$  and denote the corresponding induced norm on  $\mathfrak{g}$  by  $\|\cdot\|$ . Denote by  $\langle\langle \cdot, \cdot \rangle\rangle_S^r$  (resp.  $\langle\langle \cdot, \cdot \rangle\rangle_S^l$ ) a right-invariant (resp. left-invariant) Riemannian metric at a point  $S$  induced by  $\langle\langle \cdot, \cdot \rangle\rangle$  using right (resp. left) translation of the Lie algebra  $\mathfrak{g}$ . Denote the corresponding induced right-invariant (resp. left-invariant) norm on  $T_S G$  by  $|\cdot|_S^r$  (resp.  $|\cdot|_S^l$ ). Denote the geodesic distance with respect to (w.r.t.)  $\langle\langle \cdot, \cdot \rangle\rangle^r$  (resp.  $\langle\langle \cdot, \cdot \rangle\rangle^l$ ) between the points  $S$  and  $X$  by  $d_r(S, X)$  (resp.  $d_l(S, X)$ ). For a finite-dimensional vector space  $V$ , we denote its corresponding dual and bidual vector spaces by  $V^*$  and  $V^{**}$  respectively. A linear map  $F: V^* \rightarrow V$  is called positive definite if  $v^*[F[v^*]] > 0$  for all  $0 \neq v^* \in V^*$ . The dual of  $F$  is denoted by  $F^*: V^* \rightarrow V^{**}$  and is defined by  $F^*[v^*] = v^* \circ F$ . The linear map  $F$  is called symmetric (resp. anti-symmetric) if  $v^*[F[w^*]] = w^*[F[v^*]]$  (resp.  $v^*[F[w^*]] = -w^*[F[v^*]]$ ) for all  $v^*, w^* \in V^*$ , and it is called symmetric positive definite if it is symmetric and positive definite. We can extend the above notion of symmetry and positiveness to linear maps  $H: W \rightarrow W^*$  as well. Defining  $V := W^*$ ,  $H$  is called positive definite if  $H^*: V^* \rightarrow V$  is positive

definite and it is called symmetric if  $H^*$  is symmetric. Positive definite cost functions on manifolds are also used in the thesis and should not be mistaken with positive definite linear maps. The notations  $\underline{\sigma}(\text{Ad}_S)$  and  $\bar{\sigma}(\text{Ad}_S)$ , respectively, denotes the smallest and the largest singular value of the adjoint map w.r.t. the norm  $\|\cdot\|$ , defined by  $\underline{\sigma}(\text{Ad}_S) := \min_{v \in \mathfrak{g}, \|v\|=1} \|\text{Ad}_S v\|$  and  $\bar{\sigma}(\text{Ad}_S) := \max_{v \in \mathfrak{g}, \|v\|=1} \|\text{Ad}_S v\|$ , respectively. For a matrix  $A \in \mathbb{R}^{m \times m}$ , the notations  $\underline{\sigma}(A)$ ,  $\bar{\sigma}(A)$ , and  $\text{cond}(A)$ , respectively, denote the smallest singular value, the largest singular value, and the condition number of that matrix (with respect to the Euclidean norm on  $\mathbb{R}^m$ ). We say the trajectory  $a(t) \in \mathbb{R}^+$  converges to zero and denote  $a(t) \rightarrow 0$  if  $\lim_{t \rightarrow +\infty} a(t) = 0$ . We write  $a(t) \xrightarrow{\text{exp}} 0$  and we say  $a(t)$  converges exponentially to zero if there exist positive constants  $c$  and  $\alpha$  such that  $a(t) \leq c \exp(-\alpha t)$  for all  $t \geq 0$ .

The following standard definitions are used throughout the thesis [36, 61, 90]. The group action  $h : G \times M \rightarrow M$  is called *transitive* if there exists  $y \in M$  such that every  $y \in M$  satisfies  $y = h(X, y)$  for some  $X \in G$ . The space  $M$  is called a *homogeneous space* of  $G$  if there exists a transitive action of  $G$  on  $M$ . The Lie group  $G$  has a *faithful representation* as a finite-dimensional matrix Lie group if there exist a positive integer  $m$  and an injective Lie group homomorphism  $\Phi : G \rightarrow \text{GL}(m)$  into the group  $\text{GL}(m)$  of invertible  $m \times m$  matrices.

# Input Bias Estimation for Invariant Systems on Lie Groups with Homogeneous Outputs

---

This chapter provides a new observer design methodology for invariant systems whose state evolves on a Lie group with outputs in a collection of related homogeneous spaces and where the measurement of system input is corrupted by an unknown constant bias. The key contribution of the Chapter is to study the combined state and input bias estimation problem in the general setting of Lie groups, a question for which only case studies of specific Lie groups are available in prior literature. We show that any candidate observer (with the same state space dimension as the observed system) results in non-autonomous error dynamics, except in the trivial case where the Lie group is Abelian. This precludes the application of the standard non-linear observer design methodologies available in the literature and leads us to propose a new design methodology based on employing invariant cost functions and general gain mappings. We provide a rigorous and general stability analysis for the case where the underlying Lie group allows a faithful matrix representation. We demonstrate our theory in the example of rigid body pose estimation and show that the proposed approach unifies two competing pose observers published in prior literature. The contributions presented in this chapter are published in [85–87].

## 2.1 Related work

Many mechanical systems carry a natural symmetry or invariance structure expressed as invariance properties of their dynamical models under transformation by a symmetry group. For totally symmetric kinematic systems, the system can be lifted to an invariant system on the symmetry group [102]. Work in this area is motivated by applications in analytical mechanics, robotics and geometric control of mechanical systems [17, 32, 40, 71, 105]. Systematic observer design methodologies for deterministic state estimation of invariant systems on *general* Lie groups have been proposed that lead to strong stability and robustness properties [28, 34, 94]. All asymptotically stable observer designs for kinematic systems on Lie groups depend on a measurement of system input [28, 33, 68, 94, 102, 123, 143]. In practice, measurements of system input are often corrupted by an unknown bias that must be estimated and compensated to achieve good observer error performance. The specific cases of attitude estimation on  $SO(3)$  and pose estimation on  $SE(3)$  have been studied independently, and methods have been proposed for the concurrent estimation of state and input measurement bias [101, 135, 136]. These methods strongly depend on particular properties of the specific Lie groups  $SO(3)$  or  $SE(3)$  and do not directly generalize to general Lie groups. To our knowledge, there is no existing work on combined state and input bias estimation for general classes of invariant systems.

In this chapter, we tackle the problem of observer design for general invariant systems on Lie groups with homogeneous outputs when the measurement of system input is corrupted by an unknown constant bias. The observer is required to be implementable based on available sensor measurements; the system input in the Lie algebra, corrupted by an unknown bias, along with a collection of partial state measurements (i.e. outputs) that ensure observability of the state. For bias free input measurements, it is always possible to obtain autonomous dynamics for the standard error [92–94], and previous observer design methodologies for systems on Lie groups rely on the autonomy of the resulting error dynamics. However, for concurrent state and input measurement bias estimation, we show that *any* implementable candidate observer (with the same state space dimension as the observed system) yields non-autonomous error dynamics unless the Lie group is Abelian (Theorem 2.3.1). This

result explains why the previous general observer design methodologies for the bias-free case do not apply and why the special cases considered in prior works [65, 135] do not naturally lead to a general theory.

We go on to show that, despite the nonlinear and non-autonomous nature of the error dynamics, there is a natural choice of observer for which we can prove exponential stability of the error dynamics (Theorems 2.4.1 and 2.4.3). The approach taken employs a general gain mapping applied to the differential of a cost function rather than the more restrictive gradient-like innovations used in prior work [92–94]. We also propose a systematic method for construction of invariant cost functions based on lifting costs defined on the homogeneous output spaces (Proposition 2.5.1). To demonstrate the generality of the proposed approach we consider the problem of rigid body pose estimation using landmark measurements when the measurements of linear and angular velocity are corrupted by constant unknown biases. We show that for specific choices of gain mappings the resulting observer specializes to either the gradient-like observer of [65] or the non-gradient pose estimator proposed in [135], unifying these two state-of-the-art application papers in a single framework that applies to any invariant kinematic system on a Lie group.

The Chapter is organized as follows. We formulate the problem in Section 2.2. A standard estimation error is defined and autonomy of the resulting error dynamics is investigated in Section 2.3. We introduce the proposed observer in Section 2.4 and investigate the stability of observer error dynamics. Section 2.5 is devoted to the systematic construction of invariant cost functions. A gradient based observer design example in Section 2.6 and a non-gradient example in Section 2.7 followed by brief conclusions in Section 2.8 complete the Chapter.

## 2.2 Problem Formulation

We consider a class of left invariant systems on  $G$  given by

$$\dot{X}(t) = X(t)u(t), \quad X(t_0) = X_0, \quad (2.1)$$

where  $u \in \mathfrak{g}$  is the system input and  $X \in G$  is the state<sup>1</sup>. We assume that  $u: \mathbb{R}^+ \rightarrow \mathfrak{g}$  is continuous and hence a unique solution for (2.1) exists for all  $t \geq t_0$  [72]. In most kinematic mechanical systems,  $u$  models the velocity of physical objects. Hence, it is reasonable to assume that  $u$  is bounded and continuous.

Let  $M_i$ ,  $i = 1, \dots, n$  denote a collection of  $n$  homogeneous spaces of  $G$ , termed *output spaces*. Denote the outputs of system (2.1) by  $y_i \in M_i$ . Suppose each output provides a partial measurement of  $X$  via

$$y_i = h_i(X, \hat{y}_i) \quad (2.2)$$

where  $\hat{y}_i \in M_i$  is the constant (with respect to time) reference output associated with  $y_i$  and  $h_i$  is a right action of  $G$  on  $M_i$ , i.e.  $h_i(I, y_i) = y_i$  and  $h_i(XS, y_i) = h_i(S, h_i(X, y_i))$  for all  $y_i \in M_i$  and all  $X, S \in G^2$ . Although the ideas presented in this chapter are based on the left invariant dynamics with right output actions, they can easily be modified for right invariant systems with left output actions as was done for instance in [94]. To simplify the notation, we define the combined output  $y := (y_1, \dots, y_n)$ , the combined reference output  $\hat{y} := (\hat{y}_1, \dots, \hat{y}_n)$ , and the combined right action  $h(X, \hat{y}) := (h_1(X, \hat{y}_1), \dots, h_n(X, \hat{y}_n))$ . The combined output  $y$  belongs to the orbit of  $G$  acting on the product space  $M_1 \times M_2 \times \dots \times M_n$  containing  $\hat{y}$ , that is  $M := \{y \in M_1 \times M_2 \times \dots \times M_n \mid y = h(X, \hat{y}), X \in G\} \subset M_1 \times M_2 \times \dots \times M_n$ . Note that the combined action  $h$  of  $G$  defined above is transitive on  $M$ . Hence,  $M$  is a homogeneous space of  $G$  while  $M_1 \times M_2 \times \dots \times M_n$  is not necessarily a homogeneous space of  $G$  [91].

We assume that measurements of the system input are corrupted by a constant unknown additive bias. That is

$$u_y = u + b \quad (2.3)$$

<sup>1</sup>In Chapters 3 and 4 where we consider left-invariant, right-invariant, and mixed-invariant systems, we denote the states of those systems by  $X_l$ ,  $X_r$ , and  $X_m$ , respectively. Since in this chapter we only talk about the left-invariant system 2.1, for convenience, we omit the subscript  $l$  and simplify the notation to  $X$ .

<sup>2</sup>In Chapters 3 and 4 where we consider both right and left output actions, we denote the associated actions by  $h_r$  and  $h_l$ , respectively. Since in this chapter we only talk about the right output action 2.2, for convenience, we omit the subscript  $r$  and simplify the notation to  $h$ . The same arrangement applies to the notation for the output element  $y_i$  and its corresponding reference output  $\hat{y}_i$ .

where  $u_y \in \mathfrak{g}$  is the measurement of  $u$  and  $b \in \mathfrak{g}$  is the unknown bias. In practice, bias is slowly time-varying but it is common to assume that  $b$  is constant for observer design and analysis.

We investigate the observer design problem for concurrent estimation of  $X$  and  $b$ . The observer should be *implementable* based on sensor measurements. This is important since the actual state  $X \in G$  and the actual input  $u \in \mathfrak{g}$  are not available for measurement and only the partial measurements  $y_1, \dots, y_n$  and the biased input  $u_y$  are directly measured. We consider the following general class of implementable observers with the same state space dimension as the observed system.

$$\dot{\hat{X}} = \gamma(\hat{X}, y, \dot{y}, \hat{b}, u_y, t) \quad (2.4a)$$

$$\dot{\hat{b}} = \beta(\hat{X}, y, \dot{y}, \hat{b}, u_y, t) \quad (2.4b)$$

where  $\hat{X}$  and  $\hat{b}$  are the estimates of  $X$  and  $b$ , respectively, and  $\gamma: G \times M \times M \times \mathfrak{g} \times \mathfrak{g} \times \mathbb{R} \rightarrow TG$  and  $\beta: G \times M \times M \times \mathfrak{g} \times \mathfrak{g} \times \mathbb{R} \rightarrow \mathfrak{g}$  are parameterized vector fields on  $G$  and  $\mathfrak{g}$ , respectively. Note that  $\hat{X}, y, \dot{y}, \hat{b}, u_y$  and  $t$  are all available for implementation of the observer in practical scenarios. We refer to (2.4a) and (2.4b) as the *group estimator* and the *bias estimator*, respectively.

**Example 2.2.1** (Attitude Estimation on  $SO(3)$ ). *Attitude estimation is a classical problem which is still a popular research topic [34, 37, 55, 65, 69, 80, 101, 105, 108, 116, 120, 121, 124, 133, 136, 144]. The attitude of a rigid body can be identified by a rotation matrix belonging to the Lie group  $R \in SO(3)$  representing the rotation from body-fixed frame  $\{B\}$  to the inertial frame  $\{A\}$ . The Lie algebra of  $SO(3)$  is identified by the set of skew-symmetric  $3 \times 3$  matrices with zero trace denoted by  $\mathfrak{so}(3)$ . The attitude kinematics on  $SO(3)$  is given by*

$$\dot{R}(t) = R(t)\Omega(t) \quad (2.5)$$

where  $\Omega \in \mathfrak{so}(3)$  represents the angular velocity of  $\{B\}$  with respect to  $\{A\}$  expressed in  $\{B\}$ . Assume that partial attitude information is provided by vectorial measurements  $y_i$  in  $\{B\}$ . Such a measurement can be provided by an on-board sensor system such as a 3-axis magnetometer. We recall that in most practical cases, the constant inertial direction  $\hat{y}_i \in \{A\}$  associated with  $y_i$  is known a-priori. The measured direction  $y_i \in \{B\}$  is related to  $R$  and  $\hat{y}_i$

by

$$y_i(t) = R(t)^\top \hat{y}_i. \quad (2.6)$$

The vectors  $y_i$  and  $\hat{y}_i$  are normalized to have unit norms such that the measured output lives in  $S^2$ . The output map (2.6) defines a right action of  $SO(3)$  on the homogeneous output space  $M_i \cong S^2$  by  $h_i(R, \hat{y}_i) := R^\top \hat{y}_i$ .

The angular velocity measured by rate gyros is usually disturbed by additive unknown bias such that

$$\Omega_y(t) = \Omega(t) + b \quad (2.7)$$

where  $\Omega_y$  is the measured angular velocity and  $b$  denotes the unknown constant bias. The attitude estimation problem is to use the measurements  $\Omega_y$  and  $y_i$ ,  $i = 1, \dots, n$  together with a-priori knowledge  $\hat{y}_i$ ,  $i = 1, \dots, n$  in order to estimate the attitude matrix  $R$  and the gyro bias  $b$ .

**Example 2.2.2** (Pose Estimation on  $SE(3)$ ). Estimating the position and attitude of a rigid body has been investigated by a range of authors during the past decades, see, e.g., [25, 65, 116, 135, 138]. The full 6-DOF pose kinematics of a rigid body can be modeled as an invariant system on the special Euclidean group  $SE(3)$  [94, 116, 135, 138]. The Lie group  $SE(3)$  has a representation as a semi-direct product of  $SO(3)$  and  $\mathbb{R}^3$  given by  $SE(3) = \{(R, p) \mid R \in SO(3), p \in \mathbb{R}^3\}$  [36]. Consider the group multiplication on  $SE(3)$  given by  $R_{(S,q)}(R, p) = L_{(R,p)}(S, q) = (RS, p + Rq)$  for any  $(R, p), (S, q) \in SE(3)$ . The identity element of  $SE(3)$  is represented by  $(I_{3 \times 3}, 0_3)$  and the inverse of an element  $(R, p) \in SE(3)$  is given by  $(R, p)^{-1} = (R^\top, -R^\top p)$ . The Lie algebra of  $SE(3)$  is identified with  $\mathfrak{se}(3) = \{(\Omega, V) \mid \Omega \in \mathfrak{so}(3), V \in \mathbb{R}^3\}$ .

Let  $R$  represent the attitude matrix of a rigid body, as was discussed in Example 2.2.1, and suppose that  $p$  represents the position of the rigid body with respect to the inertial frame and expressed in the inertial frame. The left-invariant kinematics of the rigid body on  $SE(3)$  is formulated as

$$(\dot{R}, \dot{p}) = T_L L_{(R,p)}(\Omega, V) = (R\Omega, RV) \quad (2.8)$$

where  $\Omega$  resp.  $V$  represent the angular velocity resp. linear velocity of the rigid body with

respect to the inertial frame expressed in the body-fixed frame. Here, the group element is  $X = (R, p) \in SE(3)$  and the system input is  $u = (\Omega, V) \in \mathfrak{se}(3)$ . Denote the measurement of the system input by  $(\Omega_y, V_y) \in \mathfrak{se}(3)$  and assume that it is corrupted by an unknown constant bias  $(b_\omega, b_v) \in \mathfrak{se}(3)$  such that  $(\Omega_y, V_y) = (\Omega + b_\omega, V + b_v)$ . Suppose that positions of  $n$  points with respect to the body-fixed frame are measured by on-board sensors and denote these measurements by  $y_1, \dots, y_n \in \mathbb{R}^3$ . Denote the positions of these points with respect to the inertial frame by  $\hat{y}_i$ ,  $i = 1, \dots, n \in \mathbb{R}^3$  and assume these positions are known and constant. The output model for such a set of measurements is given by

$$y_i = h_i((R, p), \hat{y}_i) = R^\top \hat{y}_i - R^\top p, \quad i = 1, \dots, n \quad (2.9)$$

where  $h_i$  is a right action of  $SE(3)$  on the homogeneous output space  $M_i := \mathbb{R}^3$ . A practical example of measurements modeled by (2.9) is vision based landmark readings where the landmarks are fixed in the inertial frame, leading to constant  $\hat{y}_i$ ,  $i = 1, \dots, n$  [135]. The pose estimation problem is to estimate  $R$  and  $p$  together with the input biases  $b_\omega$  and  $b_v$ .  $\square$

### 2.3 Error Definition and Autonomy of Error Dynamics

We consider the following right-invariant group error,

$$E = \hat{X}X^{-1} \in G, \quad (2.10)$$

as was proposed in [94, 102]. The above error resembles the usual error  $\hat{x} - x$  used in classical observer theory when  $\hat{x}, x$  belong to a vector space. We also consider the following bias estimation error

$$\tilde{b} = \hat{b} - b \in \mathfrak{g}. \quad (2.11)$$

We are interested to see when an observer of the general form (2.4) produces *autonomous* error dynamics since that would enable straight-forward stability analysis. When the measurement of system input is bias free, implementable observers of the form (2.4a) have been proposed that produce autonomous group error dynamics  $\dot{E}$  [94]. In this section, we show that when the measurement of system input is corrupted by bias, any implementable observer of the form (2.4) produces

non-autonomous error dynamics for general Lie groups, and it can only produce autonomous error dynamics for Abelian Lie groups. To prove this result, we note that the observer (2.4) can be rewritten into the form

$$\dot{\hat{X}} = \hat{X}[u_y - \hat{b}] - \alpha_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t), \quad (2.12a)$$

$$\dot{\hat{b}} = \beta_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t), \quad (2.12b)$$

where  $\alpha_{\hat{y}}: G \times M \times \mathfrak{g} \times \mathfrak{g} \times \mathbb{R} \rightarrow TG$  and  $\beta_{\hat{y}}: G \times M \times \mathfrak{g} \times \mathfrak{g} \times \mathbb{R} \rightarrow \mathfrak{g}$  are parameterized vector fields on  $G$  and  $\mathfrak{g}$ , respectively, and  $\hat{y}$  is now interpreted as a parameter for  $\alpha_{\hat{y}}$  and  $\beta_{\hat{y}}$ .

**Theorem 2.3.1.** *Consider the observer (2.12) for the system (2.1)-(2.3). The error dynamics  $(\dot{E}, \dot{\tilde{b}})$  is autonomous if and only if all of the following conditions hold;*

- (a)  $\alpha_{\hat{y}}$  and  $\beta_{\hat{y}}$  do not depend on  $\hat{b}$ ,  $u_y$  and  $t$ .
- (b) The vector field  $\alpha_{\hat{y}}$  is right equivariant in the sense that  $T_{\hat{X}}R_Z\alpha_{\hat{y}}(\hat{X}, y) = \alpha_{\hat{y}}(\hat{X}Z, h(Z, y))$  for all  $\hat{X}, Z \in G$  and all  $y \in M$ .
- (c) The vector field  $\beta_{\hat{y}}$  is right invariant in the sense that  $\beta_{\hat{y}}(\hat{X}, y) = \beta_{\hat{y}}(\hat{X}Z, h(Z, y))$  for all  $\hat{X}, Z \in G$  and all  $y \in M$ .
- (d) For all  $Z \in G$  the adjoint map  $Ad_Z: \mathfrak{g} \rightarrow \mathfrak{g}$  is the identity map.  $\square$

*Proof:* In view of (2.1) and (2.12), differentiating  $E = \hat{X}X^{-1}$  and  $\tilde{b} = \hat{b} - b$  with respect to time yields

$$\dot{E} = -T_{\hat{X}}R_{X^{-1}}\alpha_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t) - T_I R_E Ad_{\hat{X}}\tilde{b} \quad (2.13a)$$

$$\dot{\tilde{b}} = \beta_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t). \quad (2.13b)$$

If the conditions (a) to (d) of the Theorem hold, the error dynamics will be simplified to

$$\dot{E} = -\alpha_{\hat{y}}(\hat{X}X^{-1}, h(X^{-1}, y)) - T_I R_E \tilde{b} = -\alpha_{\hat{y}}(E, \hat{y}) - T_I R_E \tilde{b}, \quad (2.14a)$$

$$\dot{\tilde{b}} = \beta_{\hat{y}}(\hat{X}X^{-1}, h(X^{-1}, y)) = \beta_{\hat{y}}(E, \hat{y}), \quad (2.14b)$$

which are autonomous.

Conversely, assume that the error dynamics (2.13) are autonomous. Then there exist functions  $F_{\hat{y}}: G \times \mathfrak{g} \rightarrow TG$  and  $H_{\hat{y}}: G \times \mathfrak{g} \rightarrow \mathfrak{g}$  such that for all  $X, \hat{X} \in G, y \in M, \hat{b}, u_y \in \mathfrak{g}$ ,

$$\dot{E} = -T_{\hat{X}}R_{X^{-1}}\alpha_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t) - T_I R_E \text{Ad}_{\hat{X}} \tilde{b} = F_{\hat{y}}(E, \tilde{b}) \quad (2.15a)$$

$$\dot{\tilde{b}} = \beta_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t) = H_{\hat{y}}(E, \tilde{b}). \quad (2.15b)$$

It immediately follows that  $\alpha_{\hat{y}}$  and  $\beta_{\hat{y}}$  are independent of  $u_y$  and  $t$ . Moreover, since the error  $E = \hat{X}X^{-1}$  is invariant with respect to the transformation  $(\hat{X}, X) \mapsto (\hat{X}Z, XZ)$  for all  $Z \in G$  and the error  $\tilde{b} = \hat{b} - b$  is invariant with respect to the transformation  $(\hat{b}, b) \mapsto (\hat{b} + d, b + d)$  for all  $d \in \mathfrak{g}$ , we have

$$-T_{\hat{X}}R_{X^{-1}}\alpha_{\hat{y}}(\hat{X}, y, \hat{b}) - T_I R_E \text{Ad}_{\hat{X}} \tilde{b} = F_{\hat{y}}(E, \tilde{b}) \quad (2.16a)$$

$$\begin{aligned} &= -T_{\hat{X}Z}R_{(XZ)^{-1}}\alpha_{\hat{y}}(\hat{X}Z, h(Z, y), \hat{b} + d) - T_I R_E \text{Ad}_{\hat{X}Z} \tilde{b}, \\ \beta_{\hat{y}}(\hat{X}, y, \hat{b}) &= H_{\hat{y}}(E, \tilde{b}) = \beta_{\hat{y}}(\hat{X}Z, h(Z, y), \hat{b} + d). \end{aligned} \quad (2.16b)$$

From (2.16b) it follows that  $\beta_{\hat{y}}$  is independent of  $\hat{b}$  since the right hand side of (2.16b) depends on  $d$  while the left hand side is independent of this variable. This establishes condition (a) for  $\beta_{\hat{y}}$ . It also follows that  $\beta_{\hat{y}}$  satisfies the invariance condition  $\beta_{\hat{y}}(\hat{X}, y) = \beta_{\hat{y}}(\hat{X}Z, h(Z, y))$  (condition (c) of the Theorem). We can rearrange (2.16a) to obtain

$$-T_{\hat{X}}R_{X^{-1}}\alpha_{\hat{y}}(\hat{X}, y, \hat{b}) - T_I R_E \text{Ad}_{\hat{X}} \tilde{b} + T_I R_E \text{Ad}_{\hat{X}Z} \tilde{b} = -T_{\hat{X}Z}R_{(XZ)^{-1}}\alpha_{\hat{y}}(\hat{X}Z, h(Z, y), \hat{b} + d). \quad (2.17)$$

The right hand side of (2.17) is a function of  $d$  while the left hand side is not. This implies that  $\alpha_{\hat{y}}$  is independent of  $\hat{b}$  (establishing condition (a) for  $\alpha_{\hat{y}}$ ). We can then rearrange (2.17) again to obtain

$$-T_{\hat{X}}R_{X^{-1}}\alpha_{\hat{y}}(\hat{X}, y) + T_{\hat{X}Z}R_{(XZ)^{-1}}\alpha_{\hat{y}}(\hat{X}Z, h(Z, y)) = T_I R_E \text{Ad}_{\hat{X}} \tilde{b} - T_I R_E \text{Ad}_{\hat{X}Z} \tilde{b}. \quad (2.18)$$

The right hand side of (2.18) is a linear function acting on  $\tilde{b} \in \mathfrak{g}$  while the left hand side is completely independent of the variable  $\tilde{b}$ . Since  $\tilde{b}$  is arbitrary, this implies that both sides of (2.18) are zero. In particular,  $T_{\hat{X}}R_{X^{-1}}\alpha_{\hat{y}}(\hat{X}, y) = T_{\hat{X}Z}R_{(XZ)^{-1}}\alpha_{\hat{y}}(\hat{X}Z, h(Z, y))$

and  $T_I R_E \text{Ad}_{\hat{X}Z} \tilde{b} = T_I R_E \text{Ad}_{\hat{X}} \tilde{b}$  for all  $\tilde{b} \in \mathfrak{g}$  and all  $E, \hat{X}, Z \in G$ . These equations imply  $T_{\hat{X}} R_Z \alpha_{\hat{y}}(\hat{X}, y) = \alpha_{\hat{y}}(\hat{X}Z, h(Z, y))$  and  $\text{Ad}_Z \tilde{b} = \tilde{b}$  to obtain conditions (b) and (d) imposed in the theorem, respectively. This completes the proof.  $\blacksquare$

**Remark 2.3.2.** *If  $G$  is a real, finite-dimensional, connected Lie group then condition (d) of Theorem 2.3.1 implies that  $G$  is Abelian [90, Proposition 1.91]. By the structure theorem for connected Abelian Lie groups [36, Proposition III.6.4.11], this means that  $G$  is isomorphic to a product  $\mathbb{R}^n \times (S^1)^m$  where  $\mathbb{R}^n$  is additive and  $(S^1)^m$  denotes the  $m$ -dimensional torus. This class of Lie groups is far more specific than the Lie groups that are encountered in many practical applications. For robotics applications, the Lie groups typically considered are  $SO(3)$  and  $SE(3)$  both of which are non-Abelian. Theorem 2.3.1 in particular implies that all implementable geometric bias estimators on  $SO(3)$  and  $SE(3)$  proposed in the literature produce non-autonomous standard error dynamics (see [101] and [135]).  $\square$*

## 2.4 Observer Design and Analysis

We propose the following implementable group estimator,

$$\dot{\hat{X}} = \hat{X}[u_y - \hat{b}] - K_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t)[D_1 \phi_{\hat{y}}(\hat{X}, y)], \quad (2.19)$$

with  $\hat{X}(t_0) = \hat{X}_0$  where  $\phi_{\hat{y}}: G \times M \rightarrow \mathbb{R}^+$  is a  $C^2$  cost function,  $D_1 \phi_{\hat{y}}(\hat{X}, y) \in T_{\hat{X}}^* G$  denotes the differential of  $\phi_{\hat{y}}$  with respect to its first argument evaluated at the point  $(\hat{X}, y)$  and  $K_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t)$  is a linear gain mapping from  $T_{\hat{X}}^* G$  to  $T_{\hat{X}} G$ . Note that  $\hat{y}$  is considered to be a parameter for  $K_{\hat{y}}$  and  $\phi_{\hat{y}}$ . The above group estimator matches the structure of (2.12a) where the innovation  $\alpha_{\hat{y}}$  is generated by applying the gain mapping  $K_{\hat{y}}$  to the differential  $D_1 \phi_{\hat{y}}$ . By Theorem 2.3.1, we already know that the above estimator cannot produce autonomous error dynamics for a general Lie group. Hence, there is no reason to omit the arguments  $\hat{b}, u_y$  and  $t$  of the gain mapping. If the gain mapping  $K_{\hat{y}}$  is symmetric positive definite and independent of  $\hat{b}, u_y$  and  $t$ , the above group estimator simplifies to the gradient-like observers proposed in [94] for the bias free case, and in the Author's previous work [85] for the case including bias.

We consider the following bias estimator,

$$\dot{\hat{b}} = \Gamma \circ T_L L_{\hat{X}}^* [D_1 \phi_{\hat{y}}(\hat{X}, y)], \quad \hat{b}(t_0) = \hat{b}_0, \quad (2.20)$$

where  $T_L L_{\hat{X}}^*: T_{\hat{X}}^* G \rightarrow \mathfrak{g}^*$  is the dual of the map  $T_L L_{\hat{X}}$  (see Section 1.5) and  $\Gamma: \mathfrak{g}^* \rightarrow \mathfrak{g}$  is a constant gain mapping.

We will require the following assumptions for statement of results.

- (A1) The Lie group  $G$  has a faithful representation as a finite-dimensional matrix Lie group. That is, there exist a positive integer  $m$  and an injective Lie group homomorphism  $\Phi: G \rightarrow GL(m)$  into the group  $GL(m)$  of invertible  $m \times m$  matrices. Note that  $\Phi(G)$  is a matrix subgroup of  $GL(m)$ .
- (A2) [boundedness conditions]  $\Phi(X)$ ,  $\Phi(X)^{-1}$ ,  $u$  and  $K(\hat{X}, y, \hat{b}, u_y, t)$  are bounded along the system trajectories.
- (A3) [differentiability conditions]  $\dot{u}(t)$ , the first differentials of  $h_i(X, \dot{y}_i)$  and  $K(\hat{X}, y, \hat{b}, u_y, t)$ , as well as the first and the second differential<sup>3</sup> of  $\phi(\hat{X}, y)$  with respect to all of their arguments are bounded along the system trajectories.

**Theorem 2.4.1.** *Consider the observer (2.19)-(2.20) for the system (2.1)-(2.3). Suppose that assumptions (A1), (A2) and (A3) hold. Assume moreover that the gain mappings  $K$  and  $\Gamma$ , and the cost function  $\phi$  satisfy the following properties;*

- (a) *The gain mapping  $K_{\hat{y}}: T_{\hat{X}}^* G \rightarrow T_{\hat{X}} G$  is uniformly positive definite (not necessarily symmetric). That is, there exist positive constants  $\underline{k}$  and  $\bar{k}$  and a continuous vector norm  $|\cdot|$  on  $T_{\hat{X}}^* G$  such that for all  $v^* \in T_{\hat{X}}^* G$  we have  $\underline{k}|v^*|^2 \leq v^* [K_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t)[v^*]] \leq \bar{k}|v^*|^2$ .*
- (b) *The gain mapping  $\Gamma: \mathfrak{g}^* \rightarrow \mathfrak{g}$  is symmetric positive definite.*
- (c) *The cost  $\phi_{\hat{y}}$  is right invariant, that is  $\phi_{\hat{y}}(\hat{X}Z, h(Z, y)) = \phi_{\hat{y}}(\hat{X}, y)$  for all  $\hat{X}, Z \in G$  and all  $y \in M$ .*
- (d) *The cost  $\phi_{\hat{y}}(\cdot, \dot{y}): G \rightarrow \mathbb{R}^+$ ,  $E \mapsto \phi_{\hat{y}}(E, \dot{y})$  is locally positive definite around  $E = I$  and it has an isolated critical point at  $E = I$ .*

<sup>3</sup>Second differential of  $\phi$  is either in the sense of embedding the Lie group into  $\mathbb{R}^{m \times m}$  or in the sense of employing a Riemannian metric.

Then the error dynamics  $(\dot{E}, \dot{\tilde{b}})$  is uniformly locally asymptotically stable at  $(I, 0)$ .  $\square$

*Proof:* The following result is used in the development later in this proof.

**Lemma 2.4.2.** *Let  $\phi_{\hat{y}}: G \times M \rightarrow \mathbb{R}^+$  be a right-invariant cost function in the sense defined in part (c) of Theorem 2.4.1. Then we have*

$$D_1\phi_{\hat{y}}(\hat{X}, y) = T_{\hat{X}}R_{X^{-1}}^*[D_1\phi_{\hat{y}}(E, \hat{y})] \quad (2.21)$$

$$D_1\phi_{\hat{y}}(E, \hat{y}) = D_1\phi_{\hat{y}}(\hat{X}, y) \circ T_ER_X \quad (2.22)$$

*Proof of Lemma 2.4.2:* The right-invariance property of  $\phi_{\hat{y}}$  implies  $\phi_{\hat{y}}(\hat{X}, y) = \phi_{\hat{y}} \circ R_{X^{-1}}(\hat{X}, y)$ . Differentiating both sides in an arbitrary direction  $v \in T_{\hat{X}}G$  and using the chain rule we obtain  $D_1\phi_{\hat{y}}(\hat{X}, y)[v] = D_1\phi_{\hat{y}}(E, \hat{y}) \circ T_{\hat{X}}R_{X^{-1}}[v]$ . Since  $v$  is arbitrary and by using the duality we have  $D_1\phi_{\hat{y}}(\hat{X}, y) = T_{\hat{X}}R_{X^{-1}}^*[D_1\phi_{\hat{y}}(E, \hat{y})]$  which proves (2.21). Applying  $(T_{\hat{X}}R_{X^{-1}})^{-1} = T_ER_X$  from the right to both sides of  $D_1\phi_{\hat{y}}(\hat{X}, y) = D_1\phi_{\hat{y}}(E, \hat{y}) \circ T_{\hat{X}}R_{X^{-1}}$  yields (2.22). This completes the proof of Lemma 2.4.2.

For simplicity, we denote  $K_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t)$  by  $K_{\hat{y}}(\cdot)$ . Considering (2.1), (2.3), and (2.19), the group error dynamics are given by

$$\begin{aligned} \dot{E} &= \hat{X}X^{-1} + \hat{X}X^{-1} = T_{\hat{X}}R_{X^{-1}} \circ T_IL_{\hat{X}}[u] - T_{\hat{X}}R_{X^{-1}} \circ T_IL_{\hat{X}}[\tilde{b}] \\ &\quad - T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\cdot)[D_1\phi_{\hat{y}}(\hat{X}, y)] - T_{\hat{X}}R_{X^{-1}} \circ T_IL_{\hat{X}}[u] \\ &= -T_{\hat{X}}R_{X^{-1}} \circ T_IL_{\hat{X}}[\tilde{b}] - T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\cdot) \circ T_{\hat{X}}R_{X^{-1}}^*[D_1\phi_{\hat{y}}(E, \hat{y})], \end{aligned} \quad (2.23)$$

where  $E$  is as in (2.10) and (2.21) is used in the last line of (2.23). Now, consider the candidate Lyapunov function,

$$\mathcal{L}(E, \tilde{b}) = \phi_{\hat{y}}(E, \hat{y}) + \frac{1}{2}\Gamma^{-1}[\tilde{b}][\tilde{b}]. \quad (2.24)$$

The Lyapunov candidate is at least locally positive definite due to conditions (b) and (d). The time derivative of  $\mathcal{L}$  is given by

$$\dot{\mathcal{L}}(E, \tilde{b}) = D_1\phi_{\hat{y}}(E, \hat{y})[\dot{E}] + \Gamma^{-1}[\dot{\tilde{b}}][\tilde{b}]. \quad (2.25)$$

Recalling that  $\dot{\tilde{b}} = \hat{\dot{b}}$  and substituting  $\dot{E}$  form (2.23) in (2.25), we obtain

$$\begin{aligned} \dot{\mathcal{L}}(E, \tilde{b}) = & -D_1\phi_{\hat{y}}(E, \hat{y})[T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\cdot) \circ T_{\hat{X}}R_{X^{-1}}^* [D_1\phi_{\hat{y}}(E, \hat{y})]] \\ & - D_1\phi_{\hat{y}}(E, \hat{y})[T_{\hat{X}}R_{X^{-1}} \circ T_I L_{\hat{X}}[\tilde{b}]] + \Gamma^{-1}[\hat{\dot{b}}][\tilde{b}]. \end{aligned} \quad (2.26)$$

Using (2.22), we conclude

$$\begin{aligned} \dot{\mathcal{L}}(E, \tilde{b}) = & -D_1\phi_{\hat{y}}(E, \hat{y})[T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\cdot) \circ T_{\hat{X}}R_{X^{-1}}^* [D_1\phi_{\hat{y}}(E, \hat{y})]] \\ & - D_1\phi_{\hat{y}}(\hat{X}, y) \circ T_E R_X \circ T_{\hat{X}}R_{X^{-1}} \circ T_I L_{\hat{X}}[\tilde{b}] + \Gamma^{-1}[\hat{\dot{b}}][\tilde{b}] \\ = & -D_1\phi_{\hat{y}}(E, \hat{y})[T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\cdot) \circ T_{\hat{X}}R_{X^{-1}}^* [D_1\phi_{\hat{y}}(E, \hat{y})]] \\ & - D_1\phi_{\hat{y}}(\hat{X}, y) \circ T_I L_{\hat{X}}[\tilde{b}] + \Gamma^{-1}[\hat{\dot{b}}][\tilde{b}]. \end{aligned} \quad (2.27)$$

Now, replacing  $\hat{\dot{b}}$  with (2.20) we obtain

$$\begin{aligned} \dot{\mathcal{L}}(E, \tilde{b}) = & -D_1\phi_{\hat{y}}(E, \hat{y})[T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\cdot) \circ T_{\hat{X}}R_{X^{-1}}^* [D_1\phi_{\hat{y}}(E, \hat{y})]] \\ & - D_1\phi_{\hat{y}}(\hat{X}, y) \circ T_I L_{\hat{X}}[\tilde{b}] + \Gamma^{-1} \circ \Gamma \circ T_I L_{\hat{X}}^* \circ D_1\phi_{\hat{y}}(\hat{X}, y)[\tilde{b}]. \end{aligned} \quad (2.28)$$

Duality implies  $D_1\phi_{\hat{y}}(\hat{X}, y) \circ T_I L_{\hat{X}} = T_I L_{\hat{X}}^* \circ D_1\phi_{\hat{y}}(\hat{X}, y)$  and (2.28) simplifies to

$$\dot{\mathcal{L}}(E, \tilde{b}) = -D_1\phi_{\hat{y}}(E, \hat{y})[T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\cdot) \circ T_{\hat{X}}R_{X^{-1}}^* [D_1\phi_{\hat{y}}(E, \hat{y})]]. \quad (2.29)$$

Since  $K_{\hat{y}}(\cdot)$  is assumed to be positive definite and the map  $T_{\hat{X}}R_{X^{-1}}$  is full rank, the map  $T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\cdot) \circ T_{\hat{X}}R_{X^{-1}}^*$  is positive definite. This implies that  $\dot{\mathcal{L}}(E, \tilde{b}) \leq 0$  and hence the Lyapunov function is non-increasing along the system trajectories. We adopt the proof of [78, Theorem 4.8] to prove uniformly local stability of error dynamics. Recalling assumption (A1), distance to the identity element of  $G$  is denoted by  $d(\cdot)$  and is induced by Frobenius norm on  $\Phi(G) \subset \mathbb{R}^{m \times m}$  via  $d(E) := \|\text{Id} - \Phi(E)\|_F$  where  $\text{Id}$  is the identity matrix. Define the compound error  $\tilde{x} := (E, \tilde{b}) \in G \times \mathfrak{g}$  and obtain the distance of  $\tilde{x}$  to  $(I, 0)$  by  $l(\tilde{x})^2 := d(E)^2 + \|\tilde{b}\|_{\mathfrak{g}}^2$  where  $\|\cdot\|_{\mathfrak{g}}$  denotes a norm on  $\mathfrak{g}$ . Using assumption (d), there exist a ball  $B_r := \{E \in G : d(E) \leq r\}$  such that  $\phi_{\hat{y}}(\cdot, \hat{y})$  is positive definite on  $B_r$ . Consequently  $\mathcal{L}(\tilde{x})$  is positive definite on  $\bar{B}_r := \{\tilde{x} \in G \times \mathfrak{g} : l(\tilde{x}) \leq r\}$ . Choose  $c < \min_{l(\tilde{x})=r} \mathcal{L}(\tilde{x})$  and define  $\Omega_c := \{\tilde{x} \in \bar{B}_r \mid \mathcal{L}(\tilde{x}) \leq c\}$ . Since  $\dot{\mathcal{L}}(t) \leq 0$ , any solution starting in  $\Omega_c$  at  $t_0$  remains in  $\Omega_c$  for all  $t \geq t_0$ . On the

other hand, since  $\mathcal{L}(\tilde{x})$  is positive definite on  $\Omega_c \subset \bar{B}_r$ , there exist class  $\mathcal{K}$  functions  $\eta_1$  and  $\eta_2$  such that  $\eta_1(l(\tilde{x})) \leq \mathcal{L}(\tilde{x}) \leq \eta_2(l(\tilde{x}))$  for all  $\tilde{x} \in \Omega_c$  [78, Lemma 4.3]. Consequently, we have  $l(\tilde{x}(t)) \leq \eta_1^{-1}(\mathcal{L}(\tilde{x}(t))) \leq \eta_1^{-1}(\mathcal{L}(\tilde{x}(t_0))) \leq \eta_1^{-1}(\eta_2(l(\tilde{x}(t_0))))$  which implies  $l(\tilde{x}(t)) \leq \eta_1^{-1} \circ \eta_2(l(\tilde{x}(t_0)))$ . Since  $\eta_1^{-1} \circ \eta_2$  is a class  $\mathcal{K}$  function (by [78, Lemma 4.2]), the equilibrium point  $\tilde{x} = (I, 0)$  is uniformly stable for all initial conditions starting in  $\Omega_c$  [78, Lemma 4.5]. Moreover, the error  $E$  is bounded by  $d(E(t)) \leq l(\tilde{x}(t)) \leq \eta_1^{-1}(\mathcal{L}(\tilde{x}(t_0))) \leq \eta_1^{-1}(c)$  for such initial conditions.

Boundedness of  $\tilde{x}(t)$  implies that  $E(t)$  and  $\tilde{b}(t)$  are bounded with respect to  $d(\cdot)$  and  $\|\cdot\|_{\mathfrak{g}}$ , respectively. Differentiating (2.29) with respect to time and considering the boundedness of  $(E(t), \tilde{b}(t))$  together with assumptions (A2) and (A3), one can conclude that  $\dot{\mathcal{L}}(t)$  is bounded and hence  $\dot{\mathcal{L}}(t)$  is uniformly continuous. By invoking Barbalat's lemma we conclude that  $\dot{\mathcal{L}}(t) \rightarrow 0$ . This together with condition (a) implies that  $D_1\phi_{\hat{y}}(E(t), \hat{y}) \rightarrow 0$ . Since  $\phi_{\hat{y}}(E, I)$  has an isolated critical point at  $E = I$ , there exist a ball  $B_{\bar{c}} \subset G$  such that  $E = I$  is the only point in  $B_{\bar{c}}$  where  $D_1\phi_{\hat{y}}(\cdot, \hat{y})$  is zero. We proved before that  $E(t) \in B_{\eta_1^{-1}(c)}$  for all initial conditions starting in  $\Omega_c$ . Choosing  $c < \min(\eta_1(\bar{c}), \min_{l(\tilde{x})=r} \mathcal{L}(\tilde{x}))$  ensures that  $E = I$  is the only critical point in  $B_{\eta_1^{-1}(c)}$ . This implies that  $E(t) \rightarrow I$  for all initial conditions starting in  $\Omega_c$ . Using (2.1), (2.12a), and (2.23), recalling assumptions (A2) and (A3), and using a local coordinate representation, one can verify that  $\ddot{E}(t)$  is bounded and hence  $\dot{E}(t)$  is uniformly continuous. Thus, by invoking Barbalat's lemma we have  $\dot{E}(t) \rightarrow 0$ . Considering  $E(t), \dot{E}(t) \rightarrow 0$  together with error dynamics (2.23) implies that  $\tilde{b}(t) \rightarrow 0$  for all initial errors starting in  $\Omega_c$ . This completes the proof of uniformly local asymptotic stability of the error dynamics. ■

The following theorem proposes additional conditions to guarantee local *exponential* stability of the error dynamics.

**Theorem 2.4.3.** *Consider the observer (2.19)-(2.20) for the system (2.1)-(2.3). Suppose that assumptions (A1) and (A2) and conditions (a), (b) and (c) of Theorem 2.4.1 hold. Assume moreover that;*

(d)  $D_1\phi_{\hat{y}}(I, \hat{y}) = 0$  and  $\text{Hess}_1\phi_{\hat{y}}(I, \hat{y}) : \mathfrak{g} \rightarrow \mathfrak{g}^*$  is symmetric positive definite.

(e) The condition number of  $\Phi(X(t))$  is bounded for all  $t \geq t_0$  (uniformly in  $t_0$ ).

Then, the error dynamics  $(E(t), \tilde{b}(t))$  is uniformly locally exponentially stable at  $(I, 0)$ . □

*Proof:* The group error dynamics (2.23) can be rewritten as

$$\dot{E} = -T_{\hat{X}}R_{X^{-1}} \circ K_{\hat{y}}(\hat{X}, y, \hat{b}, u_y, t) \circ T_{\hat{X}}R_{X^{-1}}^* [D_1\phi_{\hat{y}}(E, \hat{y})] - T_I L_E \text{Ad}_X[\tilde{b}]. \quad (2.30)$$

Using (2.21) and (2.20), the bias error dynamics is obtain as

$$\dot{\tilde{b}} = \Gamma \circ T_I L_X^* \circ T_{\hat{X}}R_{X^{-1}}^* [D_1\phi_{\hat{y}}(E, \hat{y})] = \Gamma \circ \text{Ad}_X^* \circ T_I L_E^* [D_1\phi_{\hat{y}}(E, \hat{y})]. \quad (2.31)$$

Defining  $\epsilon, \delta \in \mathfrak{g}$  as the first order approximation of  $E$  and  $\tilde{b}$  respectively, linearizing the error dynamics (2.30)-(2.31) around  $(I, 0)$ , noting  $D_1\phi_{\hat{y}}(I, \hat{y}) = 0$ , and neglecting all terms of quadratic or higher order in  $(\epsilon, \delta)$  yields (see e.g. [122])

$$\dot{\epsilon} = -T_X R_{X^{-1}} \circ K_{\hat{y}}(X, y, b, u_y, t) \circ T_X R_{X^{-1}}^* \circ \text{Hess}_1\phi_{\hat{y}}(I, \hat{y})[\epsilon] - \text{Ad}_X[\delta], \quad (2.32)$$

$$\dot{\delta} = \Gamma \circ \text{Ad}_X^* \circ \text{Hess}_1\phi_{\hat{y}}(I, \hat{y})[\epsilon], \quad (2.33)$$

where  $\text{Hess}_1\phi_{\hat{y}}(I, \hat{y}): \mathfrak{g} \rightarrow \mathfrak{g}^*$  denotes the Hessian operator which is intrinsically defined at the critical point of the cost [15]. In order to investigate the stability of the linearized error dynamics, we consider a basis for the involved tangent spaces and rewrite (2.32)-(2.33) in matrix format. To this end, consider a basis  $\{e_j\}$  for  $\mathfrak{g}$  and its corresponding dual basis for  $\mathfrak{g}^*$ . Obtain the basis  $\{e_j X\}$  for the vector space  $T_X G$  by right translating  $\{e_j\}$  and consider its corresponding dual basis  $\{(e_j X)^*\}$  for  $T_X^* G$ . Denote by  $[[\epsilon]], [[\delta]]$  the representation of the vectors  $\epsilon, \delta$  with respect to the basis  $\{e_j\}$ . Denote the matrix representation of the maps  $K_{\hat{y}}(X, y, b, u_y, t): T_X^* G \rightarrow T_X G$ ,  $\Gamma: \mathfrak{g}^* \rightarrow \mathfrak{g}$ ,  $\text{Hess}_1\phi_{\hat{y}}(I, \hat{y}): \mathfrak{g} \rightarrow \mathfrak{g}^*$  and  $\text{Ad}_X: \mathfrak{g} \rightarrow \mathfrak{g}$  with respect to the above bases for their corresponding domain and co-domain by  $[[K]], [[\Gamma]], [[H]]$  and  $[[\text{Ad}_X]]$  respectively. Note that the matrix representation of  $T_X R_{X^{-1}}: T_X G \rightarrow \mathfrak{g}$  with respect to the corresponding basis for its domain and co-domain is the identity matrix. Hence, the matrix representation of the error dynamics (2.32)-(2.33) is obtained as

$$\begin{bmatrix} [[\dot{\epsilon}]] \\ [[\dot{\delta}]] \end{bmatrix} = \begin{bmatrix} -[[K]][[H]] & -[[\text{Ad}_X]] \\ [[\Gamma]][[\text{Ad}_X]]^\top [[H]] & 0 \end{bmatrix} \begin{bmatrix} [[\epsilon]] \\ [[\delta]] \end{bmatrix}. \quad (2.34)$$

Since  $\Gamma$  is symmetric positive definite, there exists a full rank square matrix  $L$  such that  $[[\Gamma]] = L^\top L$ . Consider the change of coordinates  $\bar{\epsilon} := L[[\epsilon]]$  and  $\bar{\delta} := L^{-\top}[[\delta]]$ .

Using (2.34), the dynamics of the new error coordinates are obtained as

$$\begin{bmatrix} \dot{\bar{\epsilon}} \\ \dot{\bar{\delta}} \end{bmatrix} = \begin{bmatrix} -L\llbracket K \rrbracket \llbracket H \rrbracket L^{-1} & -L\llbracket \text{Ad}_X \rrbracket L^\top \\ L\llbracket \text{Ad}_X \rrbracket^\top \llbracket H \rrbracket L^{-1} & 0 \end{bmatrix} \begin{bmatrix} \bar{\epsilon} \\ \bar{\delta} \end{bmatrix}. \quad (2.35)$$

Consider initial conditions  $X(t_0)$  for system (2.1) and  $(\hat{X}(t_0), \hat{b}(t_0))$  for the estimator (2.19)-(2.20), respectively. Introducing the parameter  $\lambda = (t_0, X(t_0), \hat{X}(t_0), \hat{b}(t_0)) \in \mathcal{D}$  where  $\mathcal{D} := \mathbb{R} \times G \times G \times \mathfrak{g}$ , the trajectories of  $X, \hat{X}, \hat{b}$  and  $y$  can be viewed as functions of  $t$  and  $\lambda$ . Define  $A(t, \lambda) := -L\llbracket K \rrbracket \llbracket H \rrbracket L^{-1}$ ,  $B(t, \lambda) := -L\llbracket \text{Ad}_X \rrbracket^\top L^\top$ , and  $P := L^{-\top} \llbracket H \rrbracket L^{-1}$ . The system (2.35) belongs to the following standard class of parameterized linear time-varying systems discussed extensively in the literature [99, 110, 111].

$$\begin{bmatrix} \dot{\bar{\epsilon}} \\ \dot{\bar{\delta}} \end{bmatrix} = \begin{bmatrix} A(t, \lambda) & B(t, \lambda)^\top \\ -B(t, \lambda)P & 0 \end{bmatrix} \begin{bmatrix} \bar{\epsilon} \\ \bar{\delta} \end{bmatrix}. \quad (2.36)$$

We can now verify the conditions of [99, Theorem 1] to prove the stability of system (2.35). Both  $B(t, \lambda)$  and its time derivative are bounded due to Assumption (A2). Since  $\text{Hess}_1 \phi_{\hat{y}}(I, \hat{y})$  is symmetric positive definite and  $L$  has full rank, the matrix  $P$  is symmetric positive definite and it is bounded by  $\underline{\sigma}(H)\underline{\sigma}(L)^{-2}I \leq P \leq \bar{\sigma}(H)\bar{\sigma}(L)^{-2}I$  where  $\underline{\sigma}(\cdot)$  and  $\bar{\sigma}(\cdot)$  denote the smallest and largest singular value of a matrix respectively. Define  $-Q := \dot{P} + A(t, \lambda)^\top P + PA(t, \lambda) = -(\llbracket H \rrbracket L^{-1})^\top (\llbracket K \rrbracket^\top + \llbracket K \rrbracket) (\llbracket H \rrbracket L^{-1})$ . Using condition (a) of Theorem 2.4.1 and recalling assumption (A2), there exist positive constants  $k_1$  and  $k_2$  such that  $2k_1 \text{Id} \leq \llbracket K \rrbracket^\top + \llbracket K \rrbracket \leq 2k_2 \text{Id}$  where  $\text{Id}$  is the identity matrix. This ensures that  $Q$  is uniformly symmetric positive definite and we have  $2k_1 \underline{\sigma}(H)^2 \underline{\sigma}(L)^{-2} \text{Id} \leq Q \leq 2\bar{\sigma}(H)^2 k_2 \bar{\sigma}(L)^{-2} \text{Id}$ . It only remains to investigate whether  $B(t, \lambda)$  is  $\lambda$ -uniformly persistently exciting [99, equation (10)]. Using condition (e), there exists a positive constant  $c_0$  such that  $\text{cond}(\Phi(X)(t)) \leq c_0$ . Invoking Lemma 6.2.1 in the Appendix, we have  $\underline{\sigma}(\text{Ad}_X) \geq c_0^{-2}$ . Hence,  $\underline{\sigma}(B(t, \lambda)B(t, \lambda)^\top) = \underline{\sigma}(L\llbracket \text{Ad}_X \rrbracket^\top \llbracket \Gamma \rrbracket \llbracket \text{Ad}_X \rrbracket L^\top) \geq \underline{\sigma}(L)^2 \underline{\sigma}(\llbracket \Gamma \rrbracket) c_0^{-2} := \bar{c}_0$ . Integrating both sides yields  $\int_\tau^{t+T} B(\tau, \lambda)B(\tau, \lambda)^\top d\tau \geq \bar{c}_0 T \text{Id}$  which completes the requirements of [99, Theorem 1]. Hence, the equilibrium  $(0, 0)$  of the (2.35) is uniformly exponentially stable. This implies that the equilibrium  $(0, 0)$  of the linearized system (2.35) is uniformly exponentially stable and consequently the equilibrium  $(I, 0)$  of the nonlinear error dy-

namics (2.30)-(2.31) is uniformly locally exponentially stable [78, Theorem 4.15]<sup>4</sup>.

Owing to the parameter-dependent analysis, the obtained exponential stability is uniform with respect to the choice of all initial conditions in  $\lambda$  and not only with respect to the choice of  $E(t_0)$  and  $\tilde{b}(t_0)$  for a given  $\hat{X}$ . ■

**Remark 2.4.4.** *For the stability analysis, we assume that  $G$  allows a matrix Lie group representation (by assumption (A1)). Nevertheless, the actual formulas of the proposed observer (2.19)-(2.20) can be computed without requiring any matrix structure for the Lie group, owing to the representation-free formulation of the proposed observer. We only require the matrix Lie group representation of  $G$  to interpret the boundedness conditions on  $\Phi(X)$ ,  $\Phi(X^{-1})$ , and  $\text{cond}(\Phi(X))$ . We will illustrate this point further with an example in Section 2.7. Boundedness of  $\Phi(X(t))$  and  $\Phi(X^{-1}(t))$  are usually mild conditions in practice. Moreover, it is easy to verify that  $\text{cond}(X(t))$  is bounded (uniformly in  $t_0$ ) if  $\Phi(X(t))$  and  $\Phi(X^{-1}(t))$  are bounded (uniformly in  $t_0$ ). For the special case where the considered Lie group is  $SO(3)$ , all of these boundedness conditions are satisfied automatically since we have  $\|\Phi(X(t))\|_F^2 = \text{tr}(\Phi(X)^\top \Phi(X)) = \text{tr}(I_{3 \times 3}) = 3$  for all  $X \in SO(3)$ . In Section 2.7, we interpret the boundedness requirements for the Lie group  $SE(3)$  as well. □*

It is possible to replace the requirement for boundedness of  $\Phi(X(t))$ ,  $\Phi(X^{-1}(t))$ , and  $\text{cond}(\Phi(X(t)))$  respectively with the boundedness of  $\Phi(\hat{X}(t))$ ,  $\Phi(\hat{X}^{-1}(t))$ , and  $\text{cond}(\Phi(\hat{X}(t)))$  in Theorems 2.4.1 and 2.4.3 and still prove the same stability results. Boundedness conditions on  $\hat{X}$  are always verifiable in practice.

Theorem 2.4.1 does not necessarily require a *symmetric* gain mapping  $K_{\hat{y}}$ . Also, we do not impose any invariance condition on this gain mapping. Condition (a) of Theorem 2.4.1 only requires the symmetric part of  $K_{\hat{y}}$ , denoted by  $K_{\hat{y}}^s$ , to be uniformly positive definite. Considering a basis for  $T_{\hat{X}}G$  and the corresponding dual basis for  $T_{\hat{X}}^*G$ , condition (a) of Theorem 2.4.1 implies that the matrix representation of  $K_{\hat{y}}^s(\cdot): T_{\hat{X}}^*G \rightarrow T_{\hat{X}}G$  with respect to these bases is uniformly symmetric positive definite. In practice, we will use this property to design a suitable gain mapping and obtain the innovation term of the observer. We will illustrate this method with an example in Section 2.7.

Condition (d) of Theorem 2.4.1 is milder than condition (d) of Theorem 2.4.3 or similar conditions imposed in [94] and in the Author's previous work [85]. This

<sup>4</sup>What is referred to as uniform exponential stability here is the same as exponential stability in the sense of [78])

allows the choice of much larger class of cost functions to generate innovation terms that guarantee the asymptotic stability of error dynamics.

**Remark 2.4.5.** *In the special case where  $K_{\hat{y}}$  is uniformly symmetric positive definite and is independent of the arguments  $\hat{b}, u_y$  and  $t$ , the term  $K_{\hat{y}}(\hat{X}, y)[D_1\phi_{\hat{y}}(\hat{X}, y)]$  simplifies to  $\text{grad}_1\phi_{\hat{y}}(\hat{X}, y)$  where  $\text{grad}_1$  denotes the gradient with respect to the Riemannian metric on  $G$  induced by the gain mapping. In this case, the observer (2.19)-(2.20) simplifies to the following gradient-like observer discussed in the Author's previous work [85, equations (7)-(8)].*

$$\dot{\hat{X}} = \hat{X}(u_y - \hat{b}) - \text{grad}_1\phi_{\hat{y}}(\hat{X}, y), \quad (2.37)$$

$$\dot{\hat{b}} = -\gamma \text{Ad}_{\hat{X}}^* \left[ \text{grad}_1\phi_{\hat{y}}(\hat{X}, y)\hat{X}^{-1} \right], \quad (2.38)$$

where the gain mapping  $\Gamma$  is simply replaced by the scalar gain  $\gamma$  and  $\text{Ad}_{\hat{X}}^* : \mathfrak{g} \rightarrow \mathfrak{g}$  denotes the Hermitian adjoint of  $\text{Ad}_{\hat{X}}$  w.r.t. the induced Riemannian metric on  $\mathfrak{g}$  such that  $\langle\langle \text{Ad}[u], v \rangle\rangle = \langle\langle u, \text{Ad}^*[v] \rangle\rangle$  for all  $u, v \in \mathfrak{g}$ . If in addition we assume that  $K_{\hat{y}}$  satisfies the invariance condition  $T_{\hat{X}}R_Z \circ K_{\hat{y}}(\hat{X}, y) \circ T_{\hat{X}}R_Z^* = K_{\hat{y}}(\hat{X}Z, h(Z, y))$ , the induced Riemannian metric on  $G$  would be right-invariant. In this case, the error dynamics (2.30)-(2.31) correspond to the perturbed gradient-like error dynamics given by [85, equations (17)-(18)]. Algebraic derivations associated with deriving an observer on a special Lie groups is usually simpler by employing an invariant Riemannian metrics and working with the gradient representation (2.37)-(2.38) of the observer (see an example in Section 2.6). Nevertheless, the larger class of gain mappings together with the larger class of cost functions proposed in this chapter ensures that the proposed observer allows a much larger class of observers compared to [92, 94] and the Author's previous work [85]. The discussion presented here shows also that a non-invariant Riemannian metric can be employed for the bias-free case to design the innovation term of the gradient-like observers the in [85, 92, 94]. In this case, the resulting error dynamics would be stable as long as the conditions on the cost function are satisfied, but the error dynamics would be non-autonomous. Non-invariant gains also lead to observers that are not symmetry-preserving in the sense of [35].

## 2.5 Constructing Invariant Cost Functions on Lie Groups

In Section 2.4, we propose the observer (2.19)-(2.20) that depends on the differential of the cost function  $\phi_{\hat{y}}: G \times M \rightarrow \mathbb{R}^+$  as its innovation term. The cost function  $\phi_{\hat{y}}$  must be right invariant, and it should satisfy condition (d) of Theorem 2.4.1 (or condition (d) of Theorem 2.4.3) in order to guarantee asymptotic (or exponential) stability of the observer error. Designing such a cost function can be challenging since  $M$  is an orbit in the product of different output spaces which can generally be a complicated manifold. In this section, we propose a method for constructing a cost function  $\phi_{\hat{y}}$  on the Lie group by employing single variable cost functions on the homogeneous output spaces  $M_i$ . Finding a suitable cost function on each output space is usually easy, especially when the output spaces are embedded in Euclidean spaces.

**Proposition 2.5.1.** [102] Suppose  $f_{\hat{y}_i}^i: M_i \rightarrow \mathbb{R}^+$ ,  $y_i \mapsto f_{\hat{y}_i}^i(y_i)$  are single variable  $C^2$  cost functions on  $M_i$ ,  $i = 1, \dots, n$ . Corresponding to each  $f_{\hat{y}_i}^i$ , construct a cost function  $\phi_{\hat{y}_i}^i: G \times M_i \rightarrow \mathbb{R}^+$  using  $\phi_{\hat{y}_i}^i(\hat{X}, y_i) := f_{\hat{y}_i}^i(h_i(\hat{X}^{-1}, y_i))$ . Obtain the cost function  $\phi_{\hat{y}}(\hat{X}, y) := \sum_{i=1}^n \phi_{\hat{y}_i}^i(\hat{X}, y_i)$ .

- (a) The cost function  $\phi_{\hat{y}}$  is right invariant in the sense defined in part (c) of Theorem 2.4.1.
- (b) Assume that each  $f_{\hat{y}_i}^i$ ,  $i = 1, \dots, n$  is locally positive definite around  $\hat{y}_i \in M_i$ . Assume moreover that  $\bigcap_{i=1}^n \text{stab}_{h_i}(\hat{y}_i) = \{I\}$  where  $\text{stab}_{h_i}(\hat{y}_i)$  denotes the stabilizer of  $\hat{y}_i$  with respect to the action  $h_i$ , defined by  $\text{stab}_{h_i}(\hat{y}_i) := \{X \in G : h_i(X, \hat{y}_i) = \hat{y}_i\}$ . Then  $\phi_{\hat{y}}(\cdot, \hat{y}) : G \rightarrow \mathbb{R}^+$  is locally positive definite around  $I \in G$ .
- (c) If  $Df_{\hat{y}_i}^i(\hat{y}_i) = 0$  for all  $i = 1, \dots, n$  then  $D_1\phi_{\hat{y}}(I, \hat{y}) = 0$ . If additionally the Hessian operators  $\text{Hess}f_{\hat{y}_i}^i(\hat{y}_i): T_{\hat{y}_i}M_i \rightarrow T_{\hat{y}_i}^*M_i$  are symmetric positive definite for all  $i = 1, \dots, n$  and  $\bigcap_{i=1}^n T_I \text{stab}_{h_i}(\hat{y}_i) = \{0\}$ , then  $\text{Hess}_1\phi_{\hat{y}}(I, \hat{y})$  is symmetric positive definite.  $\square$

*Proof:*

*Part (a):* For any arbitrary  $Z \in G$  we have

$$\begin{aligned} \phi_{\hat{y}}(\hat{X}Z, h(Z, y)) &= \sum_{i=1}^n f_{\hat{y}_i}^i\left(h_i((\hat{X}Z)^{-1}, h_i(Z, y))\right) = \sum_{i=1}^n f_{\hat{y}_i}^i\left(h_i(ZZ^{-1}\hat{X}^{-1}, y)\right) \\ &= \sum_{i=1}^n f_{\hat{y}_i}^i\left(h_i(\hat{X}^{-1}, y)\right) = \phi_{\hat{y}}(\hat{X}, y). \end{aligned}$$

This shows that  $\phi_{\hat{y}}$  is right invariant.

*Part (b):* Since  $f_{\hat{y}_i}^i(y_i)$  is positive definite around  $y_i = \hat{y}_i$ , there exists a neighborhood  $N_i \subset M_i$  of  $\hat{y}_i$  such that  $f_{\hat{y}_i}^i(y_i) \geq 0$  and  $f_{\hat{y}_i}^i(y_i) = 0 \Rightarrow y_i = \hat{y}_i$  for all  $y_i \in N_i$ . Corresponding to each  $N_i$ , define the set  $\bar{N}_i := \{E \in G : h_i(E^{-1}, \hat{y}_i) \in N_i\} \subset G$  and consider the set  $\bar{N} := \bigcap_{i=1}^n \bar{N}_i$ . It is easy to verify that  $\bar{N} \subset G$  is a neighborhood of  $I$  and we have  $\phi_{\hat{y}}(E, \hat{y}) = \sum_{i=1}^n f_{\hat{y}_i}^i(h_i(E^{-1}, \hat{y}_i)) \geq 0$  for all  $E \in \bar{N}$ . Moreover, for any  $E \in \bar{N}$ ,  $\phi_{\hat{y}}(E, \hat{y}) = \sum_{i=1}^n f_{\hat{y}_i}^i(h_i(E^{-1}, \hat{y}_i)) = 0$  yields  $f_{\hat{y}_i}^i(h_i(E^{-1}, \hat{y}_i)) = 0$  for all  $i = 1, \dots, n$ . This in turn implies that  $h_i(E^{-1}, \hat{y}_i) = \hat{y}_i$ ,  $i = 1, \dots, n$  and hence  $E \in \bigcap_{i=1}^n \text{stab}_{h_i}(\hat{y}_i)$ . We assumed  $\bigcap_{i=1}^n \text{stab}_{h_i}(\hat{y}_i) = \{I\}$  which ensures that  $E = I$  and hence  $\phi_{\hat{y}}(E, \hat{y})$  is positive definite on  $\bar{N}$ .

*Part (c):* Define the map  $h_{\hat{y}_i}: G \rightarrow M_i$  by  $h_{\hat{y}_i}X := h_i(X, \hat{y}_i)$ . Differentiating both sides of  $\phi_{\hat{y}}(E, \hat{y}) = \sum_{i=1}^n f_{\hat{y}_i}^i(h_i(E^{-1}, \hat{y}_i))$  in an arbitrary direction  $v \in T_E G$  and using the chain rule we have

$$D_1 \phi_{\hat{y}}(E, \hat{y})[v] = - \sum_{i=1}^n Df_{\hat{y}_i}^i(h_i(E^{-1}, \hat{y}_i)) \circ T_{E^{-1}} h_{\hat{y}_i} \circ T_I L_{E^{-1}} \circ T_E R_{E^{-1}}[v].$$

Evaluating the later relation at  $E = I$  and omitting the arbitrary argument  $v$  we obtain  $D_1 \phi_{\hat{y}}(I, \hat{y}) = - \sum_{i=1}^n Df_{\hat{y}_i}^i(\hat{y}_i) \circ T_I h_{\hat{y}_i}$ . Hence,  $Df_{\hat{y}_i}^i(\hat{y}_i) = 0$ ,  $i = 1, \dots, n$  implies  $D_1 \phi_{\hat{y}}(I, \hat{y}) = 0$ . Under this condition, standard computations shows that

$$\text{Hess}_1 \phi_{\hat{y}}(I, \hat{y}) = \sum_{i=1}^n \text{Hess}_1 \phi_i(I, \hat{y}_i) = \sum_{i=1}^n T_I h_{\hat{y}_i}^* \circ \text{Hess} f_{\hat{y}_i}^i(\hat{y}_i) \circ T_I h_{\hat{y}_i},$$

where  $T_I h_{\hat{y}_i}^*: T_{\hat{y}_i}^* M_i \rightarrow T_I^* G$  denotes the dual of  $T_I h_{\hat{y}_i}$ . If all of  $\text{Hess} f_{\hat{y}_i}^i(\hat{y}_i)$ ,  $i = 1, \dots, n$  are symmetric positive definite, then  $\text{Hess}_1 \phi_{\hat{y}}(I, \hat{y})$  is symmetric positive semi definite with  $\ker(\text{Hess}_1 \phi_{\hat{y}}(I, \hat{y})) = \bigcap_{i=1}^n \ker(T_I h_{\hat{y}_i})$ . Since,  $\ker(T_I h_{\hat{y}_i}) = T_I \text{stab}_{h_i}(\hat{y}_i)$ , we have

$$\bigcap_{i=1}^n \ker(T_I h_{\hat{y}_i}) = \bigcap_{i=1}^n T_I \text{stab}_{h_i}(\hat{y}_i) = \{0\}.$$

Consequently,  $\ker(\text{Hess}_1 \phi_{\hat{y}}(I, \hat{y})) = \{0\}$  which implies that  $\text{Hess}_1 \phi_{\hat{y}}(I, \hat{y})$  is full rank and hence symmetric positive definite.  $\blacksquare$

Proposition 2.5.1 suggests a systematic method to construct a cost function which satisfies the requirements of Theorem 2.4.1 or Theorem 2.4.3. The differential of this function can be employed to design the innovation term of the observer. We will

illustrate this method with examples in sections 2.6 and 2.7.

The method proposed in Proposition 2.5.1 to construct the cost function  $\phi_{\hat{y}}$  is fundamentally different from the one presented in the Author's prior work [85, Proposition 2]. The method proposed in [85, Proposition 2] employs invariant cost functions on  $M_i \times M_i$  while the method presented here only requires single variable cost functions on each  $M_i$ . Implementability of the proposed observer in [85] is guaranteed when the homogeneous output spaces are reductive. The method presented in this Chapter guarantees the implementability of resulting observer without imposing any reductivity condition.

The condition  $\bigcap_{i=1}^n \text{stab}_{h_i}(\hat{y}_i) = \{I\}$  (imposed in part (b) of Proposition 2.5.1) is sufficient to ensure  $\bigcap_{i=1}^n T_I \text{stab}_{h_i}(\hat{y}_i) = \{0\}$  (imposed in part (c) of the Proposition). This condition can be interpreted as an observability criterion. In particular, for the attitude estimation problem with vectorial measurements, this condition is equivalent to the availability of two or more non-collinear reference vectors, as will be shown in Section 2.6. As will be discussed in Section 2.7, for the pose estimation problem with landmark measurements, this condition corresponds to the availability of three or more landmarks which are not located on the same line.

The method presented in this chapter suits the systems with constant reference outputs. Time varying reference outputs have been investigated in [29, 54, 80, 89, 133] for attitude estimation problem on  $\text{SO}(3)$ . Nevertheless, in most practical cases, the reference outputs are approximately constant [37, 101, 135] and the proposed observer design methodology applies.

## 2.6 Example: Attitude Estimation Using Biased Angular Velocity Measurement

Recall the attitude estimation problem discussed in Example 2.2.1. We aim to employ the gradient formulation of the observer (2.37)-(2.38) to design an attitude and gyro bias observer. Since the Lie group  $\text{SO}(3)$  is compact, the boundedness conditions imposed by Assumption (A2) and part (e) of Theorem 2.4.3 are automatically satisfied. It is easy to verify that the requirements  $\bigcap_{i=1}^n \text{stab}_{h_i}(\hat{y}_i) = \{I\}$  and  $\bigcap_{i=1}^n T_I \text{stab}_{h_i}(\hat{y}_i) = \{0\}$  (respectively, imposed by parts (b) and (c) of Proposition 2.5.1) are satisfied if there are at least two vector measurements whose correspond-

ing reference vectors are non-collinear.

Suppose  $f_{\hat{y}_i}^i: M_i \rightarrow \mathbb{R}^+$ ,  $y_i \mapsto f_{\hat{y}_i}^i(y_i)$  are single variable  $C^2$  cost functions on  $M_i$ ,  $i = 1, \dots, n$ . Corresponding to each  $f_{\hat{y}_i}^i$ , construct a cost function  $\phi_{\hat{y}_i}^i: G \times M_i \rightarrow \mathbb{R}^+$  using  $\phi_{\hat{y}_i}^i(\hat{X}, y_i) := f_{\hat{y}_i}^i(h_i(\hat{X}^{-1}, y_i))$ . Obtain the cost function  $\phi_{\hat{y}}(\hat{X}, y) := \sum_{i=1}^n \phi_{\hat{y}_i}^i(\hat{X}, y_i)$ .

Consider  $f_{\hat{y}_i}^i(y_i) = \frac{l_i}{2} \|y_i - \hat{y}_i\|^2$  where  $\|a\| = a^\top a$  is induced by the standard Euclidian norm on  $\mathbb{R}^3$  and  $l_i$  is a positive scalar. It is easy to see that  $f_{\hat{y}_i}^i$ ,  $i = 1, \dots, n$  are locally positive definite around  $\hat{y}_i \in M_i$  and they have symmetric positive definite Hessians, hence satisfying the requirements of Proposition 2.5.1. Obtain  $\phi_{\hat{y}}(\hat{R}, y) := \sum_{i=1}^n f_{\hat{y}_i}^i(h_i(\hat{X}^{-1}, y_i)) = \sum_{i=1}^n \frac{l_i}{2} \|\hat{R}y_i - \hat{y}_i\|^2 = \sum_{i=1}^n \frac{l_i}{2} \|\hat{R}^\top \hat{y}_i - y_i\|^2$ . Consider the inner product  $\langle\langle \Omega_1, \Omega_2 \rangle\rangle = \text{tr}(\Omega_1^\top \Omega_2)$ ,  $\forall \Omega_1, \Omega_2 \in \mathfrak{so}(3)$  induced by the standard inner product in  $\mathbb{R}^{3 \times 3}$  obtain the following right-invariant Riemannian metric on  $\text{SO}(3)$  using the right translation of the Lie algebra  $\mathfrak{so}(3)$ .

$$\langle\langle \Omega_1 \hat{R}, \Omega_2 \hat{R} \rangle\rangle_{\hat{R}} := \langle\langle \Omega_1, \Omega_2 \rangle\rangle = \text{tr}(\Omega_1^\top \Omega_2).$$

Recalling that  $\text{grad}_1 \phi_{\hat{y}}(\hat{X}, X)$  can be identified using the differential of  $\phi_{\hat{y}}(\hat{X}, y)$  with respect to the first coordinate in an arbitrary tangential direction  $\hat{\Omega} \hat{R} \in T_{\hat{R}} \text{SO}(3)$ , we have

$$D_1 \phi_{\hat{y}}(\hat{R}, y)[\hat{\Omega} \hat{R}] = \langle \text{grad}_1 \phi_{\hat{y}}(\hat{R}, y), \hat{\Omega} \hat{R} \rangle_{\hat{R}} = \langle\langle \text{grad}_1 \phi_{\hat{y}}(\hat{R}, y) \hat{R}^\top, \hat{\Omega} \rangle\rangle, \quad (2.39)$$

We also have

$$D_1 \phi_{\hat{y}}(\hat{R}, y)[\hat{\Omega} \hat{R}] = \sum_{i=1}^n l_i \hat{y}_i^\top \hat{\Omega} \hat{R} (\hat{R}^\top \hat{y}_i - y_i) = -\text{tr} \left( \hat{\Omega}^\top \hat{R} \sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i - y_i) \hat{y}_i^\top \right).$$

Using the notation  $\mathbb{P}_{\mathfrak{so}(3)}(A) = 0.5(A - A^\top)$  for the matrix projection from  $\mathbb{R}^{3 \times 3}$  onto  $\mathfrak{so}(3)$  we have

$$\begin{aligned} D_1 \phi_{\hat{y}}(\hat{R}, y)[\hat{\Omega} \hat{R}] &= -\text{tr} \left( \hat{\Omega}^\top \mathbb{P}_{\mathfrak{so}(3)}(\hat{R} \sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i - y_i) \hat{y}_i^\top) \right) \\ &= \langle\langle -\mathbb{P}_{\mathfrak{so}(3)}(\hat{R} \sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i - y_i) \hat{y}_i^\top), \hat{\Omega} \rangle\rangle. \end{aligned} \quad (2.40)$$

Comparing (2.39) and (2.40), we infer

$$\text{grad}_1 \phi_{\hat{y}}(\hat{R}, y) = -\mathbb{P}_{\mathfrak{so}(3)}(\hat{R} \sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i - y_i) \hat{y}_i^\top) \hat{R}. \quad (2.41)$$

Using the property  $\mathbb{P}_{\mathfrak{so}(3)}(\hat{R}A) = \hat{R}\mathbb{P}_{\mathfrak{so}(3)}(A\hat{R})\hat{R}^\top$  for  $A \in \mathbb{R}^{3 \times 3}$ , we have

$$\begin{aligned} \text{grad}_1 \phi_{\hat{y}}(\hat{R}, y) &= -\hat{R}\mathbb{P}_{\mathfrak{so}(3)}\left(\sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i - y_i) \hat{y}_i^\top \hat{R}\right) = -\hat{R}\mathbb{P}_{\mathfrak{so}(3)}\left(\sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i - y_i) (\hat{R}^\top \hat{y}_i)^\top\right) \\ &= -\hat{R} \sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i \hat{y}_i^\top - y_i \hat{y}_i^\top \hat{R}). \end{aligned} \quad (2.42)$$

Using (2.37), the attitude estimator is given by

$$\dot{\hat{R}} = \hat{R}(\Omega_y - \hat{b}) + \hat{R} \sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i \hat{y}_i^\top - y_i \hat{y}_i^\top \hat{R}) \quad (2.43)$$

Since  $\text{SO}(3) \subset \text{GL}(3)$  is a matrix Lie group and our considered Riemannian metric is induced by the standard inner product on  $\text{GL}(3)$ , we can explicitly formulate the map  $\text{Ad}_{\hat{R}}^* : \mathfrak{so}(3) \rightarrow \mathfrak{so}(3)$  by  $\text{Ad}_{\hat{R}}^* v = \mathbb{P}_{\mathfrak{so}(3)}(\hat{R}^\top v \hat{R})$  for all  $v \in \mathfrak{so}(3)$ . So, the bias estimator (2.38) becomes

$$\dot{\hat{b}} = \gamma \mathbb{P}_{\mathfrak{so}(3)}(\hat{R}^\top \hat{R} \sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i \hat{y}_i^\top - y_i \hat{y}_i^\top \hat{R}) \hat{R}^{-1} \hat{R}) = \gamma \sum_{i=1}^n l_i (\hat{R}^\top \hat{y}_i \hat{y}_i^\top - y_i \hat{y}_i^\top \hat{R}). \quad (2.44)$$

Using the property  $ab^\top - ba^\top = (a \times b)_\times$ , one can conclude that (2.43)-(2.44) corresponds to the complementary passive attitude estimator proposed in [101, 136] where almost globally asymptotic and locally exponentially convergence of the estimation error to  $(I, 0)$  has been proved.

## 2.7 Example: Pose Estimation Using Biased Angular and Linear Velocity Measurements

Recalling the pose estimation problem discussed in Example 2.2.2, here we employ our observer (2.19)-(2.20) to derive the pose estimators proposed in [65] and [135] and we generalize them. As opposed to the attitude estimation example, here we do not employ the gradient formulation of the observer and we work directly with the

observer formulation (2.19)-(2.20).

Apart from the semi-direct product representation of  $SE(3)$  discussed in Example 2.2.2, it is known that  $SE(3)$  has also a matrix Lie group representation as a subgroup of  $GL(4)$  (see e.g. [65]). We use this matrix Lie group representation only to interpret the required boundedness conditions (see Assumption (A2)) but we employ the semi-direct product representation to derive the observer formulas (see remark 2.4.4). The Lie group homomorphism  $\Phi$  which maps an element  $(R, p) \in SE(3)$  to its corresponding matrix representation in  $GL(4)$  is given by;  $\Phi : (R, p) \mapsto \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}$ . The Frobenius norm of  $\Phi((R, p)) \in GL(4)$  is given by  $\|\Phi((R, p))\|^2 = \text{tr}(\Phi((R, p))^\top \Phi((R, p))) = 4 + \|p\|^2$ . Hence,  $\Phi((R(t), p(t)))$  is bounded if  $p(t)$  is bounded. Similarly, one can verify that  $\Phi((R(t), p(t)))^{-1}$  and  $\text{cond}(\Phi((R(t), p(t))))$  are bounded (uniformly in  $t_0$ ) if  $p(t)$  is bounded (uniformly in  $t_0$ ). This characterizes the boundedness conditions imposed by Assumption (A2) and part (e) of Theorem 2.4.3.

From here after, we only consider the semi-direct product representation of  $SE(3) \simeq SO(3) \times \mathbb{R}^3$ . We aim to employ the observer developed in Section 2.4 and use the guidelines presented in Section 2.5 to design an observer to estimate the pose  $X = (R, p)$  and the bias  $b = (b_\omega, b_v)$ . Let us first evaluate the observability condition imposed by part (b) and (c) of Proposition 2.5.1. We have  $\bigcap_{i=1}^n \text{stab}_{h_i}(\hat{y}_i) = \{(R, p) \in SE(3) : R^\top \hat{y}_i - R^\top p = \hat{y}_i, i = 1, \dots, n\} = \{(R, p) \in SE(3) : R^\top p = R^\top \hat{y}_i - \hat{y}_i, R(\hat{y}_i - \hat{y}_j) = \hat{y}_i - \hat{y}_j, i, j = 1, \dots, n, i \neq j\}$  which implies that  $\hat{y}_i - \hat{y}_j$  is an eigenvector of  $R$ . Hence, a necessary and sufficient condition which guarantees  $\bigcap_{i=1}^n \text{stab}_{h_i}(\hat{y}_i) = \{(I_{3 \times 3}, 0_3)\}$  (and consequently  $\bigcap_{i=1}^n T_{(I,0)} \text{stab}_{h_i}(\hat{y}_i) = \{(0_{3 \times 3}, 0_3)\}$ ) is the existence of at least three reference outputs  $\hat{y}_i, \hat{y}_j, \hat{y}_k$  such that  $\hat{y}_i - \hat{y}_j$  is not parallel to  $\hat{y}_j - \hat{y}_k$ . Note that this condition is independent of the choice of inertial frame. Specifically, when landmark measurements are employed to provide outputs  $y_i, i = 1, \dots, n$ , this condition is equivalent to the existence of at least three landmarks which are not located on the same line [65, 135].

In order to design the innovation terms of the estimator (2.19)-(2.20), we resort to choose a basis for each tangent space to obtain matrix representations for the linear mappings  $K_{\hat{y}}, \Gamma, T_1 L_X^*$  and use simple matrix calculus. For the sake of clarity, we denote the matrix representation of a linear mapping  $F: U \rightarrow W$  with respect to the

basis  $\{u\}$  for its domain and basis  $\{w\}$  for its co-domain by the notation  $\llbracket F \rrbracket_u^w$ . Also, the  $\mathbb{R}^n$  representation of a vector  $a \in U$  with respect to the basis  $\{u\}$  is denoted by  $\llbracket a \rrbracket_u$ . Denote the standard bases of  $\mathbb{R}^3$  and  $\mathfrak{so}(3)$  by  $\{e\}$  and  $\{e_\times\}$ , respectively. Using these bases, one can obtain a standard basis for  $\mathfrak{se}(3)$  denoted by  $\{\bar{e}\}$ . We obtain a basis for  $T_{(\hat{R}, \hat{p})}\text{SE}(3)$  using the right translation of  $\{\bar{e}\}$ . Denote this basis of  $T_{(\hat{R}, \hat{p})}\text{SE}(3)$  by  $\{\bar{e}\hat{X}\}$  and its corresponding dual basis of  $T_{(\hat{R}, \hat{p})}^*\text{SE}(3)$  by  $\{(\bar{e}\hat{X})^*\}$ .

In order to use Proposition 2.5.1, we start by designing suitable costs  $f_{\hat{y}_i}^i : M_i \rightarrow \mathbb{R}^+$ . A simple cost function is constructed by  $f_{\hat{y}_i}^i(y_i) := \frac{l_i}{2} \|y_i - \hat{y}_i\|^2$ ,  $l_i > 0$  where  $\|\cdot\|$  denotes the Euclidean distance. It is straight forward to verify that  $f_{\hat{y}_i}^i$  satisfies the requirements imposed by part (c) of Proposition 2.5.1, i.e.  $Df_{\hat{y}_i}^i(\hat{y}_i) = 0$  and  $\text{Hess}f_{\hat{y}_i}^i(\hat{y}_i)$  is symmetric positive definite. The cost functions  $\phi_{\hat{y}_i}^i : \text{SE}(3) \times M_i \rightarrow \mathbb{R}^+$ ,  $i = 1, \dots, n$  are obtained as  $\phi_{\hat{y}_i}^i(\hat{X}, y_i) = \frac{l_i}{2} \|h_i((\hat{R}, \hat{p})^{-1}, y_i) - \hat{y}_i\|^2 = \frac{l_i}{2} \|\hat{R}y_i + \hat{p} - \hat{y}_i\|^2$ . Denoting an arbitrary element of  $\mathfrak{se}(3)$  by  $(\hat{\Omega}, \hat{V})$ , we have  $T_{(I, 0)}R_{(\hat{R}, \hat{p})}[(\hat{\Omega}, \hat{V})] = (\hat{\Omega}\hat{R}, \hat{\Omega}\hat{p} + V) \in T_{(\hat{R}, \hat{p})}\text{SE}(3)$ . One can obtain  $D_1\phi_{\hat{y}_i}((\hat{R}, \hat{p}), y) : T_{(\hat{R}, \hat{p})}\text{SE}(3) \rightarrow \mathbb{R}$  as

$$D_1\phi_{\hat{y}_i}((\hat{R}, \hat{p}), y)[(\hat{\Omega}\hat{R}, \hat{\Omega}\hat{p} + V)] = \sum_{i=1}^n l_i \alpha_i^\top (\hat{\Omega}\hat{R}y_i + \hat{\Omega}\hat{p} + V). \quad (2.45)$$

where

$$\alpha_i := (\hat{R}y_i + \hat{p} - \hat{y}_i) \in \mathbb{R}^3.$$

The  $\mathbb{R}^6$  representation of  $D_1\phi_{\hat{y}_i}(\hat{X}, y) \in T_{\hat{X}}^*\text{SE}(3)$  is the transpose of the matrix representation of  $D_1\phi_{\hat{y}_i}(\hat{X}, y) : T_{\hat{X}}\text{SE}(3) \rightarrow \mathbb{R}$ , i.e.  $\llbracket D_1\phi_{\hat{y}_i}(\hat{X}, y) \rrbracket_{(\bar{e}\hat{X})^*} = \left( \llbracket D_1\phi_{\hat{y}_i}(\hat{X}, y) \rrbracket_{\bar{e}\hat{X}}^1 \right)^\top$ . Employing (2.45) and using the basis  $\{\bar{e}\hat{X}\}$ , we obtain

$$\begin{aligned} \llbracket D_1\phi_{\hat{y}_i}(\hat{X}, y) \rrbracket_{\bar{e}\hat{X}}^1 &= \sum_{i=1}^n l_i \alpha_i^\top [e_\times^1 \hat{R}y_i + e_\times^1 \hat{p}, e_\times^2 \hat{R}y_i + e_\times^2 \hat{p}, e_\times^3 \hat{R}y_i + e_\times^3 \hat{p}, e^1, e^2, e^3] \\ &= \sum_{i=1}^n l_i [-\alpha_i^\top (\hat{R}y_i + \hat{p})_\times, \alpha_i^\top] = \sum_{i=1}^n l_i [\hat{y}_i^\top (\hat{R}y_i + \hat{p})_\times, \alpha_i^\top], \end{aligned} \quad (2.46)$$

where  $\{e\} = \{e^1, e^2, e^3\}$  and  $-\alpha_i^\top (\hat{R}y_i + \hat{p})_\times = -(\hat{R}y_i + \hat{p} - \hat{y}_i)^\top (\hat{R}y_i + \hat{p})_\times = \hat{y}_i^\top (\hat{R}y_i + \hat{p})_\times$  is used in the last step of (2.46). We choose  $\llbracket K_{\hat{y}_i}(\hat{X}, y, \hat{b}, u_y, t) \rrbracket_{(\bar{e}\hat{X})^*}^{\bar{e}\hat{X}} = \text{diag}(l_\omega I_{3 \times 3}, l_v I_{3 \times 3})$  where  $l_\omega, l_v$  are positive scalars and ensure that the resulting gain mapping

$$K_{\hat{y}_i}(\hat{X}, y, \hat{b}, u_y, t) : T_{(\hat{R}, \hat{p})}^*\text{SE}(3) \rightarrow T_{(\hat{R}, \hat{p})}\text{SE}(3),$$

is uniformly positive definite. Using (2.46) we have

$$\begin{aligned} \llbracket K_{\hat{y}}(\cdot)[D_1\phi_{\hat{y}}(\hat{X}, y)] \rrbracket_{\bar{e}\hat{X}} &= \llbracket K_{\hat{y}}(\cdot) \rrbracket_{(\bar{e}\hat{X})^*}^{\bar{e}\hat{X}} \llbracket D_1\phi_{\hat{y}}(\hat{X}, y) \rrbracket_{(\bar{e}\hat{X})^*} \\ &= \left( \sum_{i=1}^n l_i [l_\omega \hat{y}_i^\top (\hat{R}y_i + \hat{p})_\times, l_v \alpha_i^\top] \right)^\top \end{aligned}$$

where the argument  $(\hat{X}, y, \hat{b}, u_y, t)$  of  $K_{\hat{y}}$  has been omitted for brevity. We use (6.2), from Lemma 6.1.1 in the Appendix, to obtain

$$K_{\hat{y}}(\cdot)[D_1\phi_{\hat{y}}(\hat{X}, y)] = \sum_{i=1}^n l_i (-l_\omega((\hat{R}y_i + \hat{p})_\times \hat{y}_i)_\times \hat{R}, -l_\omega((\hat{R}y_i + \hat{p})_\times \hat{y}_i)_\times \hat{p} + l_v \alpha_i) \quad (2.47)$$

In order to design the innovation term of the bias estimator, we first need to obtain a matrix representation for the map  $[T_1L_{(\hat{R}, \hat{p})}]$ . Suppose  $[a^\top, b^\top]^\top \in \mathbb{R}^6$  as the first column of  $\llbracket T_1L_{(\hat{R}, \hat{p})} \rrbracket_{\bar{e}}^{\bar{e}\hat{X}}$  and denote by  $a_i, b_i, i = 1, \dots, 3$  the elements of  $a, b \in \mathbb{R}^3$ . We have

$$T_1L_{\hat{X}}[(e^1_\times, 0)] = (\hat{R}e^1_\times, 0) = \sum_{i=1}^3 a_i (e^i_\times \hat{R}, e^i_\times \hat{p}) + b_i (0, e^i) = \sum_{i=1}^3 (a_i e^i_\times \hat{R}, a_i e^i_\times \hat{p} + b_i e^i).$$

This implies that  $\hat{R}e^1_\times = \sum_{i=1}^3 a_i e^i_\times \hat{R}$  and  $0 = \sum_{i=1}^3 a_i e^i_\times \hat{p} + b_i e^i$  which together form 6 linear equations with 6 unknowns. Solving this set of equations yields  $a = \hat{R}e^1$  and  $b = \hat{p}_\times \hat{R}e^1$ . Consequently, the first column of  $\llbracket T_1L_{(\hat{R}, \hat{p})} \rrbracket_{\bar{e}}^{\bar{e}\hat{X}}$  is  $[(\hat{R}e^1)^\top, (\hat{p}_\times \hat{R}e^1)^\top]^\top$ . One can use the same procedure as was explained above to obtain the second and third column of  $\llbracket T_1L_{(\hat{R}, \hat{p})} \rrbracket_{\bar{e}}^{\bar{e}\hat{X}}$  as  $[(\hat{R}e^2)^\top, (\hat{p}_\times \hat{R}e^2)^\top]^\top$  and  $[(\hat{R}e^3)^\top, (\hat{p}_\times \hat{R}e^3)^\top]^\top$  respectively. Suppose  $[c^\top, d^\top]^\top$  as the fourth column of  $\llbracket T_1L_{(\hat{R}, \hat{p})} \rrbracket_{\bar{e}}^{\bar{e}\hat{X}}$ . We have,  $T_1L_{\hat{X}}[(0, e^1)] = (0, \hat{R}e^1) = \sum_{i=1}^3 (c_i e^i_\times \hat{R}, c_i e^i_\times \hat{p} + d_i e^i)$ . This implies that  $0 = \sum_{i=1}^3 c_i e^i_\times \hat{R}$  and  $\hat{R}e^1 = \sum_{i=1}^3 c_i e^i_\times \hat{p} + d_i e^i$  which again form 6 linear equations with 6 unknowns. Solving this set of equations yields  $c = 0$  and  $d = \hat{R}e^1$ . Hence the fourth column of  $\llbracket T_1L_{(\hat{R}, \hat{p})} \rrbracket_{\bar{e}}^{\bar{e}\hat{X}}$  is given by  $[0, (\hat{R}e^1)^\top]^\top$ . We can apply the same procedure to obtain the fifth and sixth column as well. Combining all of the columns together yields

$$\llbracket T_1L_{(\hat{R}, \hat{p})} \rrbracket_{\bar{e}}^{\bar{e}\hat{X}} = \begin{bmatrix} \hat{R}e^1 & \hat{R}e^2 & \hat{R}e^3 & 0_3 & 0_3 & 0_3 \\ \hat{p}_\times \hat{R}e^1 & \hat{p}_\times \hat{R}e^2 & \hat{p}_\times \hat{R}e^3 & \hat{R}e^1 & \hat{R}e^2 & \hat{R}e^3 \end{bmatrix} = \begin{bmatrix} \hat{R} & 0_{3 \times 3} \\ \hat{p}_\times \hat{R} & \hat{R} \end{bmatrix}.$$

Now, choosing the gain  $\llbracket \Gamma \rrbracket_{\bar{e}}^{\bar{e}} = \text{diag}(\gamma_\omega I_{3 \times 3}, \gamma_v I_{3 \times 3})$ , we have

$$\begin{aligned} \llbracket \Gamma \circ T_I L_{\hat{X}}^* [D_1 \phi_{\hat{y}}(\hat{X}, y)] \rrbracket_{\bar{e}}^{\bar{e}} &= \llbracket \Gamma \rrbracket_{\bar{e}}^{\bar{e}} \left( \llbracket T_I L_{(\hat{R}, \hat{p})} \rrbracket_{\bar{e}}^{\bar{e}\hat{X}} \right)^\top \llbracket D_1 \phi_{\hat{y}}(\hat{X}, y) \rrbracket_{(\bar{e}\hat{X})^*} \\ &= \begin{bmatrix} \gamma_\omega I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & \gamma_v I_{3 \times 3} \end{bmatrix} \begin{bmatrix} \hat{R} & 0_{3 \times 3} \\ \hat{p}_\times \hat{R} & \hat{R} \end{bmatrix}^\top \begin{bmatrix} -\sum_{i=1}^n l_i (\hat{R} y_i + \hat{p})_\times \hat{y}_i \\ \sum_{i=1}^n l_i \alpha_i \end{bmatrix} \\ &= \sum_{i=1}^n l_i \begin{bmatrix} \gamma_\omega y_{i \times} (\hat{R}^\top \hat{y}_i - \hat{R}^\top \hat{p}) \\ \gamma_v \hat{R}^\top \alpha_i \end{bmatrix}. \end{aligned} \quad (2.48)$$

We employ (2.48) and (6.1), from Lemma 6.1.1 in the Appendix, to obtain

$$\Gamma \circ T_I L_{\hat{X}}^* [D_1 \phi_{\hat{y}}(\hat{X}, y)] = \sum_{i=1}^n l_i (\gamma_\omega (y_{i \times} (\hat{R}^\top \hat{y}_i - \hat{R}^\top \hat{p}))_\times, \gamma_v \hat{R}^\top \alpha_i) \quad (2.49)$$

Using (2.47) and (2.49), the observer is summarized as

$$\dot{\hat{R}} = \hat{R}(\Omega_y - \hat{b}_\omega) + l_\omega \sum_{i=1}^n l_i ((\hat{R} y_i + \hat{p})_\times \hat{y}_i)_\times \hat{R} \quad (2.50a)$$

$$\dot{\hat{p}} = \hat{R}(V_y - \hat{b}_v) + \sum_{i=1}^n l_i (l_\omega ((\hat{R} y_i + \hat{p})_\times \hat{y}_i)_\times \hat{p} - l_v (\hat{R} y_i + \hat{p} - \hat{y}_i)) \quad (2.50b)$$

$$\dot{\hat{b}}_\omega = \gamma_\omega \sum_{i=1}^n l_i (y_{i \times} (\hat{R}^\top \hat{y}_i - \hat{R}^\top \hat{p}))_\times \quad (2.50c)$$

$$\dot{\hat{b}}_v = \gamma_v (\hat{R}^\top \hat{p} - \hat{R}^\top \hat{y}_i + y_i) \quad (2.50d)$$

Notice that the resulting observer formula (2.50) do not depend on the chosen basis. Omitting the bias estimator, the group estimator (2.50a)-(2.50b) has a similar form as the gradient-like observer proposed in [65, equation (35)] since the chosen gain mapping  $K_{\hat{y}}$  is symmetric positive definite and yields a gradient innovation term.

The pose estimator of [135] has a different form from (2.50). Here, we derive the observer of [135] by choosing different gain mappings and output maps. Similar to [135, equation (8)], consider the new set of outputs  $z_j$ ,  $j = 1, \dots, n$  given by

$$z_j := \sum_{i=1}^{n-1} a_{ij} (y_{i+1} - y_i), \quad j = 1, \dots, n-1 \quad (2.51a)$$

$$z_n := -\frac{1}{n} \sum_{i=1}^n y_i. \quad (2.51b)$$

We assume that  $a_{ij} \in \mathbb{R}$  are such that the matrix  $A := [a_{ij}] \in \mathbb{R}^{(n-1) \times (n-1)}$  is full rank. This requirement guarantees that no information is lost by applying the linear trans-

formation (2.51) to the measurements. Substituting  $y_i$  from (2.9) into (2.51) and defining new reference outputs  $\dot{z}_j := \sum_{i=1}^{n-1} a_{ij}(\dot{y}_{i+1} - \dot{y}_i)$ ,  $j = 1, \dots, n-1$ ,  $\dot{z}_n = -\frac{1}{n} \sum_{i=1}^n \dot{y}_i$  yields

$$z_j = g_j((R, p), \dot{z}_j) := R^\top \dot{z}_j, \quad j = 1, \dots, n-1 \quad (2.52a)$$

$$z_n = g_n((R, p), \dot{z}_n) := R^\top \dot{z}_n + R^\top p \quad (2.52b)$$

where  $g_j$ ,  $j = 1, \dots, n$  are right output actions of  $G$ . Consider the new combined output  $z := (z_1, \dots, z_n)$  and the combined reference output  $\dot{z} := (\dot{z}_1, \dots, \dot{z}_n)$ . One can show that the necessary and sufficient condition for  $\bigcap_{j=1}^n \text{stab}_{g_j}(\dot{z}_j) = \{I\}$  is the existence of at least two non-collinear reference outputs  $\dot{z}_j, \dot{z}_k$ . Assuming that  $A = [a_{ij}]$  is invertible, it is straight forward to show that the above mentioned condition on  $\dot{z}$  is equivalent to the condition on  $\dot{y}$  we derived before.

We employ the cost functions  $f_{\dot{z}_j}^j(z_j) := \frac{l_j}{2} \|z_j - \dot{z}_j\|^2$ ,  $l_j > 0$  and we choose the gain mappings  $\llbracket K_{\dot{z}}(\hat{X}, y, \hat{b}, u_y, t) \rrbracket_{(\hat{e}\hat{X})^*}^{\hat{e}\hat{X}} = \text{diag}(l_\omega I_{3 \times 3}, l_v I_{3 \times 3}) + \text{diag}(0_{3 \times 3}, (\hat{R}(\Omega_y - \hat{b}_\omega))_\times)$  and  $\llbracket \Gamma \rrbracket_{\hat{e}^*}^{\hat{e}} = \text{diag}(\gamma_\omega I_{3 \times 3}, \gamma_v I_{3 \times 3})$ . It is easy to verify that this choice of cost functions and gain mappings satisfies the requirements of our method. Notice that  $K_{\dot{z}}$  is non-symmetric and depends also on  $\Omega_y$  and  $\hat{b}_\omega$  unlike the previous part. In particular, this implies that the observer innovation is not a gradient innovation. Nevertheless, the symmetric part of  $K_{\dot{z}}$  is  $\text{diag}(l_\omega I_{3 \times 3}, l_v I_{3 \times 3})$  which implies that the resulting gain mapping  $K_{\dot{z}}$  is uniformly positive definite. Following the same procedure as was done to derive (2.50), we obtain the following observer.

$$\dot{\hat{R}} = \hat{R}(\Omega_y - \hat{b}_\omega) - l_\omega k_n (\hat{p} \times \dot{z}_n)_\times \hat{R} + l_\omega \sum_{j=1}^n l_j \hat{R} ((\hat{R}^\top \dot{z}_j)_\times z_j)_\times \quad (2.53a)$$

$$\begin{aligned} \dot{\hat{p}} = & \hat{R}(V_y - \hat{b}_v) + k_n (l_v I_{3 \times 3} + (\hat{R}(\Omega_y - \hat{b}_\omega))_\times) (\hat{R} z_n - \hat{p} - \dot{z}_n) \\ & - l_\omega k_n (\hat{p} \times \dot{z}_n)_\times \hat{p} + l_\omega \sum_{j=1}^n l_j ((\hat{R} z_j)_\times \dot{z}_j)_\times \hat{p} \end{aligned} \quad (2.53b)$$

$$\dot{\hat{b}}_\omega = \gamma_\omega k_n \left( \hat{R}^\top \hat{p}_\times (\hat{R} z_n - \hat{p}) \right)_\times + \gamma_\omega \sum_{j=1}^n l_j \left( (\hat{R}^\top \dot{z}_j)_\times z_j \right)_\times \quad (2.53c)$$

$$\dot{\hat{b}}_v = -\gamma_v k_n \hat{R}^\top (\hat{R} z_n - \hat{p} - \dot{z}_n) \quad (2.53d)$$

In [135], it is assumed that the origin of inertial frame is located at the geometric center of the landmarks. In this case we have  $\dot{z}_n = 0$  which simplifies the observer

(2.53) to the observer designed in [135]<sup>5</sup>. Compared to [135], the observer (2.53) has the advantage that it is well-defined even if only some of the measurements  $y_i$  are unavailable at some period of time. In this case, the reference output  $\dot{z}_n$  can be recalculated using the reference outputs corresponding to the remaining available measurements. Also, we only require  $A = [a_{ij}]$  to be full rank but [135] necessarily requires that  $a_{ij}$  are chosen such that  $[\dot{z}_1, \dots, \dot{z}_{n-1}][\dot{z}_1, \dots, \dot{z}_{n-1}]^\top = I_{3 \times 3}$ .

## 2.8 Summary

In this chapter, we investigate the problem of observer design for invariant systems on finite-dimensional real connected Lie groups where the measurement of system input is corrupted by an unknown constant bias. We show that the corresponding standard error dynamics are non autonomous in general. We propose an observer design methodology that guarantees the uniform local asymptotic (or exponential) convergence of the observer trajectories to the system trajectories. We employ a gain mapping acting on the differential of a cost function to design the innovation term of the group estimator. The bias estimator is then designed using a Lyapunov method. The notion of homogeneous output spaces is generalized to multiple outputs, each of which is modeled via a right action of the Lie group on an output space. A systematic method for constructing invariant cost functions on Lie groups is proposed, yielding implementable innovation terms for the observer. A verifiable condition on the stabilizer of the reference outputs associated with the output spaces ensures the stability of the observer. This condition is consistent with the observability criterion discussed in [93]. Our proposed method omits the limiting reductivity condition imposed in the Author's previous work [85] and in [92]. As a case study, pose estimation on the Lie group SE(3) was investigated where our observer design methodology unifies the state-of-the-art pose estimators of [135] and [65] into a single framework that applies to any invariant kinematic system on a Lie group. Extension of the proposed observer design methodology to the (co)tangent bundle of a Lie group could be considered by assigning a Lie group structure to the (co)tangent bundle noting that the (co)tangent bundle is trivial (see e.g. [123]).

<sup>5</sup>Here, the position vector  $p$  is expressed in the inertial frame but in [135] the position vector is expressed in the body-fixed frame. One can transform the system of [135] to the form presented here using the change of variable  $p \mapsto Rp$ .



# State Estimation for Systems with Delayed Output Measurements; Observer-predictor approach

---

This chapter proposes a state estimation methodology for invariant systems on Lie groups where outputs of the system are measured with delay. The proposed method is based on a cascade of an observer and a predictor. The observer uses delayed measurements and provides estimates of delayed states. The predictor uses those estimates together with the current inputs of the system to compensate for the delay and to provide a prediction of the current state of the system. We consider three classes of left-invariant, right-invariant, and mixed-invariant systems and propose predictors tailored to each class. The key contribution of the Chapter is to exploit the underlying symmetries of systems to design novel predictors that are computationally simple and generic, in the sense that they can be combined with any stable observer or filter. We provide a rigorous stability analysis demonstrating that the prediction of the current state converges to the current system trajectory if the observer state converges to the delayed system trajectory. The good performance of the proposed approach is demonstrated using a sophisticated Software-In-The-Loop simulator indicating the robustness of the observer-predictor methodology even when large measurement delays are present. This chapter is based on the contributions presented in [82, 84].

### 3.1 Related work

State observers for systems on general Lie groups rely on fusing the measurements of inputs and outputs of systems to estimate their internal states [28, 33, 68, 94, 102, 123, 143]. All of the above mentioned observer design methodologies require delay free measurement of the current outputs and inputs of the system. In many practical scenarios, however, measurement of the outputs of the system are inherently delayed (compared to the ideal outputs of the system) while the inputs are measured without significant delays. For example, in the velocity aided attitude estimation scenario, measurements of linear velocity (output) provided by commercial GPS units are usually delayed with respect to the actual velocity of the vehicle. The delay can be several hundred milliseconds (and even up to half a second) due to various environmental effects and in-sensor processing delays [58, 88]. In contrast, measurements of the vehicle's angular velocity and linear acceleration provided by an onboard IMU are almost instantaneous. Another example is attitude estimation for satellites using star trackers and gyros. The image processing inside a star-tracker sensor can cause significant delays in the order of tens of milliseconds while gyroscope measurements on the satellite are obtained without significant delays. Similar delay problems occur in attitude estimation for aerial robots when vision based sensors compute landmarks. Even in instrumented indoor flight environments, the attitude data from devices such as VICON or OptiTrack are delayed by the communication channel from these sensors to the onboard attitude estimation system of the vehicle. Finally, heavy filtering of noisy sensor measurements before data fusion can introduce significant delays in the filtered data. This is particularly important, for example, for obtaining air velocity measurements from noisy air pressure readings [119]. It is well understood that measurement delays can negatively affect the stability and robustness of observers or filters and degrade their performance [16, 19, 30, 41, 53, 75, 76, 134].

The classical approach to tackle sensor delay in estimation problem is to take an estimator that has the desired performance for delay free measurements, and modify its innovation term such that it compares each delayed measurement with its corresponding backward time-shifted estimate. If the delay-free estimator has a Lyapunov stability proof, the stability analysis for the modified estimator is approached by using Lyapunov-Krasovskii or Lyapunov-Razumikhin functions [16, 18, 24, 41, 134].

---

Although commonly used in practice, such modified estimators require complicated stability analyses and careful and conservative gain tuning which may lead to poor transient responses. Combined observer-predictor design is an alternative method that has received recent interests, see e.g. [19, 30, 53, 75, 76] and the references therein. These methods take observers that use delayed measurements to provide estimates of the delayed state and combine them with appropriate predictors to compensate for the delay. An observer-predictor combination is proposed in [58] for attitude, velocity, and position estimation of flying vehicles and its good performance is verified in practice. The predictor of [58] (which is called a fast simulator in the context of that paper) relies on buffering the IMU measurements and employing a nonlinear observer to approximate the velocity and position kinematics by a double integrator system. This approximation enables forward integrating the buffered IMU data each time a new GPS measurement is received to obtain predictions of the current position and velocity. To our knowledge, there is currently no state estimation methodology (with stability proof) for systems on general Lie groups when output measurements are affected by delay.

In this chapter, we tackle the problem of state estimation for systems on general matrix Lie groups when measurements of system outputs are delayed. We propose an observer-predictor methodology for three classes of systems on Lie groups; left-invariant, right-invariant, and mixed-invariant. Our proposed methodology employs observers that use the delayed output measurements and estimate the delayed state of the system. We propose dynamic predictors that use the delayed estimates from the observers together with the current estimates of the inputs in order to predict the current state of the system (see Fig. 3.1). The key contribution of the Chapter is effective use of the underlying symmetry of systems in order to design the predictors such that the observer-predictor pairs are co-stable meaning that the observer-predictor combination provides (asymptotically/exponentially) stable estimates of the current state if the observer itself provides (asymptotically/exponentially) stable estimates of the delayed state. It turns out that the convergence rate of the combined predictor-observer depends only on the convergence rate of the observer independent of the magnitude of the delay (in contrast to what is common for the predictors for general systems on  $\mathbb{R}^n$  [19, 30, 53, 75, 76, 81]). We ensure separation of the observer design problem from the predictor design problem which gives us the freedom of

employing any stable observer or filter without affecting the proposed co-stability analysis. An advantage of this approach is that the gain tuning process of the observer is independent of the magnitude of the delay, yielding robust performance of the observer-predictor pairs even with large delays. Rather than assuming the availability of current input measurements, we consider a more general case where only estimates of the current inputs of the system are available. This generalization is particularly useful when the input measurements have unknown biases and scaling factors that are adaptively estimated at the observer stage. The proposed predictors are recursive and hence computationally cheap making them ideal for embedded implementation in real-world applications. As an example, we consider the velocity aided attitude estimation problem and we provide realistic numerical simulations using a sophisticated Software-In-The-Loop (SITL) system designed for ArduPilot/APM<sup>1</sup>[2, 5, 7, 11, 14]. Using the SITL simulator, we demonstrate robustness of the observer-predictor approach in practical situations even when large GPS delays are present.

The Chapter is organized as follows. We formulate the problem in Section 3.2. The observer-predictor approach is discussed in Section 3.3 where the proposed predictors are introduced and their co-stability properties are investigated. The relation of the proposed recursive predictors to the non-recursive predictors of [113] is discussed in Section 3.4. Realistic simulation studies in Section 3.5 and brief conclusions in Section 3.6 complete the Chapter.

## 3.2 Problem formulation

We consider three classes of invariant systems on the Lie group  $G$  given by

$$\dot{X}_l(t) = X_l(t)u_l(t), \quad X_l(0) = X_{l_0} \quad (3.1)$$

$$\dot{X}_r(t) = u_r(t)X_r(t), \quad X_r(0) = X_{r_0} \quad (3.2)$$

$$\dot{X}_m(t) = X_m(t)u_l(t) + u_r(t)X_m(t), \quad X_m(0) = X_{m_0}, \quad (3.3)$$

---

<sup>1</sup>ArduPilot is a comprehensive open source autopilot software that is widely used amongst the UAV enthusiast community.

where  $X_l, X_r$  and  $X_m \in G$  are internal states of the systems (3.1), (3.2), and (3.3), respectively, and  $u_l, u_r \in \mathfrak{g}$  are input signals. We drop the subscripts  $l, r$  and  $m$  wherever this is possible without causing confusion. Similar to [28, 94], we call the systems (3.1) and (3.2) left-invariant and right-invariant, respectively. We call the system (3.3) *mixed-invariant* since its vector field is composed of a left-invariant term  $X_m(t)u_l(t)$  and a right-invariant term  $u_r(t)X_m(t)$ , but the combined vector field is in general neither right nor left invariant. We assume that the inputs  $u_l(t)$  and  $u_r(t)$  are admissible in the sense that corresponding solutions for the relevant systems exist for all initial conditions and that these solutions are unique and continuously differentiable for all time. In the context of mechanical systems, equation (3.1) is typically used to model the kinematics when the system input  $u_l$  is measured in the body-fixed reference frame of the system. Similarly, the system model (3.2) is typically employed when the system input  $u_r$  is measured in the inertial frame of reference. In the cases where some of the system inputs are measured in the body-fixed frame while other inputs are measured in the inertial frame, the system model (3.3) might be employed.

Assume that either direct measurements of current system inputs or an estimate of those inputs are available but the measurement of the outputs of the system encounter a delay of  $\tau$  seconds. Such a situation occurs in many applications involving state estimation of mobile robots where inputs of the system are typically angular velocities and accelerations obtained almost delay-free using high rate sensors such as gyros and accelerometers while partial measurements of states (i.e system outputs which are position and orientation in this example) are obtained using magnetometers, cameras, GPS, etc. which are usually subject to considerable amounts of delay. In some practical scenarios direct measurements of inputs are not available but an (asymptotically stable) estimate of those inputs can be obtained using adaptive estimation techniques. The problem we discuss in this chapter is to use the estimates of current inputs together with delayed measurements of outputs in order to estimate the current state of systems of the form (3.1), (3.2), or (3.3), assuming that the amount of the output delay is known.

**Example 3.2.1.** *Recall the attitude estimation problem discussed in Example 2.2.1. In mobile robotics applications, when a vehicle performs low acceleration maneuvers, a 3-axis accelerometer typically employed as an inclinometer to measure the Earth's gravitational di-*

rection for use as a vector measurement of the form (2.6). When a vehicle performs high acceleration maneuvers, however, accelerometer measurements do not accurately match the model (2.6). In this situation, it is common to consider a more accurate model of the accelerometer and enhance the attitude estimation algorithm by employing measurements of the linear velocity of the vehicle. Such algorithms are called velocity aided attitude estimators [20, 55, 62, 64, 107, 120]. To this end, the motion of a flying rigid body in the Earth's gravitational field is described by the following equations [62, 107, 120].

$$\dot{R}(t) = R(t)\Omega(t), \quad R(0) = R_0 \quad (3.4)$$

$$\dot{v}(t) = ge_3 + R(t)a(t), \quad v(0) = v_0 \quad (3.5)$$

where  $R \in SO(3)$  and  $\Omega \in \mathfrak{so}(3)$  are the attitude matrix and the angular velocity as in Example 2.2.1,  $v \in \mathbb{R}^3$  is the linear velocity of body-fixed frame  $\{B\}$  w.r.t. the inertial frame  $\{A\}$  expressed in  $\{A\}$ ,  $a \in \mathbb{R}^3$  is the so-called specific acceleration of the rigid body which represents the sum of all non-gravitational forces applied to the body divided by its mass and is expressed in  $\{B\}$ , and  $ge_3$  is the (constant) gravitational acceleration vector expressed in  $\{A\}$ . The internal states of the dynamical system (3.4)-(3.5) are the attitude matrix  $R(t)$  and the velocity vector  $v(t)$  and its inputs are the angular velocity vector  $\Omega(t)$  and the specific acceleration  $a(t)$ .

Similar to [28], we rewrite the dynamics (3.4)-(3.5) as a mixed-invariant system on the Lie group  $SE(3)$  with inputs living on its Lie algebra  $\mathfrak{se}(3)$ . Defining  $X = \begin{bmatrix} R & v \\ 0 & 1 \end{bmatrix} \in SE(3)$ ,  $u_l = \begin{bmatrix} \Omega & a \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3)$ , and  $u_r = \begin{bmatrix} 0 & ge_3 \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3)$ , it is straight-forward to verify that  $\dot{X}(t) = X(t)u_l(t) + u_r(t)X(t)$  where the group product is simply given by matrix multiplication. If the angular velocity  $\Omega(t)$  and the acceleration  $a(t)$  are directly measured (by a 3-axis gyro and accelerometers), then the inputs  $u_l(t)$  and  $u_r(t)$  would be available and can be used to estimate the state  $X(t)$ . However, in practice, measurements of angular velocity and linear acceleration are sometimes corrupted by unknown biases and scaling factors. As was discussed in Chapter 2, it is common practice to assume that these biases and scaling factors are constant and that they are adaptively estimated with observers [24, 55, 86, 101, 107, 137, 144]. These estimates then can be used to obtain estimates of  $\Omega(t)$  and  $a(t)$ .

*In outdoor environments, a measurement of linear velocity is usually obtained using GPS units attached to the rigid body. Due to the internal processing time of GPS chips, the velocity measurement is usually delayed w.r.t. the actual linear velocity  $v(t)$  of the vehicle. The amount of this delay can be up to hundreds of milliseconds and hence is not negligible. Assume that the GPS velocity measurement is given by  $v_m(t) = v(t - \tau_v)$  where  $\tau_v$  is a known constant delay. A magnetometer is another type of sensor usually employed for attitude estimation. A 3-axis magnetometer measures the magnetic field of the earth in the body-fixed frame. An ideal magnetometer output  $y_b(t)$  is related to the attitude matrix via  $y_b(t) = R(t)^\top \dot{y}_b$  where  $\dot{y}_b$  is the vector of the Earth's magnetic field at the position of the rigid body expressed in  $\{A\}$ . Assume that the magnetometer measurement is given by  $y_b^\tau(t) = R(t - \tau_b)^\top \dot{y}_b$  where  $\tau_b$  is a known constant delay. In practice, the magnetometer delay is much shorter compared to the GPS delay. Defining  $\tau := \max(\tau_v, \tau_b)$  both outputs (i.e. GPS velocity and magnetometer measurements) at time  $t - \tau$  are available at time  $t$ . The velocity aided attitude estimation problem is to use the estimates (or measurements) of current inputs  $\Omega(t)$  and  $a(t)$  together with the delayed GPS velocity and magnetometer measurements in order to estimate the current states  $R(t)$  and  $v(t)$ .  $\square$*

### 3.3 Observer-predictor approach

The approach that we take in this chapter to tackle the problem defined in Section 3.2 is to employ an observer that uses the delayed output measurements and artificially delayed input measurements (denoted by  $u_y(t - \tau)$ ) and provides an estimate of the delayed state  $X(t - \tau)$ , denoted by  $\hat{X}^\tau(t)$ . We then design a predictor that uses the delayed estimate  $\hat{X}^\tau(t)$  together with measurements or estimates of the current input  $u(t)$  (denoted by  $\hat{u}(t)$ ) and provides "predictions" of the current state  $X(t)$  denoted by  $X^p(t)$ . This observer-predictor scheme is illustrated in Fig. 3.1. In the special case where measurements of the true input  $u(t)$  are available, one can replace  $u_y(t - \tau)$  and  $\hat{u}(t)$ , respectively, with  $u(t - \tau)$  and  $u(t)$  in Fig 3.1. Otherwise,  $\hat{u}(t)$  can be provided by an adaptive observer which estimates bias, scaling, and other unknown parameters of the input. This adaptive observer can sometimes be a part of the observer block of Fig. 3.1, in which case  $\hat{u}(t)$  is indeed fed to the predictor from the observer block.

When outputs of the system are measured delay-free, various observer design methodologies, including those presented in Chapter 2, are available that are capa-

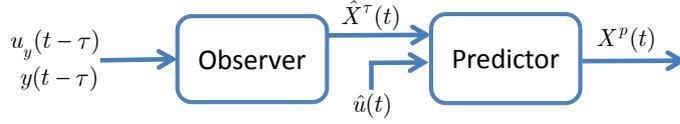


Figure 3.1: Proposed observer-predictor methodology.

ble of asymptotically/exponentially estimating the system state [20, 28, 34, 55, 56, 62, 64, 65, 69, 86, 94, 97, 101, 102, 107, 116, 120, 121, 135, 136, 144]. If a given observer is time-invariant<sup>2</sup>, one can simply feed that observer with delayed outputs and delayed inputs of the system to obtain an estimate of the delayed state  $X(t-\tau)$ . Most observers employed in control design are formulated by a set of ODEs that does not explicitly depend on time. These observers are naturally time-invariant systems and hence the method described above is applicable to obtain an estimate of  $X(t-\tau)$  for the systems (3.1)-(3.3). Hence, in this chapter, we focus only on designing the predictor block of our proposed observer-predictor approach for the case of systems in the form (3.1), (3.2), or (3.3). Roughly speaking, the predictor's job is to use the information provided by the input signal in order to compute how much the state has changed in the period  $t-\tau$  to  $t$  and use the result of this computation together with the estimate of  $X(t-\tau)$  to provide a prediction of the current state  $X(t)$ .

We propose the following predictor for the left-invariant system (3.1)

$$\dot{\Delta}_l(t) = \Delta_l(t)\hat{u}_l(t), \quad \Delta_l(0) = \Delta_{l_0}, \quad (3.6a)$$

$$X_l^p(t) = \hat{X}_l^\tau(t)\Delta_l(t-\tau)^{-1}\Delta_l(t), \quad t \geq \tau \quad (3.6b)$$

where  $\Delta_l(t) \in G$  is the internal state of the predictor,  $\hat{u}_l(t) \in \mathfrak{g}$  is an estimate of the input  $u_l(t)$ , and  $X_l^p(t) \in \mathfrak{g}$  is the prediction of  $X_l(t)$ . If direct measurements of  $u_l(t)$  are available, those measurements can replace  $\hat{u}_l(t)$ . The predictor dynamics (3.6a), which is a copy of the system dynamics (3.1), generates the trajectory of  $\Delta_l(t)$ . This trajectory is stored in a buffer for the period  $[t-\tau, t]$  in order to compute the prediction  $X_l^p(t)$  using the static equation (3.6b). The term  $\Delta_l(t-\tau)^{-1}\Delta_l(t)$  in (3.6b) is a prediction of the "increment" of the state  $X(t)$  from the time  $t-\tau$  to the time  $t$ . The predictor recursively computes this increment using the information provided

<sup>2</sup>Time-invariant is in the sense of considering the system inputs and outputs altogether as the input of the observer and considering the state estimate as the output of the observer.

by the input via (3.6a). Note that this increment (and hence  $X_l^p(t)$ ) is independent of the choice of the initial condition  $\Delta_{l_0}$  of the predictor. Since the predictor dynamics (3.6a) is left-invariant, multiplying its initial condition by an arbitrary  $S \in G$  from the left results in  $(S\Delta_l(t-\tau))^{-1}(S\Delta_l(t)) = \Delta_l(t-\tau)^{-1}\Delta_l(t)$ . This independence from the choice of initial condition is a strong property which prevents aggregation of error due to input measurement noise or numerical integration in practice. This also enables periodic resetting of the initial condition to keep  $\Delta_l(t)$  bounded for all time (see Lemma 3.3.6).

Similar to (3.6a)-(3.6b), we propose the following predictor for the system (3.2)

$$\dot{\Delta}_r(t) = \hat{u}_r(t)\Delta_r(t), \quad \Delta_r(0) = \Delta_{r_0}, \quad (3.7a)$$

$$X_r^p(t) = \Delta_r(t)\Delta_r(t-\tau)^{-1}\hat{X}_r^\tau(t), \quad t \geq \tau \quad (3.7b)$$

where  $\Delta_r(t) \in G$  and  $X_r^p(t) \in G$  are the internal state of the predictor and the prediction of  $X_r(t)$ , respectively, and  $\hat{u}_r(t) \in \mathfrak{g}$  is an estimate of  $u_r(t)$ . Again, we need to buffer  $\Delta_r(t)$  for the period  $[t-\tau, t]$ . Note that this time, the increment of the state is predicted by the term  $\Delta_r(t)\Delta_r(t-\tau)^{-1}$  such that it is compatible with the right-invariant nature of system (3.2). Similar to the left-invariant case, the predictor trajectory  $X_r^p(t)$  is independent of the choice of the initial condition  $\Delta_{r_0}$ .

The internal dynamics of the predictor that we propose for the mixed-invariant system (3.3) consists of both (3.6a) and (3.7a). The trajectories  $\Delta_l(t)$  and  $\Delta_r(t)$  are then buffered for  $[t-\tau, t]$  and are used in the following equation to provide the prediction  $X_m^p(t) \in G$  of  $X_m(t)$ .

$$X_m^p(t) = \Delta_r(t)\Delta_r(t-\tau)^{-1}\hat{X}_m^\tau(t)\Delta_l(t-\tau)^{-1}\Delta_l(t), \quad t \geq \tau \quad (3.8)$$

The term  $\Delta_l(t-\tau)^{-1}\Delta_l(t)$  in (3.8) takes into account the increment of the state  $X_m(t)$  (from  $t-\tau$  to  $t$ ) due to the left-invariant vector field  $X_m(t)u_l(t)$  while the term  $\Delta_r(t)\Delta_r(t-\tau)^{-1}$  takes into account the increment due to the right-invariant vector field  $u_r(t)X_m(t)$ . Note that the effects of the left and the right-invariant parts of the mixed-invariant vector field are combined in (3.8) through multiplicative calculus. Assuming  $u_r = 0$  in (3.1),  $\Delta_r(t)$  is constant for all  $t \geq 0$  and hence  $\Delta_r(t)\Delta_r(t-\tau)^{-1} = I$ . This simplifies the predictor equation (3.8) to (3.6b). This is because  $u_r = 0$  sim-

plifies the mixed-invariant system (3.3) to the left-invariant form (3.1). A similar observation can be made by assuming  $u_l = 0$  and simplifying the mixed-invariant case to a right-invariant form. Similar to the left and right-invariant cases, the predictor trajectory  $X_m^p(t)$  is independent of the initial conditions  $\Delta_{l_0}$  and  $\Delta_{r_0}$ .

Note that all of the above predictors can be implemented *recursively* together with *any* observer algorithm that provides the estimate  $\hat{X}^\tau(t)$ .

In the observer-predictor arrangement, the trajectory of the predicted state  $X^p(t)$  inherently depends on the trajectory of the observer  $X^\tau(t)$ . Hence, rather than discussing the stability of the predictor, we need a notion of stability that relates the stability properties of the predictor to the stability properties of its corresponding observer. The following definition formalizes this notion.

**Definition 3.3.1.** Consider a distance  $d(\cdot, \cdot)$  on  $G$ . With respect to this distance, we say an observer-predictor pair is

- *co-stable* if  $d(\hat{X}^\tau(t), X(t-\tau))$  bounded for all  $t \geq \tau$  yields that  $d(X^p(t), X(t))$  is bounded for all  $t \geq \tau$ .
- *asymptotically co-stable* if it is co-stable and  $d(\hat{X}^\tau(t), X(t-\tau)) \rightarrow 0$  yields  $d(X^p(t), X(t)) \rightarrow 0$ .
- *exponentially co-stable* if it is co-stable and  $d(\hat{X}^\tau(t), X(t-\tau)) \xrightarrow{\text{exp}} 0$  yields  $d(X^p(t), X(t)) \xrightarrow{\text{exp}} 0$ .

We say a predictor is *universally (asymptotically/exponentially) co-stable w.r.t.  $d(\cdot, \cdot)$*  if the combination of that predictor with any (asymptotically/exponentially) stable observer yields a (asymptotically/exponentially) co-stable observer-predictor pair w.r.t.  $d(\cdot, \cdot)$ .  $\square$

The following Theorem summarizes the co-stability properties of the predictor (3.6) (resp. (3.7)) for the left-invariant system (3.1) (resp. right-invariant system (3.2)).

**Theorem 3.3.2.** Consider system (3.1) with the predictor (3.6) (resp. system (3.2) with the predictor (3.7)). Assume that  $\text{cond}(X_l(t))$  and  $\|\hat{u}_l(t) - u_l(t)\|$  (resp.  $\text{cond}(X_r(t))$  and  $\|\hat{u}_r(t) - u_r(t)\|$ ) are bounded for all  $t \geq 0$ . For all  $\tau \geq 0$  and all choices of  $\Delta_{l_0} \in G$  (resp.  $\Delta_{r_0} \in G$ ) we have;

- (a) The predictor is *universally co-stable w.r.t. the distance  $d_r(\cdot, \cdot)$  (resp.  $d_l(\cdot, \cdot)$ )*.

(b) If  $\|\hat{u}_l(t) - u_l(t)\| \rightarrow 0$  (resp.  $\|\hat{u}_r(t) - u_r(t)\| \rightarrow 0$ ) then the predictor is universally asymptotically co-stable w.r.t.  $d_r(\cdot, \cdot)$  (resp.  $d_l(\cdot, \cdot)$ ).

(c) If  $\|\hat{u}_l(t) - u_l(t)\| \xrightarrow{\text{exp}} 0$  (resp.  $\|\hat{u}_r(t) - u_r(t)\| \xrightarrow{\text{exp}} 0$ ), then the predictor is universally exponentially co-stable w.r.t.  $d_r(\cdot, \cdot)$  (resp.  $d_l(\cdot, \cdot)$ ).

□

*Proof:* We prove parts (a)-(c) of Theorem 3.3.2 for the left-invariant system (3.1) with the predictor (3.6a)-(3.6b). The proof for the right-invariant case can be obtained similarly.

Consider the observer error

$$E_l^\tau(t) = \hat{X}_l^\tau(t) X_l(t-\tau)^{-1} \quad (3.9)$$

and the total prediction error

$$E_l^p(t) = X_l^p(t) X_l(t)^{-1}. \quad (3.10)$$

Using (3.6b) and (3.10) we have

$$E_l^p(t) = \hat{X}_l^\tau(t) \Delta_l(t-\tau)^{-1} \Delta_l(t) X_l(t)^{-1} = \hat{X}_l^\tau(t) X_l(t-\tau)^{-1} X_l(t-\tau) \Delta_l(t-\tau)^{-1} \Delta_l(t) X_l(t)^{-1}.$$

Defining  $\Xi_l(t) := X_l(t) \Delta_l(t)^{-1}$  and using (3.9) we have

$$E_l^p(t) = E_l^\tau(t) \Xi_l(t-\tau) \Xi_l(t)^{-1} \quad (3.11)$$

It is easy to verify that the distance  $d_r(\cdot, \cdot)$  is right-invariant such that  $d_r(XZ, YZ) = d_r(X, Y)$  for all  $X, Y, Z \in G$ . Hence, using (3.11) and (3.9) we have

$$\begin{aligned} d_r(X_l^p(t), X_l(t)) &= d_r(E_l^p(t), I) = d_r(E_l^\tau(t) \Xi_l(t-\tau) \Xi_l(t)^{-1}, I) \\ &= d_r(E_l^\tau(t), \Xi_l(t) \Xi_l(t-\tau)^{-1}) \\ &\leq d_r(E_l^\tau(t), I) + d_r(\Xi_l(t) \Xi_l(t-\tau)^{-1}, I) \\ &= d_r(X_l^\tau(t), X_l(t-\tau)) + d_r(\Xi_l(t), \Xi_l(t-\tau)) \end{aligned} \quad (3.12)$$

The above inequality relates the scalar prediction error  $d_r(X_l^p(t), X_l(t))$  to the scalar

observer error  $d_r(X_l^\tau(t), X_l(t-\tau))$  and the virtual error  $d_r(\Xi_l(t), \Xi_l(t-\tau))$  which indirectly depends on the input estimation error  $\|\hat{u}_l(t) - u_l(t)\|$ . Using (3.1) and (3.6a), the dynamics of  $\Xi_l$  is given by  $\dot{\Xi}_l(t) = \dot{X}_l(t)\Delta_l(t)^{-1} - X_l(t)\Delta_l(t)^{-1}\dot{\Delta}_l(t)\Delta_l(t)^{-1} = -[\text{Ad}_{X_l(t)}(\hat{u}_l(t) - u_l(t))]\Xi_l(t)$ . So,

$$\dot{\Xi}_l(t) = u_l^b(t)\Xi_l(t), \quad u_l^b(t) := -\text{Ad}_{X_l(t)}(\hat{u}_l(t) - u_l(t)) \in \mathfrak{g} \quad (3.13)$$

Lemma 6.4.1 in the Appendix summarizes the convergence properties of the error  $d_r(\Xi_l(t), \Xi_l(t-\tau))$  based on eq (3.13). Also, according to Lemma 6.2.1,  $\bar{\sigma}(\text{Ad}_{X_l})$  is bounded if  $\text{cond}(X_l)$  is bounded. Hence, invoking both Lemma 6.2.1 and Lemma 6.4.1, the proof of parts (a)-(c) for the left invariant system directly follows from inequality (3.12).

For the right-invariant system (3.2) with the predictor (3.7), the proof of parts (a)-(c) can be obtained by direct adaptation of the proof of the left-invariant part using the following alternative error definitions.

$$E_r^\tau(t) = X_r(t-\tau)^{-1}\hat{X}_r^\tau(t), \quad (3.14)$$

$$E_r^p(t) = X_r(t)^{-1}X_r^p(t), \quad (3.15)$$

$$E_r^p(t) = \Xi_r(t)^{-1}\Xi_r(t-\tau)E_r^\tau(t), \quad (3.16)$$

where  $\Xi_r(t) := \Delta_r(t)^{-1}X_r(t)$ . ■

In order to discuss the co-stability properties of the predictor (3.6a), (3.7a), (3.8) for the mixed-invariant system (3.3), we rewrite this system as a combination of a right-invariant and a left-invariant system. Consider the systems

$$\dot{Z}_l(t) = Z_l(t)u_l(t), \quad Z_l(0) = Z_{l_0} \in G, \quad (3.17)$$

$$\dot{Z}_r(t) = u_r(t)Z_r(t), \quad Z_r(0) = Z_{r_0} \in G \quad (3.18)$$

where the initial conditions  $Z_{l_0}$  and  $Z_{r_0}$  are chosen such that  $X_{m_0} = Z_{r_0}Z_{l_0}$ . We have  $\frac{d}{dt}(Z_r(t)Z_l(t)) = \dot{Z}_r(t)Z_l(t) + Z_r(t)\dot{Z}_l(t) = u_r(t)Z_r(t)Z_l(t) + Z_r(t)Z_l(t)u_l(t) = X_m(t)u_l(t) + u_r(t)X_m(t)$ . This proves that  $X_m(t) = Z_r(t)Z_l(t)$  for all  $t \geq 0$ . Using this decomposition, the following Theorem summarizes the co-stability properties of the proposed predictor for the mixed-invariant system.

**Theorem 3.3.3.** Consider system (3.3) with the predictor (3.6a), (3.7a), (3.8). Assume that  $\|\hat{u}_l(t) - u_l(t)\|$ ,  $\|\hat{u}_r(t) - u_r(t)\|$ ,  $\text{cond}(Z_r(t))$ ,  $\text{cond}(Z_l(t))$ , and  $\text{cond}(\Delta_r(t)\Delta_r(t-\tau)^{-1})$  (resp.  $\text{cond}(\Delta_l(t-\tau)^{-1}\Delta_l(t))$ ) are bounded for all  $t \geq 0$ . For all  $\tau \geq 0$  and all choices of  $\Delta_{l_0} \in G$  and  $\Delta_{r_0} \in G$  we have;

(a) The predictor is universally co-stable w.r.t. the distance  $d_r(\cdot, \cdot)$  (resp.  $d_l(\cdot, \cdot)$ ).

(b) If  $\|\hat{u}_l(t) - u_l(t)\| \rightarrow 0$  and  $\|\hat{u}_r(t) - u_r(t)\| \rightarrow 0$  then the predictor is universally asymptotically co-stable w.r.t.  $d_r(\cdot, \cdot)$  (resp.  $d_l(\cdot, \cdot)$ ).

(c) If  $\|\hat{u}_l(t) - u_l(t)\| \xrightarrow{\text{exp}} 0$  and  $\|\hat{u}_r(t) - u_r(t)\| \xrightarrow{\text{exp}} 0$ , then the predictor is universally exponentially co-stable w.r.t.  $d_r(\cdot, \cdot)$  (resp.  $d_l(\cdot, \cdot)$ ).  $\square$

*Proof:* We prove the Theorem for the right-invariant distance  $d_r(\cdot, \cdot)$ . The proof for the left-invariant distance follows similarly. Define the following observer and prediction errors for the mixed-invariant system (3.3).

$$E_m^\tau(t) = Z_r(t-\tau)^{-1} \hat{X}_m^\tau(t) Z_l(t-\tau)^{-1}, \quad (3.19)$$

$$E_m^p(t) = Z_r(t)^{-1} X_m^p(t) Z_l(t)^{-1}. \quad (3.20)$$

Using (3.8) and (3.20), we have

$$E_m^p(t) = Z_r(t)^{-1} \Delta_r(t) \Delta_r(t-\tau)^{-1} (Z_r(t-\tau) Z_r(t-\tau)^{-1}) \hat{X}_m^\tau(t) \\ (Z_l(t-\tau)^{-1} Z_l(t-\tau)) \Delta_l(t-\tau)^{-1} \Delta_l(t) Z_l(t)^{-1}.$$

Defining  $\Xi_l^z(t) := Z_l(t) \Delta_l(t)^{-1}$  and  $\Xi_r^z(t) := Z_r(t)^{-1} \Delta_r(t)$  and using (3.19) yields

$$E_m^p(t) = \Xi_r^z(t) \Xi_r^z(t-\tau)^{-1} E_m^\tau(t) \Xi_l^z(t-\tau) \Xi_l^z(t)^{-1} \quad (3.21)$$

Using (3.19), (3.20), (3.21), and Lemma 6.3.1 in the Appendix and dropping the argu-

ment  $t$ , we have

$$\begin{aligned}
d_r(X_m^p, X_m) &= d_r(X_m^p, Z_r Z_l) \\
&\leq \bar{\sigma}(\text{Ad}_{Z_r}) d_r(Z_r^{-1} X_m^p, Z_r^{-1} (Z_r Z_l)) \\
&= \bar{\sigma}(\text{Ad}_{Z_r}) d_r(Z_r^{-1} X_m^p Z_l^{-1}, I) = \bar{\sigma}(\text{Ad}_{Z_r}) d_r(E_m^p, I) \\
&= \bar{\sigma}(\text{Ad}_{Z_r}) d_r(\Xi_r^z \Xi_r^z(t-\tau)^{-1} E_m^r \Xi_l^z(t-\tau) \Xi_l^{z-1}, I) \\
&= \bar{\sigma}(\text{Ad}_{Z_r}) d_r(\Xi_r^z \Xi_r^z(t-\tau)^{-1} E_m^r, \Xi_l^z \Xi_l^z(t-\tau)^{-1}) \\
&\leq \bar{\sigma}(\text{Ad}_{Z_r}) \left( d_r(\Xi_r^z \Xi_r^z(t-\tau)^{-1} E_m^r, I) + d_r(\Xi_l^z \Xi_l^z(t-\tau)^{-1}, I) \right) \\
&= \bar{\sigma}(\text{Ad}_{Z_r}) \left( d_r(\Xi_r^z \Xi_r^z(t-\tau)^{-1} E_m^r \Xi_r^{z-1}, \Xi_r^z(t-\tau) \Xi_r^{z-1}) + d_r(\Xi_l^z \Xi_l^z(t-\tau)^{-1}, I) \right) \\
&\leq \bar{\sigma}(\text{Ad}_{Z_r}) \left( d_r((\Delta_r(t-\tau) \Delta_r^{-1} Z_r)^{-1} \hat{X}_m^r(Z_r(t-\tau) Z_l(t-\tau))^{-1} (\Delta_r(t-\tau) \Delta_r^{-1} Z_r), I) \right. \\
&\quad \left. + d_r(\Xi_r^z(t-\tau) \Xi_r^{z-1}, I) + d_r(\Xi_l^z, \Xi_l^z(t-\tau)) \right) \\
&= \bar{\sigma}(\text{Ad}_{Z_r}) \left( \bar{\sigma}(\text{Ad}_{Z_r^{-1} \Delta_r \Delta_r(t-\tau)^{-1}}) d_r(\hat{X}_m^r X_m(t-\tau)^{-1} \Delta_r(t-\tau) \Delta_r^{-1} Z_r, \Delta_r(t-\tau) \Delta_r^{-1} Z_r) \right. \\
&\quad \left. + d_r(\Xi_r^z(t-\tau) \Xi_r^{z-1}, I) + d_r(\Xi_l^z, \Xi_l^z(t-\tau)) \right) \\
&\leq \bar{\sigma}(\text{Ad}_{Z_r}) \left( \bar{\sigma}(\text{Ad}_{Z_r^{-1} \Delta_r \Delta_r(t-\tau)^{-1}}) d_r(\hat{X}_m^r, X_m(t-\tau)) + d_r(\Xi_r^z(t-\tau), \Xi_r^z) + d_r(\Xi_l^z, \Xi_l^z(t-\tau)) \right)
\end{aligned} \tag{3.22}$$

Similar to (3.13), one can verify that

$$\dot{\Xi}_l^z(t) = u_l^z(t) \Xi_l^z(t), \quad u_l^z(t) := -\text{Ad}_{Z_l(t)}(\hat{u}_l(t) - u_l(t)) \tag{3.23a}$$

$$\dot{\Xi}_r^z(t) = u_r^z(t) \Xi_r^z(t), \quad u_r^z(t) := \text{Ad}_{Z_r(t)^{-1}}(\hat{u}_r(t) - u_r(t)). \tag{3.23b}$$

Lemma 6.4.1 summarizes the stability properties of the scalar errors  $d_r(\Xi_r^z(t-\tau), \Xi_r^z(t))$  and  $d_r(\Xi_l^z(t), \Xi_l^z(t-\tau))$  based on eq. (3.23). Also, using Lemma 6.2.1 in the Appendix we have that if  $\text{cond}(Z_r(t))$ ,  $\text{cond}(Z_l(t))$ , and  $\text{cond}(\Delta_r \Delta_r(t-\tau)^{-1})$  are bounded, then  $\bar{\sigma}(\text{Ad}_{Z_r(t)^{-1}})$ ,  $\bar{\sigma}(\text{Ad}_{Z_l(t)})$ , and  $\bar{\sigma}(\text{Ad}_{Z_r^{-1}(t) \Delta_r(t) \Delta_r(t-\tau)^{-1}})$  are bounded. Hence, invoking both Lemma 6.2.1 and Lemma 6.4.1, proof of parts (a)-(c) for the right-invariant distance  $d_r(\cdot, \cdot)$  follows directly from inequality (3.22).  $\blacksquare$

In the following, we consider a particular case where the measurements of the actual inputs  $u_l(t)$  and  $u_r(t)$  (instead of their estimates  $\hat{u}_l(t)$  and  $\hat{u}_r(t)$ ) are used in the predictors. The following Theorem shows that in this case the total prediction error of the current state exactly equals the observer error of estimating the delayed

state. This shows that the proposed predictors do not change the estimation error behavior of the corresponding observer that feeds the predictor, even though the predictor fully compensates for the effect of sensor delays.

**Theorem 3.3.4.** *Consider system (3.1) with the predictor (3.6), or system (3.2) with the predictor (3.7), or system (3.3) with the predictor (3.6a), (3.7a), (3.8). Assume that  $\hat{u}_l(t) = u_l(t)$  and  $\hat{u}_r(t) = u_r(t)$  for all  $t \geq 0$ . Consider the observer errors (3.9), (3.14) and (3.19) and their corresponding total prediction errors (3.10), (3.15), and (3.20), respectively. We have  $E_l^\tau(t) = E_l^p(t)$ ,  $E_r^\tau(t) = E_r^p(t)$ , and  $E_m^\tau(t) = E_m^p(t)$  for all  $t \geq \tau$ , all  $\tau \geq 0$ , and all choices of  $\Delta_{l_0}, \Delta_{r_0} \in G$ .  $\square$*

*Proof:* The proof follows from the derivations made in the proof of Theorem 3.3.2. According to (3.13), if  $\hat{u}_l(t) = u_l(t)$  then  $\dot{\Xi}_l(t) = 0$  and hence  $\Xi_l(t)$  is constant which in particular implies  $\Xi_l(t) = \Xi_l(t - \tau)$  for all  $t \geq \tau$ . This together with (3.11) yields  $E_l^p(t) = E_l^\tau(t)$  for all  $t \geq \tau$ . A similar argument for the right-invariant case shows that  $\dot{\Xi}_r(t) = 0$  if  $\hat{u}_r(t) = u_r(t)$  and proves that  $E_r^p(t) = E_r^\tau(t)$  for all  $t \geq \tau$ . Also, the same argument is applicable to (3.23) and (3.21) to conclude  $E_m^p(t) = E_m^\tau(t)$  if both  $\hat{u}_l(t) = u_l(t)$  and  $\hat{u}_r(t) = u_r(t)$ .  $\blacksquare$

**Remark 3.3.5.** *Using Theorems 3.3.2 and 3.3.3 and resorting to the proof of Lemma 6.4.1, one can show that the convergence rate of the combined observer-predictor depends only on the convergence rate of the corresponding observer and is independent of the magnitude of the delay. This is straight-forward to see under the conditions of Theorem 3.3.4 where the total estimation error is exactly equal to the observer error alone. This is in contrast to most general predictors designed on  $\mathbb{R}^n$  where the total convergence rate decreases as the delay increases [19, 30, 53, 75, 76, 81].  $\blacksquare$*

The internal dynamics of the predictors (3.6a) and (3.7a) are simple forward integration. One possible drawback of using these pure forward integrators is that their internal states  $\Delta_l(t)$  and  $\Delta_r(t)$  can become larger and larger (due to the input measurement noise or numerical integration inaccuracies) as we continue the integration procedure. One possible way to overcome the above issue is to periodically reset the initial condition of dynamics (3.6a) and (3.7a) to a constant value so that the trajectories of  $\Delta_l(t)$   $\Delta_r(t)$  remain bounded for all times. An arbitrary resetting of the initial conditions of (3.6a) and (3.7a) can potentially destroy the co-stability properties of

the predictors. However, owing to the invariance of the systems (3.6a) and (3.7a), the following Lemma proposes a resetting methodology that keeps the trajectories of the internal states of the predictors bounded while maintaining their co-stability properties.

**Lemma 3.3.6.** *Consider the system*

$$\dot{\Delta}_l^s(t) = \Delta_l^s(t)\hat{u}_l(t), \quad (3.24)$$

and assume that the trajectory  $\Delta_l^s(t)$  is stored in a buffer for the period  $[t-\tau, t]$  (i.e. the last  $\tau$  seconds). Assume that the initial condition of the system (3.24) resets to a fixed value  $\Delta_{l_0}$  every  $T_s$  seconds where  $T_s \geq \tau$  (i.e.  $\Delta_l^s(nT_s) = \Delta_{l_0}$  for  $n = 0, 1, \dots$ ). For  $t \in [nT_s, (n+1)T_s)$ , compute the following variable that depends only on the buffered values of  $\Delta_l^s$ .

$$\Delta_l^b(t) := \begin{cases} \Delta_{l_0}\Delta_l^s(nT_s^-)^{-1}\Delta_l^s(t-\tau), & \text{for all } t \in [nT_s, nT_s + \tau), \\ \Delta_l^s(t-\tau), & \text{for all } t \in [nT_s + \tau, (n+1)T_s) \end{cases} \quad (3.25)$$

Then, we have  $\Delta_l^b(t)^{-1}\Delta_l^s(t) = \Delta_l(t-\tau)^{-1}\Delta_l(t)$  for all  $t \geq \tau$  (i.e. we can replace  $\Delta_l(t-\tau)^{-1}\Delta_l(t)$  with the bounded signal  $\Delta_l^b(t)^{-1}\Delta_l^s(t)$  in (3.6b)). Similarly, consider the system

$$\dot{\Delta}_r^s(t) = \hat{u}_r(t)\Delta_r^s(t), \quad (3.26)$$

and buffer the trajectory  $\Delta_r^s(t)$  for  $[t-\tau, t]$ . Reset the initial condition of the system (3.26) to a fixed value  $\Delta_{r_0}$  every  $T_s \geq \tau$  seconds. For  $t \in [nT_s, (n+1)T_s)$ , compute the following variable.

$$\Delta_r^b(t) := \begin{cases} \Delta_r^s(t-\tau)\Delta_r^s(nT_s^-)^{-1}\Delta_{r_0}, & \text{for all } t \in [nT_s, nT_s + \tau), \\ \Delta_r^s(t-\tau), & \text{for all } t \in [nT_s + \tau, (n+1)T_s) \end{cases} \quad (3.27)$$

Then, we have  $\Delta_r^s(t)\Delta_r^b(t)^{-1} = \Delta_r(t)\Delta_r(t-\tau)^{-1}$  for all  $t \geq \tau$  (i.e. we can replace  $\Delta_r(t)\Delta_r(t-\tau)^{-1}$  with the bounded signal  $\Delta_r^s(t)\Delta_r^b(t)^{-1}$  in (3.7b)).  $\square$

*Proof:* We prove the Lemma for the left-invariant case. The right-invariant case can be proved similarly. Consider the systems (3.6a) and (3.24), both with the same initial condition  $\Delta_{l_0}$ . We have  $\Delta_l^s(t) = \Delta_l(t)$  for all  $t \in [0, T_s)$  and clearly  $\Delta_l^b(t)^{-1}\Delta_l^s(t) = \Delta_l^s(t-\tau)^{-1}\Delta_l^s(t) = \Delta_l(t-\tau)^{-1}\Delta_l(t)$  holds for all  $t \in [\tau, T_s)$ . In

the following, we show that the statement of the Lemma also holds for all  $t \in [nT_s, (n+1)T_s)$ ,  $n = 1, 2, \dots$ . Using (3.6a) and (3.24), we have  $\frac{d}{dt}(\Delta_l^s(t)\Delta_l(t)^{-1}) = \Delta_l^s(t)\hat{u}_l(t)\Delta_l(t)^{-1} - \Delta_l^s(t)\hat{u}_l(t)\Delta_l(t)^{-1} = 0$  for all  $t \in [nT_s, (n+1)T_s)$ . Hence,  $\Delta_l^s(t)\Delta_l(t)^{-1}$  is constant for all  $t \in [nT_s, (n+1)T_s)$  and we have

$$\Delta_l^s(t_1)\Delta_l(t_1)^{-1} = \Delta_l^s(t_2)\Delta_l(t_2)^{-1} \quad \text{for all } t_1, t_2 \in [nT_s, (n+1)T_s). \quad (3.28)$$

Choosing  $t_1 = t \in [nT_s, (n+1)T_s)$  and  $t_2 = nT_s$  we obtain

$$\Delta_l^s(t) = \Delta_l^s(nT_s)\Delta_l(nT_s)^{-1}\Delta_l(t) = \Delta_{l_0}\Delta_l(nT_s^-)^{-1}\Delta_l(t) \quad (3.29)$$

where we used the continuity of  $\Delta_l(t)$  at  $t = nT_s$  to replace  $\Delta_l(nT_s)$  by  $\Delta_l(nT_s^-)$ . One can use the same method as was done to derive (3.28) to conclude  $\Delta_l^s(t_1)\Delta_l(t_1)^{-1} = \Delta_l^s(t_2)\Delta_l(t_2)^{-1}$  for all  $t_1, t_2 \in [(n-1)T_s, nT_s)$ . Choosing  $t_1 = t - \tau \in [nT_s - \tau, nT_s) \subset [(n-1)T_s, nT_s)$  and  $t_2 = nT_s^-$  we obtain

$$\Delta_l^s(t - \tau)^{-1}\Delta_l^s(nT_s^-) = \Delta_l(t - \tau)^{-1}\Delta_l(nT_s^-). \quad (3.30)$$

Using (3.25) and (3.29), for all  $t \in [nT_s, nT_s + \tau)$  we have

$$\Lambda(t)^{-1}\Delta_l^s(t) = \Delta_l^s(t - \tau)^{-1}\Delta_l^s(nT_s^-)\Delta_l(nT_s^-)^{-1}\Delta_l(t) \quad (3.31)$$

Substituting (3.30) into (3.31) yields  $\Delta_l^b(t)^{-1}\Delta_l^s(t) = \Delta_l(t - \tau)^{-1}\Delta_l(t)$  for all  $t \in [nT_s, nT_s + \tau)$ . For  $t \in [nT_s + \tau, (n+1)T_s)$ , we chose  $t_1 = t - \tau \in [nT_s, (n+1)T_s - \tau)$  and  $t_2 = t$  and recall (3.28) to obtain  $\Delta_l^s(t - \tau)^{-1}\Delta_l^s(t) = \Delta_l(t - \tau)^{-1}\Delta_l(t)$ . Using (3.25) for  $t \in [nT_s + \tau, (n+1)T_s)$  we have  $\Delta_l^b(t)^{-1}\Delta_l^s(t) = \Delta_l^s(t - \tau)^{-1}\Delta_l^s(t) = \Delta_l(t - \tau)^{-1}\Delta_l(t)$ . Consequently, the Lemma holds for all  $t \in [nT_s, (n+1)T_s)$ . This completes the proof. ■

Lemma 3.3.6 proposes a resetting technique that keeps the trajectories of the internal states of the predictor bounded while not affecting the trajectory of the prediction  $X^p(t)$ . This implies that the predictors with and without switching are input-output equivalent (considering  $u(t)$  and  $\hat{X}(t)$  as inputs and  $X^p(t)$  as the output of the predictors). Consequently, switching does not affect the co-stability properties of the predictors presented in Theorems 3.3.2, 3.3.3, and 3.3.4. An alternative reset-

ting technique has been proposed in the Author's previous work [82, Lemma 1] to bound the internal states of predictors while maintaining their stability. That method involves employing two copies of the internal dynamics of the predictor and appropriately resetting the initial conditions of those copies and also switching between the trajectories of those copies such that the final predictions (3.6a) and (3.7a) do not change. The method proposed here requires less computational power compared to [82, Lemma 1] since it employs only one copy of the predictor's internal dynamics.

**Remark 3.3.7.** *Theorem 3.3.3 assumes that  $\text{cond}(Z_r(t))$ ,  $\text{cond}(Z_l(t))$ ,  $\text{cond}(\Delta_r(t)\Delta_r(t-\tau)^{-1})$ , and  $\text{cond}(\Delta_l(t-\tau)^{-1}\Delta_l(t))$  are bounded for all  $t \geq 0$ . If  $G$  is a compact group, then these conditions are automatically satisfied. Nevertheless, for general Lie groups these conditions may not hold in general. One way to address this issue is to resort to Lemma 3.3.6 and periodically switch the initial conditions of (3.32), (3.34), (3.17) and (3.18) such that the trajectories of  $\Delta_l(t)$ ,  $\Delta_r(t)$ ,  $Z_l(t)$ , and  $Z_r(t)$  remain bounded for all  $t \geq 0$ . It is possible to show that switching can be done in a way that does not change the trajectory of  $X_m^p(t)$  while the trajectories of the switched  $Z_l(t)$  and  $Z_r(t)$  still satisfy  $X_m(t) = Z_r(t)Z_l(t)$  for all  $t \geq 0$ , and  $\Xi_l^z(t)$  and  $\Xi_r^z(t)$  remain continuous. That is to say, with the resetting scheme of Lemma 3.3.6, we can replace the boundedness conditions of  $\text{cond}(Z_r(t))$ ,  $\text{cond}(Z_l(t))$ ,  $\text{cond}(\Delta_r(t)\Delta_r(t-\tau)^{-1})$ , and  $\text{cond}(\Delta_l(t-\tau)^{-1}\Delta_l(t))$  with  $\text{cond}(X_m(t))$  being bounded and the results of Theorem 3.3.3 still remain valid for general Lie groups.  $\square$*

**Example 3.3.8.** *As was shown in Example 3.2.1, the velocity-aided attitude estimation problem with GPS delay can be formulated as a special case of predictor design for a mixed-invariant system on the Lie group  $SE(3)$ . Hence, one can directly employ the predictor (3.6a), (3.7a), (3.8). Nevertheless, here we show that the specific structure of dynamics (3.4)-(3.5) allows reducing the dimension of the mixed-invariant predictor. We assume that an estimate of  $X(t-\tau) \in SE(3)$  is available and we decompose it into  $\hat{X}^\tau(t) = \begin{bmatrix} \hat{R}^\tau(t) & \hat{\vartheta}^\tau(t) \\ 0 & 1 \end{bmatrix}$*

*where  $\hat{R}^\tau \in SO(3)$  and  $\hat{\vartheta}^\tau \in \mathbb{R}^3$ . Decompose  $\Delta_l$  and  $\Delta_r$  into  $\Delta_l = \begin{bmatrix} \bar{\Delta}_l & \delta_l \\ 0 & 1 \end{bmatrix}$  and*

*$\Delta_r = \begin{bmatrix} \bar{\Delta}_r & \delta_r \\ 0 & 1 \end{bmatrix}$  where  $\bar{\Delta}_l, \bar{\Delta}_r \in SO(3)$  and  $\delta_l, \delta_r \in \mathbb{R}^3$ . We need estimates of the inputs  $u_l(t)$  and  $u_r(t)$  to formulate the predictor. As was shown in Example 3.2.1,  $u_r(t)$  is a known*

constant in this scenario. So, we simply choose  $\hat{u}_r(t) = u_r = \begin{bmatrix} 0 & ge_3 \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3)$ . We assume that estimates of  $\hat{a}(t)$  and  $\hat{\Omega}(t)$  of the current specific acceleration and angular velocity are available and we choose  $\hat{u}_l(t) = \begin{bmatrix} \hat{\Omega}(t) & \hat{a}(t) \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3)$ . If accelerometers and gyros directly measure  $a(t)$  and  $\Omega(t)$ , we use those direct measurements as  $\hat{a}(t)$  and  $\hat{\Omega}(t)$ . Otherwise, if the accelerometers and gyros have constant unknown biases and scaling factors, the observer part of Fig 3.1 estimates those values and provides estimates of  $\hat{a}(t)$  and  $\hat{\Omega}(t)$ . Substituting for  $\Delta_r$  and  $u_r$  in (3.6a) we have

$$\begin{bmatrix} \dot{\bar{\Delta}}_l(t) & \dot{\delta}_l(t) \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \bar{\Delta}_l(t) & \delta_l(t) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\Omega}(t) & \hat{a}(t) \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \bar{\Delta}_l(t)\hat{\Omega}(t) & \bar{\Delta}_l(t)\hat{a}(t) \\ 0 & 0 \end{bmatrix}. \quad (3.32)$$

Also, we have

$$\begin{aligned} \Delta_l(t-\tau)^{-1}\Delta_l(t) &= \begin{bmatrix} \bar{\Delta}_l(t)^\top & -\bar{\Delta}_l(t)^\top\delta_l(t) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{\Delta}_l(t) & \delta_l(t) \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \bar{\Delta}_l(t-\tau)^\top\bar{\Delta}_l(t) & \bar{\Delta}_l(t-\tau)^\top(\delta_l(t) - \delta_l(t-\tau)) \\ 0 & 1 \end{bmatrix} \end{aligned} \quad (3.33)$$

Similarly, substituting for  $\Delta_r$  and  $u_r$  in (3.7a) we have

$$\begin{bmatrix} \dot{\bar{\Delta}}_r(t) & \dot{\delta}_r(t) \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & ge_3 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{\Delta}_r(t) & \delta_r(t) \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & ge_3 \\ 0 & 0 \end{bmatrix}. \quad (3.34)$$

This yields  $\dot{\bar{\Delta}}_r(t) = 0$  and  $\dot{\delta}_r(t) = ge_3$ , implying  $\bar{\Delta}_r(t-\tau) = \bar{\Delta}_r(t)$  and  $\delta_r(t) - \delta_r(t-\tau) = \tau ge_3$ . These together yield

$$\Delta_r(t)\Delta_r(t-\tau)^{-1} = \begin{bmatrix} \bar{\Delta}_r(t) & \delta_r(t) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{\Delta}_r(t-\tau)^\top & -\bar{\Delta}_r(t-\tau)^\top\delta_r(t) \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} I & \tau ge_3 \\ 0 & 1 \end{bmatrix}. \quad (3.35)$$

Since the right hand side of (3.35) is a known constant, we do not actually need to implement the right-invariant part of the predictor (i.e. the dynamics (3.34)). We can use the known

value given by the right hand side of (3.35) to obtain the prediction (3.8). This reduces the dimension of the predictor to the same dimension as the actual system. Decomposing  $X^p$  into  $X^p = \begin{bmatrix} R^p & v^p \\ 0 & 1 \end{bmatrix}$  and replacing (3.33) and (3.35) into (3.8) and dropping the argument  $t$ , we have

$$\begin{aligned} \begin{bmatrix} R^p & v^p \\ 0 & 1 \end{bmatrix} &= \begin{bmatrix} I & \tau g e_3 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{R}^\tau & \hat{v}^\tau \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{\Delta}_l(t-\tau)^\top \bar{\Delta}_l & \bar{\Delta}_l(t-\tau)^\top (\delta_l - \delta_l(t-\tau)) \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \hat{R}^\tau \bar{\Delta}_l(t-\tau)^\top \bar{\Delta}_l & \hat{v}^\tau + \tau g e_3 + \hat{R}^\tau \bar{\Delta}_l(t-\tau)^\top (\delta_l - \delta_l(t-\tau)) \\ 0 & 1 \end{bmatrix} \end{aligned} \quad (3.36)$$

Finally, using (3.32) and (3.36), we simplify the predictor (3.6a), (3.7a), and (3.8) to

$$\dot{\bar{\Delta}}_l(t) = \bar{\Delta}_l(t) \hat{\Omega}(t), \quad \bar{\Delta}_l(0) = \bar{\Delta}_{l_0} \in SO(3), \quad t \geq 0 \quad (3.37)$$

$$\dot{\delta}_l(t) = \bar{\Delta}_l(t) \hat{a}(t), \quad \delta_l(0) = \delta_{l_0} \in \mathbb{R}^3, \quad t \geq 0 \quad (3.38)$$

$$R^p(t) = \hat{R}^\tau(t) \bar{\Delta}_l(t-\tau)^\top \bar{\Delta}_l(t), \quad t \geq \tau \quad (3.39)$$

$$v^p(t) = \hat{v}^\tau(t) + \tau g e_3 + \hat{R}^\tau(t) \bar{\Delta}_l(t-\tau)^\top (\delta_l(t) - \delta_l(t-\tau)), \quad t \geq \tau. \quad (3.40)$$

The attitude predictor (3.37) and (3.39) has exactly the same form as the attitude predictor proposed in the Author's previous work [82, Eq. (6)-(7)], however, the velocity predictor (3.38) and (3.40) has an essentially different form compared to [82, Equ. (8)-(9)]. It is easy to verify that the predictor of [82] is an example of predictor (3.6a)-(3.6b) when the underlying Lie group is  $SO(3) \times \mathbb{R}^3$  and the group multiplication is simply given by  $(R_1, v_1)(R_2, v_2) = (R_1 R_2, v_1 + v_2)$ .

**Remark 3.3.9.** Since  $SO(3)$  is a compact manifold, there is no concern regarding the boundedness of the internal state  $\bar{\Delta}_l(t)$ . However, the state  $\delta_l(t)$  might grow larger and larger as it lives in  $\mathbb{R}^3$ . Hence, we need to employ the resetting technique proposed in Lemma 3.3.6 to bound the trajectory of  $\delta_l(t)$ . If we periodically reset the initial condition of (3.38) to  $\delta_{l_0}$  every  $T_s$  seconds (and do not reset (3.37)), Lemma 3.3.6 still applies since not resetting (3.37) is equivalent to resetting  $\bar{\Delta}_l(t)$  to its current value every  $T_s$  seconds. In this case, (3.25) simplifies to adding  $\delta_{l_0} - \delta_l(nT_s^-)$  to the term  $\delta_l(t - \tau)$  in (3.40) for all  $t \in [nT_s, nT_s + \tau)$ ,  $n = 1, 2, \dots$ , and does not require modifying 3.39. This resetting scheme ensures the co-stability of the predictor as long as  $v(t)$  is bounded (see Remark 3.3.7).  $\square$

### 3.4 Recursive implementation of non-recursive predictors

The predictor design methodologies presented in the previous sections only suit invariant systems on Lie groups. For a general system of the form

$$\dot{X}(t) = F(X(t), u(t)), \quad (3.41)$$

where  $X \in G$  and  $u$  belongs to an input manifold  $\mathcal{M}_u$  and  $F : G \times \mathcal{M}_u \rightarrow TG$ ,  $F(X, u) \in T_X G$ , a simple predictor with desirable co-stability properties was presented in [113]. At time  $t$ , this predictor uses the estimate  $\hat{X}^\tau(t)$  as the initial condition of a copy of the system dynamics at time  $t - \tau$ . Then it forward integrates the system dynamics from  $t - \tau$  to  $t$  using the information of the input  $u(t)$  to obtain a prediction of  $X(t)$ . This procedure is mathematically formulated as follows

$$\frac{d}{ds} X_t^p(s) = F(X_t^p(s), u(s)), \quad s \in [t - \tau, t], \quad (3.42a)$$

$$X_t^p(t - \tau) = \hat{X}^\tau(t), \quad (3.42b)$$

$$X^p(t) = X_t^p(t). \quad (3.42c)$$

where  $X_t^p$  is the internal state of the predictor and  $X^p$  is the prediction of  $X(t)$ . The subscript  $t$  of  $X_t^p$  emphasizes that the predictor (3.42) is *non-recursive* in the sense that at each time  $t$ , the forward integration of dynamics (3.42a) from  $s = t - \tau$  to  $s = t$  should be completely performed in order to compute the final value  $X_t^p(t)$  and obtain the prediction  $X^p(t)$ . This non-recursive nature of predictor (3.42) is in fact its major drawback which prevents its application in practical scenarios involving real-time implementations in on-board computers of robots. Despite this drawback, universally asymptotic and exponential co-stability of this predictor has been shown in [113] for systems on  $\mathbb{R}^n$  and this result can be generalized for systems whose states evolve on differentiable manifolds, including Lie groups.

There is an interesting link between the non-recursive predictor (3.42) and the recursive predictors we proposed in Section 3.3. When the underlying system (3.41) is of the form (3.1), (3.2), or (3.3), the following lemma shows that the resulting predictor (3.42) produces exactly the same prediction trajectory as is produced by the recursive predictors we proposed in this Chapter. In other words, for the case

where the vector field  $F(X, u)$  is left, right, or mixed-invariant, the non-recursive predictor (3.42) can be *recursively* realized using the method we presented in this chapter.

**Proposition 3.4.1.** *Consider the predictor (3.42) for the system (3.41).*

(a) (left resp. right-invariant case) *Assume that the system (3.41) is left-invariant (resp. right-invariant) in the sense that  $F(X, u) = XF_l^g(u)$  (resp.  $F(X, u) = F_r^g(u)X$ ) where  $F_l^g : \mathcal{M}_u \rightarrow \mathfrak{g}$  (resp.  $F_r^g : \mathcal{M}_u \rightarrow \mathfrak{g}$ ) is a known map. Define  $u_l := F_l^g(u)$  (resp.  $u_r := F_r^g(u)$ ), rewrite the system dynamics as  $\dot{X}(t) = X(t)u_l(t)$  (resp.  $\dot{X}(t) = u_r(t)X(t)$ ), and obtain the corresponding predictor (3.6) (resp. (3.7)) for this system assuming that  $\hat{u}_l = u_l$  (resp.  $\hat{u}_r = u_r$ ). If both predictors (3.42) and (3.6) (resp. (3.7)) are fed with the same observer trajectories i.e.  $\hat{X}^\tau(t) = \hat{X}_l^\tau(t)$  (resp.  $\hat{X}^\tau(t) = \hat{X}_r^\tau(t)$ ), then the prediction  $X^p(t)$  of (3.42c) equals  $X_l^p(t)$  of (3.6b) (resp.  $X_r^p(t)$  of (3.7b)) for all  $t \geq \tau$ .*

(b) (mixed-invariant case) *Assume that the system (3.41) is mixed-invariant in the sense that  $F(X, u) = XF_l^g(u) + F_r^g(u)X$  where  $F_l^g, F_r^g : \mathcal{M}_u \rightarrow \mathfrak{g}$  are known maps. Define  $u_l := F_l^g(u)$  and  $u_r := F_r^g(u)$ , rewrite the system dynamics as  $\dot{X}(t) = X(t)u_l(t) + u_r(t)X(t)$ , and obtain the corresponding predictor (3.6a), (3.7a), and (3.8) for this system (assuming  $\hat{u}_l = u_l$  and  $\hat{u}_r = u_r$ ). If both predictors are fed with the same observer trajectories (i.e.  $\hat{X}^\tau(t) = \hat{X}_m^\tau(t)$ ), then the prediction  $X^p(t)$  of (3.42c) equals  $X_r^p(t)$  of (3.8) for all  $t \geq \tau$ .*

□

*Proof:* We prove part (a) for the left-invariant system. Proof for the right-invariant case can be obtained similarly. We suggest that the solution  $X_t^p(s)$ ,  $s \geq t - \tau$  of (3.42a)-(3.42b) is given by  $X_t^p(s) = \hat{X}^\tau(t)\Delta_l(t - \tau)^{-1}\Delta_l(s)$  where  $\frac{d}{ds}\Delta_l(s) = \Delta_l(s)u_l(s)$ . To show this, we note that this solution satisfies the initial condition requirement  $X_t^p(s)|_{s=t-\tau} = \hat{X}^\tau(t)\Delta_l(t - \tau)^{-1}\Delta_l(t - \tau) = \hat{X}^\tau(t)$  and the derivative requirement  $\frac{d}{ds}X_t^p(s) = \hat{X}^\tau(t)\Delta_l(t - \tau)^{-1}\frac{d}{ds}\Delta_l(s) = \hat{X}^\tau(t)\Delta_l(t - \tau)^{-1}\Delta_l(s)u_l(s) = X_t^p(s)u_l(s) = F(X_t^p(s), u_l(s))$ . By (3.42c) we have  $X^p(t) = X_t^p(s)|_{s=t} = \hat{X}^\tau(t)\Delta_l(t - \tau)^{-1}\Delta_l(t)$  which is equal to (3.6b). Proof of part (b) is obtained similarly by verifying that the solution of (3.42a)-(3.42b) is given by  $X_t^p(s) = \Delta_r(s)\Delta_r(t - \tau)^{-1}\hat{X}^\tau(t)\Delta_l(t - \tau)^{-1}\Delta_l(s)$  with  $\frac{d}{ds}\Delta_l(s) = \Delta_l(s)u_l(s)$  and  $\frac{d}{ds}\Delta_r(s) = u_r(s)\Delta_r(s)$  and by evaluating this solution at  $s = t$ . ■

It immediately follows from Proposition 3.4.1 that the proposed recursive predictors are practically stable with respect to input measurement noise; that is any

bounded input measurement noise yields a bounded prediction error [59]. Noting that [113] implies universally asymptotic/exponential co-stability of the non-recursive predictor (3.42), Proposition 3.4.1 provides an alternative proof for asymptotic/exponential co-stability of the recursive predictors proposed in this paper, at least for the case where true input measurements are fed into the predictors. The results of [113] suggest that recursive implementability of the predictor (3.42) is essentially (up to a change of variable) restricted to the systems of the form (3.1), (3.2), or (3.3).

## 3.5 Simulations

A predictor for the velocity-aided attitude estimation problem with GPS delay is proposed by (3.37)-(3.40) in Example 3.3.8. In this section, we provide extensive simulation studies to evaluate the performance of the predictor (3.37)-(3.40) in practical situations where there are inaccuracies such as numerical integration errors, sensor noises, biases, sampling, etc. To discretize the predictor dynamics (3.37), we assume that the angular velocity is approximately constant during each IMU sampling period and simply employ the exponential map (see e.g. [65, Section V.B.]). According to Fig. 3.1, the total observer-predictor estimation error (i.e. the difference between  $X^p(t)$  and  $X(t)$ ) is a combination of the pure error due to the observer (i.e. the difference between  $\hat{X}^\tau(t)$  and  $X(t - \tau)$ ) and the pure error due to the predictor (i.e. the difference between  $X^p(t)$  and  $X(t)$  if the predictor is fed with  $X(t - \tau)$  instead of  $\hat{X}^\tau(t)$ ). We first provide a set of simulations to evaluate the pure predictor error. Then we present realistic simulations using a sophisticated Software-In-The-Loop (SITL) system to demonstrate the total observer-predictor error.

### 3.5.1 Pure prediction error

Using MATLAB<sup>®</sup>, we feed the predictor (3.37)-(3.40) with the true  $R(t - \tau)$  and  $v(t - \tau)$  instead of  $\hat{R}^\tau(t)$  and  $\hat{v}^\tau(t)$ , respectively. We add Gaussian noises with the high standard deviation of 2.5 (deg/s) and 1.5 (m/s<sup>2</sup>) to each axis of the gyro and accelerometer, respectively. We consider various amounts of delay ranging from a small delay of 0.2 (s) to a large delay of 1 (s). We consider a plane that is flying in a circular trajectory with a linear velocity of 22.5 (m/s). In order to demonstrate the

effects of both low acceleration and high acceleration maneuvers, we perform separate simulations with different values for the radius of the plane's flight trajectory (smaller radius implies higher acceleration maneuvers). For each value of GPS delay and each flight radius, we perform 100 simulations and we average the attitude prediction error<sup>3</sup> and the velocity prediction error<sup>4</sup> during each simulation. The IMU sampling time is 20 (ms) and each simulation is performed over a flight time of 100 (s) where the initial condition of (3.38) is reset every 2 seconds. Fig. 3.2 summarizes the results by showing an error bar for the results of each set of 100 simulations. According to Fig. 3.2, the average error due to the predictor is small even with large delays and even though we considered high gyro and accelerometer noises. The error increases as the amount of delay becomes larger. This is because the predictor relies on the forward integration of noisy gyro and accelerometer data from  $t - \tau$  to  $t$  and hence the larger this period is, the more noise aggregates in the final prediction. Also, the errors increase when the flight radius decreases (i.e. when the plane performs higher acceleration maneuvers). The attitude prediction error is much less sensitive to this effect than the velocity prediction error.

### 3.5.2 Total observer-predictor error

Here we aim to provide realistic simulations to demonstrate that the observer-predictor approach is indeed capable of providing good estimates in practical situations. To this end, we use a comprehensive open source SITL system designed mostly by a group of ArduPilot/APM developers and CanberraUAV team [5, 7, 11, 14]. This simulator allows building the ArduPilot/APM<sup>5</sup> autopilot code using an ordinary C++ compiler, making a native executable that allows testing the autopilot code without implementing on an actual hardware. The native executable emulates the hardware of the APM board at the register level, so the key low level hardware drivers (such as gyros, accelerometers, GPS, ADC, etc.) all run in the same way that they would run in a real flight. The SITL consists of three main modules that interact with each other to simulate the whole system. The first module is JSBSim, an open source flight

<sup>3</sup>The attitude error between two rotation matrices  $R_1$  and  $R_2$  is computed using the angle of rotation in the angle-axis representation of the error matrix  $R_1 R_2^\top$  given by  $\frac{180}{\pi} \arccos(1 - 0.5\text{tr}(I - R_1 R_2^\top))$ .

<sup>4</sup>The error between two velocity vectors is simply computed using the Euclidean norm of the difference of two velocity vectors.

<sup>5</sup>An open-source autopilot system that is widely used among the UAV enthusiasts community.

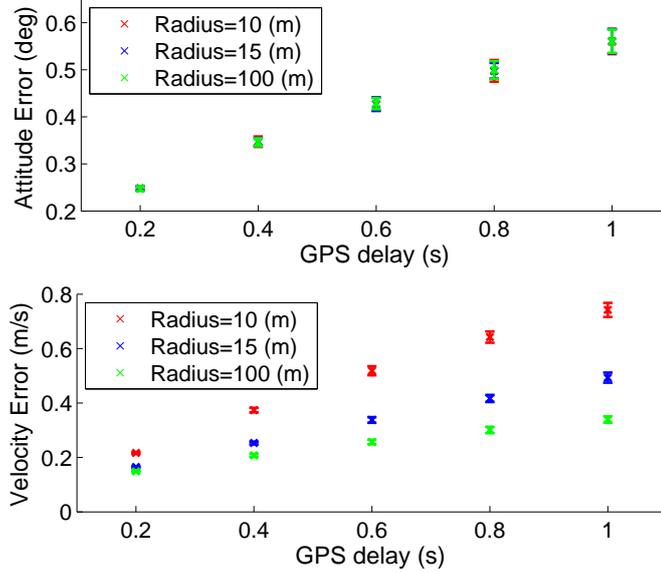


Figure 3.2: Pure attitude and velocity prediction errors. The error bars show two standard deviations. The linear velocity of the plane is 22.5 (m/s).

dynamics model [9] that simulates the trajectory of the vehicle. The second module is the ArduPilot/APM code that emulates the onboard autopilot software of the robot (including the estimation, control, navigation algorithms, etc.). The third module is MAVProxy which is a MAVLink ground station written in python [11]<sup>6</sup>.

For navigation purposes, SITL is capable of emulating various sensor measurements including GPS (longitude/latitude/linear velocity vector), accelerometers, gyros, etc. We setup the SITL parameters such that the GPS measurements are delayed. Here, we aim to demonstrate the performance of the observer-predictor approach when ArduPilot’s native EKF [2] is used as an observer that takes the delayed measurements and provides the estimates  $\hat{R}^\tau(t)$  and  $\hat{v}^\tau(t)$  to the predictor (3.37)-(3.40)<sup>7</sup>. We take the estimates of gyro and accelerometer bias from the observer and use them along with the current gyro and accelerometer measurements to obtain estimates of current inputs  $\hat{\Omega}(t)$  and  $\hat{a}(t)$  fed into the predictor. We setup SITL to use ArduPlane (a fixed wing plane simulator) which tries to follow a desired square path of about 700 (m) by 200 (m) with the desired linear velocity of about 22.5 (m/s). Except for the GPS delay, all other simulation parameters (including the sensor sampling rates,

<sup>6</sup>See Sections 4.6 and 4.6.1 for more explanations about the ArduPilot system.

<sup>7</sup>For the simulation studies of this chapter, we omit the delay compensation parts of the ArduPilot’s native EKF and turn it into a standard EKF before we combine it with the proposed predictor.

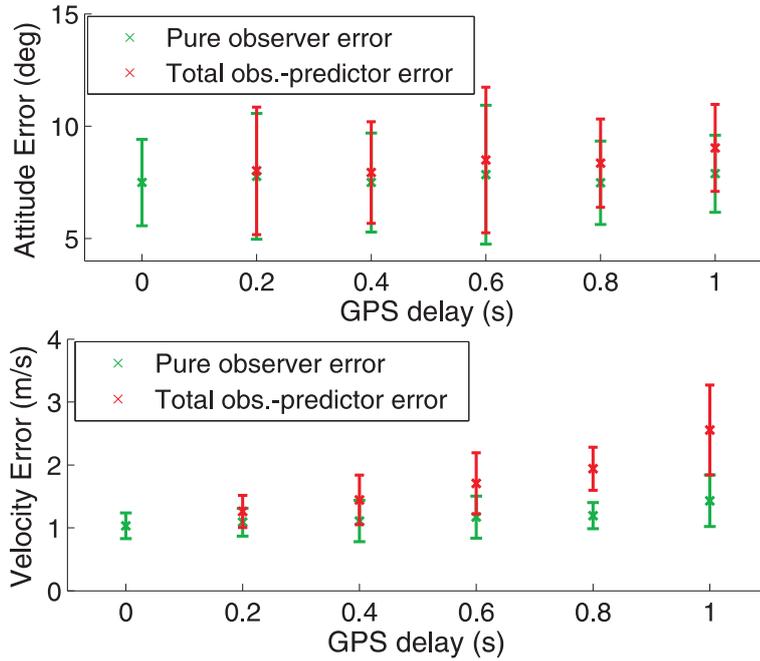


Figure 3.3: Pure observer error and total observer-predictor error. The error bars show two standard deviations.

noises, biases, EKF gain matrices, etc.) are set to the SITL's default values which correspond to typical low cost sensor suites. The flight time for each simulation is about 120 to 140 (s) during which the IMU measurements are fused every 20 (ms) and the initial condition of (3.38) is reset every 2 seconds according to Remark 3.3.9. We choose various amounts of GPS delays and for each amount, we perform the simulation 20 times where in each simulation the plane's path is slightly different from other simulations<sup>8</sup>. We average the error trajectory during each flight simulation and then we compute the mean and standard deviation of all 20 flights. Repeating this procedure for various amounts of GPS delay, Fig. 3.3 summarizes the results by showing an error bar for the results of each set of 20 simulations. This figure shows that the observer-predictor remains stable even for large amounts of delay, which demonstrates the robustness of the proposed approach. A large portion of the total observer-predictor error is due to the observer itself. The average pure error due to the predictor is roughly the difference between the means of the red and the green plots of Fig. 3.3. This difference increases as the amount of delay increases,

<sup>8</sup>This is because the ArduPilot system implements closed-loop control based on noisy data

---

which is compatible to the results of Fig. 3.2<sup>9</sup>. Nevertheless, even when large GPS delay is present, the total error is within a very reasonable range for navigation and control using low cost sensor suites. It is worth noting that alternative estimation methods that either do not compensate for the delay or compensate with Lyapunov-Krasovskii terms perform poorly (or even become unstable) with large delays while the proposed observer-predictor approach remains stable for all amounts of delay, demonstrating the robustness of the proposed approach.

### 3.6 Summary

In this chapter, we consider a state estimation problem for invariant systems on Lie groups where measurements of the outputs of the system are delayed. Given an observer or filter that has the desired stability properties when the system outputs are delay-free, we propose an estimation methodology that preserves those stability properties when the system outputs are delayed. The proposed approach relies on combining the observer with a predictor that compensates for the delay. The delayed measurements are fused in the observer to obtain estimates of the delayed state. Those delayed estimates are then fed into the predictor to obtain the prediction of the current state. We employ invariance of the underlying systems together with the Lie group structure of the state space in order to design recursive predictors that are computationally simple and demonstrate strong co-stability properties ensuring that the observer-predictor combination preserves the stability properties of the observer. Using a sophisticated Software-In-The-Loop simulator, we demonstrate the robustness of the proposed predictors in practical situations even when large sensor delay is present.

---

<sup>9</sup>The pure predictor error of Fig. 3.3 is larger than Fig. 3.2 since the observer's estimation error of gyro and accelerometer biases (in addition to the input noise and plane's acceleration) contribute to the prediction error.



---

# State Estimation for Systems with Sampled Output Measurements; Predictor-observer approach

---

In this chapter, we investigate the problem of state estimation for invariant systems on Lie groups where the output measurements are sampled and are prone to delay effects. We propose a predictor-observer approach in which the predictor takes the delayed and sampled output measurements and provides predictions of the current output. The predicted outputs can then be fed into any arbitrary observer or filter to estimate the current states. Preliminary results of this chapter is presented in [83], though we extensively rely on the theory presented in Chapter 3 to extend those results. We demonstrate the advantages of the predictor-observer approach over traditional Lyapunov-Krasovskii methods via MATLAB simulations. We present an open source C++ library of the proposed predictor for attitude, velocity, and position prediction and we provide experimental test results verifying the performance of the predictor by implementation on the autopilot system of a model plane.

## 4.1 Related work

Invariant systems on Lie groups, their importance, and their applications for state estimation and control of mechanical systems are discussed in Chapter 1. In applications involving state estimation of mechanical systems, measurements of the input are usually obtained almost instantaneously at a very high sampling rate either through odometry or via inertial sensors. In many applications, however, measure-

ments of the system outputs are obtained at much lower sampling rate compared to the sampling rate of system inputs. Output samples are also usually available to the user with some delay due to various reasons including physical properties of sensors (e.g. slow transients) or the environment, internal signal processing of sensors, extensive filtering of sensor measurements for noise reduction, and communication delays from sensors to processing units. For example, in satellite attitude estimation applications, high accuracy output sensors such as star trackers or earth sensors provide measurements at low sampling rates (0.5 to 10 Hz) [105]. In contrast, the onboard gyroscope can easily provide high bandwidth measurements at kHz rates, potentially two orders of magnitude faster than the direction information is obtained. The image processing inside a star-tracker sensor can cause significant delays in the order of tens of milliseconds, leading to the star-tracker measurement being delayed with respect to the gyroscope measurements. Similar sampling and delay problems also occur in attitude and position estimation for aerial robots when vision based sensors such as cameras and landmarks are employed. Also, in indoor flight environments, the attitude data from devices such as VICON or OptiTrack are delayed by the communication channel from these sensors to the onboard attitude estimation system of the vehicle. In outdoor environments, low cost GPS units provide measurements of position and velocity at around 5 Hz which is much lower than the rate of input measurements provide by IMUs (gyros and accelerometers) [58, 119, 132]. GPS measurements are also usually delayed by up to hundreds of milliseconds largely due to the internal processing time of GPS chips.

Sensor sampling and delays can negatively affect the stability and robustness of any observer or filter and degrade their performance if they are not compensated for properly [19, 23, 47, 57, 60, 75, 77]. Typical estimator design methodologies to tackle the measurement sampling and delay problem are; estimator design with Lyapunov-Krasovskii modification, stochastic filtering with Out-Of-Sequence Measurements (OOSM), and compound observer-predictor design. The classical approach to tackle the sensor delay is to take an estimator that has the desired performance for delay free measurements, and modify its innovation term such that it compares each delayed measurement with its corresponding backward time-shifted estimate. If the delay-free estimator has a Lyapunov stability proof, the stability analysis for the modified estimator can be undertaken using Lyapunov-Krasovskii

---

functions [18, 24, 125]. Although these modified estimators are commonly used in practice (see e.g. [2]), they require complicated stability analyses and careful and conservative gain tuning, leading to poor transient responses of the resulting estimators. Stochastic filtering with OOSM has been extensively studied [26, 27, 31, 103, 145], albeit most of this literature focuses on target tracking applications. Although OOSM filtering approaches are flexible, easily dealing with sampled and delayed data as well as out-of-sequence measurements, they usually have significant memory and processing requirements that are unrealistic for most embedded observer design applications, except for linear system models where simpler OOSM filters are available [27, 77]. For the specific problem of attitude estimation with sampled and delayed measurements, a modified extended Kalman filter with a novel real time implementation architecture is proposed in [88]. Despite its good performance in practice, this algorithm suffers from major drawbacks such as unclear convergence properties and high computational load due to the required propagation stages associated with sensor delay compensation. An alternative method is used in [58] for attitude, velocity, and position estimation of flying vehicles and its good performance is verified in practice. This method uses a corrector-predictor representation of a nonlinear observer [52] to handle the sensor sampling and benefits from an predictor to compensate for the sensor delay. The predictor of [58] (which is called a fast simulator in the context of that paper) relies on buffering the IMU measurements and employing a nonlinear observer to approximate the velocity and position kinematics by a double integrator system. This approximation enables employing the state transition matrix of the double integrator system to obtain predictions of the current position and velocity by forward integrating the buffered IMU data each time a new GPS measurement is received. Combined observer-predictor design methods for general nonlinear systems on  $\mathbb{R}^n$  have been developed in [19, 53, 75]. These methods take observers that have the desired stability properties for continuous delay-free measurements and combine them with appropriate predictors that compensate for the effects of sensor sampling and delays, such that the combined observer-predictor maintains the stability properties of the observer. In Chapter 3, we proposed a cascade observer-predictor approach to handle output measurement delay for invariant systems on general Lie groups. The proposed predictor is capable of predicting current states recursively without needing to reprocess the whole stack of buffered

sensor measurements at each time. Although the resulting observer-predictor combination is stable, this method requires continuous availability of sensor outputs and is not directly applicable to the sampled measurement case. Moreover, the observer part of this approach should be fed with delayed input and output measurements. This requires buffering of all of the sensor measurements up to the largest amount of delay amongst all of the sensors. Hence, large memory should be allocated to this buffering. In addition, it is not usually straightforward in practice to modify the time horizon off-the-shelf observers developed for industrial systems such that they provide an estimate of the delayed state required for the observer part of the observer-predictor. This is because industrial observers usually include additional code such as consistency checks and fail safe modes that relies on the assumption that the observer estimates correspond to the current state. Consequently, modifying the time horizon of those observers requires implementing compatible modifications to the additional code as well. To our knowledge, there is no recursive state estimation methodology (with stability proof) for systems on general Lie groups available that considers both sampled and delayed measurements.

In this chapter, we consider the state estimation problem for invariant systems on Lie groups where sampled and delayed output measurements are available. We propose a cascade predictor-observer approach in which the predictor takes the sampled and delayed output measurements and provides predictions of the current outputs. The predicted outputs are then fed into an observer or filter to estimate the system states. Although we make use of the predictors designed in Chapter 3 to develop parts of the theory in this chapter, the approach that we take here to employ the predictors is essentially different from the one proposed in Chapter 3 in that we predict system outputs in this chapter (rather than system states). The main contribution of this chapter is to effectively employ the symmetries of the underlying system dynamics and the output maps to predict the current outputs of the system in a computationally cheap way. The predicted outputs can then be fed into any observer or filter that has asymptotically stable estimation error in ideal conditions (i.e. when it is fed with continuous time delay-free output measurements) and the predictor-observer combination maintains those stability properties when delay and sampling effects do exist. We assume that the delay in each output measurement is known, that is we require accurate time-stamping of data, however, this is the

---

only condition on the data. Given this assumption, the gain tuning process and the stability of the observer is independent of the size of the delay and valid even for time varying delays or out-of-sequence measurements. The proposed approach directly extends to the multi-rate measurement case without further modification. Via a MATLAB simulation example, we show that our predictor-observer method performs significantly better than approaches based on Lyapunov-Krasowskii modifications. Through a collaboration with the developers of the ArduPilot system, we implement the developed predictor as a C++ library and verify its performance in practice. Our experimental test results indicate that combining the predictors with the original navigation algorithm of the ArduPilot system does indeed increase the robustness of state estimation against gain tuning, making it suitable for a wider range of sensor sets and environmental conditions.

The structure of this chapter is as follows. A simple framework for modeling sensor sampling and delays is given in Section 4.2. The problem formulation and motivating examples are given in Section 4.3. The proposed predictor-observer approach is described in Section 4.4 where the main result of this chapter is given by Theorem 4.4.1. The performance of our method is demonstrated via MATLAB simulations in Section 4.5. Section 4.6 is devoted to the predictor designed for the ArduPilot system and to the experimental test results.

## **4.2 Sensor modeling with samples and delays**

In this section, we describe two ways for modeling the sampling and delay effects on sensor measurements. The first model is inspired by physical sources of delays in real sensors, while the second model is a simplified model which is input-to-output equivalent to the first model while it is simpler to use in design and analysis.

### **4.2.1 Physically inspired modeling of sampling and delays**

We propose the model illustrated in Fig. 4.1 to include the effect of sampling and delays on the output of sensors. This model is inspired by the physical process that takes place in sensors during measuring a physical quantity. This model consists of a zero-order-hold (ZOH) block that models the effect of sampling and two delay blocks before and after the ZOH that, respectively, model the pre-sampling and

post-sampling delays. The pre-sampling delay on the left side of Fig. 4.1 models  $\rho_i$  seconds of delay from when the physical quantity  $y_i(t)$  occurs to when it is observed by the  $i$ -th sensor. We have  $y_i^o(t) = y_i(t - \rho_i)$  for all  $t$ . In practice, this delay is usually due to the physical properties of the environment or the sensors. For instance, a star tracker requires that its imaging sensor is exposed to light from stars for a specific amount of time so that it can produce an image of the stars. This is known as exposure time and can be as large as hundreds of milliseconds [129]. The ZOH block in Fig. 4.1 takes the delayed signal and produces a sample at time  $t_{k_i}$ . This sample is latched at the output of ZOH until the next sample is taken at time  $t_{k_i+1}$ . Hence we have  $z_i^o(t) = y_i^o(t_{k_i}) = y_i(t_{k_i} - \rho_i)$  for  $t \in [t_{k_i}, t_{k_i+1})$ . For clarity in presentation, we assume that the sequence  $(t_{k_i})_{k_i=1}^{\infty}$  is an ordered monotonically non-decreasing sequence, i.e.  $t_{k_i-1} \leq t_{k_i} \leq t_{k_i+1}$ . However, this assumption is not necessary for our proposed method and our method is also applicable to the case where the measurements are out-of-sequence, although the necessary modifications to the notation are rather cumbersome. For a star tracker, the sequence  $(t_{k_i})_{k_i=1}^{\infty}$  corresponds to the specific times when the star tracker obtains an image of stars. This sampling frequency can be as low as only 0.5 Hz up to 10 Hz for practical star trackers. The post-sampling delay on the right side of Fig. 4.1 models  $\sigma_i$  seconds of delay from when a sample of the physical variable becomes available to the sensor to the time when the new output  $z_i(t)$  becomes available to the user. We have

$$z_i(t) = z_i^o(t - \sigma_i) = y_i(t_{k_i} - \rho_i), \quad t \in [t_{k_i} + \sigma_i, t_{k_i+1} + \sigma_i) \quad (4.1)$$

In practice, the post-sampling delay models the delay due to the internal signal processing of sensors or due to the communication delay for transmitting information from the sensor to the user. For a star tracker, the post-sampling delay is mainly due to the processing time associated with image processing algorithms that analyze the images taken by the star tracker to recognize stars in the image and associate each recognized star with its corresponding star in the on-board star catalog. The post-sampling delay can also model the lag due to the communication delay from VICON or OptiTrack systems to the onboard attitude estimation system of flying vehicles in indoor flight environments. It also can model the measurement delay associated with internal processing in GPS modules, which in practice is the main source of



Figure 4.1: Modelling the effect of sampling and delays in attitude sensors

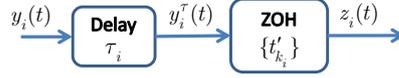


Figure 4.2: Simplified input-to-output equivalent model of Fig. 4.1

delay affecting GPS measurements.

#### 4.2.2 An input-to-output equivalent model

Although the model illustrated in Fig. 4.1 is suitable at the modeling stage to carefully describe the effect of sampling and various delays, it might not be convenient to be employed for compensation of delays and sampling effects at the design stage. The main disadvantage of this model is that even if the user knows the value of  $\sigma_i$ , the value of  $t_{k_i}$  is only available from the time  $t_{k_i} + \sigma_i$  onwards. That is, the sensors usually do not inform the user when exactly they obtain samples of measurements. Instead, they inform the user when they finish processing the samples and the processing result (i.e. the sensor output) is ready to be collected by the user. This is why we discuss a simpler model here which is input-to-output equivalent to Fig. 4.1, but is more convenient to use for compensation of the effect of sampling and delays. Later on in this section, we discuss the condition under which these two models are equivalent.

Assume that at time  $t'_{k_i}$ ,  $k_i = 1, 2, \dots$ , we receive the most recent output of the  $i$ -th sensor denoted by  $z_i(t'_{k_i})$ . We assume that this output is delayed  $\tau_i$  seconds with respect to the measured physical quantity  $y_i(t'_{k_i})$ . This output is latched until the next output arrives at  $t'_{k_{i+1}}$ . This procedure is equivalent to a cascade combination of a delay operator and a ZOH, as sketched in Fig. 4.2. We have

$$z_i(t) = z_i(t'_{k_i}) = y_i(t'_{k_i} - \tau_i), \quad t \in [t'_{k_i}, t'_{k_{i+1}}). \quad (4.2)$$

Each sequence  $(t'_{k_i})_{k_i=1}^{\infty}$  is ordered monotonically non-decreasing (again, this assumption is not necessary for our method but it is imposed for clarity in presentation). The

main difference compared to the model discussed in Section 4.2.1 is that here the sequence  $(t'_{k_i})$  is known to the user and can be used for compensating the effect of sampling and delays. It is obvious that the outputs of Fig. 4.1 and Fig. 4.2 can be different in general. However, the following simple calculations show that both models are input-to-output equivalent. If the output  $z_i(t)$  of both (4.1) and (4.2) are the same for all  $t$ , then we have  $y_i(t_{k_i} - \rho_i) = y_i(t'_{k_i} - \tau_i)$  and  $[t_{k_i} + \sigma_i, t_{k_i+1} + \sigma_i) = [t'_{k_i}, t'_{k_i+1})$ . These equalities hold if and only if  $\tau_i = \rho_i + \sigma_i$  and  $(t'_{k_i})_{k_i=1}^\infty = (t_{k_i} + \sigma_i)_{k_i=1}^\infty$ . That is the output  $z_i(t)$  of both models are equal for all  $t$ , where both models measure the same physical quantity  $y_i(t)$ , if and only if  $\tau_i = \rho_i + \sigma_i$  and  $(t'_{k_i})_{k_i=1}^\infty = (t_{k_i} + \sigma_i)_{k_i=1}^\infty$ . The delay  $\tau_i$  in Fig. 4.2 represents the combination of the delays  $\rho_i$  and  $\sigma_i$  of Fig. 4.1 and the sampling sequence  $(t'_{k_i})_{k_i=1}^\infty$  in Fig. 4.1 is equivalent to the sequence of times by which the user receives the outputs  $z_i(t)$  of Fig. 4.2. Given a model in the form of Fig. 4.1, we are always able to simplify that model to the form of Fig. 4.2 by proper choice of the delay and sampling sequence in Fig. 4.2. This in particular means that, as far as the input-to-output characteristics of the sensors are concerned, there is no need to separately know the value of the pre-sampling and post-sampling delays. In fact, as we show in Section 4.4, only the knowledge of the total delay  $\tau_i$  between the physical value  $y_i(t)$  and the sensor output suffice to reproduce the physical quantity  $y_i(t)$  from the sampled and delayed sensor output.

### 4.3 Problem formulation

Consider the left-invariant, right-invariant, and mixed-invariant systems given by (3.1), (3.2), and (3.3), respectively. Assume that the output measurement models corresponding to the systems (3.1) and (3.2) are, respectively, given by

$$y_{r_i} = h_{r_i}(X_{l_i}, \hat{y}_{r_i}), \quad (4.3)$$

$$y_{l_i} = h_{l_i}(X_{r_i}, \hat{y}_{l_i}), \quad (4.4)$$

where  $\hat{y}_{r_i} \in M_{r_i}$  (resp.  $\hat{y}_{l_i} \in M_{l_i}$ ) is the constant<sup>1</sup> reference output associated with  $y_{r_i}$  (resp.  $y_{l_i}$ ) and  $h_{r_i}$  (resp.  $h_{l_i}$ ) is a right (resp. left) action of  $G$  on the homogeneous output manifold  $M_{r_i}$  (resp.  $M_{l_i}$ ), i.e.  $h_{r_i}(I, y_{r_i}) = y_{r_i}$  and  $h_{r_i}(XS, y_{r_i}) = h_{r_i}(S, h_{r_i}(X, y_{r_i}))$

<sup>1</sup>Constant with respect to time.

(resp.  $h_{l_i}(I, y_{l_i}) = y_{l_i}$  and  $h_{l_i}(XS, y_{l_i}) = h_{l_i}(S, h_{l_i}(X, y_{l_i}))$ ) for all  $y_{r_i} \in M_{r_i}$  (resp.  $y_{l_i} \in M_{l_i}$ ) and all  $X, S \in G$ . Note that we consider the left-invariant system with right output actions or right-invariant systems with left output actions. These types of symmetries correspond to the type II symmetry discussed in [102] and are known to be beneficial for observer design and estimation purpose. Assume that the outputs of system (3.3) are such that they can be modeled by both right and left actions. That is, for a given output  $y_{m_i}$  belonging to the output manifold  $M_{m_i}$ , there exist a reference output  $\mathring{y}_{m_i} \in M_{m_i}$  and right and left actions  $h_r$  and  $h_l$  such that the output  $y_{l_i}$  is modeled by

$$y_{m_i} = h_{r_i}(X_m, \mathring{y}_{m_i}) = h_{l_i}(X_r, \mathring{y}_{m_i}), \quad (4.5)$$

for all  $X_m \in G$ . The output model (4.5) characterizes a special class of measurements. We will provide an example of a physical system with such an output model later (see Example 4.3.2).

The sampled and delayed output measurements corresponding to (4.3)-(4.4) are, respectively, given by

$$z_{r_i}(t) = y_{r_i}(t'_{k_i} - \tau_i), \quad t \in [t'_{k_i}, t'_{k_i+1}), \quad (4.6)$$

$$z_{l_i}(t) = y_{l_i}(t'_{k_i} - \tau_i), \quad t \in [t'_{k_i}, t'_{k_i+1}), \quad (4.7)$$

$$z_{m_i}(t) = y_{m_i}(t'_{k_i} - \tau_i), \quad t \in [t'_{k_i}, t'_{k_i+1}), \quad (4.8)$$

where  $\tau_i \geq 0$  is the delay corresponding to the  $i$ -th output measurement and  $\{t'_{k_i}\}_{k=1}^{\infty}$  is its corresponding measurement sequence. We drop the subscript  $i$  if we only have one output.

The problem at hand is to design an estimation methodology that uses the continuous measurements of the input  $u_l(t)$  and  $u_r(t)$  of the systems (3.1), (3.2), and (3.3) together with their corresponding sampled and delayed output measurements  $z_{r_i}(t)$ ,  $z_{l_i}(t)$ , and  $z_{m_i}(t)$  and provides continuous estimates of the system states  $X_l(t)$ ,  $X_r(t)$ , and  $X_m(t)$ .

**Example 4.3.1** (Attitude estimation problem with vector measurements). *Recall the attitude estimation problem discussed in Example 2.2.1. The attitude kinematics (3.4) belongs to the class of left-invariant systems (3.1) on the Lie group  $SO(3)$ . In practice, the angular*

velocity  $\Omega(t)$  is measured at a high sampling rate. Partial attitude information is obtained in the form of vector measurements (2.6) which belongs to the class of output models (4.3). One can replace  $y_i(t)$  from (2.6) into (4.6) to obtain the sampled and delayed vector measurement model  $z_i(t)$  corresponding to Fig. 4.2. The problem at hand is to use the sampled and delayed vector measurements together with the continuous measurement of the angular velocity to estimate the attitude of the vehicle.  $\square$

**Example 4.3.2** (Velocity aided attitude estimation). We recall the velocity aided attitude estimation problem discussed in Example 3.2.1 where the underlying system dynamics are given by (3.4)-(3.5) as a mixed-invariant system of the form (3.3) on the Lie group  $SE(3)$ . The system inputs  $\Omega(t)$  and  $a(t)$  are measured at a very high sampling rate using accelerometers and gyros. In indoor flight environments, measurement samples of attitude and velocity are obtained using a VICON system and are passed to the onboard autopilot system of the flying vehicle via a communication link. These samples are received by the autopilot system with some delay mostly due to the communication link (but also due to the processing time of the VICON system). Both the sampling rate at which the measurements are received by the onboard autopilot system and the delays associated with measurements are time-varying. Nevertheless, it is possible to accurately compute the sampling sequence and the corresponding delays if the VICON measurements are carefully time-stamped before passing to the vehicle through the communication link (this requires synchronization of the clock of the onboard autopilot system with the VICON ground station). In this case, the underlying measurement model is simply

$$Y_m(X(t)) = X(t) = \begin{bmatrix} R(t) & v(t) \\ 0 & 1 \end{bmatrix}. \quad (4.9)$$

Assuming the reference output  $\dot{Y}_m = y_m(I) = I$ , the output model (4.9) can be thought of as either the right action of the Lie group  $SE(3)$  on itself via  $h_r(X, \dot{Y}_m) = \dot{Y}_m X$  or a left action  $h_l(X, \dot{Y}_m) = X \dot{Y}_m$ . Hence, the output model (4.9) is an example that satisfies the condition (4.5). The sampled and delayed measurements received by the onboard autopilot system are modeled by (4.8). Denote the sampled and delayed attitude and velocity measurements by  $R_z(t) = R(t'_k - \tau)$  and  $v_z(t) = v(t'_k - \tau)$  for all  $t \in [t'_k, t'_{k+1})$ , respectively. The output measurements model is then given by  $Z_m(t) = \begin{bmatrix} R_z(t) & v_z(t) \\ 0 & 1 \end{bmatrix}$ . The estimation problem

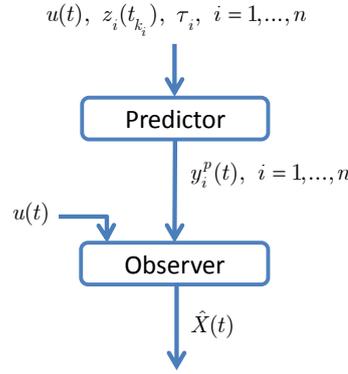


Figure 4.3: Illustration of the proposed predictor-observer approach.

is to use the sampled and delayed measurements  $Z_m(t)$  together with the continuous measurements of the inputs  $(\Omega(t), a(t))$  to obtain continuous estimates of the system states (i.e. attitude and velocity).

#### 4.4 Predictor-observer Approach

Due to the reasons discussed in Section 4.2.2, we opt to work with the simplified sensor model (4.2) to design an algorithm that compensates for the effects of sampling and both pre and post-sampling delays combined. The approach that we propose here to tackle the problem formulated in Section 4.3 is illustrated in Fig. 4.3. We first propose a predictor that takes the sampled and delayed measurements  $z_i(t)$  and provides current predictions of  $y_i(t)$  denoted by  $y_i^p(t)$ . The predictor relies on the knowledge of the input  $u(t)$  in continuous time (or practically at high frequency) and the total delay  $\tau_i = \rho_i + \sigma_i$  to predict the outputs such that  $y_i^p(t) = y_i(t)$  for all  $t \geq t'_{1_i}$  in noise-free conditions (i.e. when there is no measurement noise in  $z_i(t'_{k_i})$  or  $u(t)$  and the integration procedure within the predictor is also exact). The predicted outputs  $y_i^p(t)$ ,  $i = 1, \dots, n$  are then fed into an observer to compute an estimate of the state denoted by  $\hat{X}(t)$ . Our proposed predictor is generic in the sense that it is independent of the employed observer algorithm, i.e., the predictor can be coupled with any asymptotically stable observer or filter to estimate the state.

We propose the following output predictor for the system (3.1) with the outputs

(4.6).

$$\dot{\Delta}_l(t) = \Delta_l(t)u_l(t), \quad \Delta_l(0) = \Delta_{l_0}, \quad (4.10)$$

$$y_{r_i}^p(t) = h_{r_i}(\Delta_l(t'_{k_i} - \tau_i)^{-1}\Delta_l(t), z_{r_i}(t)), \quad t \in [t'_{k_i}, t'_{k_i+1}), \quad (4.11)$$

where  $\Delta_l \in G$  is the internal state of the predictor and  $\Delta_{l_0} \in G$  is an arbitrary initial condition. The trajectory  $\Delta_l(t)$  of the predictor dynamics (4.10) needs to be stored in a buffer for the previous  $t'_{k_i+1} - t'_{k_i} + \tau_i$  seconds in order to compute the prediction  $y_{r_i}^p(t)$  (4.11) at each time. Similarly, we propose the following output predictor for the system (3.2) with the outputs (4.7).

$$\dot{\Delta}_r(t) = \Delta_r(t)u_r(t), \quad \Delta_r(0) = \Delta_{r_0}, \quad (4.12)$$

$$y_{l_i}^p(t) = h_{l_i}(\Delta_r(t)\Delta_r(t'_{k_i} - \tau_i)^{-1}, z_{r_i}(t)), \quad t \in [t'_{k_i}, t'_{k_i+1}), \quad (4.13)$$

where  $\Delta_r \in G$  is the internal state of the predictor. For the mixed-invariant system (3.3) with the outputs (4.8), the dynamics of our propose predictor consists of both (4.10) and (4.12). The trajectories of  $\Delta_l(t)$  and  $\Delta_r(t)$  are buffered and used in the following static output predictor.

$$y_{m_i}^p(t) = h_{r_i}(\Delta_l(t'_{k_i} - \tau_i)^{-1}\Delta_l(t), h_{l_i}(\Delta_r(t)\Delta_r(t'_{k_i} - \tau_i)^{-1}, z_{m_i}(t))), \quad t \in [t'_{k_i}, t'_{k_i+1}). \quad (4.14)$$

Due to the assumption (3.3), the output predictor (4.14) can equivalently be written as

$$y_{m_i}^p(t) = h_{l_i}(\Delta_r(t)\Delta_r(t'_{k_i} - \tau_i)^{-1}, h_{r_i}(\Delta_l(t'_{k_i} - \tau_i)^{-1}\Delta_l(t), z_{m_i}(t))), \quad t \in [t'_{k_i}, t'_{k_i+1}). \quad (4.15)$$

The following theorem summarizes the properties of the proposed predictors.

**Theorem 4.4.1.** *Consider*

- (a) *the predictor (4.10)-(4.11) for the system (3.1) and the output measurements (4.6) with (4.3).*
- (b) *the predictor (4.12)-(4.13) for the system (3.2) and the output measurements (4.7) with (4.4).*

(c) the predictor (4.10), (4.12), and (4.14) (or (4.15)) for the system (3.3) and the output measurements (4.8) with (4.5).

The predicted outputs  $y_{r_i}^p(t)$ ,  $y_{l_i}^p(t)$ , and  $y_{m_i}^p(t)$  are, respectively, equal to the ideal outputs  $y_{r_i}(t)$ ,  $y_{l_i}(t)$ , and  $y_{m_i}(t)$  for all  $i = 1, \dots, n$ , all  $t > t_{1_i}$ , all  $\tau_i \geq 0$ , and all choices of  $\Delta_{l_0}, \Delta_{r_0} \in G$ .  $\square$

*Proof:* The proof is obtained by adapting the proof of Theorem 3.3.2 and Theorem 3.3.4. To prove part (a), use (3.1) and (4.10) to obtain  $\frac{d}{dt}(X_l(t)\Delta_l(t)^{-1}) = X_l(t)u_l(t)\Delta_l(t)^{-1} - X_l(t)\Delta_l(t)^{-1}\Delta_l(t)u_l(t)\Delta_l(t)^{-1} = 0$ . This implies that  $X_l(t_1)\Delta_l(t_1)^{-1} = X_l(t_2)\Delta_l(t_2)^{-1}$  or equivalently  $X_l(t_2) = X_l(t_1)\Delta_l(t_1)^{-1}\Delta_l(t_2)$  for all  $t_1$  and  $t_2$ . Using (4.3) we have

$$\begin{aligned} y_{r_i}(t_2) &= h_{r_i}(X_l(t_2), \dot{y}_{r_i}) = h_{r_i}(X_l(t_1)\Delta_l(t_1)^{-1}\Delta_l(t_2), \dot{y}_{r_i}) \\ &= h_{r_i}(\Delta_l(t_1)^{-1}\Delta_l(t_2), h_{r_i}(X_l(t_1), \dot{y}_{r_i})) = h_{r_i}(\Delta_l(t_1)^{-1}\Delta_l(t_2), y_{r_i}(t_1)). \end{aligned} \quad (4.16)$$

Choosing  $t_1 \in \{t_{k_i}\}_{k=1}^{\infty}$  and  $t_2 = t$  and using (4.6) we have  $y_{r_i}(t) = h_{r_i}(\Delta_l(t_1)^{-1}\Delta_l(t_2), z_{r_i}(t))$  for all  $t \in [t_{k_i}, t_{k_i+1})$ . Resorting to (4.11), this implies that  $y_{r_i}^p(t) = y_{r_i}(t)$  for all  $t \in [t_{k_i}, t_{k_i+1})$  and hence for all  $t \geq t_{1_i}$ . Part (b) is proved similarly. In order to prove part (c), we employ (3.3), (4.10), and (4.12) to obtain

$$\begin{aligned} &\frac{d}{dt}(\Delta_r(t)^{-1}X_m(t)\Delta_l(t)^{-1}) \\ &= -\Delta_r(t)^{-1}\dot{\Delta}_r(t)\Delta_r(t)^{-1}X_m(t)\Delta_l(t)^{-1} \\ &\quad + \Delta_r(t)^{-1}\dot{X}_m(t)\Delta_l(t)^{-1} - \Delta_r(t)^{-1}X_m(t)\Delta_l(t)^{-1}\dot{\Delta}_l(t)\Delta_l(t)^{-1} \\ &= -\Delta_r(t)^{-1}u_r(t)\Delta_r(t)\Delta_r(t)^{-1}X_m(t)\Delta_l(t)^{-1} \\ &\quad + \Delta_r(t)^{-1}(X_m(t)u_l(t) + u_r(t)X_m(t))\Delta_l(t)^{-1} - \Delta_r(t)^{-1}X_m(t)\Delta_l(t)^{-1}\Delta_l(t)u_l(t)\Delta_l(t)^{-1} \\ &= 0. \end{aligned}$$

Hence  $\Delta_r(t_1)^{-1}X_m(t_1)\Delta_l(t_1)^{-1} = \Delta_r(t_2)^{-1}X_m(t_2)\Delta_l(t_2)^{-1}$  or equivalently  $X_m(t_2) =$

$\Delta_r(t_2)\Delta_r(t_1)^{-1}X_m(t_1)\Delta_l(t_1)^{-1}\Delta_l(t_2)$  for all  $t_1$  and  $t_2$ . Using (4.5) we have

$$\begin{aligned} y_{m_i}(t_2) &= h_{r_i}(X_m(t_2), \dot{y}_{m_i}) = h_{r_i}(\Delta_r(t_2)\Delta_r(t_1)^{-1}X_m(t_1)\Delta_l(t_1)^{-1}\Delta_l(t_2), \dot{y}_{m_i}) \\ &= h_{r_i}(\Delta_l(t_1)^{-1}\Delta_l(t_2), h_{r_i}(\Delta_r(t_2)\Delta_r(t_1)^{-1}X_m(t_1), \dot{y}_{m_i})) \\ &= h_{r_i}(\Delta_l(t_1)^{-1}\Delta_l(t_2), h_{l_i}(\Delta_r(t_2)\Delta_r(t_1)^{-1}X_m(t_1), \dot{y}_{m_i})) \\ &= h_{r_i}(\Delta_l(t_1)^{-1}\Delta_l(t_2), h_{l_i}(\Delta_r(t_2)\Delta_r(t_1)^{-1}, h_{l_i}(X_m(t_1), \dot{y}_{m_i}))) \\ &= h_{r_i}(\Delta_l(t_1)^{-1}\Delta_l(t_2), h_{l_i}(\Delta_r(t_2)\Delta_r(t_1)^{-1}, y_{m_i}(t_1))). \end{aligned}$$

Choosing  $t_1 \in \{t_{k_i}\}_{k=1}^{\infty}$  and  $t_2 = t$  and using (4.8) concludes the proof.  $\blacksquare$

Note that in order to implement the proposed predictor-observer methodology, it is only required to implement one copy of the predictor dynamics even though we might have several output measurements  $z_i(t)$ ,  $i = 1, \dots, n$  with possibly different delays  $\tau_i$  and possibly different sampling sequences  $(t'_{k_i})_{k=1}^{\infty}$ . Only a fixed duration buffer for the predictor state  $\Delta(t)$  is needed.

**Remark 4.4.2.** *Our proposed method is also applicable to the case where the delay  $\tau_i$  is time-varying and out-of-sequence measurements do potentially occur. In this case, we should replace the notation  $\tau_i$  with  $\tau_{k_i}$  (forming the sequence  $(\tau_{k_i})_{k=1}^{\infty}$ ) and each measurement delay  $\tau_{k_i}$  should be known to the user at time  $t'_{k_i}$ . It is easy to verify that Theorem 4.4.1 still holds in this case.  $\square$*

**Remark 4.4.3.** *For the very special case where the underlying Lie group is  $\mathbb{R}^n$  and the group multiplication is addition, the kinematic system is simply the linear integrator  $\dot{x}(t) = u(t)$  where  $x(t) \in \mathbb{R}^n$  is the state and  $u(t) \in \mathbb{R}^n$  is the input. The output is given by  $y(t) = Cx(t) \in \mathbb{R}^m$  where  $C \in \mathbb{R}^{m \times n}$ , and the sensors provide the delayed measurement  $z(t) = y(t - \tau)$ . In this case, the proposed predictor simplifies to*

$$\dot{\delta}(t) = u(t), \quad (4.17)$$

$$y^p(t) = C(\delta(t) - \delta(t - \tau)) + z(t). \quad (4.18)$$

*This corresponds to the well-known Smith predictor [127] originally designed for output feedback control of linear systems with delayed measurements. Note however that the Smith predictor is in fact a model predictive control approach for linear systems and not a predictor aimed at a state estimation problem outside of a control loop. In a general linear system setup,*

an application of a sole Smith predictor outside a control feedback loop might not yield a stable method for compensation of delay for state estimation.  $\square$

**Example 4.4.4.** Recalling Example 4.3.1, one can directly apply (4.10)-(4.11) to obtain the following predictor for vector measurements.

$$\Delta(t) = \Delta(t)\Omega(t), \quad \Delta(0) = \Delta_0 \in SO(3) \quad (4.19)$$

$$y_i^p(t) = \Delta(t)^\top \Delta(t'_k - \tau_i) z_i(t), \quad t \in [t'_k, t'_{k+1}) \quad (4.20)$$

where  $\Delta(t) \in SO(3)$  is the internal state of the predictor and  $y_i^p(t) \in \mathbb{R}^3$  is the prediction of current output  $y_i(t)$ .

**Example 4.4.5.** Recalling Example 4.3.2, here we aim to apply the predictor (4.10), (4.12), and (4.14) proposed for general mixed-invariant systems and obtain a predictor tailored for predicting the current attitude and velocity of the vehicle using delayed and sampled attitude and velocity measurements (provided for instance by a VICON system). As explained in Example 4.3.2, only the predictor dynamics (4.10) need to be implemented since the input signal to the predictor dynamics (4.12) is a known constant and hence the predictor state  $\Delta_r(t)$  can be computed analytically. Hence, the predictor dynamics simplifies to the dynamics (3.37)-(3.38). Using the output model (4.9) and resorting to the definition of the right and left output action models explained in Example 4.3.2, the output predictor (4.14) simplifies to  $Y_m^p(t) = \Delta_r(t)\Delta_r(t'_k - \tau)^{-1}Z_m(t)\Delta_l(t'_k - \tau)^{-1}\Delta_l(t)$  for all  $t \in [t'_k, t'_{k+1})$ . Using the same simplifications as done to derive (3.37)-(3.40), one can summarize the resulting attitude and velocity predictor as follows.

$$\dot{\bar{\Delta}}_l(t) = \bar{\Delta}_l(t)\Omega(t), \quad \bar{\Delta}_l(0) = \bar{\Delta}_{l_0} \in SO(3), \quad t \geq 0 \quad (4.21)$$

$$\dot{\delta}_l(t) = \bar{\Delta}_l(t)a(t), \quad \delta_l(0) = \delta_{l_0} \in \mathbb{R}^3, \quad t \geq 0 \quad (4.22)$$

$$R^p(t) = R_z(t)\bar{\Delta}_l(t'_k - \tau)^\top \bar{\Delta}_l(t), \quad t \in [t'_k, t'_{k+1}) \quad (4.23)$$

$$v^p(t) = v_z(t) + (t - t'_k + \tau)ge_3 + R_z(t)\bar{\Delta}_l(t'_k - \tau)^\top (\delta_l(t) - \delta_l(t'_k - \tau)), \quad t \in [t'_k, t'_{k+1}). \quad (4.24)$$

## 4.5 Simulation results

In this section, we provide a set of simulations to illustrate the performance of our proposed predictor-observer methodology. Recalling Example 4.3.1, we simulate the

attitude kinematics (3.4) with  $\Omega(t) = \omega_{\times}$  with  $\omega = [0 \ 0 \ 8]^{\top}$  (deg/s) and we choose the initial attitude  $R_0$  corresponding to the initial roll 14 (deg), pitch 0 (deg), and yaw 0 (deg) to generate the trajectory of  $R(t)$ . We suppose that the attitude sensors provide the vector measurements corresponding to the reference directions  $\hat{y}_1 = [1 \ 0 \ 0]^{\top}$  and  $\hat{y}_2 = [0 \ 1 \ 0]^{\top}$ . Although in practice the number of vector measurements can be high and their directions are not necessarily pairwise perpendicular (e.g. for star trackers), here we consider only two vector measurements with perpendicular directions to avoid unnecessary discussions on gain tuning and focus only on the sampling and delay effects. To model  $z_1(t)$  and  $z_2(t)$ , the ideal vector measurements  $y_1(t)$  and  $y_2(t)$  are obtained by (2.6) and then fed to the block diagram of Fig. 4.1 with pre and post-sampling delays of  $\rho_1 = \rho_2 = 0.1$  (s) and  $\sigma_1 = \sigma_2 = 0.3$  (s), respectively, yielding a total delay of  $\tau_1 = \tau_2 = 0.4$  (s), and a sampling rate of 5 (Hz). Zero mean Gaussian noise processes with a standard deviation of 0.01 are added to each axis of the vector measurements  $z_1(t)$  and  $z_2(t)$  which approximately adds perturbations with the standard deviation of 1 (deg) to the directions of  $z_1(t)$  and  $z_2(t)$ . The angular velocity  $\Omega(t)$  is sampled at 100 (Hz) and perturbed by an additive noise of 0.05 (deg/s) in each axis.

For the simulation, we combine the predictor (4.19)-(4.20) with a bias-free version of the constant gain geometric observer developed in Example 2.6 which is similar to those proposed in [34, 101, 116, 136]. The observer dynamics reads

$$\dot{\hat{R}}(t) = \hat{R}(t)(\Omega(t) + (l_1 y_1^p(t) \times \hat{y}_1(t) + l_2 y_2^p(t) \times \hat{y}_2(t))_{\times}) \quad (4.25)$$

with  $\hat{y}_i(t) := \hat{R}(t)^{\top} \hat{y}_i(t)$  and  $l_i > 0$ ,  $i = 1, 2$ . We compare the performance of this combined predictor-observer with an ad-hoc adaptation of the constant gain observer to the case of sampled and delayed vector measurements. The dynamics of the ad-hoc observer is given by  $\dot{\hat{R}}_{\text{ad}}(t) = \hat{R}_{\text{ad}}(t)(\Omega(t) + \alpha(t)_{\times})$  where  $\hat{R}_{\text{ad}}(t)$  is the estimate of  $R(t)$  and  $\alpha(t)$  is the innovation term. When the attitude sensor provides the measured sample  $z_1(t'_{k_1})$  at time  $t = t'_{k_1}$ , the innovation term of the ad-hoc observer is inspired by the constant gain observer as  $\alpha(t'_{k_1}) = \bar{l}_1 z_1(t'_{k_1})_{\times} \hat{R}_{\text{ad}}(t'_{k_1} - \tau_1)^{\top} \hat{y}_1$  with  $\bar{l}_1 > 0$ . This innovation term compares the newly received measurement  $z_1(t'_{k_1})$  with its estimate  $\hat{R}_{\text{ad}}(t'_{k_1} - \tau_1)^{\top} \hat{y}_1$  in which the effect of the measurement delay  $\tau_1$  is considered<sup>2</sup>.

<sup>2</sup>Due to the consideration of the effect of delay in the innovation term, it can be thought of as a

Similarly, at time  $t = t'_{k_2}$  when the measurement  $z_2(t'_{k_2})$  is delivered by the attitude sensors, the innovation term is  $\alpha(t'_{k_2}) = \bar{l}_2 z_2(t'_{k_2}) \times \hat{R}_{\text{ad}}(t'_{k_2} - \tau_2)^\top \hat{y}_2$  with  $\bar{l}_2 > 0$ . If  $t'_{k_2}$  happens to be equal to  $t'_{k_1}$  for some pair  $(k_1, k_2)$ , then the innovation term is simply the sum  $\bar{l}_1 z_1(t'_{k_1}) \times \hat{R}_{\text{ad}}(t'_{k_1} - \tau_1)^\top \hat{y}_1 + \bar{l}_2 z_2(t'_{k_2}) \times \hat{R}_{\text{ad}}(t'_{k_2} - \tau_2)^\top \hat{y}_2$ . For the times where no sample of any vector measurement is available (i.e. for all  $t \notin (t_{k_1})_{k_i=1}^\infty \cup (t_{k_2})_{k_i=1}^\infty$ ), the innovation term is zero which simplifies the observer to a forward integration of attitude kinematics. This innovation term is mathematically formulated as follows.

$$\alpha(t) = \begin{cases} \bar{l}_1 z_1(t'_{k_1}) \times \hat{R}_{\text{ad}}(t'_{k_1} - \tau_1)^\top \hat{y}_1, & t = t'_{k_1} \neq t'_{k_2} \\ \bar{l}_2 z_2(t'_{k_2}) \times \hat{R}_{\text{ad}}(t'_{k_2} - \tau_2)^\top \hat{y}_2, & t = t'_{k_2} \neq t'_{k_1} \\ \sum_{i=1}^2 \bar{l}_i z_i(t'_{k_i}) \times \hat{R}_{\text{ad}}(t'_{k_i} - \tau_i)^\top \hat{y}_i, & t = t'_{k_1} = t'_{k_2} \\ 0 & t \neq t'_{k_i} \end{cases}$$

This ad-hoc method adaptation of observers is commonly used in engineering applications to handle sensor sampling and delay effects (see e.g. [2] for an EKF example). Also, the Corrector-Predictor Representation of Nonlinear Observers (extensively used in implementation of EKFs) uses similar technique to cope with multi-rate sensors [52].

The initial conditions of the combined predictor-observer (i.e.  $\hat{R}(0.4)$  and  $\Delta(0)$ ) and the initial condition of the ad-hoc observer (i.e.  $\hat{R}_{\text{ad}}(t)$ ,  $t \in [0, 0.4]$ ) are set to the identity matrix. The attitude estimation error of the combined predictor-observer is illustrated in Fig. 4.4, where the observer gains are chosen as  $l_1 = l_2 = 0.5$ . In this figure, the error  $\tilde{\theta}$  is the angle of rotation in the angle-axis representation of the attitude estimate error  $\hat{R}(t)R(t)^\top$  and is given by  $\tilde{\theta}(t) = \frac{180}{\pi} \arccos(1 - 0.5\text{tr}(I - \hat{R}(t)R(t)^\top))$ . Note that the observer trajectories are available after the first sample of the vector measurements have been provided by the attitude sensors. The red plot shows the steady state estimation error which illustrates the good performance of our proposed method even with high sensor delay, low sampling rate, and high noise. Fig. 4.5 shows the estimation error  $\tilde{\theta}_{\text{ad}}(t) = \frac{180}{\pi} \arccos(1 - 0.5\text{tr}(I - \hat{R}_{\text{ad}}(t)R(t)^\top))$  of the ad-hoc observer when its gains are chosen as  $\bar{l}_1 = \bar{l}_2 = 42.5$  such that the error trajectory of this observer has approximately the same transient convergence rate as Fig. 4.4. Comparing Fig. 4.4 and Fig. 4.5, the steady state error of our predictor-

---

Lyapunov-Krasovskii term [18, 24, 53].

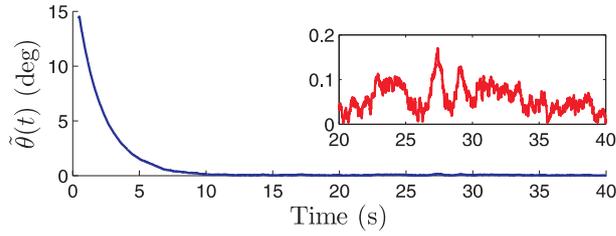


Figure 4.4: Attitude estimation error of the combined predictor-observer. The red plot is the enlarged steady state estimation error.

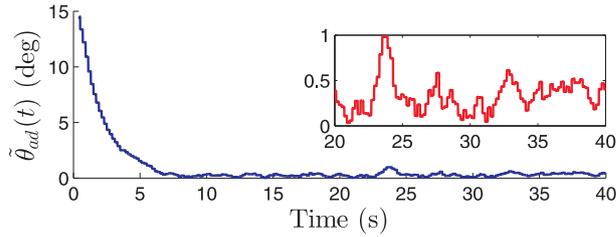


Figure 4.5: Attitude estimation error of the ad-hoc observer. The red plot is the enlarged steady state estimation error.

observer is almost an order of magnitude less than the steady state error of the ad-hoc observer. Next, we increase the sensor delays to  $\rho_1 = \rho_2 = 0.5$  (s) and  $\sigma_1 = \sigma_2 = 1.5$  (s) yielding a total sensor delay of 2 (s). With the same gains and initial conditions as in the previous simulation, the error trajectories of the predictor-observer and the ad-hoc observer are illustrated in Fig. 4.6 and Fig. 4.7, respectively. These plots show the convergence of the estimation error of our proposed predictor-observer while the estimation error of the ad-hoc observer diverges. The small degradation of the steady state estimation error of Fig. 4.6 compared to Fig. 4.4 is due to the fact that the predictor relies on noisy gyro measurements to compensate for the delay in vector measurements. Hence, a larger delay means longer integration of gyro noise which increases the estimation error<sup>3</sup>. Nevertheless, the steady state estimation error of Fig. 4.6 is less than twice the corresponding error in Fig. 4.4 even though the sensor delay is increased by a factor of five.

Next, consider the same condition as the first simulation scenario, but, assume that there is uncertainty in knowledge of the amount of delay. To this end, we consider the sensor model of Fig. 4.1 with the same parameters as the first simulation but we consider two examples where the amount of the total delay that is used in

<sup>3</sup>The effect of input noise on the performance of the predictors is discussed via simulation in Chapter 3, Section 3.5.

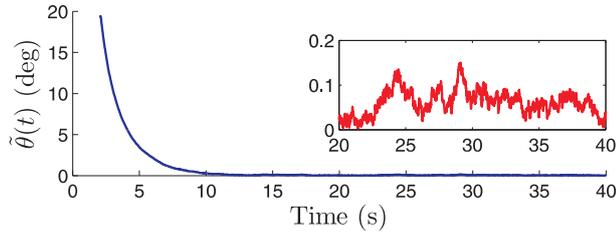


Figure 4.6: Attitude estimation error of the combined predictor-observer (large sensor delay). The red plot is the enlarged steady state estimation error.

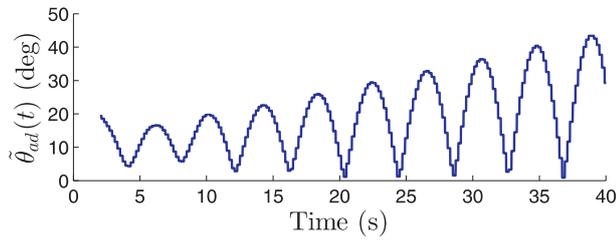


Figure 4.7: Attitude estimation error of the ad-hoc observer (large sensor delay).

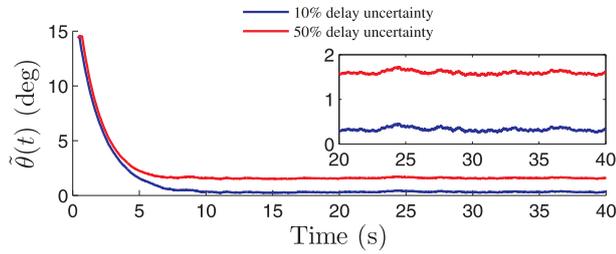


Figure 4.8: Attitude estimation error of the combined predictor-observer (with delay uncertainty). The small plots are the steady state estimation errors.

the predictor (4.10)-(4.11) is either 10 or 50 percent more than the total delay in the simulated sensor model (i.e.  $\tau_1 = \tau_2 = 0.44$  (s) or  $\tau_1 = \tau_2 = 0.6$  (s), respectively). Fig. 4.8 shows that the estimation error is practically stable in both cases although the steady state estimation error is increased compared to the previous simulation (see Fig 4.4). The steady state estimation errors are less than 0.5 (deg) and 1.8 (deg) respectively for 10% and 50% delay uncertainties which still demonstrates a very good performance considering the high values of noise and delay uncertainties.

## 4.6 Embedded Software Development and Experimental Verification

In this section, we provide experimental results to demonstrate the performance of the predictor-observer approach proposed in this chapter. To this end, we introduce a predictor based on Example 4.4.5 and we implement it in the embedded autopilot system of a plane. We use ArduPilot/APM as the autopilot software system which is developed mostly by a group of developers at 3DRobotics and CanberraUAV team [1, 5, 7, 11, 14]. ArduPilot is compatible with multiple autopilot boards such as Pixhawk, PX4FMU, Arsov AUAV-X2, APM2, etc, and is by far the most popular autopilot system used among the UAV enthusiast community including both hobbyist and commercial users [3]. This autopilot system is also equipped with software packages for off-line testing of newly developed software before real flight tests. A comprehensive open source software-in-the-loop (SITL) simulator and a *Replay* facility are examples of these software packages for testing ArduPilot's Attitude and Heading Reference Systems (AHRS). As far as testing the estimation algorithms for navigation is concerned, the difference between SITL and Replay is that SITL relies on simulating the sensor outputs while the Replay module uses experimentally gathered sensor logs from real flights to test the navigation algorithm. An advantage of SITL is that it is capable of testing the estimation algorithm in a closed loop scenario where the output of the estimation algorithm is fed into the controller to perform the mission. The advantage of the Replay facility is that it uses real flight data (instead of simulated sensor data) which is more accurate, but it is unable to perform closed loop tests. The code developed in this section is tested with both SITL and Replay modules before performing real flight tests.

The experimental results presented in this section verify effective performance of the predictor-observer approach in real flight tests. We also demonstrate advantages of the predictor-observer approach over classical Lyapunov-Krasovskii modifications for sensor delay compensation using off-line processing of flight data with the Replay module.

Through a collaboration with the ArduPilot developers, the predictor code developed in this section will be merged into the master code of the ArduPilot/APM system and will be publicly available as a C++ library. A copy of the ArduPilot



Figure 4.9: Photo taken at Canberra Model Aircraft Club. People in the photo (right to left): Alireza Khosravian, Sean O'Brien (an undergraduate student at ANU who helped with development of the predictor library during his internship with 3DRobotics), Jack Pittar (a pilot with Canberra Model Aircraft Club).



Figure 4.10: Photo taken at Canberra Model Aircraft Club. People in the photo (right to left): Sean O'Brien, Andrew Tridgell (the main developer of the ArduPilot), Adam Kroll (a pilot with Canberra Model Aircraft Club).

code including the primary version of the predictor library is available in the CD attached to this thesis and is also accessible online at [13]. Brief explanations about the predictor library are given in the Appendix.

### 4.6.1 Autopilot software and experimental platform

In this section, we briefly explain the navigation algorithm of the ArduPilot system. For more detailed explanations, the reader is referred to the ArduPilot developers' website [3].

ArduPilot's state estimation consists of two separate algorithms that run in parallel; a geometric attitude estimator on  $SO(3)$  (which is called DCM<sup>4</sup> in ArduPilot), and an Extended Kalman Filter (EKF). Generally speaking, the geometric observer is more robust than the EKF (meaning that it is less prone to divergence with various environmental or sensor setups) but is less accurate. Using the MAVLink<sup>5</sup> parameter "AHRS\_EKF\_USE", it is possible to choose which algorithm is fed into the flight controller for instant using MAVProxy<sup>6</sup>. By default, ArduPilot uses the EKF as the primary state estimation method while it still runs the DCM in the background for initializing the EKF, monitoring its convergence and triggering a failsafe mode if the EKF diverges. For the study in this section, we use the EKF as the observer that we combine with the predictor. For this reason, we explain the ArduPilot EKF in more detail.

The EKF is able to fuse sensory information from the following types of sensors; primary IMU (3-axis gyro and accelerometer), secondary IMU, primary GPS, secondary GPS, barometer, magnetometer, airspeed sensor, down facing camera (optical flow data). The user is able to select which types of sensors are available by setting the corresponding MAVLink parameters. Only North and East (NE) components of the GPS position measurements are used in the EKF while a barometer provides measurements of the Down component of the position<sup>7</sup>.

The state vector of the ArduPilot EKF consists of 34 elements which provide estimates of the following variables [4]; quaternion vector representing the vehicle's attitude (4 elements), linear velocity in the North-East-Down (NED) frame<sup>8</sup> (3 ele-

---

<sup>4</sup>Direction Cosine Matrix

<sup>5</sup>MAVLink is a very lightweight, header-only message marshalling library for micro air vehicles [10].

<sup>6</sup>MAVProxy is a command line based ground station for UAVs. The intent is for a minimalist, portable and extendable ground control system for any UAV supporting the MAVLink protocol [11].

<sup>7</sup>This is because the height measurement of the barometer is more accurate than the GPS height measurement.

<sup>8</sup>If two IMUs are available, this velocity is updated in the filter using a weighted average of both IMUs.

---

ments), position vector in the NED frame<sup>9</sup> (3 elements), gyro bias<sup>10</sup> (3 elements), bias of the z-axis of the first accelerometer (1 element), wind velocity in North and East directions (2 elements), magnetic field vector in the NED frame (3 elements), bias of the magnetometer (3 elements), bias of the z-axis of the second accelerometer (1 element), velocity vector using only the first IMU measurements (3 elements), down element of the position vector using the first IMU measurements (3 elements), velocity vector using only the second IMU measurements (3 elements), down element of the position vector using the second IMU measurements (3 elements), angular rate vector<sup>11</sup> (3 elements).

We tested the code with various model planes, although in this chapter we only provide the test results done with an electric motor drive foam plane. The plane is equipped with Pixhawk as the autopilot hardware and ArduPilot is used as the autopilot software. The available navigation sensors on the plane are two IMUs, a magnetometer, a GPS, and a barometer.

#### 4.6.2 Employed predictor

We adapt the predictors proposed by (4.19)-(4.20) and (4.21)-(4.24) and combine them with the ArduPilot's EKF. Using those predictors, we aim to compensate for the delays of linear velocity and position measurements from GPS, the magnetometer delay, and the barometer delay. We rewrite the predictors using minor modifications in order to take full advantage of the predictors in practice. The resulting predictor is divided into two parts. The first part is the predictor dynamics which update the internal states of the predictor and add the updated state to a buffer. This dynamic update is performed each time a new IMU measurement is received and the prediction part of the EKF is performed. The second part of the predictor is the static equations which take output measurements (i.e. GPS measurements, magnetometer measurements, and barometer measurements) and compensate for their delays using the buffered states of the predictor and provide a prediction of the current values of those measurements. These static equations are employed at the fusion stage of the

---

<sup>9</sup>If two IMUs are available, this position is updated in the filter using a weighted average of both IMUs.

<sup>10</sup>When two IMUs are used, the gyro bias estimate represents the estimate of the bias in the weighted average of both gyros.

<sup>11</sup>This estimated angular rate is based on blending both IMUs and is used in the optical flow fusion.

EKF where new output measurements are fused. This separation of the dynamics of the predictor from its static part allows performing data fusion only when new output measurements are received (as opposed to performing the fusion whenever new output predictions are available). This reduces the computational complexity of the whole predictor-observer and results in more or less the same computational complexity as the original ArduPilot EKF. As an additional advantage, this separation allows the user to easily switch between the original innovation terms of the EKF and the modified innovation terms provided by the predictor without needing to modify the rest of the algorithm.

Apart from the above explained modifications, a few minor modifications are also done on the predictor itself. These modifications are explained after providing the predictor equations. Denote the current time by  $t$ . This is the time at which the most recent IMU measurement is used in the prediction stage of the EKF. The predictor dynamics are given by

$$\dot{\Delta}(t) = \Delta(t)\hat{\Omega}(t), \quad \Delta(0) = \Delta_0 \in \text{SO}(3), \quad t \geq 0 \quad (4.26)$$

$$\dot{\delta}_v(t) = \Delta(t)\hat{a}(t), \quad \delta_v(0) = \delta_{v_0} \in \mathbb{R}^3, \quad t \geq 0, \quad (4.27)$$

$$\dot{\delta}_p(t) = \hat{v}(t), \quad \delta_p(0) = \delta_{p_0} \in \mathbb{R}^3, \quad t \geq 0, \quad (4.28)$$

were  $\Delta$ ,  $\delta_v$ , and  $\delta_p$  are the internal states of the predictor associated with the prediction of attitude, velocity, and position, respectively.  $\hat{\Omega}$  and  $\hat{a}$  are angular velocity and linear acceleration measurements that are obtained by compensating the bias of IMU measurements using the bias estimates provided by the EKF<sup>12</sup>.  $\hat{v}$  is the estimate of the linear velocity of the vehicle provided by the EKF. For the embedded implementation in ArduPilot, a very simple discrete time quaternion based formulation of the predictor dynamics (4.26)-(4.28) is implemented as follows.

$$q_{k+1} = q_k \otimes \begin{bmatrix} \cos(0.5\|\tilde{\Omega}_{k+1}\|) \\ \frac{\sin(0.5\|\tilde{\Omega}_k\|)}{\|\tilde{\Omega}_k\|} \tilde{\Omega}_k \end{bmatrix}, \quad (4.29)$$

$$\delta_{v_{k+1}} = \delta_{v_k} + T\Delta_k\hat{a}_k, \quad (4.30)$$

$$\delta_{p_{k+1}} = \delta_{p_k} + T\hat{v}_k. \quad (4.31)$$

<sup>12</sup>As the test vehicle is equipped with two IMUs, we use the blended IMU measurements rather than individual IMU measurements.

where  $T$  is the update interval of predictor states<sup>13</sup>,  $k = 1, 2, 3, \dots$ ,  $\otimes$  denote the quaternion multiplication, and  $\tilde{\Omega}_k := \int_{t-T}^t \text{vex}(\hat{\Omega}(s)) ds \approx T \text{vec}(\hat{\Omega}_k) \in \mathbb{R}^3$  where  $\text{vex}(\cdot) : \mathfrak{so}(3) \rightarrow \mathbb{R}^3$  is the inverse of the transformation  $(\cdot)_\times$ . Here,  $q$  denotes the quaternion representation of  $\Delta \in \text{SO}(3)$ <sup>14</sup> with the conversion equation  $\Delta = I_3 + 2q^s q^v + 2q_\times^v{}^2$  where  $q^s$  and  $q^v$  denote the scalar and vector parts of the quaternion vector  $q$ , respectively. The discretization is obtained using the exponential map assuming that  $\hat{\Omega}$ ,  $\hat{a}$ , and  $\hat{v}$  are approximately constant in an interval of  $T$  seconds<sup>15</sup>.

Denote by  $t_b$  the time at which the new magnetometer measurement  $y_b(t_b) \in \mathbb{R}^3$  is received and denote the magnetometer delay by  $\tau_b$ , meaning that the magnetometer measurement is taken at time  $t_b - \tau_b$ . Assume that the fusion of the magnetometer data is performed at  $\bar{t}_b$ <sup>16</sup>. As proposed by (4.20), the prediction  $y_b^p \in \mathbb{R}^3$  of the magnetic field at time  $\bar{t}_b$  is given by

$$y_b^p(\bar{t}_b) = \Delta(\bar{t}_b)^\top \Delta(t_b - \tau_b) y_b(t_b). \quad (4.32)$$

The prediction  $y_b^p(\bar{t}_b)$  is then used in the fusion stage of the EKF (instead of  $y_b(t_b)$ ). Similarly, denote by  $t_g$  the time at which new GPS velocity  $v^m(t_g) \in \mathbb{R}^3$  and NE position  $p_{\text{ne}}^m(t_g) \in \mathbb{R}^2$  measurements are received. Assume that the measured GPS velocity and NE position are delayed by  $\tau_v$  and  $\tau_p$  seconds, respectively<sup>17</sup>. Assume that the fusion of GPS data is performed at  $\bar{t}_g$ . We adapt (4.24) to obtain the following prediction of the velocity  $v^p \in \mathbb{R}^3$ , and we extend the prediction to NE position  $p_{\text{ne}}^p \in \mathbb{R}^2$ .

$$v^p(\bar{t}_g) = v^m(t_g) + (t_g - \tau_g g) e_3 + \hat{R}(t_g - \tau_g) \Delta(t_g - \tau_g)^\top (\delta_v(\bar{t}_g) - \delta_v(t_g - \tau_g)), \quad (4.33)$$

$$p_{\text{ne}}^p(\bar{t}_g) = p_{\text{ne}}^m(t_g) + \delta_p(\bar{t}_g) - \delta_p(\bar{t}_g - t_g + \tau_g), \quad (4.34)$$

<sup>13</sup>For our experimental tests on a plane, the update interval is 20 (ms) since the navigation algorithm of ArduPlane runs at 50 Hz.

<sup>14</sup>The quaternion vector is renormalized every  $T$  seconds to maintain its unit norm.

<sup>15</sup>More accurate discretization can be obtained using Lie group variational integrators and assuming higher order variations of the input signals  $\hat{\Omega}$ ,  $\hat{a}$ , and  $\hat{p}$  [67, 95, 112, 124].

<sup>16</sup>ArduPilot's EKF performs fusion of outputs axis by axis. This means that the fusion time associated with the first axis of an output sensor (such as the magnetometer) will be different from the second axis. Here, we use the same notation for the fusion time of all the axes of the same sensor, but, in practice we use a separate fusion time for each axis.

<sup>17</sup>Depending on the GPS chip, the delay associated with the velocity measurement can be different from the position measurement delay.

where  $\hat{R}$  denotes the estimated attitude provided by the EKF,  $g$  is the constant gravitational acceleration of the Earth, and  $e_3 = [0 \ 0 \ 1]^\top$ . Denote by  $t_d$  the time at which a new down element of the position is measured (for instance by barometer) and denote its associated measurement delay by  $\tau_d$ . Similar to (4.34), a prediction  $p_d^p$  of the down element of the position  $p_d^m$  (which is negative of the height) at time  $\bar{t}_d$  is obtained as

$$p_d^p(\bar{t}_d) = p_d^m(t_d) + \delta_p(\bar{t}_d) - \delta_p(t_d - \tau_d). \quad (4.35)$$

### 4.6.3 Experimental results

In order to test the predictor-observer approach, several flight tests are done with various model planes at the Canberra Model Aircraft Club [6]. In the following, we provide test results of the electric motor drive foam plane shown in Fig. 4.10. The original EKF of ArduPilot runs in parallel to a similar EKF combined with the predictor onboard the plane. The original EKF of ArduPilot uses a Lyapunov-Krasovskii method to compensate for sensor delays. In the Lyapunov-Krasovskii approach, the output measurements (with known delays) are compared with their delayed estimate to form the innovation term of the EKF in the fusion stage. This approach is similar to what is explained as the "ad-hoc" approach in Section 4.5. In the implemented predictor-observer, a similar EKF as the original ArduPilot EKF is employed. The only difference is that the Lyapunov-Krasovskii innovation term of the original EKF is replaced by a new innovation term that compares the prediction of the outputs at the current time (given by equations (4.32)-(4.35)) with their current estimate. Although both EKFs run in parallel, the estimate provided by the predictor-observer is fed to the onboard flight controller of the plane during the entire flight (and the estimates of the original EKF are logged only for comparison and analysis after the flight). For the EKF that is combined with our predictor, we use the same default gains of the original ArduPilot EKF (i.e. we do not do any gain tuning for our EKF). For comparison purposes, this condition favors the original EKF because the default gains have been very carefully tuned for the original EKF (using thousands of flight logs analyzed by the ArduPilot developers over the past two years).

For this flight test, IMU measurements are taken at high rate but are passed to the EKFs at 50 Hz (this is the rate at which the prediction stage of the EKF is performed).

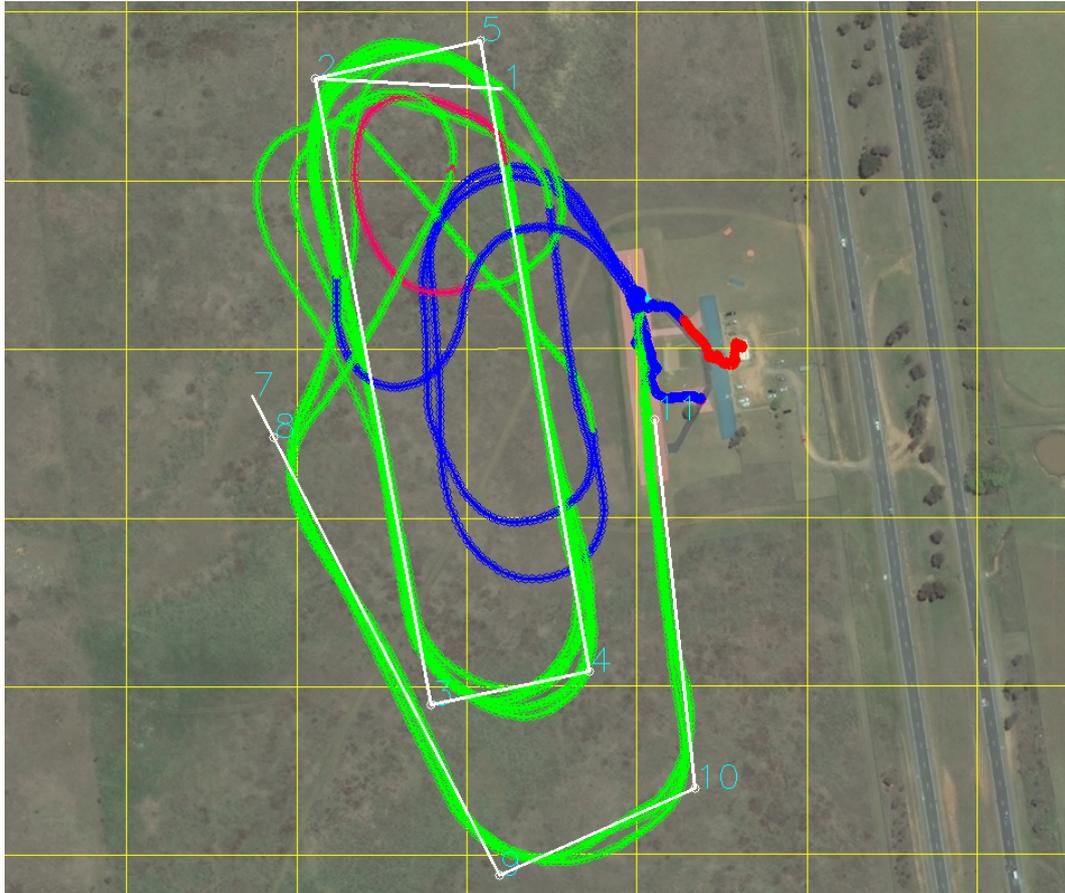


Figure 4.11: Flight path of the plane according to GPS measurements. Way-points 1 to 5 are used for autonomous flight and Way-points 7 to 11 are used for autonomous landing. Different colors indicate various flight modes; green: AUTO, red: MANUAL, blue: FLY BY WIRE A (FBWA).

GPS measurements are taken approximately at 5 Hz and the GPS delay is set to 220 (ms) in both EKFs<sup>18</sup>. Magnetometer and barometer measurements are both passed to the EKFs at 10 Hz and their associated delays are considered as 40 (ms) and 60 (ms) respectively.

Fig 4.11 shows the flight path of the vehicle using GPS measured positions (different colors indicate various flight modes; green: AUTO<sup>19</sup>, red: MANUAL<sup>20</sup>, orange:

<sup>18</sup>The value of 220 (ms) for GPS delay is suggested by the ArduPilot developers according to their post-flight analysis of the employed GPS chip.

<sup>19</sup>In AUTO mode the plane flies fully autonomously and follows a mission (a set of GPS waypoints and other commands) set by the ground station configuration.

<sup>20</sup>In MANUAL mode, the pilot is able to control the plane just like a regular RC control with no stabilization. That is, all RC control inputs are passed to the motors and actuators of the plane.

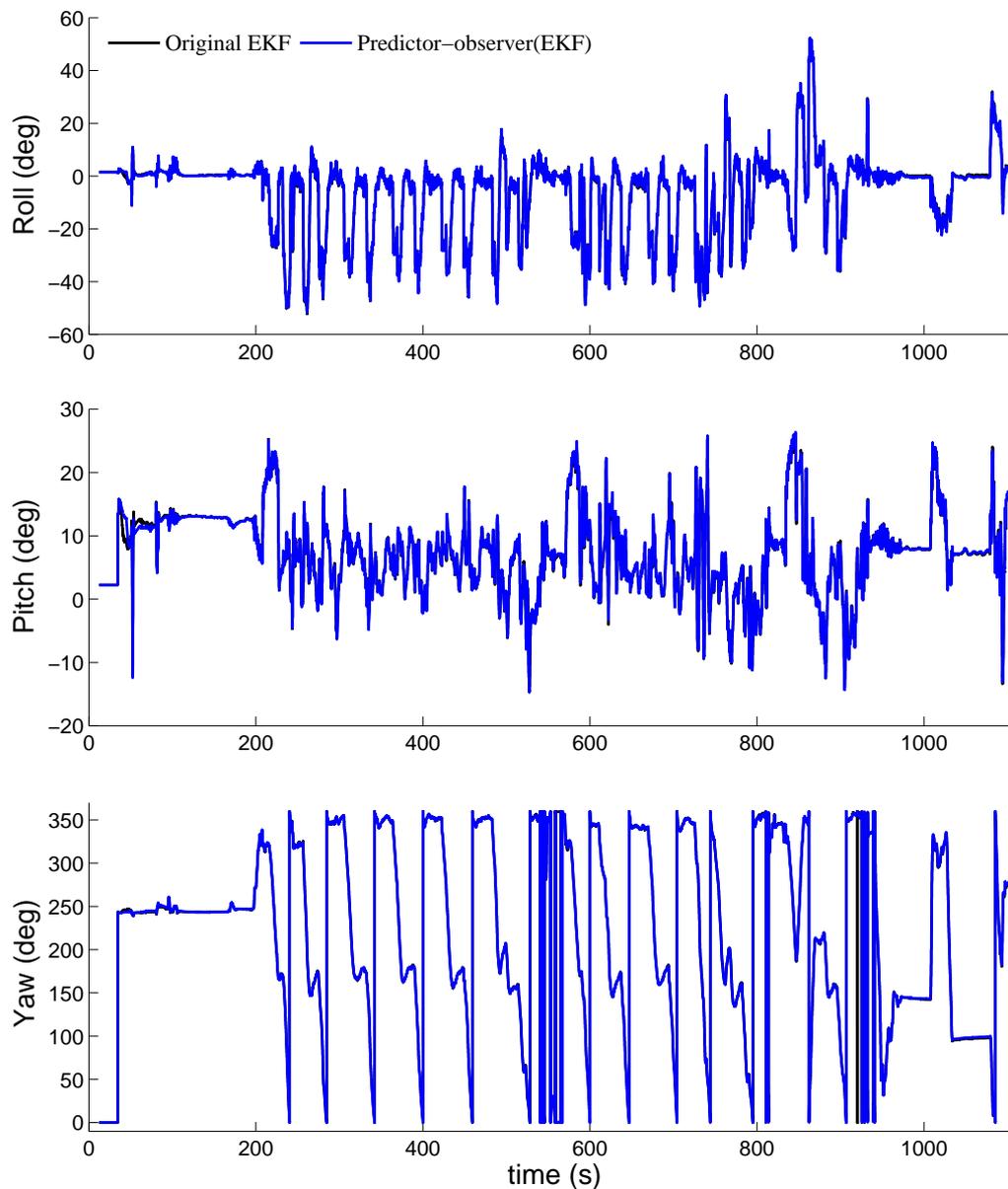


Figure 4.12: Estimates of the Euler angles of the plane provided by the original EKF versus the estimates provided by the predictor-observer.

FLY BY WIRE A (FBWA)<sup>21</sup>). Fig. 4.12 compares the estimates of the Euler angles

<sup>21</sup>FBWA is the most popular mode for assisted flying of planes, and is the best mode for inexperienced pilots. In this mode, the plane will hold the roll and pitch specified by the control sticks. So if the pilot holds the aileron stick hard right then the plane will hold its pitch level and will bank right by the angle specified in the LIM\_Roll\_CD option. It is not possible to roll the plane past the roll limit specified in LIM\_ROLL\_CD, and it is not possible to pitch the plane beyond the LIM\_PITCH\_MAX/LIM\_PITCH\_MIN settings.

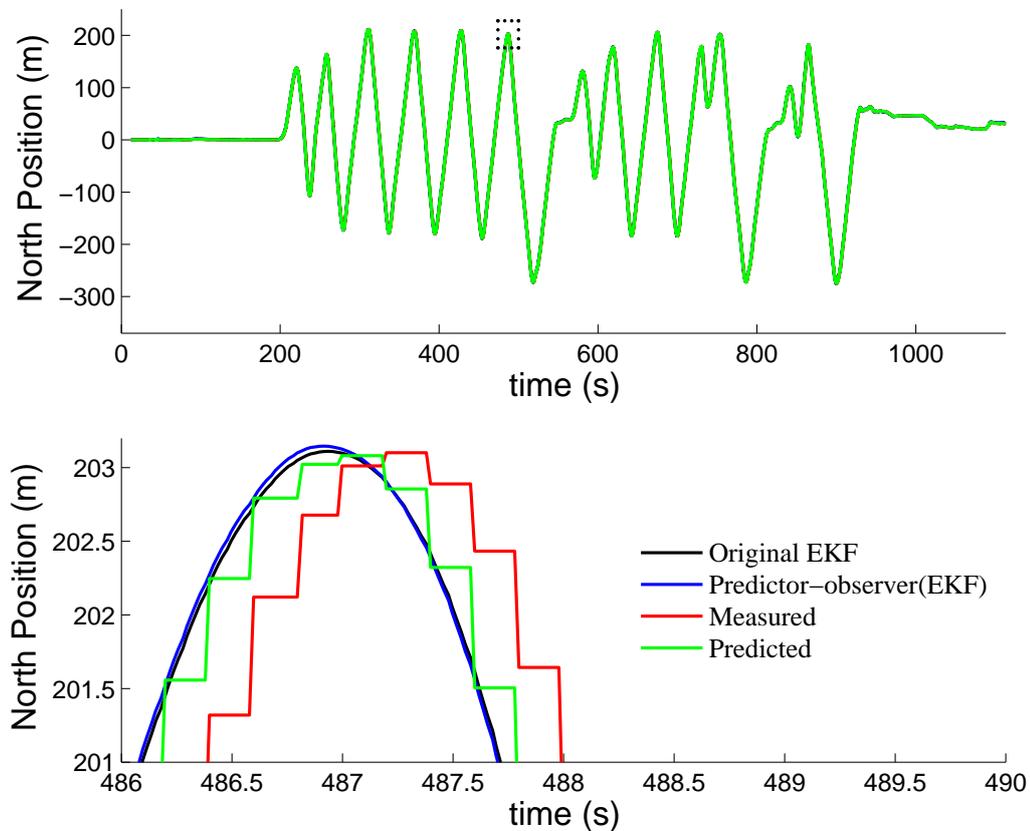


Figure 4.13: North position of the plane provided by GPS versus its prediction provided by (4.34), and its estimate provided by the combined predictor-EKF. The bottom figure is an enlarged portion of the top figure specified by the dashed black rectangle.

of the vehicle provided by the original ArduPilot EKF with the estimates provided by the predictor-observer. The estimates provided by both EKFs are very close together (the difference is less than 1 degrees in each axis for all times). We recall that the EKF gains are tuned for the original ArduPilot EKF. Fig. 4.12 shows that the predictor-observer approach works as good as the original EKF (we later on provide examples of cases where the predictor-observer demonstrates advantages over the original EKF).

In order to show the performance of the predictor alone, Fig. 4.13 and Fig. 4.14 show the North and East positions of the plane as measured by GPS versus their predicted values and the estimates provided by the combined predictor-observer and the original ArduPilot EKF. Note that the predicted values are computed only at the

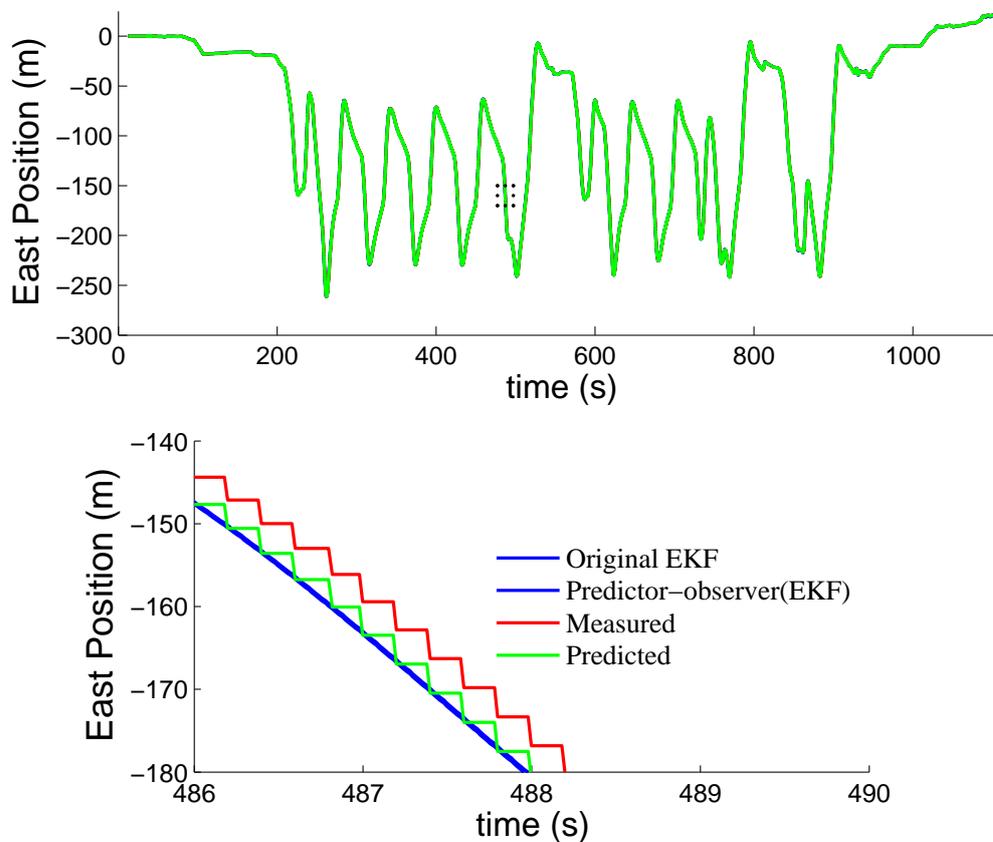


Figure 4.14: East position of the plane provided by GPS versus its prediction provided by (4.34), and its estimate provided by the combined predictor-EKF. The bottom figure is an enlarged portion of the top figure specified by the dashed black rectangle.

fusion stage of the EKF (i.e. when new sensor measurements arrive approximately every 200 (ms)). The top figures show that the estimates provided by the original EKF as well as the combined predictor-observer both are close to GPS measurements meaning that both filtering methods are stable during the whole flight. The bottom figures show that the predicted positions are shifted forward in time (by about 22 (ms)) compared to the measured positions, indicating that the predictor has effectively compensated for the GPS delay. Notice that the predicted positions are closer to the estimated positions (which are much more accurate because they are the result of fusing multiple sensor measurements) compared to the measured positions.

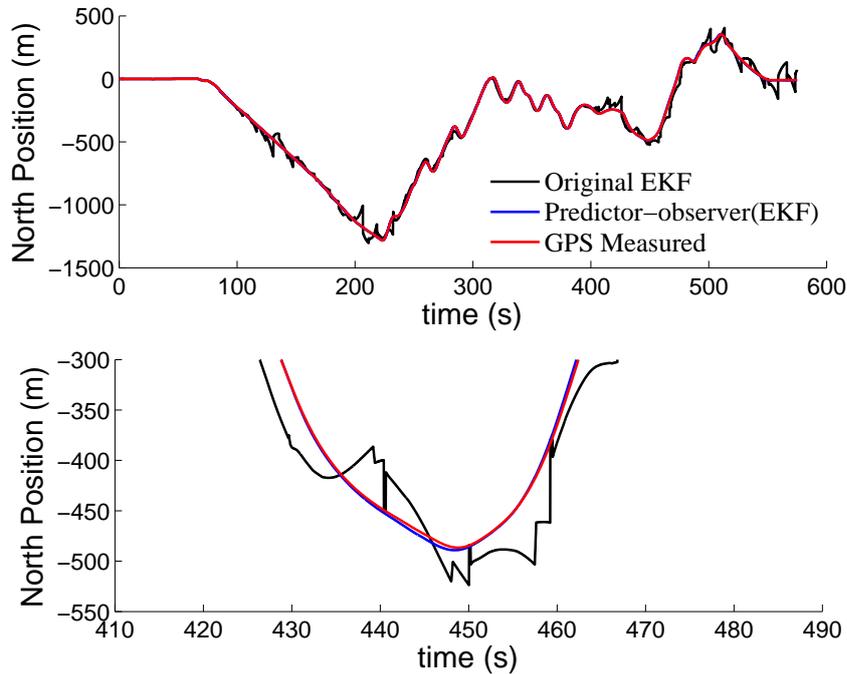


Figure 4.15: North position estimates provided by the original ArduPilot EKF versus the combined predictor-EKF and GPS measurements (default EKF gains are used). The bottom plot shows an enlarged portion of the top plot.

#### 4.6.4 Predictor-observer approach versus original ArduPilot EKF: robust gain tuning

In the previous Section, the difference between the estimates provided by the original ArduPilot EKF and the proposed predictor-observer was negligible in our experimental tests. As was explained before, the innovation terms of the original ArduPilot EKF use a Lyapunov-Krasovskii modification to cope with the sensor delay. It is known that Lyapunov-Krasovskii modifications reduce the stability margin of observers or filters and require careful and conservative gain tuning [16, 18, 24, 41, 134]. The reason that we do not observe any sign of instability of the original ArduPilot EKF in the experimental tests of the previous Section is that the EKF gains have been very carefully tuned for specific sensor boards and also separately for each type of vehicle using tens of hundreds of flight logs that the ArduPilot developers and users gathered over the past two years. This gain tuning process is very time consuming and can also be challenging, especially for the ArduPilot system where there is a very large body of users that fly their UAVs in different environments and with various

hardware. Problems may arise when users employ the original ArduPilot EKF in environmental conditions or with sensor setups that have different characteristics (e.g. noise, bias, etc.) than what matches the EKF gains. In this section, we demonstrate that the use of the predictor-observer approach instead of the Lyapunov-Krasovskii modification largely reduces the dependency of the EKF stability on the gain tuning. To this end, we employ a flight log that has been shown to have high sensitivity to the tuning of the EKF<sup>22</sup>. We process the flight log off-line using the Replay facility<sup>23</sup> of the ArduPilot system. With this flight log, the original ArduPilot EKF becomes unstable when its default gains are used. We show that retuning the EKF gains resolves the problem of the original EKF (though the new gains might not suit a large portion of other ArduPilot users). We then show that the predictor-observer preserves its stability both with the default gains and with the new gains, demonstrating the robustness of the proposed method against gain tuning.

To simplify the comparison, we only show the estimates of the vehicle's North and East positions. Fig. 4.15 and Fig. 4.16, respectively, show the North and East position estimates when the default EKF gains are used. The original ArduPilot EKF estimates are up to 150 (m) away from the GPS measurements while the predictor-observer provides estimates close to the GPS measurements. The discontinuities in the black plot are due to the fail-safe system of ArduPilot AHRS that reinitializes EKF states when it recognizes that the filter has diverged. Without the fail-safe system, the original EKF estimates would have moved much further away from the GPS measurements.

We re-tune the original EKF gains by increasing some of the gains associated with gyro measurement noise so that the original EKF becomes stable<sup>24</sup>. We then use the same re-tuned gains for the EKF part of the predictor-observer combined. Fig.

---

<sup>22</sup>The employed flight log has been provided to us by Andrew Tridgell [131] (the main developer of the ArduPilot System) and Paul Riseborough [118] (the main developer of the ArduPilot EKF).

<sup>23</sup>The Replay facility allows building the ArduPilot/APM autopilot code using an ordinary C++ compiler, making a native executable that allows testing the attitude and heading reference system (AHRS) of ArduPilot without implementing on an actual hardware. The native executable emulates the AHRS (including the EKFs), processes the flight log off-line, and feeds the sensor measurements to the AHRS as though the AHRS was implemented onboard the vehicle during the flight. A secondary log is generated by Replay containing the results of the off-line processed data.

<sup>24</sup>For the example demonstrated in Fig. 4.15 and Fig. 4.16, the following EKF gain variables are used in both "AP\_NavEKF.cpp" (original EKF code) and "AP\_NavEKF2.cpp" (the EKF combined with our predictor) library of ArduPilot; "GBIAS\_PNOISE\_DEFAULT=10<sup>-7</sup>" and "GYRO\_BIAS\_UNCERTAINTY=0.01". For the results of Fig. 4.17, the values "GBIAS\_PNOISE\_DEFAULT=10<sup>-6</sup>" and "GYRO\_BIAS\_UNCERTAINTY=0.1" are chosen.

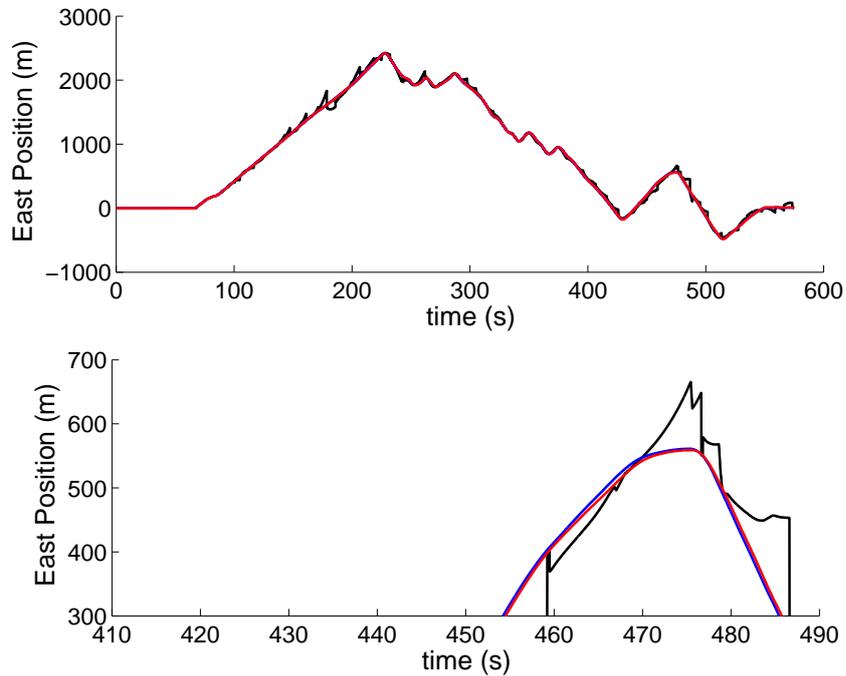


Figure 4.16: East position estimates provided by the original ArduPilot EKF versus the combined predictor-EKF and GPS measurements (default EKF gains are used). The bottom plot shows an enlarged portion of the top plot.

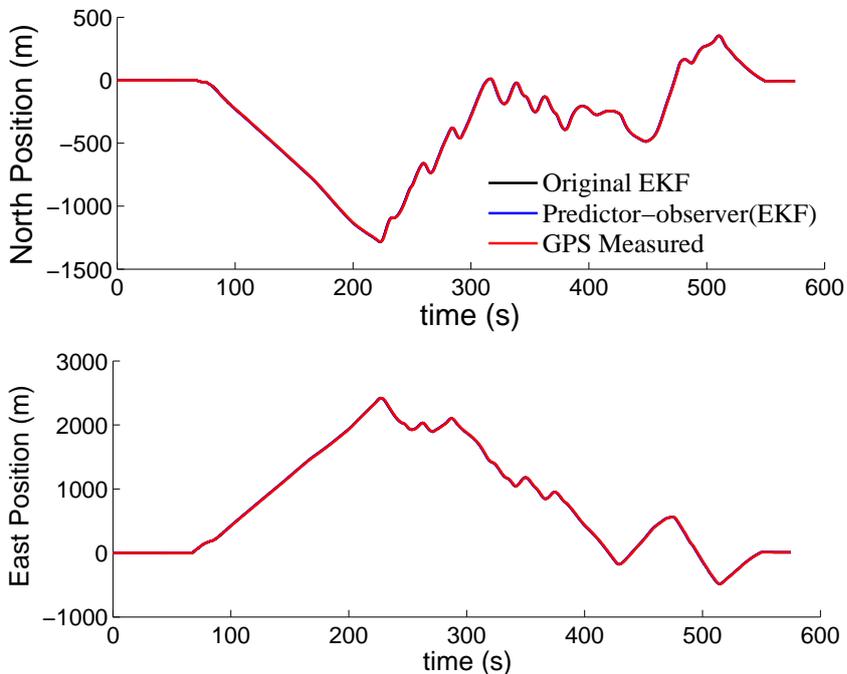


Figure 4.17: North and East position estimates provided by the original ArduPilot EKF versus the combined predictor-EKF and GPS measurements (re-tuned EKF gains are used).

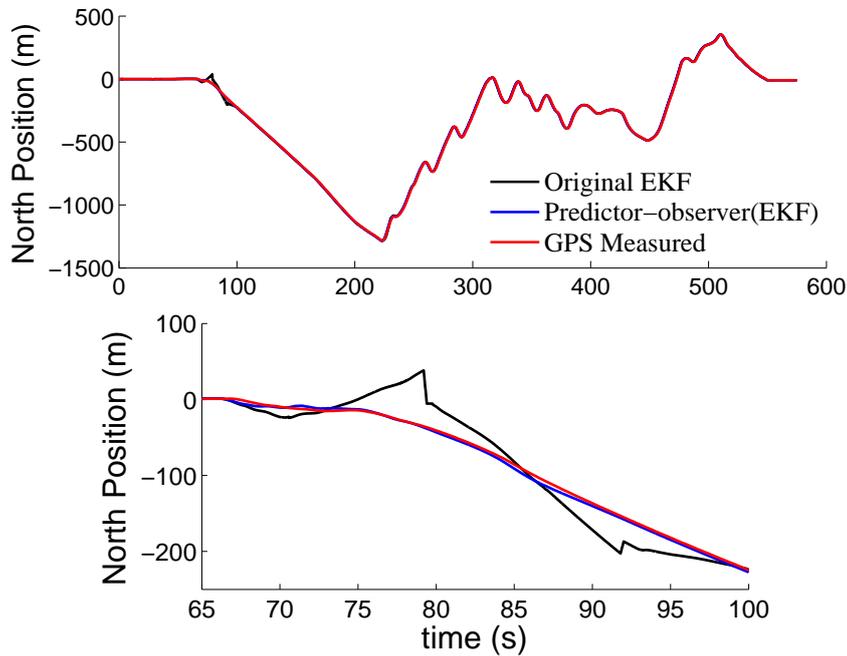


Figure 4.18: North position estimates provided by the original ArduPilot EKF versus the combined predictor-EKF and GPS measurements (high EKF gains are used). The bottom plot shows an enlarged portion of the top plot.

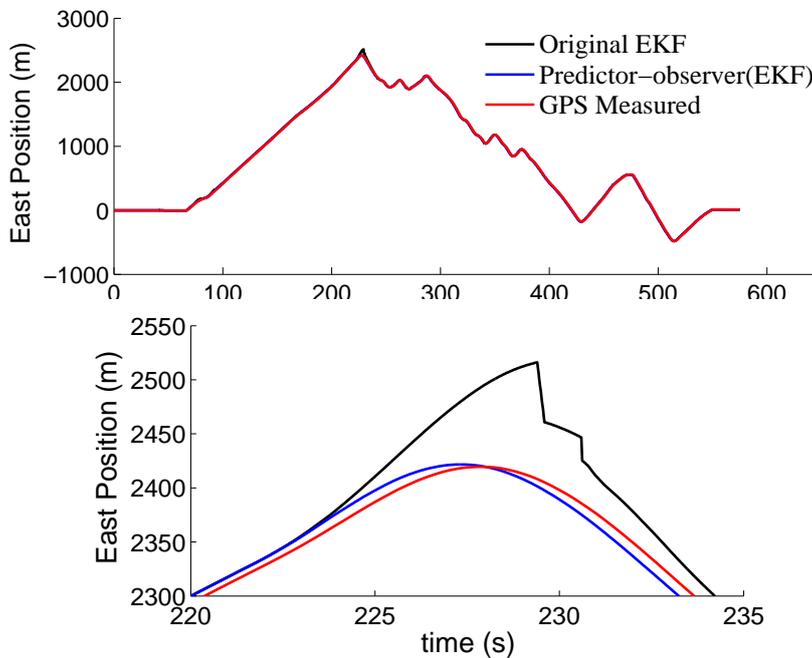


Figure 4.19: East position estimates provided by the original ArduPilot EKF versus the combined predictor-EKF and GPS measurements (high EKF gains are used). The bottom plot shows an enlarged portion of the top plot.

---

4.17 demonstrates the North and East position estimates versus GPS measurements showing that both filters are stable.

We continue to increase the gains for both the original ArduPilot EKF and the predictor-observer<sup>25</sup>. Fig. 4.18 and 4.19 show that the original EKF becomes unstable and produces large errors at some points<sup>26</sup> while the predictor-observer provides stable estimates for high gains as well. This shows that the predictor-observer demonstrates more robust behavior and provides a larger margin of stability.

## 4.7 Summary

In this chapter, we propose a combined predictor-observer methodology for state estimation of invariant systems on Lie groups in the presence of sampled and delayed output measurements. Exploiting the symmetries of the underlying system and output maps, our proposed predictor is capable of providing predictions of the current outputs. The proposed predictor is generic and can be combined with arbitrary observers or filters. When combined with a geometric attitude observer, our proposed predictor-observer approach shows improved performance in simulation compared to Lyapunov-Krasovskii methods. We develop the predictor as an open source C++ library for the ArduPilot system and we verify its performance in practice. Our experimental results show that the predictor-observer approach is more robust against gain tuning compared to traditional approaches, making it applicable to a wider class of vehicles with different sensor setups, various environmental conditions, and diverse flight trajectories.

---

<sup>25</sup>For the example of Fig. 4.18 and 4.19, we choose the following EKF gain variables in both "AP\_NavEKF.cpp" (original EKF code) and "AP\_NavEKF2.cpp" (the EKF combined with our predictor) library of ArduPilot; "GBIAS\_PNOISE\_DEFAULT=  $8 \times 10^{-6}$ " and "GYRO\_BIAS\_UNCERTAINTY= 1".

<sup>26</sup>It is worth noting that such large errors if fed into the control algorithm cause instability and might cause the vehicle to crash. In the off-line re-processing of sensor data, of course the estimates are not fed back into the control.



---

## Conclusions and Future Work

---

In this thesis, we investigate the state estimation problem for invariant systems on Lie groups by fusing the measurements of the input, living in the associated Lie algebra, and outputs, living in homogeneous spaces of the Lie group. Most of the prior literature on observer design for invariant systems on general Lie groups assumes availability of the direct measurements of the system inputs and outputs. This is not a case in many practical scenarios where nonideal measurements due to the sensor bias or measurement delays are present. Inspired by those practical scenarios, this thesis aimed at filling the gap in the literature by providing a framework for state estimation where the following nonideal measurements are present; input measurement with additive unknown bias, output measurement delays, and output sampling effects.

Assuming that the input measurements are corrupted by an unknown additive bias, we propose an observer design methodology for adaptively eliminating the bias. Such a problem is motivated by attitude and position of mechanical systems where measurements of inertial sensors (gyros or accelerometers) are prone to unknown bias. We show that the corresponding standard error dynamics are non autonomous in general, except for the trivial case where the underlying Lie group is Abelian. We design the innovation terms of the observer by systematically constructing cost functions on the associated homogeneous spaces. A verifiable condition on the stabilizer of the reference outputs associated with the output spaces ensures the stability of the observer.

To tackle the output measurement delay problem, we propose an observer-predictor methodology. This problem is motivated by real world applications such as GPS delay problem in pose estimation of vehicles, attitude estimation using optical sensors with significant delay, or indoor navigation using a VICON system or OptiTracks

with communication delays. Given an observer that has the desired stability properties when the system outputs are delay-free, we effectively employ the invariance of the system dynamics and propose novel predictors such that the combined observer-predictor preserves those stability properties when the measurements are delayed.

We also investigate the problem of state estimation with sampled output measurements when the system input is measured with a significantly larger sampling rate compared to the outputs. Such a scenario is motivated by pose estimation of UAVs equipped with high rate IMUs and low cost GPS units with low sampling rates. We propose output predictors to tackle the sensor sampling problem. The output predictors take the sampled (and delayed) measurements, employ the continuous input measurements, and provide predictions of the current outputs of the system. The proposed output predictors can then be fed into an observer in a cascade predictor-observer arrangement to estimate the current state.

The proposed state estimation methodologies in this thesis are computationally very cheap and are robust in practice, making them ideal for embedded implementation on low cost robotics systems. We demonstrate applications of the proposed methods by applying them to real world application scenarios involving attitude, velocity, and position estimation of UAVs. We provide test results with realistic simulations using Matlab and the software-in-the-loop system of ArduPilot, post-processing of offline sensory data, as well as implementation on real UAVs and performing real flight test with the ArduPilot system.

In the following, we discuss three problems that can be investigated as potential future works.

## 5.1 Reverse predictor theory

In Chapter 3, we identify three classes of systems on Lie groups for which we are able to propose recursive predictors. Those classes are; left-invariant, right invariant, and mixed-invariant systems. One might now ask, are there more general classes of systems with symmetry for which recursive predictors exist? What are the symmetry requirements of a dynamical system and the structure of its underlying symmetry groups that yield existence of a recursive predictor? Tackling these questions requires development of a reverse theory for predictor design. In this section, we formulate

the reverse predictor design problem in a general case and we provide highlights on possible solutions to this problem in a simplified case.

Let  $\mathcal{X}$  and  $\mathcal{U}$  be finite dimensional smooth real manifolds that are termed, respectively, the state and input spaces. Consider the system

$$\dot{x}(t) = f_x(x(t), u(t)), \quad x(0) = x_0, \quad (5.1)$$

where  $f_x : \mathcal{X} \times \mathcal{U} \rightarrow T\mathcal{X}$  is a smooth vector field. A recursive predictor that takes the delayed state  $x(t - \tau)$  and the input measurements  $u(t)$  (together with the past trajectories of these measurements) and predicts the current state is given by

$$\dot{\delta}(t) = f_\delta(\delta|_{[0,t]}, x|_{[0,t-\tau]}, u|_{[0,t]}), \quad (5.2)$$

$$x^p(t) = h_p(\delta|_{[0,t]}, x|_{[0,t-\tau]}, u|_{[0,t]}), \quad (5.3)$$

where  $\delta \in \mathcal{M}_\delta$  is the internal state of the predictor (living in the manifold  $\mathcal{M}_\delta$ ) and  $x^p(t) \in \mathcal{X}$  is the prediction of  $x(t)$ . Note that in general the dynamics (5.2) is a delayed differential equation since its right hand side may depend on the past trajectory of  $\delta$ . Also, note that all of the available information, including the history of the trajectories  $x|_{[0,t-\tau]}$  and  $u|_{[0,t]}$ , are used in the structure of the predictor. The predictors that we proposed in this thesis are all special cases of the general form (5.2)-(5.3). In fact, the right-invariant, the left-invariant, and the mixed-invariant predictors proposed in Chapter 3 are all examples where the function  $f_\delta$  is independent of the trajectory  $x|_{[0,t-\tau]}$  and depends only on the current input measurement  $u(t)$  and the current predictor state  $\delta(t)$ . In this case, the predictor dynamics (5.2) simplifies to an ordinary differential equation. Also, the function  $h_p$  depends only on  $x(t - \tau)$ ,  $\delta(t)$ , and  $\delta(t - \tau)$ . It is worth mentioning that the predictor proposed in the Author's previous work [81] for systems on  $\mathbb{R}^n$  is also a special case of the general form (5.2)-(5.3) where the function  $f_\delta$  depends on  $x(t - \tau)$ ,  $\delta(t)$ ,  $\delta(t - \tau)$ , and  $u(t)$  yielding a delayed differential equation as the predictor's internal dynamics. In that example, the function  $h_p$  depends only on  $x(t - \tau)$ ,  $\delta(t)$ , and  $\delta(t - \tau)$ .

Obtaining general classes of system (5.1) for which a general predictor of the form (5.2)-(5.3) exist is a complicated problem. Let us consider a more specific case where the manifolds  $\mathcal{X}$  and  $\mathcal{M}_\delta$  are copies of the same Lie group, the predictor dynamics

depend only on the variables  $\delta(t)$ ,  $u(t)$ , and  $u(t - \tau)$ , and the function  $h_p$  is a simple group multiplication. In this conditions, interesting insight for tackling the recursive predictor theory is obtained by adopting the ideas presented in [28, Theorem 1] for the observer design problem. Consider the system

$$\dot{X}(t) = f(X(t), u(t)), \quad X(0) = X_0, \quad (5.4)$$

where  $X \in G$ ,  $u \in \mathcal{M}_u$ , and  $f(X, u) \in T_X G$ . Assume that the system (5.4) is controllable in a sense that for any given initial condition  $X_0 \in G$  and final state  $X_f \in G$  there exist an input trajectory that can take the state from  $X_0$  to  $X_f$  in finite time.

Choose a trajectory  $X(t)$  of system (5.4) for  $t \geq \tau$  (with a constant  $\tau$ ) and obtain the trajectory  $X(t - \tau)$ . Define

$$\eta(t) := X(t - \tau)^{-1}X(t), \quad t \geq \tau. \quad (5.5)$$

In general, dynamics of  $\eta(t)$  may depend on the variables  $X(t)$ ,  $X(t - \tau)$ ,  $\eta(t)$ ,  $u(t)$ , and  $u(t - \tau)$ . If the dynamics of  $\eta(t)$  depends only on the variables  $\eta(t)$ ,  $u(t)$ , and  $u(t - \tau)$  as

$$\dot{\eta}(t) = g(\eta(t), u(t), u(t - \tau)), \quad (5.6)$$

one can employ this dynamics to generate the trajectory of  $\eta(t)$  (using only the input information) and then resort to (5.5) to obtain the prediction of the current state by  $X^p(t) = X(t - \tau)\eta(t)$ . Of course this condition on the dynamics of  $\eta(t)$  is only a necessary condition for existence of a recursive predictor since one should also ensure that the initial condition of (5.6) can be chosen appropriately (independent of the knowledge of the current state) such that  $X^p(t) = X(t)$  for all  $t \geq \tau$ . This is an additional requirement that should be investigated later. Let us now focus only on the requirement that the dynamics of  $\eta(t)$  is of the form (5.6). We show that, dynamics of  $\eta(t)$  has the form (5.6) for all  $\tau \geq 0$ , all  $t \geq \tau$ , and all given trajectories  $X(t)$  of the system (5.4) with their corresponding input signal  $u(t)$ , if  $f$  satisfies

$$\begin{aligned} & f(X(t - \tau)\eta(t), u(t)) \\ &= f(X(t - \tau), u(t - \tau))\eta(t) - X(t - \tau)f(I, u(t - \tau))\eta(t) + X(t - \tau)f(\eta(t), u(t)). \end{aligned} \quad (5.7)$$

Moreover, the function  $g$  (defining the dynamics of  $\eta(t)$ ) is given by

$$g(\eta(t), u(t), u(t - \tau)) = f(\eta(t), u(t)) - f(I, u(t - \tau))\eta(t). \quad (5.8)$$

To show the above claims, we differentiate the sides of (5.5) with respect to time to obtain

$$\begin{aligned} g(\eta(t), u(t), u(t - \tau)) &= \dot{\eta}(t) \\ &= -X(t - \tau)^{-1}f(X(t - \tau), u(t - \tau))X(t - \tau)^{-1}X(t) + X(t - \tau)^{-1}f(X(t), u(t)). \end{aligned} \quad (5.9)$$

This should hold for all choices of  $u(t)$  and all  $\tau \geq 0$ . Since the system is assumed controllable and  $\tau$  is arbitrary, we can choose  $X(t - \tau) = I$  in (5.9) while  $X(t)$  and hence  $\eta(t)$  are still arbitrary. This yields

$$-f(I, u(t - \tau))\eta(t) + f(\eta(t), u(t)) = g(\eta(t), u(t), u(t - \tau)).$$

as claimed in (5.8). Substituting for  $g(\eta(t), u(t), u(t - \tau))$  into (5.9) we have

$$\begin{aligned} -X(t - \tau)^{-1}f(X(t - \tau), u(t - \tau))X(t - \tau)^{-1}X(t) + X(t - \tau)^{-1}f(X(t), u(t)) \\ = -f(I, u(t - \tau))\eta(t) + f(\eta(t), u(t)). \end{aligned} \quad (5.10)$$

Multiplying the sides by  $X(t - \tau)$  from the left and rearranging the variables yields (5.7).

It is not easy to temperate the condition (5.7) as the trajectory of  $\eta(t)$  depends on  $X(t - \tau)$  and also  $u(t - \tau)$  is certainly not independent of  $u(t)$ . Nevertheless, the condition (5.7) is a good starting point to investigate the recursive predictor theory. For instance, if we know that  $f(X(t), u(t))$  is linear on  $u(t)$  (i.e. (5.4) is a kinematic system), and  $u(t)$  is arbitrary signal such that  $u(t) \neq u(t - \tau)$  for some  $t$ , then the condition (5.7) implies that  $f$  is essentially a left-invariant vector field. To prove this, we rearranging (5.7) to obtain

$$\begin{aligned} f(X(t - \tau)\eta(t), u(t)) - X(t - \tau)f(\eta(t), u(t)) \\ = f(X(t - \tau), u(t - \tau))\eta(t) - X(t - \tau)f(I, u(t - \tau))\eta(t). \end{aligned} \quad (5.11)$$

The left hand side is a linear function of  $u(t)$  while the right hand side is a linear function on  $u(t - \tau)$ . Since  $u(t)$  is arbitrary such that  $u(t) \neq u(t - \tau)$  for some  $t$ , (5.11) can hold only if both sides are equal to zero. Hence we have

$$f(X(t - \tau)\eta(t), u(t)) = X(t - \tau)f(\eta(t), u(t)) \quad (5.12)$$

which by choosing  $\eta(t) = I$  yields that  $f(X, u) = XF(u)$  for  $F(u) := f(I, u(t)) \in \mathfrak{g}$ . Notice that this automatically yields that the right hand side of (5.11) is zero.

We can extend the above methodology by assuming that the predictor state has two parts, denoted by  $\eta_1(t) \in G$  and  $\eta_2(t) \in G$ , such that the predicted state is given by

$$X^p(t) = \eta_2(t)X(t - \tau)\eta_1(t). \quad (5.13)$$

Now, one can investigate the conditions ensuring the existence of some functions  $g_1 : G \times G \times \mathcal{M}_u \times \mathcal{M}_u \rightarrow TG$  and  $g_2 : G \times G \times \mathcal{M}_u \times \mathcal{M}_u \rightarrow TG$  such that the predictor dynamics are given by

$$\begin{aligned} \dot{\eta}_1(t) &= g_1(\eta_1(t), \eta_2(t), u(t), u(t - \tau)), \\ \dot{\eta}_2(t) &= g_2(\eta_1(t), \eta_2(t), u(t), u(t - \tau)). \end{aligned}$$

One could try to replicate the computations done to drive (5.7) and (5.8) to obtain conditions on the system dynamics and propose general structure of the functions  $g_1$  and  $g_2$  based on a given  $f$ . Our expectation is that by imposing further assumptions such as linearity of  $f$  w.r.t.  $u$ , one might be able to conclude that (5.4) has a similar form as a mixed-invariant system (3.3). Note that in this case, the functions  $g_1$  and  $g_2$  are respectively independent of  $\eta_2$  and  $\eta_1$  and both do not depend on  $u(t - \tau)$ .

## 5.2 Stochastic propagation properties of predictors

In ideal conditions where there is no measurement noise and the numerical integrations are exact, the predictors proposed in Chapters 3 and 4 are able to predict the current state for arbitrary large measurement delay or arbitrary low sampling rate. In practice, however, the larger the delay is and the lower the sampling rate is the

larger is the prediction error of the current state. This is because the proposed predictors rely on forward integration of the input measurements to obtain a prediction of the current state (see Proposition 3.4.1). Large delays or low sampling rates imply that the predictor should rely on the forward integration of the input for a longer period of time, yielding aggregation of more errors due to the input noise and numerical integration errors. Although in the previous chapters, we formulated the problem in deterministic setup, it is practically very useful to investigate the effect of stochastic input measurement noise on the accuracy of the predicted state. For example, consider the predictor (4.19)-(4.20) and assume that the vector measurements  $z_i$  are obtained using magnetometer. The covariance of the measurement noise of magnetometers and gyros can be obtained by off-line processing of sensor data. If the predicted measurements  $y^p$  are used instead of  $z(t)$  in a Kalman filter (in a predictor-observer arrangement), the covariance noise associated with  $y^p$  should be used to optimally tune the gains of the Kalman filter. Such a covariance matrix necessarily depends on the noise covariance of the gyro as well as magnetometer measurements and also on the delay and sampling time of the magnetometer. Having an analytical relationship for the covariance noise of  $y^p$  as a function of the mentioned quantities also help system engineer designers to choose between faster magnetometer with higher measurement noise and slower magnetometer with lower measurement noise. It also provides guidelines to system engineers for choosing gyros with lower measurement noise to compensate for the effect of magnetometers with low sampling rate. Yet another application of such an analytical relation is in sensor fault detection and isolation. When a new magnetometer measurements is obtained, one can compare that with the predicted value of that measurement using the previous (valid) magnetometer measurement. If the new measurement falls within the uncertainty ellipsoid of the predicted value (which is obtained using the noise covariance of the predicted output), the new measurement is identified as being valid. Otherwise, it is identified as being faulty and will not be used<sup>1</sup>.

Although the structure of the predictors proposed in this thesis are simple, formulating their statistical propagation properties might not be straight-forward since their underlying state space is a Lie group (and not an Euclidean space). There is

---

<sup>1</sup>A similar fault detection system is used in the ArduPilot EKF for identifying faulty sensor measurements.

a rich literature for stochastic filtering on Lie groups [48, 49, 139], including numerous references with focus on the attitude/pose estimation application [44, 46, 66, 98, 104, 105, 114, 126], and text books are available that provide guideline for analysis of stochastic systems on general Lie groups [42, 43]. Some guidelines tailored to the specific Lie groups  $SO(3)$  and  $SE(3)$  are given in [115, 128]. Here, we provide an analysis when the underlying Lie group is simply  $\mathbb{R}^n$ . Consider the system

$$\dot{x}(t) = u(t), \quad (5.14)$$

where  $x \in \mathbb{R}^n$  and  $u \in \mathbb{R}^n$  are the state and input, respectively. Assume that the input measurement is given by

$$y_u(t) = u(t) + n_u(t), \quad (5.15)$$

where  $n_u(t)$  is a zero mean Gaussian noise with covariance matrix  $P_n \in \mathbb{R}^{n \times n}$ . Assume that the delayed measurement of state is obtained as

$$y_x(t) = x(t - \tau) + n_x(t), \quad (5.16)$$

where  $n_x(t)$  is a zero mean Gaussian noise with covariance matrix  $P_x \in \mathbb{R}^{n \times n}$ . Resorting to Remark 4.4.3, a predictor for the current state  $x(t)$  is given by

$$\dot{\delta}(t) = y_u(t), \quad (5.17)$$

$$x^p(t) = \delta(t) - \delta(t - \tau) + y_x(t). \quad (5.18)$$

We aim to obtain the statistics of  $x^p(t)$ . According to Section 3.4, the prediction  $x^p(t)$  of the recursive predictor (5.17)-(5.18) is equivalent to the predictions provided by the following non-recursive predictor.

$$\frac{d}{ds} x_t^p(s) = y_u(s), \quad s \in [t - \tau, t], \quad (5.19)$$

$$x_t^p(t - \tau) = y_x(t), \quad (5.20)$$

$$x^p(t) = x_t^p(t). \quad (5.21)$$

The dynamics (5.19) is an example of a linear time-invariant stochastic system with

a stochastic initial condition (5.20). Statistics of such a system is found in any classic stochastic estimation textbook [21]. Namely, it can be known that the covariance of the stochastic variable  $x_t^p(t)$  follows the dynamics

$$\frac{d}{ds} P_t^p(s) = P_u, \quad (5.22)$$

$$P_t^p(t-\tau) = P_x. \quad (5.23)$$

Assuming that  $P_u$  is constant, the solution of (5.22)-(5.23) is given by  $P_t^p(s) = (s - t + \tau)P_u + P_x$ . Evaluating this solution at  $s = t$  yields the covariance of the  $x^p(t)$  as

$$P_p := P_t^p(t) = \tau P_u + P_x. \quad (5.24)$$

Hence, the covariance of  $x^p$  linearly depend on the amount of delay  $\tau$  as well as the covariance of the input measurement noise and the output (delayed state) measurement noise.

Solving the statistic propagation problem is not straight-forward in the general case when the underlying Lie group is not  $\mathbb{R}^n$ . This is however possible to obtain insight into what the answer might look like without completely solving the problem. compared to the  $\mathbb{R}^n$  scenario, one important difference in the solution of this problem in the general case is that we expect that the statistics of  $X^p(t)$  depend on the trajectory of  $X(t)$  (or equivalently the input signal  $u(t)$ ) as well. This intuition comes from the simulations undertaken in Section 3.5.1 where pure prediction error due to the input measurement noise is presented. According to Fig. 3.2, the prediction error does depend on the trajectory of the state  $X(t)$ . This dependency is verified by recalling the equations (3.11)-(3.13). In fact, assuming  $\hat{u}_l(t) = u_l(t) + n_u(t)$  where  $u_n(t)$  is the input measurement noise, and using the same derivation that is done to drive (3.11)-(3.13), one can show that when the underlying Lie group is Abelian (e.g.  $\mathbb{R}^n$ ), the term  $\text{Ad}_{X_l}$  is canceled out in (3.13) and hence the trajectory of  $X_l(t)$  does not affect the resulting statistics of the prediction error  $E_l^p(t)$  for the left-invariant case. Similar result is obtained for the right-invariant case or the mixed invariant case.

### 5.3 Predictor-observer with input measurement bias

In Chapter 4, we presented a predictor-observer methodology that is able to compensate the effect of output sampling and delays for state estimation of invariant systems on Lie groups. Co-stability of the proposed predictor is proved when direct measurements of system inputs are available. In some applications, however, measurements of system inputs are corrupted by unknown biases that should be compensated adaptively, as was explained in Chapter 2. Although the presented stability proofs in Chapter 4 do not cover the case where input measurement bias is present, we demonstrated both via simulation and practical experiments in Section 4.6 that the resulting predictors provide good estimates when we feed the predictors with bias compensated inputs provided by the observers used in the predictor-observer arrangement. Stability analysis of the resulting system is not straight forward since injecting the bias estimates of the observer back to the predictor in the predictor-observer arrangement creates a feedback loop which might potentially cause instability (though we did not observe this potential instability in our tests)<sup>2</sup>. This raises the possibility that the resulting feedback loop might be indeed stable, at least locally and for careful tuning of the observer gains. So, as a future work in this section, we propose to theoretically investigate the stability of the resulting feedback system<sup>3</sup>. Let us recall the simplified example of the integrator system on  $\mathbb{R}^n$  given by (5.14). Assume that measurements of  $x(t)$  are delay-free, but measurements of  $u(t)$  are corrupted by unknown constant bias as  $y_u(t) = u(t) + b$ . The simple linear adaptive observer

$$\dot{\hat{x}}(t) = y_u(t) - \hat{b}(t) - k(\hat{x}(t) - x(t)), \quad (5.25)$$

$$\dot{\hat{b}}(t) = \gamma(\hat{x}(t) - x(t)) \quad (5.26)$$

<sup>2</sup>Note that this feedback loop does not appear in the observer-predictor arrangement and that is why we are able to provide stability analysis even in presence of input bias (and scaling) in Chapter 3.

<sup>3</sup>There is also another point of view for tackling the input bias problem; is there an alternative method for compensating the input bias in the predictor-observer arrangement, other than directly feeding the bias estimates of the observer back into the predictor? Is there a method that does not create a feedback loop or has a simple stability analysis? The answer to these questions are not clear to the Author at the time of writing this chapter, but it would be very interesting to consider designing a bias estimation methodology that is independent of the structure of the observer employed in the predictor-observer arrangement. Such a methodology may benefit from the inherent input to output stability properties of the predictors of Chapter 4 for stability analysis. Note that the mentioned input to output stability properties can be mathematically formulated by employing the inequalities (3.12) and (3.22) and resorting to Lemma 6.4.1.

with  $k, \gamma > 0$  ensures globally exponential convergence of  $\hat{x}(t)$  and  $\hat{b}(t)$  to  $x(t)$  and  $b$ , respectively. Where the delayed measurements  $x(t - \tau)$  are available (instead of  $x(t)$ )<sup>4</sup>, we use this observer in a predictor-observer arrangement as follows.

$$\dot{\delta}(t) = y_u(t) - \hat{b}(t), \quad (5.27)$$

$$x^p(t) = x(t - \tau) + \delta(t) - \delta(t - \tau), \quad (5.28)$$

$$\dot{\hat{x}}(t) = y_u(t) - \hat{b}(t) - k(\hat{x}(t) - x^p(t)), \quad (5.29)$$

$$\dot{\hat{b}}(t) = \gamma(\hat{x}(t) - x^p(t)) \quad (5.30)$$

Defining the estimation errors  $\tilde{x}(t) = \hat{x}(t) - x(t)$  and  $\tilde{b}(t) = \hat{b}(t) - b$  we have

$$\dot{\tilde{x}}(t) = -\tilde{b}(t) - k(\hat{x}(t) - x(t - \tau) + \delta(t) - \delta(t - \tau)), \quad (5.31)$$

$$\dot{\tilde{b}}(t) = \gamma(\hat{x}(t) - x(t - \tau) + \delta(t) - \delta(t - \tau)). \quad (5.32)$$

Integrating the sides of (5.14) and (5.27) we have  $x(t - \tau) = x(t) - \int_{t-\tau}^t u(s)ds$  and  $\delta(t) - \delta(t - \tau) = \int_{t-\tau}^t u(s)ds - \int_{t-\tau}^t \tilde{b}(s)ds$ , respectively. Substituting for  $\hat{x}(t) - x(t - \tau) + \delta(t) - \delta(t - \tau) = \tilde{x}(t) - \int_{t-\tau}^t \tilde{b}(s)ds$  into (5.31) and (5.32) yields the following error dynamics.

$$\dot{\tilde{x}}(t) = -\tilde{b}(t) - k\tilde{x}(t) + k \int_{t-\tau}^t \tilde{b}(s)ds, \quad (5.33)$$

$$\dot{\tilde{b}}(t) = \gamma\tilde{x}(t) - \gamma \int_{t-\tau}^t \tilde{b}(s)ds. \quad (5.34)$$

Interestingly, the error dynamics (5.33)-(5.34) has the form of a Lyapunov-krasovskii type dynamics where the right hand side depends on the integral of the error in addition to linear error terms. Stability of such error dynamics can be investigated by employing the Lyapunov-krasovskii function  $W(t) = \frac{1}{2}\|\tilde{x}(t)\|^2 + \frac{1}{\gamma}\|\tilde{x}(t)\|^2 + w \int_{t-\tau}^t (s - t + \tau)\|\tilde{b}(s)\|^2 ds$  with  $w > 0$  (see e.g. [134]). One can try to show that the error dynamics (5.33)-(5.34) is asymptotically stable to  $(\tilde{x}(t), \tilde{b}(t)) = (0, 0)$  if the constants  $k, \gamma, w$  are chosen appropriately. Extending the above analysis to the general case where the underlying Lie group is not  $\mathbb{R}^n$  is obviously not straight-forward. The major technical difficulty in this case is that an analytical solution to the dynamics (3.1), (3.2), or (3.3) does not exist in general. Nevertheless, one may be able to resort

<sup>4</sup>For simplicity, we don't consider sampled outputs for the analysis.

to the derivations made in the proofs of Theorem 3.3.2 and 3.3.3 and try to replicate the above proof without employing the analytical solution of the system dynamics.

## 5.4 Application examples to other Lie groups

From the application point of view, a strong focus of this thesis is applications to navigation and control of mechanical systems evolving on the Lie groups  $SO(3)$  and  $SE(3)$ , although the presented theory is developed for general Lie groups. Investigation of systems on  $SO(n)$  and  $SE(n)$  seem to be very popular in the geometric observer design literature, and have demonstrated successful applications. There has been efforts in investigating application examples of other Lie groups, such as the homography estimation on the special linear group  $SL(3)$  [56] and the state estimation on the multiplicative group  $\mathbb{R}_+^*$  with application to a chemical reactor [34]. Nevertheless, investigate application examples of the developed state estimation theory to systems evolving on other Lie groups would be a very interesting topic. Lie groups surely have wide range of applications in areas other than navigation and control of mechanical systems. For instance, the Special Linear group  $SL(2)$  arises in some computer vision applications [44, 73, 96]. Also, complex-valued Lie groups and unitary groups arise in multiantenna transceiver techniques [142], wave propagation and scattering involving polarized waves [45], holographic memory design and analysis [146], complex-valued artificial neural networks learning [50], and in signal processing is blind source separation [51], to mention a few. Investigating applications of the geometric observer design techniques to such problems is of high value and is highly encouraged as a potential future work.

---

# Appendix

---

## 6.1 Lemma 6.1.1

**Lemma 6.1.1.** *Suppose that the  $\mathbb{R}^6$  representation of  $u \in \mathfrak{se}(3)$  and  $w \in T_{(\hat{R}, \hat{p})}SE(3)$  with respect to the basis  $\{\bar{e}\}$  and  $\{\bar{e}\hat{X}\}$  are respectively given by  $\llbracket u \rrbracket_{\bar{e}} = [u_\omega^\top, u_v^\top]^\top$  and  $\llbracket w \rrbracket_{\bar{e}\hat{X}} = [w_\omega^\top, w_v^\top]^\top$  where  $u_\omega, u_v, w_\omega, w_v \in \mathbb{R}^3$ . Then  $u$  and  $w$  can be written in terms of their  $\mathbb{R}^6$  representation as follows.*

$$u = (u_{\omega \times}, u_v) \tag{6.1}$$

$$w = (w_{\omega \times} \hat{R}, w_{\omega \times} \hat{p} + w_v). \tag{6.2}$$

□

*Proof:*

$$\begin{aligned} w &= w_\omega^\top e^1 (e_\times^1 \hat{R}, e_\times^1 \hat{p}) + w_\omega^\top e^2 (e_\times^2 \hat{R}, e_\times^2 \hat{p}) + w_\omega^\top e^3 (e_\times^3 \hat{R}, e_\times^3 \hat{p}) \\ &\quad + w_v^\top e^1 (0_{3 \times 3}, e^1) + w_v^\top e^2 (0_{3 \times 3}, e^2) + w_v^\top e^3 (0_{3 \times 3}, e^3) \\ &= \left( (w_\omega^\top e^1 e_\times^1 + w_\omega^\top e^2 e_\times^2 + w_\omega^\top e^3 e_\times^3) \hat{R}, \right. \\ &\quad \left. (w_\omega^\top e^1 e_\times^1 + w_\omega^\top e^2 e_\times^2 + w_\omega^\top e^3 e_\times^3) \hat{p} + w_v^\top e^1 e^1 + w_v^\top e^2 e^2 + w_v^\top e^3 e^3 \right) \\ &= (w_{\omega \times} \hat{R}, w_{\omega \times} \hat{p} + w_v) \end{aligned}$$

where we used the standard equation  $a = a^\top e^1 e^1 + a^\top e^2 e^2 + a^\top e^3 e^3$  once for  $a = w_\omega$  and once for  $a = w_v$  to obtain the last line. This proves (6.2). Choosing  $(\hat{R}, \hat{p}) = (I_{3 \times 3}, 0)$ , it is easy to verify that (6.1) holds too. ■

## 6.2 Lemma 6.2.1

**Lemma 6.2.1.** *We have  $\underline{\sigma}(\text{Ad}_X) \geq \text{cond}(X)^{-1}$  and  $\bar{\sigma}(\text{Ad}_X) \leq \text{cond}(X)$  for all  $X \in G$ .  $\square$*

*Proof:* Embed  $\mathfrak{g}$  into  $\mathbb{R}^{m \times m}$ . Denote the matrix representation of  $\text{Ad}_X : \mathbb{R}^{m \times m} \rightarrow \mathbb{R}^{m \times m}$  w.r.t. the standard basis for its domain and co-domain by  $\llbracket \text{Ad}_X \rrbracket$ . Since  $\dim(\mathfrak{g}) \leq m^2$  we have  $\underline{\sigma}(\text{Ad}_X) \geq \underline{\sigma}(\llbracket \text{Ad}_X \rrbracket)$  and  $\bar{\sigma}(\text{Ad}_X) \leq \bar{\sigma}(\llbracket \text{Ad}_X \rrbracket)$ . Invoking the property  $\text{vec}(XwX^{-1}) = X^{-\top} \otimes X \text{vec}(w)$  where  $\text{vec}(w) \in \mathbb{R}^{m^2}$  is the vectorization of the matrix  $w \in \mathbb{R}^{m \times m}$  and  $\otimes$  denotes the Kronecker product, one can conclude that  $\llbracket \text{Ad}_X \rrbracket = X^{-\top} \otimes X$ . This implies that  $\underline{\sigma}(\text{Ad}_X) \geq \underline{\sigma}(\llbracket \text{Ad}_X \rrbracket) = \underline{\sigma}(\Phi(X)^{-\top}) \underline{\sigma}(\Phi(X)) = \bar{\sigma}(\Phi(X))^{-1} \underline{\sigma}(\Phi(X)) = \text{cond}(\Phi(X))^{-1}$ . Similarly, we have  $\bar{\sigma}(\text{Ad}_X) \leq \bar{\sigma}(\llbracket \text{Ad}_X \rrbracket) = \bar{\sigma}(X^{-\top}) \bar{\sigma}(X) = \underline{\sigma}(X)^{-1} \bar{\sigma}(X) = \text{cond}(X)$ . This completes the proof.  $\blacksquare$

## 6.3 Lemma 6.3.1

**Lemma 6.3.1.** *We have  $d_r(X, Y) \leq \bar{\sigma}(\text{Ad}_Z) d_r(Z^{-1}X, Z^{-1}Y)$  for all  $X, Y, Z \in G$ .  $\square$*

*Proof:* Define a right-invariant distance between  $X, Y \in G$  as the infimum of the lengths of the curves connecting  $X$  and  $Y$  w.r.t. the metric  $\langle \cdot, \cdot \rangle_r$ , such that  $d_r(X, Y) = \inf_{\gamma(0)=X, \gamma(1)=Y} \int_0^1 |\dot{\gamma}(s)|_r^{\gamma(s)} ds$ . Using a change of variable  $\tilde{\gamma}(s) := Z^{-1}\gamma(s)$  we have

$$\begin{aligned}
 d_r(X, Y) &= \inf_{\tilde{\gamma}(0)=Z^{-1}X, \tilde{\gamma}(1)=Z^{-1}Y} \int_0^1 |Z\dot{\tilde{\gamma}}(s)|_r^{Z\tilde{\gamma}(s)} ds \\
 &= \inf_{\tilde{\gamma}(0)=Z^{-1}X, \tilde{\gamma}(1)=Z^{-1}Y} \int_0^1 \|Z\dot{\tilde{\gamma}}(s)\tilde{\gamma}(s)^{-1}Z^{-1}\| ds \\
 &\leq \bar{\sigma}(\text{Ad}_Z) \inf_{\tilde{\gamma}(0)=Z^{-1}X, \tilde{\gamma}(1)=Z^{-1}Y} \int_0^1 \|\dot{\tilde{\gamma}}(s)\tilde{\gamma}(s)^{-1}\| ds \\
 &= \bar{\sigma}(\text{Ad}_Z) \inf_{\tilde{\gamma}(0)=Z^{-1}X, \tilde{\gamma}(1)=Z^{-1}Y} \int_0^1 |\dot{\tilde{\gamma}}(s)|_r^{\tilde{\gamma}(s)} ds \\
 &= \bar{\sigma}(\text{Ad}_Z) d_r(Z^{-1}X, Z^{-1}Y).
 \end{aligned}$$

$\blacksquare$

## 6.4 Lemma 6.4.1

**Lemma 6.4.1.** Consider the system  $\dot{\Xi}_l(t) = u_l^b(t)\Xi_l(t)$  (resp.  $\dot{\Xi}_r(t) = \Xi_r(t)u_r^b(t)$ ) with  $\Xi_l(0) = \Xi_0 \in G$  (resp.  $\Xi_r(0) = \Xi_0 \in G$ ) where  $u_l^b(t), u_r^b(t) \in \mathfrak{g}$ .

- (I) If  $u_l^b(t)$  (resp.  $u_r^b(t)$ ) is bounded for all  $t \geq 0$ , then  $d_l(\Xi_r(t-\tau), \Xi_r(t))$  (resp.  $d_l(\Xi_r(t-\tau), \Xi_r(t))$ ) is bounded for all  $\tau \geq 0$  and all  $t \geq \tau$ .
- (II) If  $\|u_l^b(t)\| \rightarrow 0$  (resp.  $\|u_r^b(t)\| \rightarrow 0$ ), then  $d_r(\Xi_l(t-\tau), \Xi_l(t)) \rightarrow 0$  (resp.  $d_l(\Xi_r(t-\tau), \Xi_r(t)) \rightarrow 0$ ) for all  $\tau \geq 0$ .
- (III) If  $\|u_l^b(t)\| \xrightarrow{\text{exp}} 0$  (resp.  $\|u_r^b(t)\| \xrightarrow{\text{exp}} 0$ ), then  $d_r(\Xi_l(t-\tau), \Xi_l(t)) \xrightarrow{\text{exp}} 0$  (resp.  $d_l(\Xi_r(t-\tau), \Xi_r(t)) \xrightarrow{\text{exp}} 0$ ) for all  $\tau \geq 0$ .

□

*Proof:* We prove the Lemma for the right-invariant system  $\dot{\Xi}_l(t) = u_l^b(t)\Xi_l(t)$  and the right-invariant distance  $d_r(\cdot, \cdot)$ . The proof for the left-invariant case can be obtained similarly. The length of the curve  $s \mapsto \Xi_l(s)$ ,  $s \in [t-\tau, t]$  connecting  $\Xi_l(t-\tau)$  to  $\Xi_l(t)$  w.r.t. the metric  $\langle \cdot, \cdot \rangle^r$  is given by  $L_r(\Xi_l(t-\tau), \Xi_l(t)) = \int_{t-\tau}^t |\dot{\Xi}_l(s)|_r^{\Xi_l(s)} ds = \int_{t-\tau}^t |u_l^b(s)\Xi_l(s)|_r^{\Xi_l(s)} ds = \int_{t-\tau}^t \|u_l^b(s)\| ds$ .

If  $u_l^b(t)$  is bounded, there exists a constant  $c_u$  such that  $\|u_l^b(t)\| \leq c_u$  for all  $t \geq 0$ . Hence  $L_r(\Xi_l(t-\tau), \Xi_l(t)) \leq \tau c_u$  and consequently  $d_r(\Xi_l(t-\tau), \Xi_l(t))$  is bounded by  $d_r(\Xi_l(t-\tau), \Xi_l(t)) \leq L_r(\Xi_l(t-\tau), \Xi_l(t)) \leq \tau c_u$ . This proves part (I).

If  $\|u_l^b(t)\| \rightarrow 0$ , then for all  $\epsilon_u > 0$  there exist a  $T_u$  such that for all  $t \geq T_u$  we have  $\|u_l^b(t)\| < \epsilon_u$  and hence,  $L_r(\Xi_l(t-\tau), \Xi_l(t)) < \tau \epsilon_u$ . Hence, we have  $d_r(\Xi_l(t-\tau), \Xi_l(t)) \leq L_r(\Xi_l(t-\tau), \Xi_l(t)) < \tau \epsilon_u$  for all  $t \geq T_u$ . Consequently,  $d_r(\Xi_l(t-\tau), \Xi_l(t)) \rightarrow 0$ . This proves part (II).

If  $\|u_l^b(t)\| \xrightarrow{\text{exp}} 0$ , then there exist positive constants  $c$  and  $\alpha > 0$  such that  $\|u_l^b(t)\| \leq c \exp(-\alpha t)$  for all  $t > 0$ . We have  $L_r(\Xi_l(t-\tau), \Xi_l(t)) = \int_{t-\tau}^t \|u_l^b(s)\| ds \leq \int_{t-\tau}^t c \exp(-\alpha s) ds = \frac{c}{\alpha} (\exp(\alpha\tau) - 1) \exp(-\alpha t)$ . Hence  $d_r(\Xi_l(t-\tau), \Xi_l(t)) \leq L_r(\Xi_l(t-\tau), \Xi_l(t)) \xrightarrow{\text{exp}} 0$ . This proves part (III). ■

## 6.5 Predictor C++ library for ArduPilot

This section provides explanations on the C++ code developed for the experimental tests presented in Section 4.6. A copy of the full ArduPilot software including the

developed code in this thesis is give in the CD attached to the thesis and is also available online at [13]. In the ArduPilot directory, there are two directories for the EKFs. The original EKF of ArduPilot is at the path `\ardupilot\libraries\AP_NavEKF` (which is referred to by EKF1) while the developed predictor library in this thesis along with its combined EKF is at the path `\ardupilot\libraries\AP_NavEKF2` (which is referred to by EKF2). In the directory `AP_NavEKF2`, `AP_Predictors.h` and `AP_Predictors.cpp` are the header file and the body of the predictor library, respectively, while `AP_NavEKF2.h` and `AP_NavEKF2.cpp` contain the header and body of EKF2, respectively. The code has been developed in a object oriented manner, meaning that `AP_Predictors` and `AP_NaveEKF2` are two classes. An object belonging to the class `AP_Predictors` is defined in `AP_NaveEKF2` to facilitate combining the developed predictor with EKF2 (see Section 6.5.2). Explanations and comments are given in the code to help the reader better understand the code. In the following, we briefly explain the correspondence between the functions within the predictor library and the predictor equations given in Section 4.6.2. We also discuss how the predictor library is combined with EKF2.

### 6.5.1 AP\_Predictors

As was explained in Sections 4.4 and 4.6.2, the states of the predictor need to be buffered. The length of the buffer should be at least equal to the maximum amount of the measurement delay amongst all of the sensors. Size of the buffer in the developed code is determined by the variable `BUFFER_SIZE` in `AP_Predictors.h`. Currently, this variable is set to 50. Noting that ArduPilot runs at 50 Hz on planes<sup>1</sup>, the maximum amount of delay allowed in the current configuration is  $50 \times \frac{1}{50}$  (ms) = 1000 (ms)<sup>2</sup>. This is well beyond the usual delay of 220 (ms) assumed for commercial GPS units and set in EKF2. The functions within `AP_Predictors.cpp` are as follows.

- `UpdatePredictorStates` : This function calls three functions to update the internal states of the predictor according to the discretized predictor dynamics (4.29)-(4.31). Namely, the function `AttitudeModel`, `VelModel`, and `PosModel` implement the attitude kinematics (4.29), velocity kinematics (4.30), and the

<sup>1</sup>Recall that we used model planes for the experimental tests of Section 4.6

<sup>2</sup>Note that ArduPilot runs at 200 Hz on platforms other than planes, i.e. copters and rovers. For those platforms, the maximum allowed delay drops to  $50 \times \frac{1}{200}$  (ms) = 250(ms)

---

position kinematics (4.31), respectively. The function `UpdatePredictorStates` is called in the prediction stage of EKF2 yielding the predictor states to be updated at the same rate as EKF2 states.

- `VectorPredictor` : This function implements the static predictor equation (4.32) used for predicting the current magnetic field vector. This function is called at the update stage of EKF2 where magnetic field measurement should be fused.
- `VelPredictor` : This function implements the static predictor equation (4.33) used for predicting the current linear velocity vector of vehicle. This function is called at the update stage of EKF2 where velocity measurement should be fused.
- `PosNEPredictor` : This function implements the static predictor equation (4.34) used for predicting the current North and East elements of the position of vehicle. This function is called at the update stage of EKF2 where North and East position measurement should be fused.
- `HgtPredictor` : This function implements the static predictor equation (4.35) used for predicting the current Down elements of the position of vehicle (i.e. negative of height). This function is called at the update stage of EKF2 where Down position measurement should be fused.
- The functions `storeDataVector` and `storeDataQuaternion` are used for storing 3D vector elements (i.e.  $\delta v_{k+1}$  and  $\delta p_{k+1}$ ) and quaternion elements (i.e.  $q_{k+1}$ ) of the predictor states into buffers. These functions also store the associated time stamps of those vector and quaternion elements into separate buffers. These functions are used in `AttitudeModel`, `VelModel`, and `PosModel` to store the predictor states after each update.
- The function `BestIndex` reads a given time stamp buffer and finds the time stamp that is closest to a given time. This function is used in `VectorPredictor`, `VelPredictor`, `PosNEPredictor`, and `HgtPredictor`.

### 6.5.2 Combining AP\_Predictors with AP\_NavEKF2

In order to combine the predictor with EKF2, the object `ANU_Predictor` belonging to the class `AP_Predictors` is defined in `AP_NavEKF2.h` as a member of the class `AP_NavEKF2` (see the line 98 of `AP_NavEKF2.h`). This way, the functions defined in the class `AP_Predictors` can be used within the class `AP_NavEKF2`. One can simply search for "`ANU_Predictor`" in `AP_NavEKF2.cpp` to find the exact lines of the code where the functions of the predictor library are called. The predictor states are updated by calling the function `ANU_Predictor.UpdatePredictorStates` in the function `UpdateStrapdownEquationsNED` which is the prediction stage of EKF2 (see the line 1213 of `AP_NavEKF2.cpp`). The functions `VelPredictor`, `PosNEPredictor`, and `HgtPredictor` are all called within the function `FuseVelPosNED` where the velocity and position measurements are fused (note that the predicted measurements are fused instead of the actual measurements). The function `VectorPredictor` is called within `FuseMagnetometer` where the predicted magnetic field is fused.

---

# Bibliography

---

1. 3DRobotics. <https://3dr.com/>. (visited in September 2016). (cited on pages 4 and 88)
2. . APM: Navigation extended Kalman filter overview. <http://copter.ardupilot.com/wiki/common-apm-navigation-extended-kalman-filter-overview/>. (visited in September 2014). (cited on pages 44, 65, 71, and 85)
3. . ArduPilot Developers. <http://dev.ardupilot.com/>. (visited in July 2015). (cited on pages 88 and 90)
4. . ArduPilot NavEKF Library Header. [https://github.com/diydrones/ardupilot/blob/master/libraries/AP\\_NavEKF/AP\\_NavEKF.h](https://github.com/diydrones/ardupilot/blob/master/libraries/AP_NavEKF/AP_NavEKF.h). (visited in July 2015). (cited on pages 2 and 90)
5. . ArduPilot project. <https://github.com/diydrones/ardupilot>. (visited in January 2015). (cited on pages 3, 44, 64, and 88)
6. Canberra Model Aircraft Club. <http://www.cmac.org.au/>. (visited in July 2015). (cited on page 94)
7. CanberraUAV. <http://www.canberrauav.com/>. (visited in January 2015). (cited on pages 3, 44, 64, and 88)
8. IMU data fusing: Complementary, Kalman, and Mahony filter. <http://www.olliw.eu/2013/imu-data-fusing/>. (visited in October 2015). (cited on page 3)
9. JSBSim. <http://jsbsim.sourceforge.net/>. (visited in January 2015). (cited on page 65)
10. MAVLink. <http://qgroundcontrol.org/mavlink/start>. (visited in July 2015). (cited on page 90)
11. MAVProxy. <https://github.com/ArduPilot/MAVProxy>. (visited in September 2016). (cited on pages 44, 64, 65, 88, and 90)

12. Open source IMU and AHRS algorithms. <http://www.x-io.co.uk/open-source-imu-and-ahrs-algorithms/>. (visited in October 2015). (cited on page 3)
13. Predictor C++ library. <https://github.com/alirezakhosravian/ardupilot/tree/tridge-predictor-observer-wip>. (visited in July 2015). (cited on pages 89 and 120)
14. . Setting up SITL on Linux. <http://dev.ardupilot.com/wiki/setting-up-sitl-on-linux/>. (visited in January 2015). (cited on pages 44, 64, and 88)
15. ABSIL, P.-A.; MAHONY, R.; AND SEPULCHRE, R., 2009. *Optimization algorithms on matrix manifolds*. Princeton University Press. (cited on page 25)
16. AGGOUNE, W.; BOUTAYEB, M.; AND DAROUACH, M., 1999. Observers design for a class of nonlinear systems with time-varying delay. In *Proc. IEEE Conf. on Decision and Control*, vol. 3, 2912–2913. (cited on pages 5, 42, and 99)
17. AGRACHEV, A. A. AND SACHKOV, Y., 2004. *Control theory from the geometric viewpoint*, vol. 2. Springer. (cited on pages 1 and 12)
18. AHMED-ALI, T.; CHERRIER, E.; AND LAMNABHI-LAGARRIGUE, F., 2012. Cascade high gain predictors for a class of nonlinear systems. *IEEE Trans. Autom. Control*, 57, 1 (2012), 221–226. (cited on pages 5, 42, 71, 85, and 99)
19. AHMED-ALI, T.; KARAFYLLIS, I.; AND LAMNABHI-LAGARRIGUE, F., 2013. Global exponential sampled-data observers for nonlinear systems with delayed measurements. *Systems & Control Letters*, 62, 7 (2013), 539–549. (cited on pages 5, 42, 43, 55, 70, and 71)
20. ALLIBERT, G.; ABEYWARDENA, D.; BANGURA, M.; AND MAHONY, R., 2014. Estimating body-fixed frame velocity and attitude from inertial measurements for a quadrotor vehicle. In *Proc. IEEE Conference on Control Applications*, 978–983. (cited on pages 46 and 48)
21. ANDERSON, B. D. AND MOORE, J. B., 2012. *Optimal filtering*. Courier Corporation. (cited on page 113)
22. ANDERSON, R. J. AND SPONG, M. W., 1989. Bilateral control of teleoperators with time delay. *IEEE Trans. Autom. Control*, 34, 5 (1989), 494–501. (cited on page 2)

- 
23. ARČAK, M. AND NEŠIĆ, D., 2004. A framework for nonlinear sampled-data observer design via approximate discrete-time models and emulation. *Automatica*, 40, 11 (2004), 1931–1938. (cited on page 70)
  24. BAHRAMI, S. AND NAMVAR, M., 2014. Delay compensation in global estimation of rigid-body attitude under biased velocity measurement. In *Proc. IEEE Conf. on Decision and Control*, 481–486. (cited on pages 5, 42, 46, 71, 85, and 99)
  25. BALDWIN, G.; MAHONY, R.; TRUMPF, J.; HAMEL, T.; AND CHEVIRON, T., 2007. Complementary filter design on the special euclidean group SE(3). In *Proc. European Control Conference*. (cited on pages 1 and 16)
  26. BAR-SHALOM, Y., 2002. Update with out-of-sequence measurements in tracking: exact solution. *IEEE Trans. Aerospace and Electronic Systems*, 38, 3 (2002), 769–777. (cited on page 71)
  27. BAR-SHALOM, Y.; CHEN, H.; AND MALLICK, M., 2004. One-step solution for the multistep out-of-sequence-measurement problem in tracking. *IEEE Trans. Aerospace and Electronic Systems*, 40, 1 (2004), 27–37. (cited on page 71)
  28. BARRAU, A. AND BONNABEL, S., 2014. The invariant extended Kalman filter as a stable observer. *arXiv preprint arXiv:1410.1465*, (2014). (cited on pages 2, 4, 12, 42, 45, 46, 48, and 108)
  29. BATISTA, P.; SILVESTRE, C.; AND OLIVEIRA, P., 2012. A GES attitude observer with single vector observations. *Automatica*, 48 (2012), 388–395. (cited on page 31)
  30. BATTILOTTI, S., 2014. Nonlinear predictors for systems with bounded trajectories and delayed measurements. In *Proc. IFAC World Congr.*, 6812–9817. (cited on pages 5, 42, 43, and 55)
  31. BEARD, R. W., 2014. private communication. (cited on page 71)
  32. BLOCH, A. M.; BAILLIEUL, J.; CROUCH, P.; MARSDEN, J. E.; AND ZENKOV, D., 2003. *Nonholonomic mechanics and control*, vol. 24. Springer, second edn. (cited on pages 1 and 12)

33. BONNABEL, S.; MARTIN, P.; AND ROUCHON, P., 2008. Non-linear observer on Lie groups for left-invariant dynamics with right-left equivariant output. In *Proc. IFAC World Congr.*, 8594–8598. (cited on pages 4, 12, and 42)
34. BONNABEL, S.; MARTIN, P.; AND ROUCHON, P., 2008. Symmetry-preserving observers. *IEEE Trans. Autom. Control*, 53, 11 (2008), 2514–2526. (cited on pages 1, 2, 12, 15, 48, 84, and 116)
35. BONNABEL, S.; MARTIN, P.; AND ROUCHON, P., 2009. Non-linear symmetry-preserving observers on Lie groups. *IEEE Trans. Autom. Control*, 54, 7 (2009), 1709–1713. (cited on pages 4 and 28)
36. BOURBAKI, N., 1989. *Lie groups and Lie algebras: chapters 1-3*. Springer, 2nd edn. (cited on pages 10, 16, and 20)
37. BRÁS, S.; CUNHA, R.; VASCONCELOS, J. F.; SILVESTRE, C.; AND OLIVEIRA, P., 2011. A nonlinear attitude observer based on active vision and inertial measurements. *IEEE Trans. Robot.*, 27, 4 (2011), 664–677. (cited on pages 15 and 31)
38. BROCKETT, R. W., 1972. System theory on group manifolds and coset spaces. *SIAM Journal on control*, 10, 2 (1972), 265–284. (cited on page 2)
39. BROCKETT, R. W., 1973. Lie theory and control systems defined on spheres. *SIAM Journal on Applied Mathematics*, 25, 2 (1973), 213–225. (cited on page 2)
40. BULLO, F. AND LEWIS, A. D., 2005. *Geometric control of mechanical systems*, vol. 49. Springer. (cited on pages 1 and 12)
41. CACACE, F.; GERMANI, A.; AND MANES, C., 2010. An observer for a class of non-linear systems with time varying observation delay. *Systems & Control Letters*, 59, 5 (2010), 305–312. (cited on pages 5, 42, and 99)
42. CHIRIKJIAN, G. AND KOBILAROV, M., 2014. Gaussian approximation of non-linear measurement models on lie groups. In *IEEE Conf. Decision and Control*, 6401–6406. (cited on pages 2 and 112)
43. CHIRIKJIAN, G. S., 2012. *Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications*, vol. 2. Springer Science & Business Media. (cited on page 112)

- 
44. CHIRIKJIAN, G. S. AND KYATKIN, A. B., 2016. *Harmonic Analysis for Engineers and Applied Scientists: Updated and Expanded Edition*. Courier Dover Publications. (cited on pages 1, 2, 112, and 116)
  45. CLOUDE, S. R., 1994. Special unitary groups in polarimetry theory. In *SPIE's International Symposium on Optics, Imaging, and Instrumentation*, 292–303. International Society for Optics and Photonics. (cited on page 116)
  46. COMISEL, H.; FÖRSTER, M.; GEORGESCU, E.; CIOBANU, M.; TRUHLIK, V.; AND VOJTA, J., 1997. Attitude estimation of a near-Earth satellite using magnetometer data. *Acta Astronautica*, 40, 11 (1997), 781–788. (cited on pages 2 and 112)
  47. DEZA, F.; BUSVELLE, E.; GAUTHIER, J.; AND RAKOTOPARA, D., 1992. High gain estimation for nonlinear systems. *Systems & Control letters*, 18, 4 (1992), 295–299. (cited on page 70)
  48. DUNCAN, T., 1988. An estimation problem in compact Lie groups. *Systems & Control Letters*, 10, 4 (1988), 257–263. (cited on pages 2 and 112)
  49. DUNCAN, T. E., 1979. Stochastic systems in Riemannian manifolds. *Journal of Optimization theory and applications*, 27, 3 (1979), 399–426. (cited on pages 2 and 112)
  50. FIORI, S., 2008. A study on neural learning on manifold foliations: The case of the lie group  $SU(3)$ . *Neural Computation*, 20, 4 (2008), 1091–1117. (cited on page 116)
  51. FIORI, S., 2014. Auto-regressive moving average models on complex-valued matrix Lie groups. *Circuits, Systems, and Signal Processing*, 33, 8 (2014), 2449–2473. (cited on pages 2 and 116)
  52. FOSSEN, T. I., 2011. *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons. (cited on pages 71 and 85)
  53. GERMANI, A.; MANES, C.; AND PEPE, P., 2002. A new approach to state observation of nonlinear systems with delayed output. *IEEE Trans. Autom. Control*, 47, 1 (2002), 96–101. (cited on pages 2, 5, 42, 43, 55, 71, and 85)

54. GRIP, H. F.; FOSSEN, T. I.; JOHANSEN, T. A.; AND SABERI, A., 2012. Attitude estimation using biased gyro and vector measurements with time-varying reference vectors. *IEEE Trans. Autom. Control*, 57, 5 (2012), 1332–1338. (cited on page 31)
55. GRIP, H. F.; FOSSEN, T. I.; JOHANSEN, T. A.; AND SABERI, A., 2015. Globally exponentially stable attitude and gyro bias estimation with application to GNSS/INS integration. *Automatica*, 51 (2015), 158–166. (cited on pages 1, 15, 46, and 48)
56. HAMEL, T.; MAHONY, R.; TRUMPF, J.; MORIN, P.; AND HUA, M.-D., 2011. Homography estimation on the special linear group based on direct point correspondence. In *Proc. IEEE Conf. Decision and Control and European Control Conf. (CDC-ECC)*, 7902–7908. (cited on pages 1, 48, and 116)
57. HAMMOURI, H.; NADRI, M.; AND MOTA, R., 2006. Constant gain observer for continuous-discrete time uniformly observable systems. In *Proc. IEEE Conf. Decision and Control*, 5406–5411. (cited on page 70)
58. HANSEN, J. M.; FOSSEN, T. I.; AND JOHANSEN, T. A., 2015. Nonlinear observer for INS aided by time-delayed GNSS measurements: Implementation and UAV experiments. In *Proc. IEEE International Conf. Unmanned Aircraft Systems (ICUAS)*, 157–166. (cited on pages 2, 5, 42, 43, 70, and 71)
59. HARTMAN, P., 2002. *Ordinary differential equations*. Society for Industrial and Applied Mathematics, second edn. (cited on page 63)
60. HEEMELS, W. H.; TEEL, A. R.; VAN DE WOUW, N.; AND NESIC, D., 2010. Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance. *IEEE Trans. Autom. Control*, 55, 8 (2010), 1781–1796. (cited on page 70)
61. HELMKE, U. AND MOORE, J. B., 1996. *Optimization and dynamical systems*. Springer-Verlag, second edn. (cited on page 10)
62. HUA, M.-D., 2010. Attitude estimation for accelerated vehicles using GPS/INS measurements. *Control Engineering Practice*, 18, 7 (2010), 723–732. (cited on pages 46 and 48)

- 
63. HUA, M.-D.; HAMEL, T.; MAHONY, R.; AND TRUMPF, J., 2015. Gradient-like observer design on the Special Euclidean group  $SE(3)$  with system outputs on the real projective space. In *Proc. IEEE Conf. on Decision and Control*, 2139–2145. (cited on page 1)
  64. HUA, M.-D.; MARTIN, P.; AND HAMEL, T., 2014. Velocity-aided attitude estimation for accelerated rigid bodies. *arXiv preprint arXiv:1411.3953*, (2014). (cited on pages 46 and 48)
  65. HUA, M.-D.; ZAMANI, M.; TRUMPF, J.; MAHONY, R.; AND HAMEL, T., 2011. Observer design on the special euclidean group  $SE(3)$ . In *Proc. IEEE Conf. on Decision and Control and the European Control Conf.*, 8169–8175. (cited on pages 1, 13, 15, 16, 33, 34, 37, 39, 48, and 63)
  66. IDAN, M., 1996. Estimation of Rodrigues parameters from vector observations. *IEEE Trans. Aerospace and Electronic Systems*, 32, 2 (1996), 578–586. (cited on pages 2 and 112)
  67. ISERLES, A.; MUNTHER-KAAS, H. Z.; NØRSETT, S. P.; AND ZANNA, A., 2000. Lie-group methods. *Acta Numerica*, 9 (2000), 215–365. (cited on page 93)
  68. IZADI, M.; SANYAL, A.; SAMIEL, E.; AND VISWANATHAN, S., 2015. Discrete-time rigid body attitude state estimation based on the discrete Lagrange-d'Alembert principle. In *Proc. American Control Conference*, 3392–3397. (cited on pages 12 and 42)
  69. IZADI, M. AND SANYAL, A. K., 2014. Rigid body attitude estimation based on the Lagrange-d'Alembert principle. *Automatica*, 50, 10 (2014), 2570–2577. (cited on pages 1, 4, 15, and 48)
  70. IZADI, M.; SANYAL, A. K.; BARANY, E.; AND VISWANATHAN, S. P., 2015. Rigid body motion estimation based on the Lagrange-d'Alembert principle. *arXiv preprint arXiv:1509.04744*, (2015). (cited on page 4)
  71. JURDJEVIC, V., 1997. *Geometric control theory*. Cambridge university press. (cited on pages 1 and 12)

72. JURDJEVIC, V. AND SUSSMANN, H. J., 1972. Control systems on Lie groups. *Journal of Differential equations*, 12, 2 (1972), 313–329. (cited on page 14)
73. KANATANI, K., 1990. *Group-theoretical methods in image understanding*. Springer-Verlag. (Softcover edition 2011). (cited on pages 1 and 116)
74. KARAFYLLIS, I. AND JIANG, Z.-P., 2011. *Stability and stabilization of nonlinear systems*. Springer. (cited on pages 2 and 5)
75. KARAFYLLIS, I. AND KRAVARIS, C., 2009. From continuous-time design to sampled-data design of observers. *IEEE Trans. Autom. Control*, 54, 9 (2009), 2169–2174. (cited on pages 42, 43, 55, 70, and 71)
76. KAZANTZIS, N. AND WRIGHT, R. A., 2005. Nonlinear observer design in the presence of delayed output measurements. *Systems & Control letters*, 54, 9 (2005), 877–886. (cited on pages 2, 5, 42, 43, and 55)
77. KHALEGI, B.; KHAMIS, A.; KARRAY, F. O.; AND RAZAVI, S. N., 2013. Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 14, 1 (2013), 28–44. (cited on pages 70 and 71)
78. KHALIL, H. K., 2002. *Nonlinear Systems*. Prentice Hall, third edn. (cited on pages 23, 24, and 27)
79. KHOSRAVIAN, A., 2013. Stability analysis and near optimal gain tuning of an attitude estimator on the special orthogonal group. In *Proc. IEEE Conf. Decision and Control*, 5060–5065. (cited on page iv)
80. KHOSRAVIAN, A. AND NAMVAR, M., 2010. Globally exponential estimation of satellite attitude using a single vector measurement and gyro. In *Proc. IEEE Conf. Decision and Control*, 364–369. (cited on pages 1, 15, and 31)
81. KHOSRAVIAN, A.; TRUMPF, J.; AND MAHONY, R., 2015. State estimation for nonlinear systems with delayed output measurements. In *Proc. IEEE Conf. Decision and Control*, 6330–6335. (cited on pages iv, 43, 55, and 107)
82. KHOSRAVIAN, A.; TRUMPF, J.; MAHONY, R.; AND HAMEL, T., 2014. Velocity aided attitude estimation on  $SO(3)$  with sensor delay. In *Proc. IEEE Conf. on Decision and Control*, 114–120. (cited on pages iv, 8, 41, 58, and 60)

- 
83. KHOSRAVIAN, A.; TRUMPF, J.; MAHONY, R.; AND HAMEL, T., 2015. Recursive attitude estimation in the presence of multi-rate and multi-delay vector measurements. In *Proc. American Control Conference*, 3199–3205. (cited on pages iv, 8, and 69)
  84. KHOSRAVIAN, A.; TRUMPF, J.; MAHONY, R.; AND HAMEL, T., 2016. State estimation for invariant systems on Lie groups with delayed output measurements. *Automatica*, 68 (2016), 254–265. (cited on pages iv, 8, and 41)
  85. KHOSRAVIAN, A.; TRUMPF, J.; MAHONY, R.; AND LAGEMAN, C., 2013. Bias estimation for invariant systems on Lie groups with homogeneous outputs. In *Proc. IEEE Conf. on Decision and Control*. (cited on pages iv, 7, 11, 20, 27, 28, 31, and 39)
  86. KHOSRAVIAN, A.; TRUMPF, J.; MAHONY, R.; AND LAGEMAN, C., 2015. Observers for invariant systems on Lie groups with biased input measurements and homogeneous outputs. *Automatica*, 55 (2015), 19–26. (cited on pages iv, 46, and 48)
  87. KHOSRAVIAN, A.; TRUMPF, J.; MAHONY, R.; AND LAGEMAN, C., 2015. Observers for invariant systems on Lie groups with biased input measurements and homogeneous outputs. <http://arxiv.org/abs/1507.03770>. (cited on pages iv, 7, and 11)
  88. KINGSTON, D. B. AND BEARD, R. W., 2004. Real-time attitude and position estimation for small UAVs using low-cost sensors. In *Proc. AIAA 3rd Unmanned Un-  
limited Technical Conference, Workshop and Exhibit*. (cited on pages 2, 42, and 71)
  89. KINSEY, J. C. AND WHITCOMB, L. L., 2007. Adaptive identification on the group of rigid-body rotations and its application to underwater vehicle navigation. *IEEE Trans. on Robotics*, 23, 1 (2007), 124–136. (cited on pages 1 and 31)
  90. KNAPP, A. W., 2002. *Lie groups beyond an introduction*, vol. 140. Springer. (cited on pages 10 and 20)
  91. KOBAYASHI, S. AND NOMIZU, K., 1996. *Foundations of differential geometry Volume II*. Wiley-Interscience. (cited on page 14)
  92. LAGEMAN, C.; TRUMPF, J.; AND MAHONY, R., 2008. Observer design for invariant

- systems with homogeneous observations. *arXiv preprint arXiv:0810.0748*, (2008). (cited on pages 4, 12, 13, 28, and 39)
93. LAGEMAN, C.; TRUMPF, J.; AND MAHONY, R., 2009. Observers for systems with invariant outputs. In *Proc. the European Control Conf.*, 4587–4592. (cited on pages 4 and 39)
94. LAGEMAN, C.; TRUMPF, J.; AND MAHONY, R., 2010. Gradient-like observers for invariant dynamics on a Lie group. *IEEE Trans. Autom. Control*, 55, 2 (2010), 367–377. (cited on pages 2, 4, 12, 13, 14, 16, 17, 20, 27, 28, 42, 45, and 48)
95. LEE, T.; LEOK, M.; AND McCLAMROCH, N. H., 2007. Lie group variational integrators for the full body problem in orbital mechanics. *Celestial Mechanics and Dynamical Astronomy*, 98, 2 (2007), 121–144. (cited on page 93)
96. LENZ, R., 1990. *Group theoretical methods in image processing*. Lecture Notes in Computer Science, Springer-Verlag. (cited on pages 1 and 116)
97. LEONARD, N. E. AND KRISHNAPRASAD, P. S., 1995. Motion control of drift-free, left-invariant systems on Lie groups. *IEEE Trans. Autom. Control*, 40, 9 (1995), 1539–1554. (cited on pages 1 and 48)
98. LO, J.-H., 1986. Optimal estimation for the satellite attitude using star tracker measurements. *Automatica*, 22, 4 (1986), 477–482. (cited on pages 2 and 112)
99. LORÍA, A. AND PANTELEY, E., 2002. Uniform exponential stability of linear time-varying systems: revisited. *Systems & Control Letters*, 47, 1 (2002), 13–24. (cited on page 26)
100. LUCK, R. AND RAY, A., 1990. An observer-based compensator for distributed delays. *Automatica*, 26, 5 (1990), 903–908. (cited on page 2)
101. MAHONY, R.; HAMEL, T.; AND J.M. PFLIMLIN, 2008. Nonlinear complementary filters on the special orthogonal group. *IEEE Trans. Autom. Control*, 53, 5 (2008), 1203–1218. (cited on pages 1, 2, 3, 4, 12, 15, 20, 31, 33, 46, 48, and 84)
102. MAHONY, R.; TRUMPF, J.; AND HAMEL, T., 2013. Observers for kinematic systems with symmetry. In *Proc. IFAC Symposium on Nonlinear Control Systems*, 617–633. (cited on pages 4, 12, 17, 29, 42, 48, and 77)

- 
103. MALLICK, M.; KRANT, J.; AND BAR-SHALOM, Y., 2002. Multi-sensor multi-target tracking using out-of-sequence measurements. In *Proc. IEEE International Conf. Information Fusion*, vol. 1, 135–142. (cited on page 71)
  104. MARKLEY, F. L., 2006. Attitude filtering on SO(3). *The Journal of the Astronautical Sciences*, 54, 3-4 (2006), 391–413. (cited on pages 2 and 112)
  105. MARKLEY, F. L. AND CRASSIDIS, J. L., 2014. *Fundamentals of Spacecraft Attitude Determination and Control*. Springer. (cited on pages 1, 2, 3, 12, 15, 70, and 112)
  106. MÁRQUEZ-MARTÍNEZ, L. A.; MOOG, C. H.; AND VELASCO-VILLA, M., 2002. Observability and observers for nonlinear systems with time delays. *Kybernetika*, 38, 4 (2002), 445–456. (cited on page 2)
  107. MARTIN, P. AND SALAÜN, E., 2010. Design and implementation of a low-cost observer-based attitude and heading reference system. *Control Engineering Practice*, 18, 7 (2010), 712–722. (cited on pages 46 and 48)
  108. METNI, N.; PFLIMLIN, J.-M.; HAMEL, T.; AND SOUERES, P., 2006. Attitude and gyro bias estimation for a VTOL UAV. *Control Engineering Practice*, 14, 12 (2006), 1511–1520. (cited on page 15)
  109. MICHIELS, W. AND NICULESCU, S.-I., 2007. *Stability and Stabilization of Time-Delay Systems*. Society for Industrial and Applied Mathematics. (cited on page 2)
  110. MORGAN, A. AND NARENDRA, K., 1977. On the stability of nonautonomous differential equations  $\dot{x} = A + B(t)x$ , with skew symmetric matrix  $B(t)$ . *SIAM Journal on Control and Optimization*, 15, 1 (1977), 163–176. (cited on page 26)
  111. MORGAN, A. AND NARENDRA, K., 1977. On the uniform asymptotic stability of certain linear nonautonomous differential equations. *SIAM Journal on Control and Optimization*, 15, 1 (1977), 5–24. (cited on page 26)
  112. NORDKVIST, N. AND SANYAL, A. K., 2010. A Lie group variational integrator for rigid body motion in SE(3) with applications to underwater vehicle dynamics. In *Proc. IEEE Conf. Decision and Control*, 5414–5419. (cited on page 93)

113. O'BRIEN, S. Predictors for systems on manifolds with time-delayed measurements. Australian National University. Bachelor's Thesis, 2014. (cited on pages 44, 61, and 63)
114. OSHMAN, Y. AND MARKLEY, F. L., 1998. Minimal-parameter attitude matrix estimation from vector observations. *Journal of Guidance, Control, and Dynamics*, 21, 4 (1998), 595–602. (cited on pages 2 and 112)
115. PIGGOTT, M. J. AND SOLO, V., 2014. Stochastic numerical analysis for brownian motion on  $SO(3)$ . In *Proc. IEEE Conf. Decision and Control*, 3420–3425. (cited on page 112)
116. REHBINDER, H. AND GHOSH, B. K., 2003. Pose estimation using line-based dynamic vision and inertial sensors. *IEEE Trans. Autom. Control*, 48, 2 (2003), 186–199. (cited on pages 1, 2, 3, 15, 16, 48, and 84)
117. RICHARD, J.-P., 2003. Time-delay systems: an overview of some recent advances and open problems. *Automatica*, 39, 10 (2003), 1667–1694. (cited on page 2)
118. RISEBOROUGH, P. <https://github.com/priseborough>. (visited in July 2015). (cited on page 100)
119. RISEBOROUGH, P., 2015. 3DRobotics, private communication. (cited on pages 42 and 70)
120. ROBERTS, A. AND TAYEBI, A., 2011. On the attitude estimation of accelerating rigid-bodies using GPS and IMU measurements. In *Proc. IEEE Conf. Decision and Control and European Control Conf. (CDC-ECC)*, 8088–8093. (cited on pages 1, 15, 46, and 48)
121. RODRIGUES, S. S.; CRASTA, N.; AGUIAR, A. P.; AND LEITE, F. S., 2010. State estimation for systems on  $SE(3)$  with implicit outputs: An application to visual servoing. In *Proc. IFAC Symposium on Nonlinear Control Systems*. (cited on pages 1, 15, and 48)
122. SACCON, A.; HAUSER, J.; AND AGUIAR, A. P., 2013. Optimal control on Lie groups: The projection operator approach. *IEEE Trans. Automatic Control*, 58, 9 (2013), 2230–2245. (cited on page 25)

- 
123. SACCON, A.; TRUMPF, J.; MAHONY, R.; AND AGUIAR, A. P., 2013. Second-order-optimal filters on Lie groups. In *Proc. IEEE Conf. Decision and Control*, 4434–4441. (cited on pages 4, 12, 39, and 42)
  124. SANYAL, A. K.; LEE, T.; LEOK, M.; AND McCLAMROCH, N. H., 2008. Global optimal attitude estimation using uncertainty ellipsoids. *Systems & Control Letters*, 57, 3 (2008), 236–245. (cited on pages 1, 15, and 93)
  125. SEURET, A. AND GOUAISBAUT, F., 2015. Hierarchy of LMI conditions for the stability analysis of time-delay systems. *Systems & Control Letters*, 81 (2015), 1–7. (cited on pages 2 and 71)
  126. SHUSTER, M. D. AND OH, S. D., 1981. Three-axis attitude determination from vector observations. *Journal of Guidance and Control*, 4, 1 (1981), 70–77. (cited on pages 2 and 112)
  127. SMITH, O. J., 1959. A controller to overcome dead time. *ISA Journal of Instrument Society of America*, 6, 2 (1959), 28–33. (cited on page 82)
  128. SOLO, V., 2014. An approach to stochastic system identification in Riemannian manifolds. In *Proc. IEEE Conf. Decision and Control*, 6510–6515. (cited on page 112)
  129. SUN, T.; XING, F.; YOU, Z.; AND WEI, M., 2013. Motion-blurred star acquisition method of the star tracker under high dynamic conditions. *Optics express*, 21, 17 (2013), 20096–20110. (cited on page 74)
  130. SWENSEN, J. P. AND COWAN, N. J., 2012. An almost global estimator on  $SO(3)$  with measurement on  $S^2$ . In *American Control Conference*, 1780–1786. (cited on page 1)
  131. TRIDGELL, A. <https://github.com/tridge>. (visited in July 2015). (cited on pages 3 and 100)
  132. TRIDGELL, A., 2015. 3DRobotics, private communication. (cited on page 70)
  133. TRUMPF, J.; MAHONY, R.; HAMEL, T.; AND LAGEMAN, C., 2012. Analysis of non-linear attitude observers for time-varying reference measurements. *IEEE Trans. Autom. Control*, 57, 11 (2012), 2789–2800. (cited on pages 15 and 31)

134. VAN ASSCHE, V.; AHMED-ALI, T.; HANN, C.; AND LAMNABHI-LAGARRIGUE, F., 2011. High gain observer design for nonlinear systems with time varying delayed measurements. In *Proc. IFAC World congress*, 692–696. (cited on pages 5, 42, 99, and 115)
135. VASCONCELOS, J.; CUNHA, R.; SILVESTRE, C.; AND OLIVEIRA, P., 2010. A nonlinear position and attitude observer on SE(3) using landmark measurements. *Systems & Control Letters*, 59, 3 (2010), 155–166. (cited on pages 1, 3, 4, 12, 13, 16, 17, 20, 31, 33, 34, 37, 38, 39, and 48)
136. VASCONCELOS, J.; SILVESTRE, C.; AND OLIVEIRA, P., 2008. A nonlinear observer for rigid body attitude estimation using vector observations. In *Proc. IFAC World Congr.* (cited on pages 1, 2, 3, 4, 12, 15, 33, 48, and 84)
137. VASCONCELOS, J. F.; SILVESTRE, C.; AND OLIVEIRA, P., 2008. A nonlinear GPS/IMU based observer for rigid body attitude and position estimation. In *Proc. IEEE Conf. Decision and Control*, 1255–1260. (cited on page 46)
138. VIK, B. AND FOSSEN, T. I., 2001. A nonlinear observer for GPS and INS integration. In *Proc. IEEE Conf. on Decision and Control*, vol. 3, 2956–2961. (cited on page 16)
139. WANG, Y. AND CHIRIKJIAN, G. S., 2008. Nonparametric second-order theory of error propagation on motion groups. *The International Journal of Robotics Research*, 27, 11-12 (2008), 1258–1273. (cited on pages 2 and 112)
140. WILLSKY, A. S., 1973. *Dynamical systems defined on groups: structural properties and estimation*. Ph.D. thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology. (cited on page 2)
141. WILLSKY, A. S., 1973. Some estimation problems on Lie groups. In *Geometric Methods in System Theory*, 305–314. Reidel Publishing Company. (cited on page 2)
142. YANG, T. AND MIKHAEL, W. B., 2004. Baseband image rejection for diversity superheterodyne receivers. In *IEEE Wireless Communications and Networking Conference*, vol. 4, 2232–2234. (cited on page 116)

- 
143. ZAMANI, M. Deterministic attitude and pose filtering, an embedded Lie groups approach. Australian National University. PhD Thesis, 2013. (cited on pages 4, 12, and 42)
  144. ZAMANI, M.; TRUMPF, J.; AND MAHONY, R., 2013. Minimum-energy filtering for attitude estimation. *IEEE Trans. Autom. Control*, 58 (2013), 2917–2921. (cited on pages 1, 15, 46, and 48)
  145. ZHANG, K.; LI, X. R.; AND ZHU, Y., 2005. Optimal update with out-of-sequence measurements. *IEEE Trans. Signal Processing*, 53, 6 (2005), 1992–2004. (cited on page 71)
  146. ZHANG, X.; BERGER, G.; DIETZ, M.; AND DENZ, C., 2006. Cross-talk in phase encoded volume holographic memories employing unitary matrices. *Applied Physics B Lasers Optics*, 85, 4 (2006), 575–579. (cited on page 116)