Ultra-wideband for Robot Navigation: Calibration, Communication, and State Estimation

Mohammed Ayman Shalaby

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Doctor of Philosophy

Department of Mechanical Engineering
McGill University
December 2023

Abstract

This thesis examines the utilization of ultra-wideband (UWB) radio for robot navigation in indoor applications. UWB, a wireless communication technology, offers a means of generating distance (or *range*) measurements and establishing communication channels among mobile robots and fixed anchors at known locations. When fixed anchors are present, UWB can be harnessed for precise localization, enabling, for example, the tracking of robots in warehouse applications. In unexplored environments lacking a fixed infrastructure, UWB transceivers installed on different robots facilitate inter-robot ranging and communication, thereby enabling relative localization, an essential prerequisite for tasks such as maintaining a group formation, collaboratively mapping an area, or ensuring effective collision avoidance. In order to maximize the potential of UWB for localization purposes, it becomes imperative to tackle fundamental challenges of UWB, such as clock synchronization and the choice of ranging protocol, concurrently with the development of the state estimation algorithm. Conventionally, these two issues have been addressed separately.

The primary objective of this thesis is to resolve both the underlying low-level UWB ranging and communication performance challenges, as well as the state estimation problem, ultimately providing a practical, robust, and implementable solution for indoor navigation. To this end, this thesis delves into the matter of UWB measurement **calibration**, aiming to enhance ranging accuracy and characterize measurement uncertainty for use in a probabilistic framework. Subsequently, the selection of an appropriate ranging protocol is motivated to facilitate localization and efficient inter-robot **communication**. Lastly, this thesis employs UWB measurements within a filtering framework to tackle the problem of **state estimation**, encompassing both a scenario involving a single robot performing a closed-loop teach-and-repeat experiment and a multi-robot scenario where the robots perform on-manifold relative localization and attitude estimation using preintegration techniques. Comprehensive testing of all proposed algorithms is conducted through simulation and real-world experiments employing custom-built UWB modules fitted onto quadcopters.

Résumé

Cette thèse examine l'utilisation de la radio à bande ultra-large (UWB) pour la navigation robotique en l'absence de GPS. L'UWB, une technologie de communication sans fil, permet d'obtenir des mesures de distance et de communiquer entre des robots mobiles et des ancres fixes à des positions connues. En présence d'ancres fixes, l'UWB peut être exploité pour une localisation précise, ce qui permet par exemple de localiser des robots dans des entrepôts. Dans les environnements inconnus dépourvus d'infrastructure fixe, les émetteurs-récepteurs UWB installés sur différents robots facilitent la télémétrie et la communication inter-robots, permettant ainsi une localisation relative, nécessaire pour des tâches telles que le maintien de la formation d'un groupe, la cartographie collaborative d'une zone ou l'évitement des collisions. Afin de maximiser le potentiel de l'UWB à des fins de localisation, il est impératif de relever les défis fondamentaux de l'UWB, tels que la synchronisation des horloges et le choix du protocole de télémétrie, en même temps que le développement de l'algorithme d'estimation de l'état. Traditionnellement, ces deux questions ont été traitées séparément.

L'objectif principal de cette thèse est de résoudre à la fois les défis sous-jacents de bas niveau en matiére de performance de communication et de télémétrie UWB, ainsi que le probléme d'estimation d'état, pour finalement fournir une solution pratique, robuste et implémentable pour la navigation intérieure. À cette fin, cette thèse se penche d'abord sur la question de la **calibration** des mesures UWB, dans le but d'améliorer la précision de la télémétrie et de caractériser l'incertitude des mesures pour une utilisation dans un cadre probabiliste. Ensuite, un protocole de télémétrie approprié est sélectionné pour faciliter la localisation et des **communications** inter-robots efficaces. Enfin, cette thèse utilise les mesures UWB dans un cadre de filtrage pour aborder le problème **d'estimation d'état**, d'abord pour un scénario impliquant un seul robot effectuant une tâche de « teach-and-repeat » en boucle fermée, puis pour un scénario dans lequel plusieurs robots estiment leurs position et attitude relatives à l'aide de techniques de pré-intégration et de filtrage sur des groupes de Lie. Des tests complets des algorithmes proposés sont effectués par simulation et au moyen d'expériences physiques utilisant des modules UWB fabriqués sur mesure et montés sur des quadricoptéres.

Acknowledgements

As is with every doctorate degree, this thesis would not have been possible without the unwavering support of a lot of individuals. First and foremost, I am immensely grateful to my two supervisors, James R. Forbes and Jérôme Le Ny, for their guidance over the past 4 years. They are outstanding supervisors who have made my Ph.D. the most insightful and enjoyable.

Another individual who deserves a lot of credit is Charles C. Cossette. Charles and I worked hand-in-hand throughout my 4 years at McGill, and I could not have asked for a better colleague. He is a brilliant researcher who has contributed significantly to the work presented in this thesis through general productive discussions. A lot of the experiments in this thesis would also have not been possible without his brains and coding skills.

I would also like to thank my family for their endless support throughout my Ph.D., and for their willingness to hear about state estimation and UWB even when it makes absolutely no sense to them. The same applies to my friends who made these 4 years a lot more fun.

I also have special gratitude to my loving partner, Salma Tawfik, for her unwavering support, patience, and belief in me. She really made this Ph.D. journey a lot more enjoyable and I am forever grateful for her unconditional love and support.

Lastly, special credit goes to the NSERC Alliance Grant program, the NSERC Discovery Grant program, the CFI JELF program, and the FRQNT for supporting the work in this thesis.

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

AHRS	attitude and heading reference system
CIR	channel impulse response
CRLB	Cramer-Rao Lower Bound
CSRPE	clock-synchronization and relative-pose estimator
DS-TWR	double-sided two-way ranging
EKF	extended Kalman filter
FPP	first-path power
GPS	global positioning system
IMU	inertial measurement unit
KF	Kalman filter
MAC	media-access control
NEES	normalized estimation error squared
NIS	normalized innovation squared
RMSE	root-mean-square error
ROS	robot operating system
SLAM	simultaneous localization and mapping
SS-TWR	single-sided two-way ranging
TDMA	time-division multiple-access
ToF	time-of-flight
TWR	two-way ranging
UWB	ultra-wideband

Notation

x	font used for real scalars
$\mathbf{x}$	font used for real column matrices
$\mathbf{X}$	font used for real matrices
$\mathbf{1}$	appropriately-sized identity matrix
$\mathbf{0}$	appropriately-sized zero matrix
$\mathcal{F}_a$	arbitrary reference frame a
$\underrightarrow{x}$	physical vector in three dimensions
${}^{i}\underrightarrow{x}$	derivative of a physical vector $\underrightarrow{x}$ with respect to frame i
$\mathbf{x}_a$	physical vector $\underrightarrow{x}$ resolved in $\mathcal{F}_a$
$\underrightarrow{r}^{zw}$	physical vector in three dimensions from point w to point z
$\mathbf{C}_{ab}$	3×3 rotation matrix such that $\mathbf{x}_a = \mathbf{C}_{ab}\mathbf{x}_b$. An element of $SO(3)$.
$\check{(\cdot)}$	*a priori* or *predicted* quantity
$\hat{(\cdot)}$	*a posteriori* or *estimated* quantity
$p(\mathbf{a})$	probability density of $\mathbf{a}$
$p(\mathbf{a}\|\mathbf{b})$	probability density of $\mathbf{a}$ conditioned on $\mathbf{b}$
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Gaussian probability density with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
$\mathbb{E}[\cdot]$	expectation operator

Preface

The problems explored in this thesis are mainly motivated by real-world challenges faced while trying to use UWB to estimate the position of robots relative to one another. The main issues that arose are

- the observability of relative positions, which are vector quantities, from just scalar distance measurements,
- the UWB measurements suffering from systematic-bias and environment-dependent errors,
- the lack of UWB communication protocols that allow robots to broadcast information, and thus hindering the scalability of UWB-based navigation, and finally
- the need to utilize matrix Lie groups to implement on-manifold state-estimation algorithms that better model the distributions generated from the range measurements.

This thesis addresses these issues by taking incremental steps towards the final goal of realizing a practical and scalable UWB-based localization algorithm. The contributions and novelties presented in this thesis can be categorized per chapter as follows.

- Chapter 3 [1]
  - A rigidity-theory-based observability analysis is presented for any number of robots estimating relative positions from range measurements.
  - A sufficient condition that is independent of the relative motion of the robots is derived for the observability of the three-dimensional relative positions when only range measurements are available.
  - The utilization of a two-tag framework in multi-robot systems is discussed to allow the estimation of three-dimensional relative positions using just the range measurements, an inexpensive IMU, and a magnetometer.
- Chapter 4 [2]
  - A novel ranging protocol is proposed that is shown to mitigate the clock-skew-induced bias.
  - A scalable antenna-delay calibration algorithm is presented that is robust to outliers and pose-dependent bias.

  - The bias and variance of the measurements are modelled as a function of first-path power, and the proposed ranging protocol is utilized to overcome the need to estimate the clock skew.
  - The code for the full calibration procedure is attached as an open-access online repository, which can be found at https://github.com/decargroup/uwb_calibration.

- Chapter 5 [3]
  - An analytical model of the variance of ranging protocols is derived as a function of the timing of message transmissions.
  - The derived analytical variance is compared to the Cramer-Rao Lower Bound (CRLB).
  - An optimization problem is formulated as a function of the signal timings to maximize the information collected in one unit of time.
  - The effect of relative motion during ranging is analyzed.

- Chapter 6 [4]
  - A ranging protocol is introduced that extends classical ranging protocols by allowing robots to passively listen to the measurements between neighbours and timestamp receptions, with no assumptions or imposed constraints on the robots' hierarchy.
  - The extended pose state is represented as an element of $SE_2(3)$, and an on-manifold tightly-coupled simultaneous clock-synchronization and relative-pose estimator (CSRPE) is proposed.
  - The concept of preintegration is developed for relative pose states on $SE_2(3)$ as an alternative to sharing high-frequency IMU readings with neighbours.
  - The communicated IMU measurements are incorporated in the CSRPE, where the theory behind filtering with delayed inputs is developed as the preintegrated IMU measurements arrive asynchronously from neighbouring robots.

- Chapter 7 [5]
  - A probabilistic approximation of the propagated cross-covariance terms is presented when fusing information from loosely-coupled filters that do not necessarily share a common set of states.

  - The performance of the proposed filter is compared to a sigma-point-based Covariance Intersection (SPCI) estimator.

- Chapter 8 [6]
  - A ranging protocol is proposed to allow a 3-tag robot to synchronize the clocks of its tags and receive 3 range measurements from a fixed anchor with only one ranging transaction.
  - A novel UWB-based *teach & repeat* framework is presented for a vehicle moving in an environment with spaced-out fixed anchors at unknown locations over a large area.

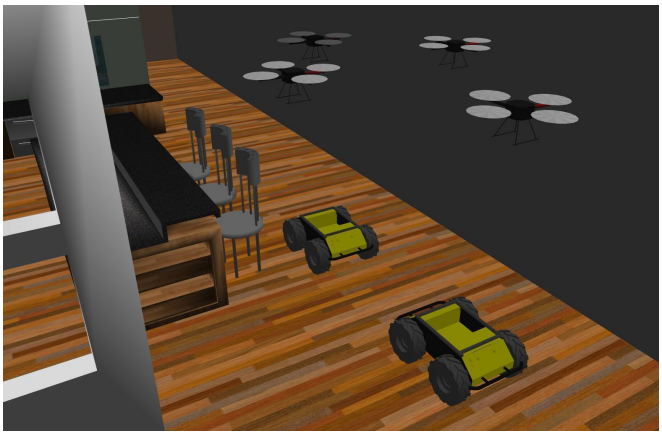
Another contribution of this thesis is the evaluation of all the proposed algorithms and protocols in simulation and in experiments. Millions of range measurements are collected and hours of experimental flights are conducted to evaluate the performance and validate the practicality of the proposed algorithms.

Additionally, special credit is given to Charles C. Cossette for his part in making the experiments in this thesis possible and for the productive discussions that led to many of the ideas presented in this thesis. Nonetheless, all the contributions claimed above are independently developed by the Author of this thesis.

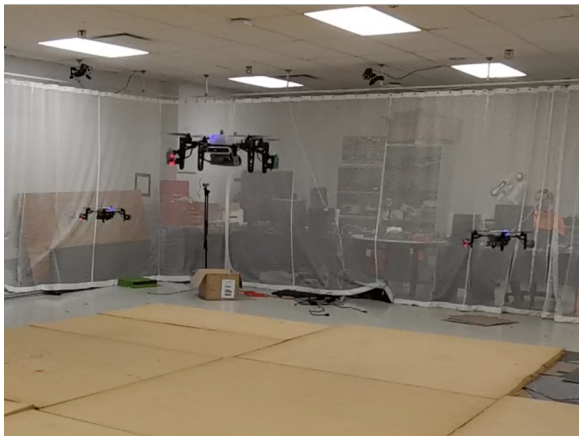
The topics addressed and the solutions presented in this thesis have a wide possibility of applications. It is foolish to turn a blind eye to the possibility of evil with such technologies; nevertheless, the Author hopes that this work instead acts as a catalyst for the change and betterment of the world.

Chapter 1

Introduction



(a) A simulated indoor scenario with 4 quadcopters and 2 ground vehicles.



(b) Three Uvify IFO-S quadcopters in a triangle formation.

Figure 1.1: Two examples of teams of robots in an indoor setting.

The overall goal of this thesis is to develop a practical navigation solution for robots utilizing ultra-wideband (UWB) radio. Robot navigation is the process in which a robot understands where it is and perceives its environment, and this thesis mainly focuses on the localization aspect of robot navigation. UWB is particularly attractive for robot navigation as it is an inexpensive and low-power sensor that provides a means for robots to obtain distance measurements and communicate with other robots at a high rate. As a result, UWB is oftentimes used for localization in GPS-denied environments and as a communication medium when deploying multi-robot systems. This is crucial for applications such as automated surveillance, search and rescue, infrastructure and mine inspections, collaborative simultaneous localization and mapping (SLAM), and agricultural monitoring. Two examples of team of robots in an indoor setting are shown in Figure 1.1. However, such applications in-

volve an additional layer of complexity, as topics such as formation architectures and control, communication protocols, calibration, collaborative localization, decentralization, and task allocation must be addressed. Localization in particular is a core prerequisite for performing any other task, such as control and motion planning.

Ever since the legalization of the unlicensed use of UWB radio in some parts of the world in 2002, research into the potential uses of UWB in commercial systems has been rapidly increasing. This has been particularly accelerated by the development of low-cost and low-power single-chip wireless transceivers by Qorvo (formerly Decawave) [7]. Due to the properties of UWB signals, different transceivers can communicate at high frequencies and *range* with one another through time-of-flight (ToF) measurements. UWB transceivers have been extensively utilized in lab environments for research purposes, usually in the presence of fixed transceivers with known positions, or *anchors*, to evaluate range-based localization algorithms. Additionally, UWB transceivers have been used in multi-robot applications where each robot is equipped with a transceiver, allowing the robots to range with one another.

The main limitation of current approaches associated with UWB-based localization is that improving the performance of UWB range measurements is treated separately from the choice of the localization algorithm. This therefore induces an unnecessary limitation at the higher level when designing localization approaches. Furthermore, neglecting the limitations of UWB systems oftentimes leads to impractical localization solutions.

To realize the full potential of UWB-based localization, this thesis simultaneously addresses the calibration procedure and protocols for UWB ranging and communication as well as the problem of localization using state estimation algorithms. Without being subject to UWB-based constraints, novel approaches are introduced that are more practical, more scalable to larger teams of robots, and lead to better localization performance as compared to the traditional counterparts. Additionally, this flexibility in designing ranging and communication protocols allows the exploration of new applications of UWB-based localization. Heavy emphasis is placed on the practicality of proposed solutions; therefore, all proposed approaches are implemented on real experimental data. Significant effort has been allocated to implementing proposed UWB protocols on custom-made UWB transceivers, which are then fitted to quadcopters. More details on the hardware used in this thesis can be found in Appendix A. Lastly, Lie group theory is utilized to represent the state of the quadcopters, leading to the adoption of on-manifold state estimation tools.

1.1 Background and Related Work

The problem of robot navigation has been addressed using a wide range of sensors. Outdoor applications typically rely on GPS receivers to localize a mobile robot using a constellation of satellites [8, Chapter 11]; however, GPS measurements are known to be unreliable in urban settings, dense environments, and indoors [9]. This has led to the development of other localization solutions, such as ones that utilize cameras [10] or LIDARs [11]. Cameras are particularly attractive due to their low cost, but they are prone to failure in low-light conditions and when there are few features in the environment. LIDARs, on the other hand, are more robust to lighting conditions, but are expensive, bulky, and require heavy computational processing, thus not particularly suited for aerial vehicles.

These limitations have accelerated the deployment of wireless technologies such as Bluetooth [12], WiFi [13], and UWB [14] for localization purposes, particularly as they have the additional benefit of allowing communication between robots. UWB is particularly attractive due to its low cost, low power consumption, compactness, and high ranging accuracy and precision [15]. Most wireless technologies rely on fixed infrastructure such as WiFi access points, Bluetooth beacons, or UWB anchors to localize the robots. However, this is not always feasible, particularly in applications where the robots need to be deployed in unknown environments. This necessitates the development of solutions that rely on inter-robot ranging and communication, which is the focus of this thesis.

The building blocks associated with achieving efficient UWB-based localization span a wide range of avenues. Range-based multi-robot localization in the absence of anchors is challenging on its own even in the presence of a central computer that processes all the information and in the absence of all the complications that arise with UWB systems. With anchors, the common solution is to perform ranging between the anchors and the robots being localized [14], [16], [17]. However, realizing an *observable* system with only range measurements between robots and in the absence of anchors is non-trivial, as there is an infinite number of positions that result in the same range measurements [1]. One way to obtain an observable system is through utilization of a motion model, where the requirement for observability then comes down to satisfying a *persistency of excitation* condition [18]. However, this requires consistent relative motion between the robots, which is limiting in applications where, for example, the robots need to maintain a specific formation or be static for a period of time. Alternatively, the use of multiple UWB transceivers per robot has also been considered [19], [20].

Adding another layer of complexity are the practical complications associated with the use of UWB systems. Although oftentimes simplified to just a system that outputs range

measurements when designing localization approaches, UWB ranging is in fact a very involved process. The range measurements obtained from UWB signals are a function of the ToF between transmission and reception of a message between two distinct transceivers each with their own clock. In order to provide centimetre-level accuracy, the clocks of the two transceivers must be synchronized to within sub-nanosecond levels of accuracy. This is a main factor that must be considered when designing the ranging protocol, and common approaches include the *time-difference-of-arrival* (TDoA) and *two-way ranging* (TWR) protocols [15]. However, even these methods have their own limitations, prompting the recent development of more advanced ranging protocols such as the double-sided TWR (DS-TWR) protocol [21] or filtering approaches for clock synchronization [22].

Even when the clocks are perfectly synchronized, the use of UWB presents other potential sources of error. Systematic delays in the timestamping of messages are common due to manufacturing inaccuracies and can result in significant bias in the range measurements [23]. Additionally, there exists a correlation between the errors in the range measurements and the pose of the robot due to the lack of symmetry in the radiation pattern of the UWB antennae [24]. Furthermore, the presence of objects or robots in the vicinity of the region traveled by the UWB messages oftentimes results in positively-biased measurements [15], which results in error distributions poorly modelled using Gaussian assumptions often adopted in standard filtering applications such as the Kalman filter [25]. To address this, some solutions involve attempting to detect the biased measurements using statistical tools [14] or supervised learning algorithms [26], [27], while others model the distribution of the noise to be non-Gaussian [28], [29]. More recently, a data-driven model of the relation between the bias in the measurements and the received signal power has been proposed [30].

UWB-reliant teams of robots also have their own distinct set of problems. In order to prevent message collision, only one pair of UWB transceivers can range at a time. This is a major limitation that deems a centralized approach infeasible, as the robots cannot all simultaneously broadcast information but rather ranging pairs have to communicate sequentially. This poses three problems. First of all, some form of media-access control (MAC) protocol needs to be implemented, such as the *time-division multiple-access* (TDMA) MAC protocol [31]. Furthermore, the majority of the proposed multi-robot UWB-based systems are restricted to a few number of robots as sequential pair-by-pair ranging generally does not scale well to many robots. Lastly, when there is a need for broadcasting high-rate sensor information such as from an *inertial measurement unit* (IMU), existing solutions rely on a centralization assumption [1] or a limit of only two robots [32], [33]. Additionally, this problem of efficient IMU-measurement sharing has not been considered for on-manifold pose estimation using matrix Lie groups.

Further deeming central processing infeasible is that it requires exponentially-growing communication and processing capabilities; therefore, decentralization is a key factor in achieving real-world applicability. The main challenge of decentralization is the cross-correlations that develop between the state estimates of neighbouring robots, which, if neglected, might lead to poor estimation performance [5], [34]. This is a classic decentralized filtering problem and has been addressed from many different viewpoints, such as using *consensus* algorithms [35] or *Covariance Intersection* (CI) [36]. These methods have been used in obtaining relative pose estimates under certain assumptions [34], [37]–[39]. More recently, general decentralization algorithms have been proposed using optimization algorithms [40] and pseudomeasurements with CI [41]. However, to date, there has been no general decentralization approach that is computationally efficient and scalable for range-based multi-robot localization without strict assumptions on the measurements or topology of the team, or relying on the CI approach, which incurs significant loss of information.

1.2 Outline

This thesis is a summary of the journey taken to achieve a practical localization approach for robots utilizing UWB radio. The general necessary mathematical preliminaries are presented in Chapter 2, but can be skipped by expert readers.

The body of this thesis starts with Chapter 3, where the problem of performing relative position estimation in multi-robot systems is first presented. In this chapter, a rigidity-theory-based observability analysis is performed for any number of robots in order to understand the minimum amount of information required to be able to perform relative position estimation in multi-robot systems using just range measurements. This leads to the derivation of a sufficient condition that can be satisfied by placing multiple UWB tags per robot alongside an IMU and a magnetometer.

Despite showing the promise of multi-tag robots for UWB-based relative localization, the performance achieved on experimental data proved the need for improving the quality of the range measurements, which appeared to be biased and noisy. To address the bias in the noise, in Chapter 4, a ranging protocol is proposed that is based on the DS-TWR protocol, which is shown to yield less bias than standard algorithms. Additionally, a novel bias calibration procedure is presented that takes into account both timestamping delays and the relative-pose-dependent bias as a function of the received signal power. The uncertainty of the range measurements are also characterized as a function of the received signal power.

On the thrust of improving the UWB range measurements, the goal of Chapter 5 is to provide an understanding of the effect of varying the delay between message reception and

message transmission in TWR on the variance of the measurement. To do so, an analytical model of the variance of DS-TWR is derived as a function of the timing of message transmissions, which is then compared to the *Cramer-Rao Lower Bound* (CRLB). An optimization problem is then formulated as a function of the variance and the frequency of the measurements for different message-timing values, allowing the choice of delays that maximize the amount of information obtained from the range measurements in any unit of time.

Now, equipped with more accurate range measurements, Chapter 6 revisits the relative pose estimation problem with practicality, scalability, and feasibility in mind. This involves addressing the problems associated with the communication constraints introduced by the transceivers' inability to range in parallel. Firstly, in order to allow broadcasting ability of messages, a ranging protocol is proposed that allows robots to "listen-in" on neighbouring robots communicating with one another. This requires simultaneously estimating the clock states of UWB transceivers alongside the relative pose states, which led to the development of a novel on-manifold tightly-coupled simultaneous clock-synchronization and relative-pose estimator (CSRPE) based on an extended Kalman filter (EKF). In order to share the IMU information efficiently between robots, the concept of preintegration is developed for relative pose states on $SE_2(3)$, and is used as a means of efficient data logging and communication between robots.

Robots estimating relative poses of neighbours end up estimating similar states as their neighbours, and it might be tempting to *naïvely* share and fuse these estimates. The issue with such approaches, which is due to unmodelled cross-correlations, is addressed in Chapter 7. A common approach to this problem is to use covariance intersection (CI), but an alternative approach is presented that relies on approximating these cross-covariances using sigma points based on knowledge of the neighbours' process models. The performance of this approach is shown to be far superior to CI as it is significantly more confident in its estimates as less information is discarded.

Lastly, the knowledge acquired by studying the minimum requirements for range-based localizability, the calibrated range measurements, and the ability to implement custom ranging protocols led to the development of a new application for UWB: a lightweight teach-and-repeat algorithm presented in Chapter 8. This relies on a novel ranging protocol with fixed transceivers at unknown locations, where despite global unobservability of the system, the local observability ensures that the robot can retrace a taught trajectory to within sub-metre accuracy.

All the developed algorithms are implemented on real hardware and evaluated on experimental data. Details on the different hardware used and the evolution of the experimental setup over the course of this thesis is presented in Appendix A.

Chapter 2

Preliminaries

Summary

Before delving into the complexities of UWB-based state estimation, this chapter will present some of the more general mathematical preliminaries necessary for understanding the subsequent algorithms, protocols, and derivations. Specific preliminaries required for a specific topic in a future chapter will then be discussed within the chapter. Therefore, the expert reader familiar with basic probability and filtering theory, matrix Lie groups, and UWB may safely skip this chapter.

2.1 Probability Theory

Real-life sensor data is noisy. Utilizing noisy sensor data for tasks such as navigation requires properly characterizing the noise using probabilistic tools. Given that any estimate of the state of a robot is typically conditioned on measurements from noisy sensors, there is always uncertainty in the state estimate. Probabilistic tools are a perfect fit for understanding the level of confidence one can associate with a state estimate.

2.1.1 Gaussian Distributions

The Gaussian distribution is perhaps the most famous and commonly used distribution. The use, and therefore commonality, is justified by the *Central Limit Theorem* [42, Theorem 5.5.14]. The Gaussian distribution is further preferred as its properties allow for concise mathematical derivations and efficient computational algorithms.

An n-dimensional random variable $\mathbf{x}$ has a Gaussian distribution with mean $\boldsymbol{\mu} \in \mathbb{R}^n$ and

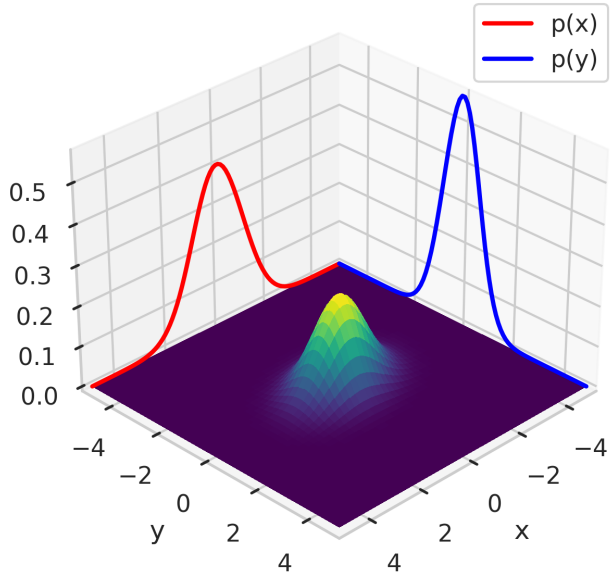


Figure 2.1: An example of the joint distribution $p(x, y)$ of two scalar Gaussian random variables x and y.

covariance $\mathbf{\Sigma} \in \mathbb{R}^{n \times n}$ if its multivariate probability density has the form

$$p(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma}) = \frac{1}{\sqrt{(2\pi)^n \det \mathbf{\Sigma}}} \exp\Big(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\mathsf{T} \mathbf{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \Big). \tag{2.1}$$

Consider now an n-dimensional random variable $\mathbf{x}$ and an m-dimensional random vector $\mathbf{y}$. If these random variables have the joint Gaussian probability distribution

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\mu}_y \end{bmatrix}, \begin{bmatrix} \mathbf{\Sigma}_{xx} & \mathbf{\Sigma}_{xy} \\ \mathbf{\Sigma}_{yx} & \mathbf{\Sigma}_{yy} \end{bmatrix} \right), \tag{2.2}$$

then the marginal and conditional distributions of $\mathbf{x}$ and $\mathbf{y}$ are given as [43, Lemma A.2]

$$p(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_x, \mathbf{\Sigma}_{xx}), \tag{2.3}$$

$$p(\mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_y, \mathbf{\Sigma}_{yy}), \tag{2.4}$$

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_x + \mathbf{\Sigma}_{xy}\mathbf{\Sigma}_{yy}^{-1}(\mathbf{y} - \boldsymbol{\mu}_y), \mathbf{\Sigma}_{xx} - \mathbf{\Sigma}_{xy}\mathbf{\Sigma}_{yy}^{-1}\mathbf{\Sigma}_{yx}), \tag{2.5}$$

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_y + \mathbf{\Sigma}_{yx}\mathbf{\Sigma}_{xx}^{-1}(\mathbf{x} - \boldsymbol{\mu}_x), \mathbf{\Sigma}_{yy} - \mathbf{\Sigma}_{yx}\mathbf{\Sigma}_{xx}^{-1}\mathbf{\Sigma}_{xy}). \tag{2.6}$$

An example of a joint distribution and the corresponding marginal distributions is shown in Figure 2.1.

2.1.2 The Sigma Point Transform

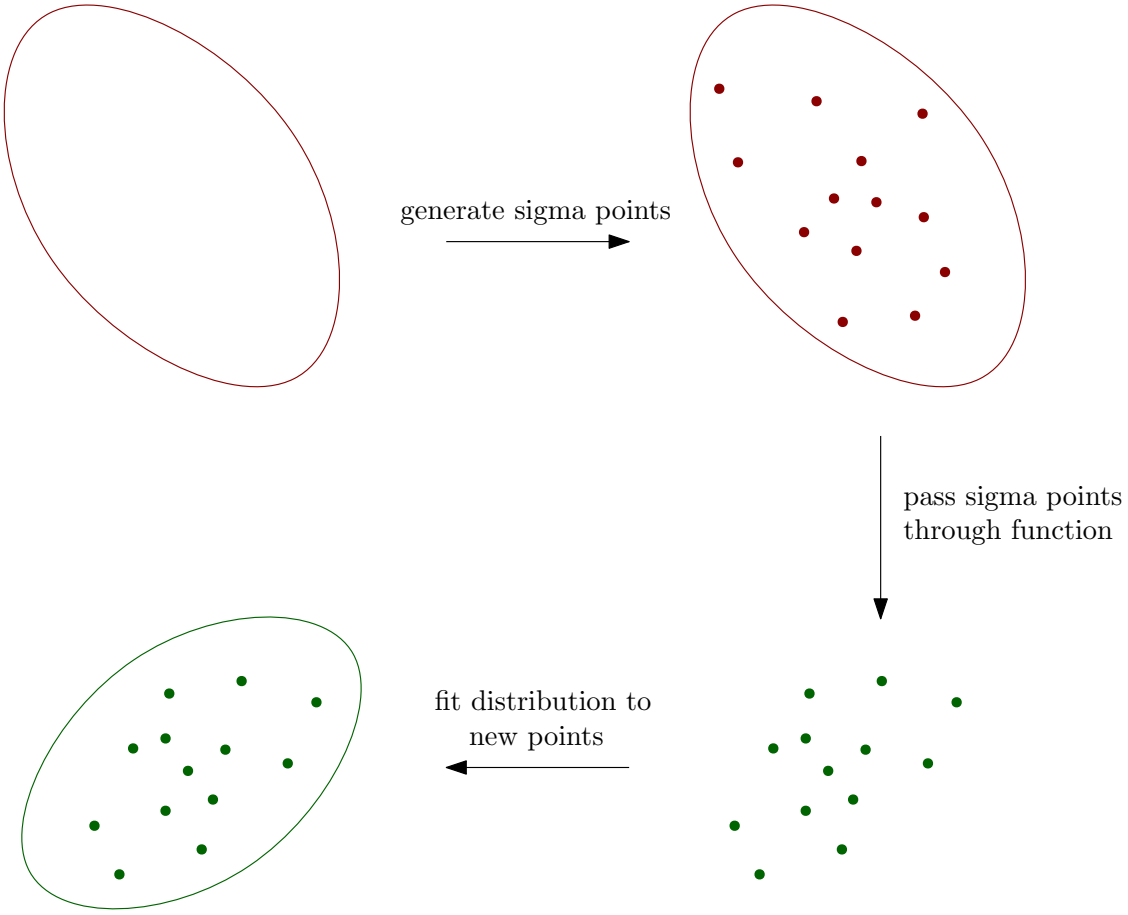


Figure 2.2: A visualization of how the sigma point transformation can be used to compute the posterior distribution (in green) from the prior distribution (in red).

Oftentimes a need arises to pass a random variable through a function. The sigma point transform is a method used to approximate the nonlinear transformation of a distribution. By generating a set of sigma points from the *a priori* distribution and passing them through the nonlinear function, the newly generated points are used to approximate the nonlinearly transformed distribution [43, Section 5.5].

Consider a nonlinear function $\mathbf{z} = \mathbf{h}(\mathbf{x}, \mathbf{y})$, where $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$ is a p-dimensional random variable and $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)$ a q-dimensional one, and $\mathbf{x}$ and $\mathbf{y}$ are assumed to be mutually uncorrelated random variables. To find the distribution of $\mathbf{z}$ using a sigma point transformation, the random variables are augmented into one vector $\mathbf{v} \sim \mathcal{N}(\boldsymbol{\mu}_v, \boldsymbol{\Sigma}_v)$ of dimension $p + q$, where

$$\boldsymbol{\mu}_v = \begin{bmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\mu}_y \end{bmatrix}, \quad \boldsymbol{\Sigma}_v = \begin{bmatrix} \boldsymbol{\Sigma}_x & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_y \end{bmatrix}. \tag{2.7}$$

The next step involves generating the sigma points from (2.7), which can be done using different approaches. The spherical cubature rule [43, Section 6.5] will be used throughout this thesis. Define $L \triangleq \dim(\boldsymbol{\mu}_v)$ and the Cholesky decomposition $\mathbf{L}\mathbf{L}^\mathsf{T} \triangleq \boldsymbol{\Sigma}_v$, where $\mathbf{L}$ is a lower-triangular matrix. The spherical cubature rule results in a total number of L sigma point pairs, where the i^{th} pair is defined as

$$\mathbf{s}_i \triangleq \boldsymbol{\mu}_v + \sqrt{L}\,\mathrm{col}_i(\mathbf{L}), \qquad \mathbf{s}_{i+L} \triangleq \boldsymbol{\mu}_v - \sqrt{L}\,\mathrm{col}_i(\mathbf{L}),$$

and $\mathrm{col}_i(\mathbf{L})$ is the i^{th} column of the matrix $\mathbf{L}$. By unstacking the i^{th} sigma point into the two components $\mathbf{x}_i$ and $\mathbf{y}_i$, the i^{th} transformed point is $\mathbf{z}_i = \mathbf{h}(\mathbf{x}_i, \mathbf{y}_i)$. The new transformed distribution can then be approximated using the $2L$ transformed points through

$$\boldsymbol{\mu}_z = \frac{1}{2L}\sum_{i=1}^{2L} \mathbf{z}_i, \qquad \boldsymbol{\Sigma}_z = \frac{1}{2L}\sum_{i=1}^{2L} (\mathbf{z}_i - \boldsymbol{\mu}_z)(\mathbf{z}_i - \boldsymbol{\mu}_z)^\mathsf{T}. \tag{2.8}$$

This process is illustrated in Figure 2.2.

2.2 Ultra-wideband Ranging

UWB ranging between two transceivers relies on ToF measurements, which are deduced from timestamps recorded by a clock on each transceiver. However, these clocks are unsychronized. Denoting $t_i(t)$ as the time t resolved in Transceiver i's clock gives

$$t_i(t) = t + \tau_i(t), \tag{2.9}$$

where $\tau_i(t)$ defines the time-varying *offset* of clock i. The evolution of the offset $\tau_i(t)$ of clock i can be modelled using a third-order ordinary differential equation (ODE) [44]. However, [22] shows that a second-order ODE of the form

$$\begin{bmatrix} \dot{\tau}_i \\ \dot{\gamma}_i \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tau_i \\ \gamma_c \end{bmatrix} + \mathbf{w}_i \tag{2.10}$$

is sufficient for localization purposes, where $\gamma_i(t)$ is called the *clock skew*, $\mathbf{w}_i$ is a continuous-time zero-mean white Gaussian process noise with $\mathbb{E}[\mathbf{w}_i(t_1)\mathbf{w}_i(t_2)] = \boldsymbol{\mathcal{Q}}\delta(t_1 - t_2)$,

$$\boldsymbol{\mathcal{Q}} = \begin{bmatrix} \mathcal{Q}^\tau & \\ & \mathcal{Q}^\gamma \end{bmatrix},$$

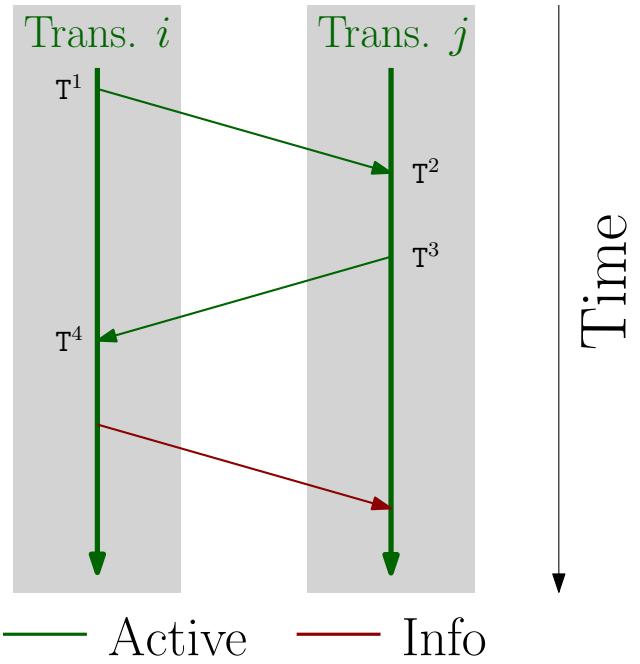


Figure 2.3: The SS-TWR protocol.

$\delta(\cdot)$ is the *Dirac's delta function*, and $\mathcal{Q}^\tau$ and $\mathcal{Q}^\gamma$ are the clock offset and skew process-noise *power spectral densities* (PSDs), respectively, which can be tuned based on the user's knowledge of the quality of the clocks. Additionally, note that the time-dependence has been omitted from the notation in (2.10) for conciseness.

To obtain a range measurement, the ToF between two transceivers must be measured. Consider the first message in Figure 2.3. Transceiver i transmits a message at time instance $\mathtt{T}^1$ that is received by Transceiver j at time instance $\mathtt{T}^2$, and the true ToF is then $t_f = \mathtt{T}^2 - \mathtt{T}^1$. In the absence of other sources of sensor noise and in the absence of any clock offsets (i.e., $\tau_i = \tau_j = 0$), the ToF t_f can be found simply by subtracting the timestamps recorded by the clocks.

However, the clocks are not synchronized in practice, and the time instances $\mathtt{T}^1$ and $\mathtt{T}^2$ as timestamped by Transceiver i's and Transceiver j's clocks are

$$\mathtt{T}_i^1 = \mathtt{T}^1 + \tau_i, \tag{2.11}$$

$$\mathtt{T}_j^2 = \mathtt{T}^2 + \tau_j, \tag{2.12}$$

respectively, under a constant clock-offset model. This means that taking the difference

yields a biased range measurement, since

$$\mathtt{T}_j^2 - \mathtt{T}_i^1 = \mathtt{T}^2 + \tau_j - \mathtt{T}^1 - \tau_i \tag{2.13}$$
$$= t_f + \tau_j - \tau_i. \tag{2.14}$$

The bias here can be significant, as 1 ns in synchronization error translates to c [m/s] $\times$ 10^{-9} [s] ≈ 30 [cm] in localization error, where c is the speed of light.

The most standard solution to this problem is to perform single-sided two-way ranging (SS-TWR), where, as shown in Figure 2.3, Transceiver j responds with another message. Under an assumption that the transceivers have not moved relative to one another, $t_f = \mathtt{T}^4 - \mathtt{T}^3 = \mathtt{T}^2 - \mathtt{T}^1$. Also, under the constant-offset assumption, the time instances $\mathtt{T}^3$ and $\mathtt{T}^4$ are timestamped as

$$\mathtt{T}_j^3 = \mathtt{T}^3 + \tau_j, \tag{2.15}$$
$$\mathtt{T}_i^4 = \mathtt{T}^4 + \tau_i, \tag{2.16}$$

respectively. Taking the average of the two ToF measurements such that

$$\frac{1}{2}((\mathtt{T}_j^4 - \mathtt{T}_i^3) + (\mathtt{T}_j^2 - \mathtt{T}_i^1)) = \frac{1}{2}((\mathtt{T}^4 + \tau_i - \mathtt{T}^3 - \tau_j) + (\mathtt{T}^2 + \tau_j - \mathtt{T}^1 - \tau_i)) \tag{2.17}$$
$$= t_f \tag{2.18}$$

causes the offsets to cancel out, thus outputting an unbiased ToF measurement.

Note that the last message shown in red represents an "information message" used to share the timestamps recorded by Transceiver i such that the robot with Transceiver j can also output a range measurement. Additionally, note that the above derivation follows from a constant offset assumption, which is in itself another source of error in the above model as will be further discussed in Chapter 4.

2.3 Matrix Lie Groups

Rather than using the conventional vector-space representation, this thesis uses *matrix Lie groups* to represent the state of the robot. This approach is taken because the state of a robot is often constrained to a manifold, and matrix Lie groups are a natural way to represent such a constrained state. For example, representing attitude using Euler angles is problematic due to the non-unique nature of such representation of attitude, as well as singularities in the attitude kinematics. Matrix Lie groups provide a more natural representation of attitude.

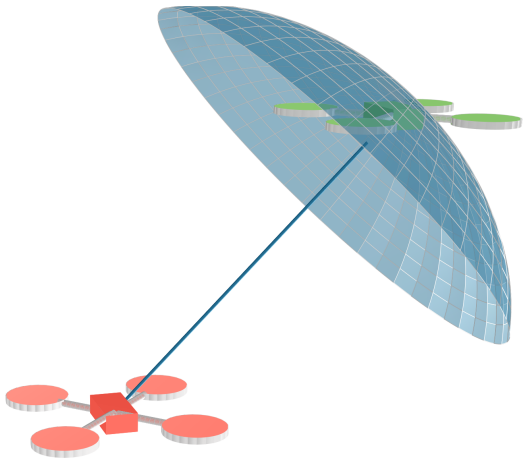


Figure 2.4: The distribution of the posterior position of the green robot given a position prior and a single range measurement with the red robot.

Further motivating the use of matrix Lie groups is the ability to compactly and more accurately represent the state of a robot and its uncertainty. For example, consider the problem of estimating the position of a robot given a prior position and a single range measurement to a landmark or another robot. The posterior distribution of the position of the robot's centre is shown in Figure 2.4. Under a Gaussian assumption, this banana-shaped distribution cannot be accurately represented using traditional vector-space-based state representations, as this would require fitting an ellipsoid to the banana-shaped distribution. However, using matrix Lie groups, the banana-shaped distribution can be accurately represented using a Gaussian distribution, leading to better-performing state estimators.

This thesis only considers matrix Lie groups, which is a subset of the more general *Lie groups*. A Lie group is a group that is also a smooth manifold, where a group is a set with a binary operation that satisfies what is referred to as the the *group axioms*, which is beyond the scope of this thesis. The interested reader can refer to [45]. Matrix Lie groups, as compared to the more general Lie groups, have the additional property that the elements of the group can be represented as square, invertible matrices, and that the group operation is matrix multiplication.

2.3.1 Definitions and Useful Identities

The relevant mathematical foundations for manipulating matrix Lie groups will be addressed next. Consider two elements $\mathbf{X}, \mathbf{Y} \in G$ of a matrix Lie group G, where $G \subset \mathbb{R}^{n \times n}$. Any matrix Lie group is closed under matrix multiplication, meaning that $\mathbf{XY} \in G$. Additionally, $\mathbf{X}^{-1} \in G$, and the identity element of the group is just the matrix identity $\mathbf{1}$ of dimension n.

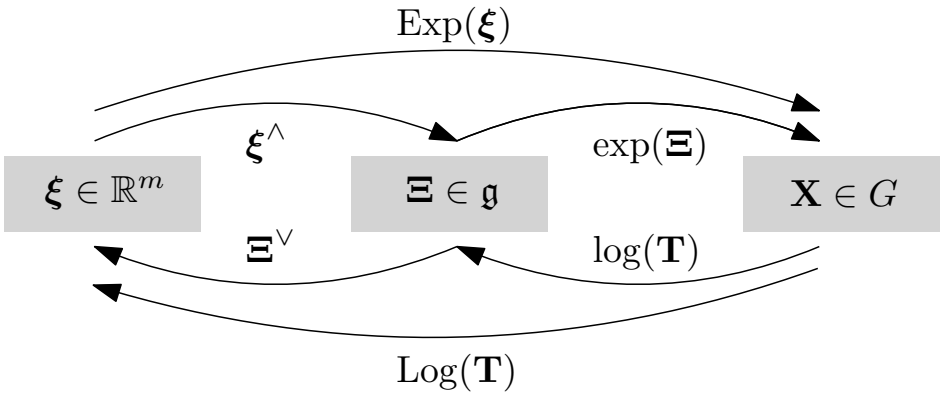


Figure 2.5: A summary of the operators between elements of the different spaces associated with matrix Lie groups.

Associated with every matrix Lie group is a *matrix Lie algebra* $\mathfrak{g}$, whose elements consist of matrices that also belong to $\mathbb{R}^{n\times n}$. The matrix Lie algebra is formally defined as the tangent space of the matrix Lie group at the identity element. Elements of the matrix Lie algebra $\boldsymbol{\Xi} \in \mathfrak{g}$ can also be represented as elements of $\mathbb{R}^m$ by defining a linear mapping $(\cdot)^\wedge : \mathbb{R}^m \to \mathfrak{g}$ such that

$$\boldsymbol{\Xi} = \boldsymbol{\xi}^\wedge, \qquad \boldsymbol{\xi} \in \mathbb{R}^m.$$

Similarly, the inverse of $(\cdot)^\wedge$, denoted $(\cdot)^\vee : \mathfrak{g} \to \mathbb{R}^m$, is defined such that $\boldsymbol{\xi} = \boldsymbol{\Xi}^\vee$.

Mapping elements of the matrix Lie algebra $\mathfrak{g}$ to the matrix Lie group G is the *exponential map* $\exp : \mathfrak{g} \to G$, thus yielding the definition

$$\mathbf{X} = \exp(\boldsymbol{\xi}^\wedge) \triangleq \mathrm{Exp}(\boldsymbol{\xi}),$$

where $\mathrm{Exp} : \mathbb{R}^m \to G$ is defined for conciseness. The inverse of the exponential map is the *logarithmic map* $\log : G \to \mathfrak{g}$, yielding the definition

$$\boldsymbol{\xi} = \log(\mathbf{X})^\vee \triangleq \mathrm{Log}(\mathbf{X}),$$

where $\mathrm{Log} : G \to \mathbb{R}^m$ is also defined for conciseness. Owing to the fact that G is a matrix Lie group, the exponential map and logarithmic map are the same as the matrix exponential and matrix logarithm, respectively. A summary of these operators is shown in Figure 2.5.

Another useful identity of matrix Lie groups comes from the *Adjoint matrix* Ad, defined as

$$\mathrm{Ad}(\mathbf{X})\boldsymbol{\xi} \triangleq (\mathbf{X}\boldsymbol{\xi}^\wedge\mathbf{X}^{-1})^\vee, \tag{2.19}$$

which can be similarly written as

$$\mathrm{Exp}(\mathrm{Ad}(\mathbf{X})\boldsymbol{\xi}) \triangleq \mathbf{X}\,\mathrm{Exp}(\boldsymbol{\xi})\mathbf{X}^{-1}.$$

Lastly, another identity that will be useful for the derivations in this thesis is the *odot* operator $(\cdot)^{\odot} : \mathbb{R}^n \to \mathbb{R}^{n\times m}$, defined such that

$$\mathbf{p}^{\odot}\boldsymbol{\xi} \triangleq \boldsymbol{\xi}^{\wedge}\mathbf{p} \tag{2.20}$$

for any vector $\mathbf{p} \in \mathbb{R}^n$.

2.3.2 Perturbing Matrix Lie Group Elements

The non-commutativity of matrix multiplication means that matrix Lie group elements can be perturbed in two ways, from the left

$$\mathbf{X} = \mathrm{Exp}(\delta\boldsymbol{\xi})\bar{\mathbf{X}}$$

or the right

$$\mathbf{X} = \bar{\mathbf{X}}\,\mathrm{Exp}(\delta\boldsymbol{\xi}),$$

where the overbar denotes a nominal value. Typically, when dealing with random elements of a group, $\bar{\mathbf{X}}$ is assumed to be the mean, and the perturbation from the nominal value is modelled by assuming that the quantity $\delta\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ is a zero-mean Gaussian random variable with covariance $\boldsymbol{\Sigma}$. Additionally, the first-order approximation

$$\mathrm{Exp}(\delta\boldsymbol{\xi}) \approx \mathbf{1} + \delta\boldsymbol{\xi}^{\wedge} \tag{2.21}$$

will often be used when linearizing nonlinear models.

2.3.3 Common Matrix Lie Groups Used in Robotics

Having formalized the basic mathematical foundations of matrix Lie groups, this section will now present some of the more common matrix Lie groups used in robotics, particularly when states are an element of three-dimensional space (3D). This section only provides the formal definition of each group; the exact expression of the operators on the different groups can be found in [46, Chapters 2.2-2.4].

Special Orthogonal Group $SO(3)$

Attitude is typically parametrized using a rotation matrix or a *direction cosine matrix* (DCM) $\mathbf{C}$, which is an element of the *special orthogonal group* $SO(3)$. The group is formally defined as

$$SO(3) = \left\{ \mathbf{C} \in \mathbb{R}^{3\times 3} \mid \mathbf{C}^{\mathsf{T}}\mathbf{C} = \mathbf{1},\ \det \mathbf{C} = +1 \right\}. \tag{2.22}$$

Special Euclidean Group $SE(3)$

Another common matrix Lie group is the *special Euclidean group* $SE(3)$, which is the group of rigid-body transformations in 3D. The *pose* of one rigid body relative to another is defined using the relative attitude and position $(\mathbf{C}, \mathbf{r})$. The group is formally defined as

$$SE(3) = \left\{ \mathbf{T} \in \mathbb{R}^{4\times 4} \mid \mathbf{T} = \begin{bmatrix} \mathbf{C} & \mathbf{r} \\ \mathbf{0} & 1 \end{bmatrix},\ \mathbf{C} \in SO(3),\ \mathbf{r} \in \mathbb{R}^3 \right\}. \tag{2.23}$$

Extended Special Euclidean Group $SE_2(3)$

Lastly, extending $SE(3)$ to include velocity is the *extended special Euclidean group* $SE_2(3)$. The *extended pose* of one rigid body relative to another is defined using the relative attitude, velocity, and position $(\mathbf{C}, \mathbf{v}, \mathbf{r})$. The extended pose can be represented using an extended pose transformation matrix [47], and the group is formally defined as

$$SE_2(3) = \left\{ \mathbf{T} \in \mathbb{R}^{5\times 5} \mid \mathbf{T} = \begin{bmatrix} \mathbf{C} & \mathbf{v} & \mathbf{r} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix},\ \mathbf{C} \in SO(3),\ \mathbf{v}, \mathbf{r} \in \mathbb{R}^3 \right\}. \tag{2.24}$$

2.4 Kalman Filtering

Consider a discrete-time linear system given by

$$\mathbf{x}_k = \mathbf{A}_{k-1}\mathbf{x}_{k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1} + \mathbf{w}_{k-1}, \quad \mathbf{w}_{k-1} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_{k-1}\right), \tag{2.25}$$

$$\mathbf{y}_k = \mathbf{C}_k\mathbf{x}_k + \boldsymbol{\nu}_k, \qquad \boldsymbol{\nu}_k \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_k\right), \tag{2.26}$$

where $\mathbf{x}$ is some state that is to be estimated, $\mathbf{u}$ is some *interoceptive* measurement, $\mathbf{y}$ is an *exteroceptive* measurement, and the subscript denotes the time-step. Additionally, assume that there is some known prior estimate, $\mathbf{x}_0 \sim \mathcal{N}\left(\check{\mathbf{x}}_0, \check{\mathbf{P}}_0\right)$. The minimum mean square error (MMSE) estimator of $\mathbf{x}_k$ given past measurements is given by the Kalman filter (KF) [25],

where the prediction or propagation step is given by

$$\check{\mathbf{x}}_k = \mathbf{A}_{k-1}\hat{\mathbf{x}}_{k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1}, \tag{2.27}$$

$$\check{\mathbf{P}}_k = \mathbf{A}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{A}_{k-1}^{\mathsf{T}} + \mathbf{Q}_{k-1}, \tag{2.28}$$

and the correction or update step is given by

$$\mathbf{S}_k = \left(\mathbf{C}_k\check{\mathbf{P}}_k\mathbf{C}_k^{\mathsf{T}} + \mathbf{R}_k\right), \tag{2.29}$$

$$\mathbf{K}_k = \check{\mathbf{P}}_k\mathbf{C}_k^{\mathsf{T}}\mathbf{S}_k^{-1}, \tag{2.30}$$

$$\mathbf{z}_k = \left(\mathbf{y}_k - \mathbf{C}_k\check{\mathbf{x}}_k\right), \tag{2.31}$$

$$\hat{\mathbf{x}}_k = \check{\mathbf{x}}_k + \mathbf{K}_k\mathbf{z}_k, \tag{2.32}$$

$$\hat{\mathbf{P}}_k = \left(\mathbf{1} - \mathbf{K}_k\mathbf{C}_k\right)\check{\mathbf{P}}_k. \tag{2.33}$$

The prediction step typically comes from a motion model that is a function of interoceptive measurements, and the correction step typically comes from a measurement or observation model that is a function of exteroceptive measurements. Note that $\mathbf{z}_k$ is typically referred to as the *innovation*, $\mathbf{S}_k$ as the *innovation covariance*, and $\mathbf{K}_k$ as the *Kalman gain*.

In order to use the KF with nonlinear models, the nonlinear models are linearized by perturbing the states to give the *extended Kalman filter* (EKF). Additionally, when estimating states that are elements of matrix Lie groups, the update given in (2.32) is replaced with a multiplicative update, yielding an on-manifold EKF [48]. Although this thesis uses the KF and its variants in their standard form, it is worth noting that there exists a large body of literature on numerical methods for implementing the KF and its variants [49, Chapters 7 and 9].

2.4.1 Consistency

The most obvious way to evaluate state estimators is to compare the estimated state with the true state. However, equally important is the ability to evaluate how well the state estimator characterizes the uncertainty in the state estimate. An overconfident estimator will underestimate its uncertainty, which can be catastrophic in safety-critical applications. On the other hand, an underconfident estimator will overestimate its uncertainty, which can lead to suboptimal performance. An example of this is shown in Figure 2.6.

The term *consistency* of a filter is typically used to evaluate how well a filter estimates its own level of confidence. When considering filters that assume that the state distribution is a Gaussian, a filter's estimated mean $\hat{\mathbf{x}}_k \in \mathbb{R}^n$ and corresponding estimated uncertainty

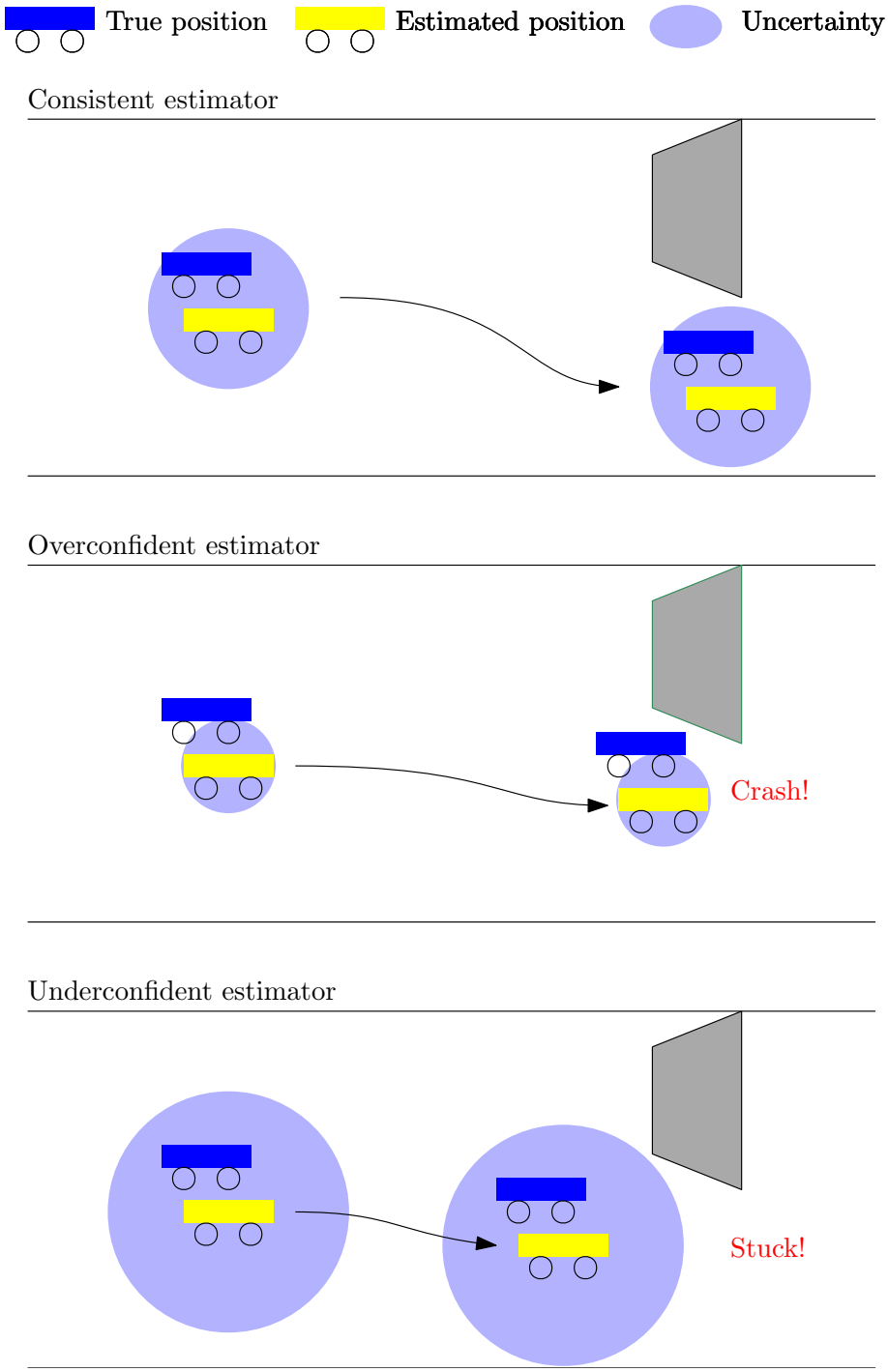


Figure 2.6: A visualization of the difference between an overconfident estimator and an underconfident estimator using a ground vehicle trying to use its state estimate to assess if it is safe to pass an obstacle. A consistent estimator will allow the robot to pass the obstacle without colliding, while an overconfident estimator might result in a crash with the obstacle. Meanwhile, an underconfident estimator is typically too conservative and will result in the robot either not passing the obstacle or taking a suboptimal path where it needs to deviate too much from its original trajectory.

$\hat{\mathbf{P}}_k \in \mathbb{R}^{n\times n}$ represent the filter's knowledge of a true state $\mathbf{x}_k \in \mathbb{R}^n$ at time-step k. In [50], a filter is formally defined as consistent if the following conditions are satisfied,

$$\mathbb{E}[\mathbf{x}_k - \hat{\mathbf{x}}_k] = \mathbf{0}, \tag{2.34}$$

$$\mathbb{E}\left[(\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^\mathsf{T}\right] = \hat{\mathbf{P}}_k. \tag{2.35}$$

The former is an *unbiasedness* requirement, and the latter is a *covariance-matching* requirement. A more lenient definition of consistency is presented in [36], where a filter is defined as consistent if it satisfies the condition

$$\hat{\mathbf{P}}_k - \mathbb{E}\left[(\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^\mathsf{T}\right] \succeq \mathbf{0}, \tag{2.36}$$

meaning that a filter is said to be consistent as long as the filter is not overconfident.

Having defined consistency, a consistency metric is now required. The normalized estimation error squared (NEES) metric is a scalar metric that is used to evaluate the consistency of a filter, and is defined as

$$\epsilon_k = (\mathbf{x}_k - \hat{\mathbf{x}}_k)^\mathsf{T}\hat{\mathbf{P}}_k^{-1}(\mathbf{x}_k - \hat{\mathbf{x}}_k). \tag{2.37}$$

If the filter is indeed consistent, then $\epsilon_k \sim \chi^2_n$, where χ^2_n is the chi-squared distribution with n degrees of freedom. The NEES test can be thought of as a state-estimation application of the *chi-squared test* [51].

If N Monte Carlo trials are executed, the N-run average NEES can be defined as

$$\bar{\epsilon}_k = \frac{1}{N}\sum_{i=1}^{N}\epsilon_k^{(i)}, \tag{2.38}$$

where $\epsilon_k^{(i)}$ is the NEES of the i^{th} Monte Carlo trial. If the filter is indeed consistent, then $N\bar{\epsilon}_k \sim N\chi^2_n$. More details regarding the evaluation of the consistency of a filter can be found in [50, Ch. 5.4].

2.4.2 Statistical Outlier Rejection

The NEES test requires knowledge of the ground truth; therefore, it cannot be used in real-time when a filter is running on-board a robot. Another statistical test, the *normalized innovation squared* (NIS) test, relies on the fact that the innovation $\mathbf{z}_k \in \mathbb{R}^m$ at time-step k, as defined by (2.31), should satisfy the estimate covariance $\mathbf{S}_k \in \mathbb{R}^{m\times m}$, as computed in

(2.29). Similar to the NEES metric ϵ_k, the NIS metric is defined

$$\kappa_k = \mathbf{z}_k^\mathsf{T} \mathbf{S}_k^{-1} \mathbf{z}_k, \tag{2.39}$$

and if the measurement is not an outlier, then $\kappa_k \sim \chi^2_m$. The NIS test can be thought of as an outlier-rejection application of the chi-squared test that can be implemented in real-time as it only relies on quantities computed by the filter and the available measurements. Again, for more information regarding the NIS test, see [50, Ch. 5.4].

2.5 Kinematics

This thesis utilizes 3D motion extensively to model the motion of quadcopters in both simulation and real-world experiments. This section presents the motion model of the quadcopters in 3 dimensions as a function of the measurements of the IMU to be used in a filtering framework alongside the range measurements. The motion model presented here is shown from the perspective of one robot as resolved in some arbitrary fixed reference frame a, denoted $\mathcal{F}_a$, often referred to as the "absolute" or "global" frame. This will be used in the remaining chapters to derive the relative-motion model between quadcopters, and to formulate a "robocentric" viewpoint, where states are resolved in a robot's body-fixed frame, denoted $\mathcal{F}_b$. Furthermore, the motion model will be reformulated directly on the $SE_2(3)$ manifold in Chapter 6.

2.5.1 IMU Motion Model in 3D

The standard IMU motion model in 3D is as follows [8, Chapter 11]. The attitude of a robot is parametrized using a DCM $\mathbf{C}_{ab}$. The attitude process model is given as

$$\dot{\mathbf{C}}_{ab} = \mathbf{C}_{ab} \left(\boldsymbol{\omega}_b^{ba}\right)^\times, \tag{2.40}$$

where time-dependence is omitted from the notation for conciseness, $\boldsymbol{\omega}_b^{ba}$ is the angular velocity of $\mathcal{F}_b$ relative to $\mathcal{F}_a$ as resolved in $\mathcal{F}_b$, and $(\cdot)^\times$ is the skew-symmetric cross operator, which is in fact the $(\cdot)^\wedge : \mathbb{R}^3 \to \mathfrak{so}(3)$ operator on $SO(3)$.

Meanwhile, the robot's velocity process model is given as

$$^a\dot{\underrightarrow{v}}^{zw/a} = \underrightarrow{a}^{zw/a}, \tag{2.41}$$

where $\underrightarrow{v}^{zw/a}$ is the velocity of the robot at point z relative to some arbitrary point w with

respect to $\mathcal{F}_a$, and ${}^a\underset{\rightarrow}{\dot{v}}^{zw/a}$ is its derivative with respect to $\mathcal{F}_a$. Similarly, the robot's position process model is given as

$$ {}^a\underset{\rightarrow}{\dot{r}}^{zw} = \underset{\rightarrow}{v}^{zw/a}. \tag{2.42} $$

The interoceptive or *proprioceptive* sensor used is an IMU, which consists of a 3-axis gyroscope and a 3-axis accelerometer. Assuming no bias and measurement noise, the gyroscope measures $\boldsymbol{\omega}_b^{ba}$ directly, while the accelerometer measures the specific acceleration

$$ \underset{\rightarrow}{\alpha} \triangleq \underset{\rightarrow}{a}^{zw/a} - \underset{\rightarrow}{g} \tag{2.43} $$

in the body frame, where $\underset{\rightarrow}{g}$ is the gravitational acceleration vector. Substituting (2.43) back in (2.41) and resolving the state and the gravitational acceleration vector in $\mathcal{F}_a$ and the accelerometer measurement in $\mathcal{F}_b$ gives

$$ {}^a\dot{\mathbf{v}}_a^{zw/a} = \mathbf{C}_{ab}\boldsymbol{\alpha}_b + \mathbf{g}_a, \tag{2.44} $$

where $\mathbf{x}_i$ is the vector $\underset{\rightarrow}{x}$ resolved in $\mathcal{F}_i$. Similarly,

$$ {}^a\dot{\mathbf{r}}_a^{zw} = \mathbf{v}_a^{zw/a}. \tag{2.45} $$

2.5.2 Transport Theorem

In order to derive the robocentric relative motion model, the *transport theorem* will be needed. The transport theorem is a generalization of the chain rule that relates the derivatives of a physical vector with respect to two different reference frames [52, Chapter 2.10]. The transport theorem for a vector $\underset{\rightarrow}{r}$ and two reference frames $\mathcal{F}_a$ and $\mathcal{F}_b$ is formally given as

$$ {}^a\underset{\rightarrow}{\dot{r}} = {}^b\underset{\rightarrow}{\dot{r}} + \underset{\rightarrow}{\omega}^{ba} \times \underset{\rightarrow}{r}, \tag{2.46} $$

where $\underset{\rightarrow}{\omega}^{ba}$ is the angular velocity of $\mathcal{F}_b$ relative to $\mathcal{F}_a$.

Chapter 3

Relative Position Estimation in Multi-Agent Systems Using Attitude-Coupled Range Measurements

Summary

The task of three-dimensional relative position estimation using range measurements in multi-agent systems suffers from *unobservabilities*. The contributions of this chapter include the derivation of a sufficient condition for the observability of the relative positions, and satisfying the condition using a simple framework with only range measurements, an accelerometer, a rate gyro, and a magnetometer on each robot. The framework is tested in simulation and in experiments, where 40-50 cm positioning accuracy is achieved using inexpensive off-the-shelf hardware.

3.1 Introduction

Estimating three-dimensional relative positions between robots in a global frame using range measurements is a non-trivial task, as there is an infinite number of possible solutions given a single range measurement. This unobservability arises from the fact that any group of agents can be collectively rotated in three-dimensional space while maintaining constant inter-agent distances, as no bearing information is available. There exist a multitude of approaches that attempt to fuse the range measurements with additional information to achieve an observable problem.

Most indoor localization approaches traditionally assume the existence of an infrastruc-

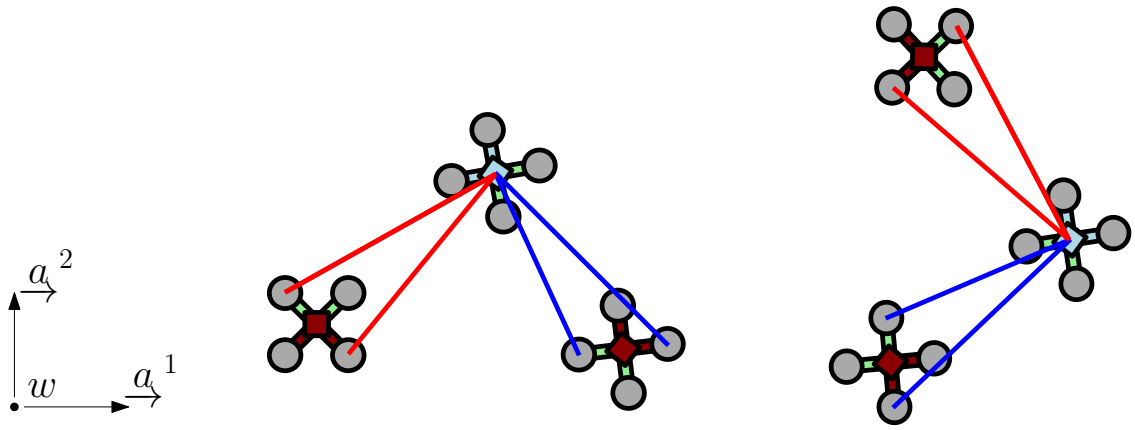


Figure 3.1: A schematic of the two-tag agent framework, where the red agents possess two ranging *tags*, and are referred to as two-tag agents, and the blue agent is a single-tag agent.

ture of 4 or more *anchors* with known positions [14], [17], [22]. However, the problem of relative localization has also been addressed in the presence of a single anchor. In [53], only a single anchor is used for position tracking, with a constant velocity assumption, while a range-based SLAM approach is utilized for relative localization in [16] and [54]. Other approaches that eliminate the need for anchors in two-dimensional relative position estimation include [55] and [56], which assume that displacement measurements are available using an optical flow sensor. In [18], a sliding window filter is able to estimate the three-dimensional relative position between two agents using just single-range measurements and 9-axis inertial measurement units (IMUs), where a 9-axis IMU is an IMU with a 3-axis magnetometer alongside the gyroscope and accelerometer. All these single-range-based localization approaches usually require persistent relative motion between the anchor and the agent [16], [53], [54] or the two agents [18], [55], [56], as outlined in [57]. In the presence of many agents, [58] and [59] show how the nonlinear observability matrix associated with a two-dimensional relative localization problem is dependent on both the rigidity matrix and the relative motion of the agents. Alternative approaches include the implementation of a particle filter when only an IMU and range measurements are available, as in [60], which is computationally expensive.

More recently, the idea of using multi-tag agents, as shown in Figure 3.1, has been proposed. In [61], UWB range sensors are used for relative positioning of trucks fitted with two tags, and in [62], an agent is equipped with three tags. Both these methods extract two-dimensional relative position information in the body frame of the computing agent. In [19], the users are capable of tracking a person in two dimensions using a special ranging protocol with 4 tags on an agent, while in [20], multiple tags on a moving platform allow an agent to approach using range and relative displacement measurements, and land using vision and range measurements. Lastly, in [63], the use of two two-tag agents is coupled

with an altimeter and optical flow velocity measurements for relative localization, and the results are validated in an experiment with limited motion. The main limitation of the results in [19], [20], [61]–[63] is that the analysis is mainly restricted to only two agents, and no observability analysis is considered.

The contributions of this chapter are threefold. The first contribution is a rigidity-theory-based observability analysis for any number of agents, where a sufficient condition that is independent of the relative motion of the agents is derived for the observability of the three-dimensional relative positions when only range measurements are available. This motivates the second contribution, which is an extension of the two-tag framework to multi-agent systems that allows the estimation of three-dimensional relative positions using just the range measurements and a low-cost 9-axis IMU per agent, for any number of agents, provided at least two two-tag agents are present. This framework is shown to be *instantaneously locally observable*, as per the sufficient condition, which means that the system is locally observable at any given point in time without any specific trajectory requirements. Lastly, the performance of this framework in simulations and in experiments using multiple agents equipped with inexpensive sensors is presented.

The remainder of this chapter is organized as follows. Graph theoretic and observability concepts are reviewed in Section 3.2. A sufficient condition for observability of a three-dimensional relative localization problem is addressed in Section 3.3. The two-tag framework is discussed in Section 3.4 and is validated in simulation and experimentally in Sections 3.5 and 3.6, respectively.

3.2 Rigidity Theory and Instantaneous Local Observability

This chapter uses graph theory as one of the fundamental tools for observability analysis. Consider an *undirected graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting of a set of n vertices and m edges, representing the n tags and m distance measurements, respectively. As such, let $\mathbf{r}_a^{f_i f_0} \in \mathbb{R}^3$ represent the unknown location of Tag f_i relative to Tag f_0, resolved in the 3-dimensional reference frame $\mathcal{F}_a$. Additionally, let $y^{f_i f_j}(\mathbf{r}_a^{f_i f_0}, \mathbf{r}_a^{f_j f_0}) \in \mathbb{R}$, $(i, j) \in \mathcal{E}$ be the range measurement between Tags f_i and f_j, where $(f_i, f_i) \notin \mathcal{E}$, $\forall f_i \in \mathcal{V}$. All graphs defined in this chapter are assumed to have this property. Define a column matrix

$$\mathbf{x} \triangleq \left[\left(\mathbf{r}_a^{f_1 f_0}\right)^\mathsf{T} \; \cdots \; \left(\mathbf{r}_a^{f_{n-1} f_0}\right)^\mathsf{T} \right]^\mathsf{T} \in \mathbb{R}^{3(n-1)} \tag{3.1}$$

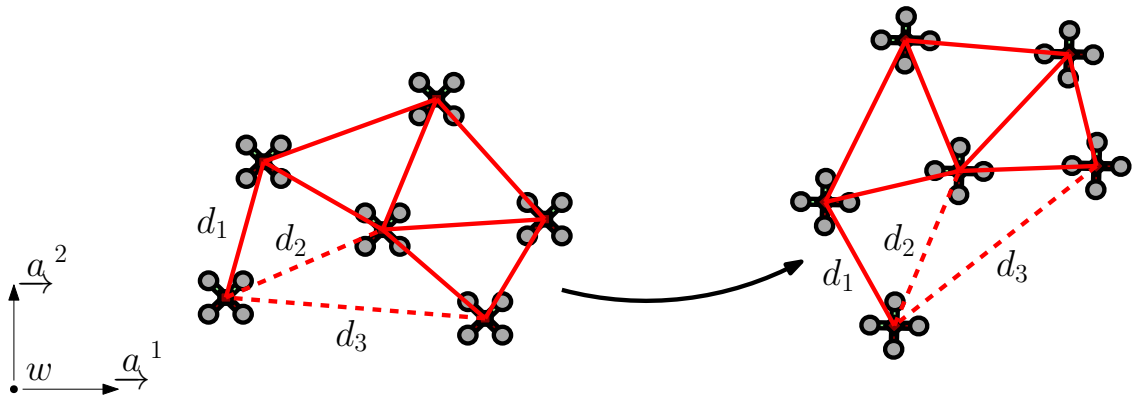


Figure 3.2: An example of a rigid graph. However, when edges d_2 and d_3 are removed, the graph becomes *flexible*, as edge d_1 can rotate about its upper vertex without breaking any constraints. Additionally, both the rigid and the flexible graphs can translate and rotate without breaking any constraints.

of the $n-1$ relative positions relative to Tag f_0, and a column matrix $\mathbf{y}(\mathbf{x}) \in \mathbb{R}^m$ of all the known measurements $y^{f_i f_j}$.

Parametrize the vector space spanned by the system state by a new arbitrary variable t. To analyze the behaviour of the measurements $\mathbf{y}(\mathbf{x}(t))$ for any infinitesimal change in the states $\mathbf{x}(t)$ with respect to t, the derivative

$$\frac{\mathrm{d}\mathbf{y}(\mathbf{x}(t))}{\mathrm{d}t} = \frac{\partial \mathbf{y}(\mathbf{x}(t))}{\partial \mathbf{x}} \frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} \triangleq \mathbf{R}\dot{\mathbf{x}}(t) \tag{3.2}$$

is computed, where $\mathbf{R} \in \mathbb{R}^{m \times 3(n-1)}$ is the *rigidity matrix*.

Traditionally, rigidity theory in $\mathbb{R}^3$ is concerned with the notion of *infinitesimal rigidity* by achieving rank $\mathbf{R} = 3n-6$ for n absolute position states, where the 6 degrees of freedom are associated with the translations and rotations of the graph as a whole [64], as shown in Figure 3.2. Similarly, when dealing with $n-1$ three-dimensional relative states as in (3.2), infinitesimal rigidity is achieved when rank $\mathbf{R} = 3(n-1)-3 = 3n-6$, where the 3 degrees of freedom are associated with the rotations of the graph as a whole.

Therefore, when addressing relative position states, the only non-trivial solutions to $\mathbf{R}\dot{\mathbf{x}} = \mathbf{0}$ of an infinitesimally rigid graph are of the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{r}_a^{f_i f_0} = \boldsymbol{\omega}^{\times}\mathbf{r}_a^{f_i f_0} = -\left(\mathbf{r}_a^{f_i f_0}\right)^{\times}\boldsymbol{\omega}, \quad \forall f_i \in \mathcal{V}\backslash\{f_0\}, \tag{3.3}$$

where $\boldsymbol{\omega}$ denotes a common overall angular velocity of the graph and $(\cdot)^{\times}$ denotes the skew-symmetric cross product matrix operator in $\mathbb{R}^3$ as defined in Section 2.5. Without loss of generality, the linearly independent canonical basis vectors $\mathbf{e}_i$ of $\mathbb{R}^3$ are chosen as a basis for $\boldsymbol{\omega}$. Therefore, from (3.3), the null space of the rigidity matrix of an infinitesimally rigid

graph is

$$\text{null}\left(\mathbf{R}\right) = \text{span}\left\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\right\}, \tag{3.4}$$

where

$$\mathbf{v}_i \triangleq \begin{pmatrix} -\left(\mathbf{r}_a^{f_1 f_0}\right)^{\times} \mathbf{e}_i \\ \vdots \\ -\left(\mathbf{r}_a^{f_{n-1} f_0}\right)^{\times} \mathbf{e}_i \end{pmatrix} \in \mathbb{R}^{3(n-1)}. \tag{3.5}$$

Note that the invariance of the measurements due to common translational motion is a trivial solution when dealing with relative states, since the relative position states are also invariant to common translational motion, meaning $\dot{\mathbf{x}} = \mathbf{0}$.

A more stringent condition as compared to infinitesimal rigidity is instantaneous local observability, which requires that there is no local trajectory of the states $\mathbf{x}$ at any instant in time, excluding the trivial trajectory $\dot{\mathbf{x}} = \mathbf{0}$, that results in no change in the measurements [65, Section 6.1]. Consequently, as per (3.2), a system of n agents consisting of $n-1$ relative position states is instantaneously locally observable if $\mathbf{R}$ is full rank, that is if $\text{rank}\,\mathbf{R} = 3(n-1)$.

The aim of this section is to disambiguate the aforementioned 3 degrees of freedom corresponding to rotations of the graph as a whole, thus achieving instantaneous local observability for the relative localization problem. In the remainder of this chapter, the term *local observability* is used to refer to instantaneous local observability for conciseness.

3.3 Sufficient Condition for Local Observability

Consider a group of $n > 3$ ranging tags navigating 3-dimensional space. The $n-1$ relative position vectors $\mathbf{r}_a^{f_i f_0}$, $f_i \in \{f_1, \ldots, f_{n-1}\}$, are considered. Tag f_0 takes the role of the arbitrary reference point, and is referred to as the *reference tag*. The position between any two tags can be computed using these $n-1$ position vectors relative to the reference tag. Additionally, there is no loss of generality in assuming Tag f_0 as the reference tag, since in practice any tag can be set as the reference tag.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an infinitesimally rigid undirected graph representing the interconnection topology of the sensor network consisting of the n tags, where the edges represent the distance measurements between pairs of tags. As is, the range measurements are invariant to translations and rotations of the group of tags as a whole, as shown in Figure 3.2, while the relative position states are invariant to the translations only.

Theorem 1 *Consider an infinitesimally rigid undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ and its rigidity matrix $\mathbf{R}$, where the state vector consists of $n-1$ relative position vectors $\mathbf{r}_a^{f_i f_0}$, $\forall f_i \in \mathcal{V} \backslash \{f_0\}$,*

and the edges represent range measurements. The graph $\tilde{\mathcal{G}}(\mathcal{V},\tilde{\mathcal{E}})$ constructed from $\mathcal{G}$ with two extra edges representing the direct measurement of two linearly independent relative position vectors $\mathbf{r}_a^{f_j f_0}, \mathbf{r}_a^{f_\ell f_0} \in \mathbb{R}^3\backslash\{\mathbf{0}\}$, $f_j, f_\ell \in \mathcal{V}\backslash\{f_0\}$ corresponds to a locally observable system.

Proof: Given the infinitesimal rigidity assumption on $\mathcal{G}$, the null space of the rigidity matrix $\mathbf{R}$ is as defined in (3.4). Local observability as discussed in Section 3.2 requires that $\mathbf{R}\dot{\mathbf{x}} = \mathbf{0}$ if and only if $\dot{\mathbf{x}} = \mathbf{0}$. Therefore, new knowledge should modify $\mathbf{R}$ to generate a new rigidity matrix $\tilde{\mathbf{R}}$, such that

$$\tilde{\mathbf{R}}\boldsymbol{\zeta} \neq \mathbf{0}, \quad \forall \boldsymbol{\zeta} \in \operatorname{null}\mathbf{R}\backslash\{\mathbf{0}\}. \tag{3.6}$$

Consider the system corresponding to the graph $\tilde{\mathcal{G}}$, where two relative position vectors $\mathbf{r}_a^{f_j f_0}$ and $\mathbf{r}_a^{f_\ell f_0}$ are measured. The rigidity matrix $\tilde{\mathbf{R}}$ is then of the form

$$\tilde{\mathbf{R}} = \begin{bmatrix} \mathbf{R}_1^\mathsf{T} & \mathbf{R}^\mathsf{T} \end{bmatrix}^\mathsf{T}, \tag{3.7}$$

where $\mathbf{R}_1 \in \mathbb{R}^{6\times 3(n-1)}$ is the permutation matrix that extracts the measured relative positions $\mathbf{r}_a^{f_j f_0}$, $\mathbf{r}_a^{f_\ell f_0}$ from $\mathbf{x}$, and $\mathbf{x}$ is defined as per (3.1). That is,

$$\begin{bmatrix} \mathbf{r}_a^{f_j f_0} \\ \mathbf{r}_a^{f_\ell f_0} \end{bmatrix} = \mathbf{R}_1\mathbf{x}. \tag{3.8}$$

Given that $\mathbf{R}\boldsymbol{\zeta} = \mathbf{0}$ by the definition of $\boldsymbol{\zeta}$, it is sufficient to show that $\mathbf{R}_1\boldsymbol{\zeta} \neq \mathbf{0}$ to achieve (3.6). This can be rewritten as

$$\mathbf{R}_1 \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \end{bmatrix} \mathbf{a} \neq \mathbf{0}, \quad \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \neq \mathbf{0}, \tag{3.9}$$

where a_i represent arbitrary scalar parameters, since $\boldsymbol{\zeta}$ is a linear combination of the vectors $\mathbf{v}_i$ as shown in (3.4). By replacing the matrix

$$\mathbf{R}_1 \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \end{bmatrix} = \begin{bmatrix} -\left(\mathbf{r}_a^{f_j f_0}\right)^\times \\ -\left(\mathbf{r}_a^{f_\ell f_0}\right)^\times \end{bmatrix} \tag{3.10}$$

into (3.9) and assuming that $\mathbf{r}_a^{f_j f_0}, \mathbf{r}_a^{f_\ell f_0} \neq \mathbf{0}$, the expression in (3.9) does not hold if and only if $\left(\mathbf{r}_a^{f_j f_0}\right)^\times \mathbf{a} = \mathbf{0}$ and $\left(\mathbf{r}_a^{f_\ell f_0}\right)^\times \mathbf{a} = \mathbf{0}$, which, due to a property of the cross product, necessitates that the vectors $\mathbf{r}_a^{f_j f_0}$, $\mathbf{r}_a^{f_\ell f_0}$, and $\mathbf{a}$ be collinear. However, $\mathbf{r}_a^{f_j f_0}$ and $\mathbf{r}_a^{f_\ell f_0}$ are

linearly independent and are non-zero by assumption. Therefore, the system represented by the graph $\tilde{\mathcal{G}}$ is locally observable. ■

When a system is locally observable, all the relative position vectors are locally unique given the known measurements. Possible approaches to satisfying the minimum-knowledge requirement specified by Theorem 1 in multi-agent localization is to fit a small subset of the agents with a GPS or a stereo camera, in addition to the ranging tags. The GPS extracts the relative position information by subtracting the absolute position information, and stereo cameras can extract relative position information using attitude estimates and depth perception. To satisfy the linear independence assumption, agents fitted with a GPS must not lie in a straight line, and the relative position vectors between agents fitted with stereo cameras and the agents they detect must not be all collinear. An alternative approach that does not require additional new sensors is discussed in the next section.

3.4 Attitude-Coupled Range Measurements

3.4.1 Overview

The problem of three-dimensional multi-agent navigation using an IMU and range measurements is now discussed. In the subsequent analysis in this section, an IMU is assumed to consist of 3-axis accelerometers, gyroscopes, and magnetometers, which is commonly referred to as a 9-axis IMU. Typically, the attitude of each agent is observable using the 9-axis IMU data, as is the case when using an attitude and heading reference system (AHRS) to estimate attitude [8], but the relative positions require further measurements, such as a GPS as discussed in the end of Section 3.3. The standard way of utilizing range measurements is usually invariant to each agent's attitude, and thus having access to an agent's attitude provides no additional information regarding its instantaneous position.

This section discusses an approach to couple the range measurements with the attitude estimates to satisfy the conditions of Theorem 1. This allows the integration of an IMU with range measurements for attitude and relative position estimation of a team of robots. A key component of this approach is the use of *two-tag agents*, which are agents equipped with two non-collocated ranging tags. The notation f_i and s_i will be used to denote the *first* and *second* UWB tags of Agent i.

3.4.2 Two-Tag Agents

Rather than having one ranging tag on each agent, consider two two-tag agents as shown in Figure 3.3, while the remaining agents are single-tag agents. A single-tag agent is a

conventional agent with only one ranging tag. The range measurement between Tags f_0 and s_1 tag is then

$$y^{f_0 s_1} = \left\| \mathbf{r}_a^{10} + \mathbf{C}_{a1}\mathbf{r}_1^{s_1 1} - \mathbf{C}_{a0}\mathbf{r}_0^{f_0 0} \right\|, \tag{3.11}$$

where $\mathbf{C}_{ai} \in SO(3)$ is the direction cosine matrix (DCM) representing the rotation from the body reference frame $\mathcal{F}_i$ to the absolute frame $\mathcal{F}_a$, $\mathbf{r}_a^{10}$ represents the position of Agent 1 relative to Agent 0 as resolved in the absolute frame, $\mathbf{r}_1^{s_1 1}$ represents the known position of Tag s_1 of Agent 1 relative to the IMU of Agent 1, resolved in the Agent 1 body frame $\mathcal{F}_1$, and similarly for $\mathbf{r}_0^{f_0 0}$. This shows the consequent coupling of the range measurements and the attitude of the agents when considering two-tag agents. The subsequent corollary follows from Theorem 1, where it is assumed that for the i^{th} two-tag agent the relative position vectors $\mathbf{r}_i^{f_i i}, \mathbf{r}_i^{s_i i} \in \mathbb{R}^3 \backslash \{\mathbf{0}\}$ are known, being the position of the two tags relative to the IMU of Agent i, as resolved in the body frame $\mathcal{F}_i$. This also means that $\mathbf{r}_i^{s_i f_i} = \mathbf{r}_i^{s_i i} - \mathbf{r}_i^{f_i i}$ is known.

Corollary 1 *Consider a team of $n_t \geq 2$ two-tag agents, each with known attitude, and n_s single-tag agents. Assume that the undirected graph composed of the $2n_t + n_s$ vertices and the range measurements between the tags is infinitesimally rigid. Given that there are at least two two-tag agents j and ℓ where $\mathbf{r}_a^{s_j f_j}$ and $\mathbf{r}_a^{s_\ell f_\ell}$ are linearly independent, the underlying system representing the relative localization problem is locally observable.*

Proof: When the attitude of any two-tag agent i is known, the vector $\mathbf{r}_a^{s_i f_i}$ resembling the relative position vector between the two tags of agent i in the absolute frame $\mathcal{F}_a$ is found through

$$\mathbf{r}_a^{s_i f_i} = \mathbf{C}_{ai}\mathbf{r}_i^{s_i f_i}. \tag{3.12}$$

Consequently, since at least two linearly independent relative position vectors are known in $\mathcal{F}_a$, the conditions of Theorem 1 are satisfied. Hence, the system is locally observable. ■

Remark 1 *Whenever a team of agents equipped with IMUs includes at least two two-tag agents, the problem of finding the relative position of the agents becomes observable using just range measurements within the team. However, this also assumes that the known relative position vectors are linearly independent. Therefore, if the two two-tag agents orient themselves such that the known relative position vectors are parallel, the system becomes unobservable. A user must therefore place the tags in a strategic way to minimize the possibility of these occurrences based on the application. For example, quadcopters rarely go from level flight to a 90° pitch or roll orientation, and by placing the two-tags vertically on one agent and horizontally on the other, it is unlikely that the system becomes unobservable. This issue is also mitigated when using more than two two-tag agents, or by fitting more than two*

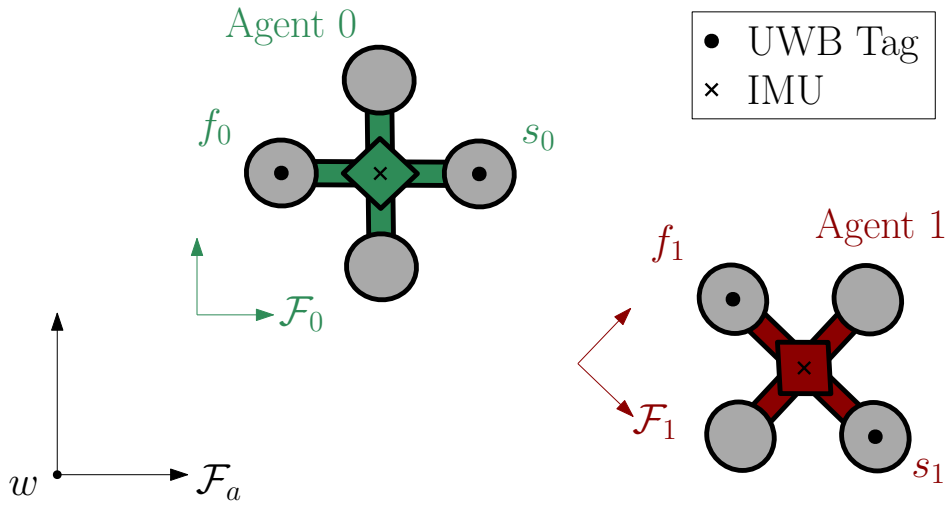


Figure 3.3: Two two-tag agents, with each agent having two ranging tags at f_0, s_0 and f_1, s_1, respectively, and an IMU at their center. The absolute frame is $\mathcal{F}_a$, and the body frame of the i^{th} agent is $\mathcal{F}_i$.

tags on an agent, which however adds hardware and congestion on the UWB communication space.

Note that the rigidity matrix only addresses whether or not ambiguities arise due to the graph being continuously deformable. Two types of ambiguities not considered by the notion of local observability are discontinuous flex ambiguities and flip ambiguities, as discussed in [66], [67]. For the two-tag agents framework, the user must be aware of the possible occurrence of such ambiguities in the presence of attitude uncertainty, as shown in Figs. 3.4 and 3.5.

3.4.3 Relative Position and Attitude Estimator

Corollary 1 requires the presence of at least two two-tag agents to achieve local observability. Therefore, any single-tag agent needs to communicate with at least two two-tag agents, and any two-tag agent needs to communicate with at least one other two-tag agent. Additionally, each two-tag agent i must compute $\hat{\mathbf{r}}_a^{s_i f_i}$ based on its attitude estimate $\hat{\mathbf{C}}_{ai}$ and the known vector $\mathbf{r}_i^{s_i f_i}$. Therefore, by satisfying these minimum ranging conditions, and with IMU measurements on two-tag agents for attitude estimation, any agent can estimate its relative position in a framework similar to the one shown in Figure 3.6. The relative position estimator can just be a simple nonlinear least squares algorithm, or a more complex filtering algorithm. In what follows, a centralized framework is considered to demonstrate the use of $n_t \geq 2$ two-tag agents and n_s single-tag agents for relative positioning, and decentralization is reserved for future work.

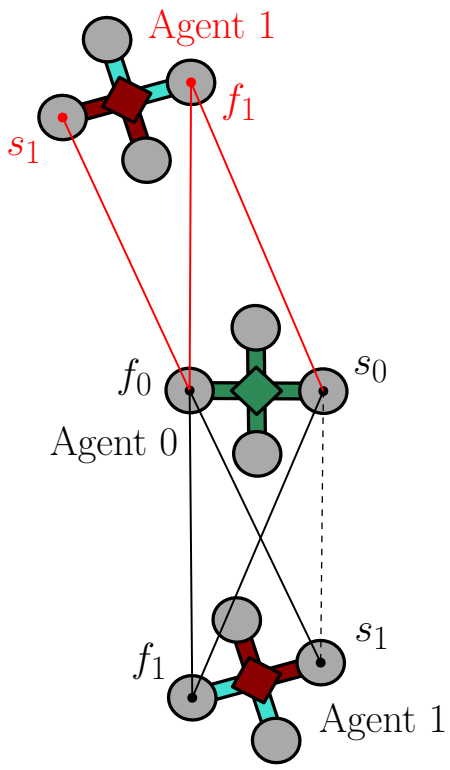


Figure 3.4: A flex ambiguity with two two-tag agents in $\mathbb{R}^2$. The flex ambiguity does not exist if the dashed edge exists, which might otherwise be assumed redundant in $\mathbb{R}^2$.

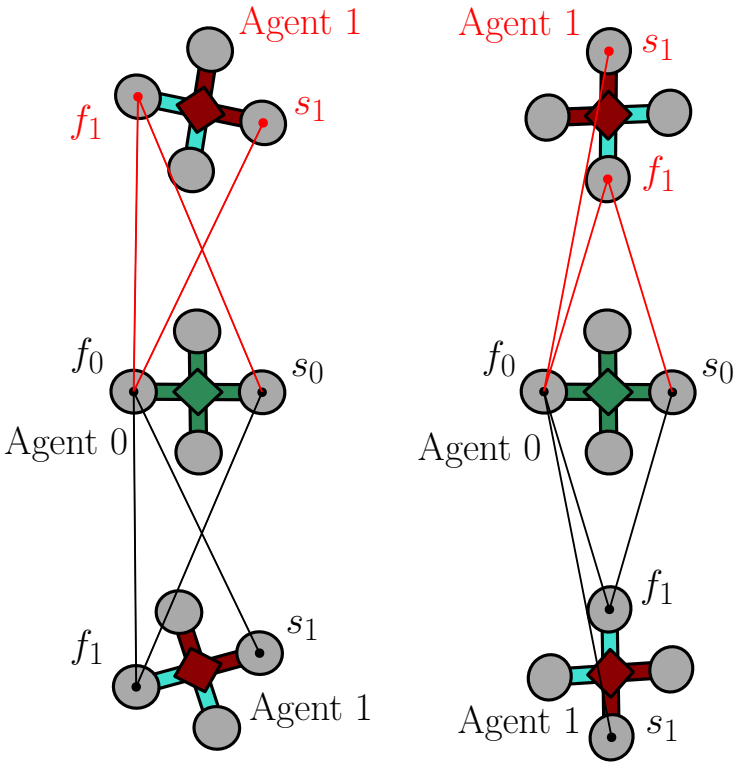


Figure 3.5: A flip ambiguity with two two-tag agents in $\mathbb{R}^2$. The more parallel the two agents are, the more likely that attitude uncertainty results in a flip ambiguity, since the attitude error resulting from the flip ambiguity is smaller.

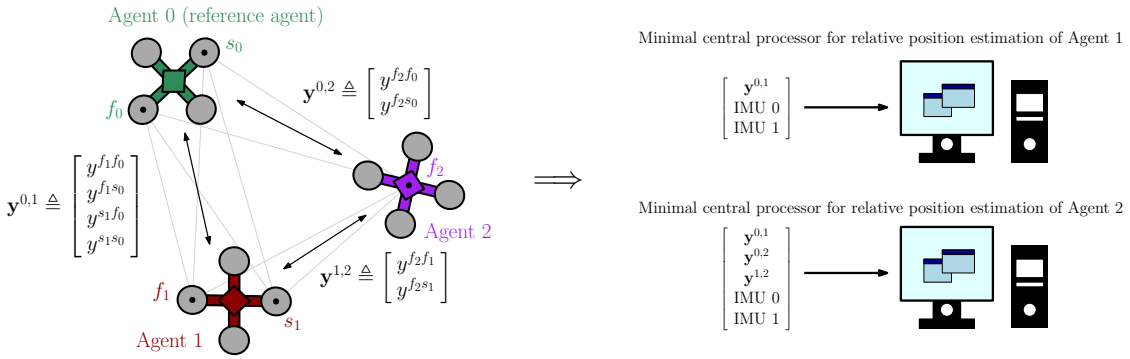


Figure 3.6: A schematic representing the minimum information required for relative position estimation of two-tag Agent 1 and single-tag Agent 2 relative to the two-tag reference agent Agent 0, in a centralized framework. The tag locations are shown using the black markers, and the light grey lines in the background represent all the range measurements between the ranging tags. The notation $\mathbf{y}^{i,j}$ is used to denote all the range measurements between Agents i and j. As per Corollary 1, the relative position vectors $r_0^{s_0 f_0}$ and $r_1^{s_1 f_1}$ are assumed to be known. In the minimal central processors, note that the IMU of agents 0 and 1 are required as the relative localization problem is dependent on estimating their attitude, while the IMU of Agent 2 is neglected as the attitude of Agent 2 is irrelevant. However, filtering approaches do require the IMU of Agent 2 as well for prediction.

The centralized relative position and attitude estimation problem for a team of $N = n_t + n_s$ agents considered herein involves estimating the state vector

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{r}_a^{10}(t) \\ \vdots \\ \mathbf{r}_a^{N-1,0}(t) \\ \mathbf{v}_a^{10/a}(t) \\ \vdots \\ \mathbf{v}_a^{N-1,0/a}(t) \\ \boldsymbol{\phi}_1(t) \\ \vdots \\ \boldsymbol{\phi}_N(t) \end{bmatrix} \in \mathbb{R}^{6(N-1)+3N}, \tag{3.13}$$

where $\mathbf{v}_a^{i0/a} \in \mathbb{R}^3$ is the velocity of the IMU of Agent i relative to the IMU of the reference agent Agent 0 with respect to $\mathcal{F}_a$, resolved in $\mathcal{F}_a$, and $\boldsymbol{\phi}_i \in \mathbb{R}^3$ is the rotation vector associated with the DCM of Agent i. Note that this approach involves the estimation of the attitude of the n_s single-tag agents as well.

Let $\mathbf{u}_i^{\text{acc}}, \mathbf{u}_i^{\text{gyr}} \in \mathbb{R}^3$ denote the accelerometer and gyroscope readings of Agent i, respectively. Starting from Section 2.5, the process model of the relative states of Agent i is then modelled as

$$ {}^a\dot{\mathbf{r}}_a^{i0}(t) = \mathbf{v}_a^{i0/a}(t), \tag{3.14}$$

$$ {}^a\dot{\mathbf{v}}_a^{i0/a}(t) = \mathbf{C}_{ai}(t)\left(\mathbf{u}_i^{\text{acc}}(t) + \mathbf{w}_i^{\text{acc}}(t)\right) \tag{3.15}$$

$$ - \mathbf{C}_{a0}(t)\left(\mathbf{u}_0^{\text{acc}}(t) + \mathbf{w}_0^{\text{acc}}(t)\right), \tag{3.16}$$

where $\mathbf{w}_i^{\text{acc}} \in \mathbb{R}^3$ denotes the white Gaussian noise associated with the accelerometer measurement of the i^{th} agent. The process model of Agent i's attitude is modelled as

$$\dot{\mathbf{C}}_{ai}(t) = \mathbf{C}_{ai}(t)\left(\mathbf{u}_i^{\text{gyr}}(t) + \mathbf{w}_i^{\text{gyr}}(t)\right)^{\times}, \tag{3.17}$$

where $\mathbf{w}_i^{\text{gyr}} \in \mathbb{R}^3$ denotes the white Gaussian noise associated with the gyroscope measurement of the i^{th} agent, and as before, the $(\cdot)^{\times}$ denotes the skew-symmetric cross product matrix operator in $\mathbb{R}^3$.

To estimate the state vector (3.13) in a centralized framework, all agents communicate their measurements to a master agent, which could be any of the N agents, along with noisy range measurements between the i^{th} tag of Agent k and the j^{th} tag of Agent ℓ, which for

Table 3.1: Simulation parameters used in the Monte Carlo trials.

Specification	Value
Accelerometer std. dev. (m/s^2)	0.026
Gyroscope std. dev. (rad/s)	0.0025
Magnetometer std. dev. (μF)	0.85
UWB std. dev. (m)	0.1
IMU rate (Hz)	100
UWB rate (Hz)	20
No. of UWB freq. channels	3
Initial relative position std dev. (m)	0.45
Initial relative velocity std dev. (m/s)	0.45
Initial attitude std dev. (rad)	0.1

example for Tags f_1 and s_2 would be of the form

$$y^{s_2 f_1} = \left\| \mathbf{r}_a^{21} + \mathbf{r}_a^{s_2 2} - \mathbf{r}_a^{f_1 1} \right\| + \nu^{s_2 f_1}, \tag{3.18}$$

where $\nu^{s_2 f_1} \in \mathbb{R}$ represents the white Gaussian noise associated with the range measurement $y^{s_2 f_1}$. In addition to the range measurements, accelerometer aiding [8] and magnetometer measurements are implemented to correct attitude drift.

The process models (3.14)-(3.17) are discretized using a forward Euler discretization scheme, and the process models and measurement models are linearized using a first-order Taylor series approximation. A centralized multiplicative extended Kalman filter (MEKF) in the spirit of [48] is then designed and evaluated in simulations in Section 3.5, and in experiments in Section 3.6.

3.5 Simulation Results

Consider 6 fully-connected aerial robots equipped with an IMU and ultra-wideband (UWB) ranging tags, where Agents 0, 1, and 2 are two-tag agents, and Agents 3, 4, and 5 are single-tag agents. Let the 3 known relative tag positions be

$$\mathbf{r}_0^{s_0 f_0} = \begin{bmatrix} 0.3 \\ 0 \\ 0 \end{bmatrix}, \mathbf{r}_1^{s_1 f_1} = \begin{bmatrix} 0 \\ 0.3 \\ 0 \end{bmatrix}, \mathbf{r}_2^{s_2 f_2} = \begin{bmatrix} 0 \\ 0 \\ 0.3 \end{bmatrix},$$

where all values are given in metres. Additionally, let Agent 0 be the elected *reference agent*, where a reference agent is specified similarly to a reference tag in Section 3.3. In this section,

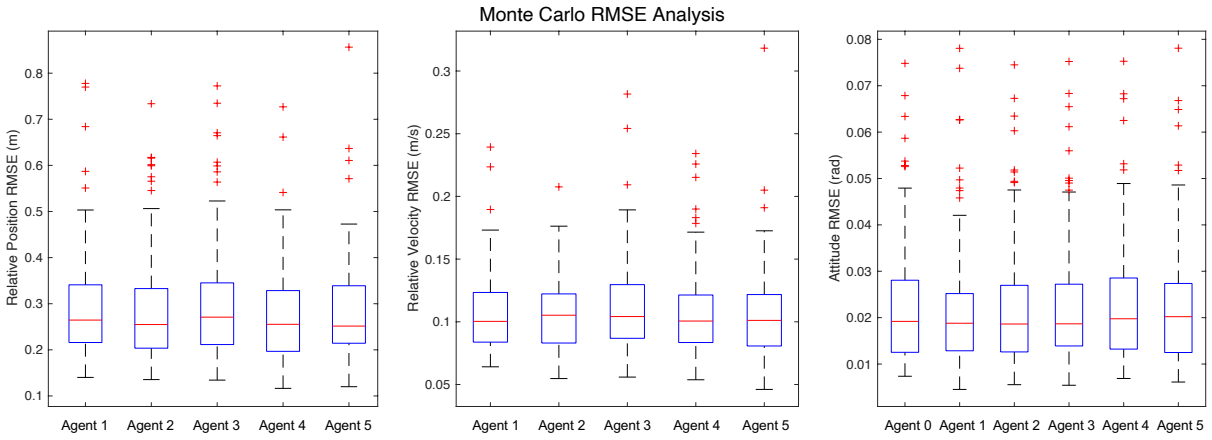


Figure 3.7: A box plot for the RMSE on 100 Monte Carlo trials. Although there are 3 to 8 outliers in the relative position estimates of the agents, the estimator still achieves an RMSE below 1 m accuracy for all runs, below 0.35 m/s for the velocity estimates, and below 0.08 rad for the attitude estimates.

the developed framework is evaluated by fusing the range measurements with an IMU using an MEKF to find the position of agents 1-5 relative to the reference agent as they move in 3-dimensional space. The centralized state estimator discussed in Section 3.4.3 is assumed to be on Agent 0. The simulation parameters are given in Table 3.1.

To assess the performance of the MEKF with the two-tag framework, 100 Monte Carlo trials with different initial conditions and noise realizations are performed, and the corresponding root-mean-squared-error (RMSE) on the relative position, relative velocity, and attitude states are shown in Figure 3.7. When considering all 100 runs, an average RMSE of 0.2887 m, 0.1080 m/s, and 1.306° for the position, velocity, and attitude states respectively are achieved. The normalized estimation error squared (NEES) test from Section 2.4.1 is performed as shown in Figure 3.8 to verify the consistency of the estimator.

3.6 Experimental Results

Experimental data is collected for a set-up with two two-tag agents and a single-tag agent, where a fully-connected graph is assumed. The prototypes of a single-tag agent and a two-tag agent are shown in Figure 3.9. The two two-tag agents are set to be Agents 0 and 1, and Agent 0 is set to be the reference agent, with

$$\mathbf{r}_0^{s_0 f_0} = \begin{bmatrix} -0.0067 \\ 0.3172 \\ -0.0185 \end{bmatrix}, \quad \mathbf{r}_1^{s_1 f_1} = \begin{bmatrix} 0.0043 \\ 0.3213 \\ 0.0224 \end{bmatrix},$$

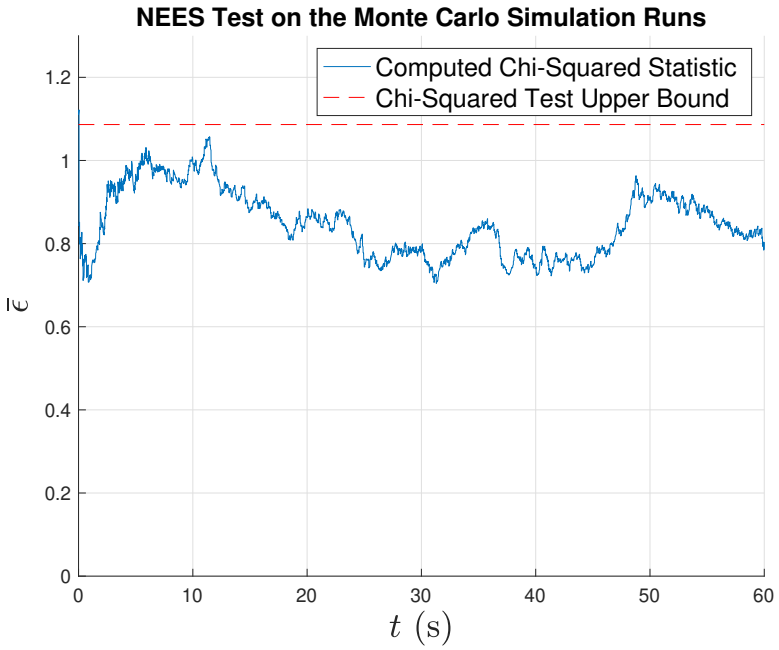


Figure 3.8: A plot representing the NEES test. The computed chi-squared statistic is below the upper bound, indicating the estimator is consistent.

where all the values are in metres. The IMU data is collected at 240 Hz using a Raspberry Pi Sense HAT device, and Pozyx UWB Developer Tags discussed in Appendix A.1.1 are used for ranging. Only one frequency channel is used at a communication rate of 16 Hz; therefore, each range measurement is collected at a frequency of only 2 Hz. Additionally, ground truth position and attitude measurements are collected at 120 Hz using an OptiTrack optical motion capture system.

The data is collected by moving all three hand-held agents indoors in random 3-dimensional rotational and translational motion, in a volume of approximately 5 m $\times$ 5 m $\times$ 2 m. The magnetometers are affected both by perturbations from the surroundings and the other agents, making estimation, especially attitude estimation, more difficult. Despite that, and with such a low ranging frequency and an inexpensive IMU, a relative position RMSE of 0.4890 m is achieved for Agent 1, and 0.42813 m for Agent 2, with the error and $\pm 3\sigma$ confidence bounds plotted in Figure 3.10. This asserts the potential of the two-tag framework on indoor self-localization without the need for expensive hardware or computationally expensive algorithms, such as visual odometry.

On flying quadcopters, vibrations affecting the IMU readings might result in worse relative position estimates. Additionally, a larger volume might degrade the performance of the algorithm as the measurements to the two tags from another agent become less geometrically distinct. However, by implementing more than just 3 agents and/or by increasing

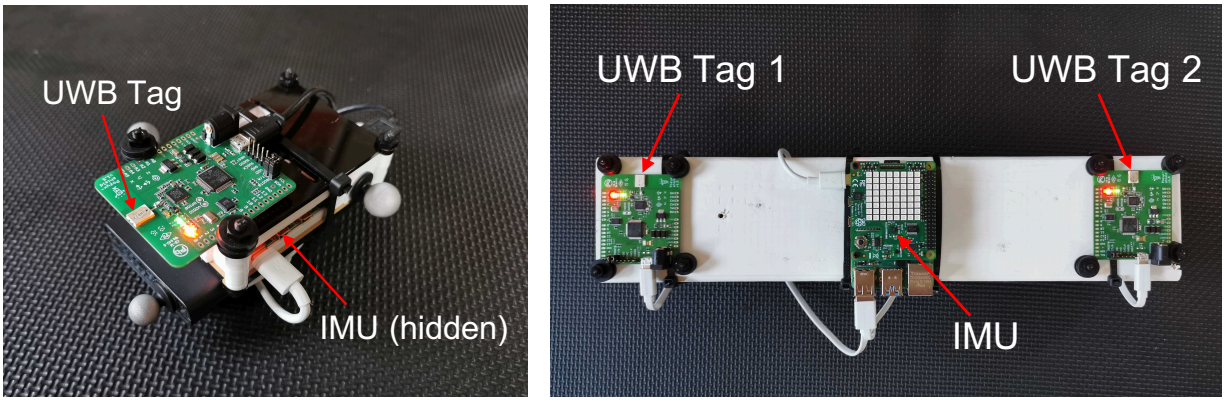


Figure 3.9: The experimental set-up, showing a single-tag agent (left) and a two-tag agent (right).

the distance between the two tags of the two-tag agents, the performance of the estimator improves and might compensate for worse attitude estimates or larger distance between the agents.

Figure 3.10: The performance of the relative position estimator on experimental data for both the two-tag Agent 1 and the single-tag Agent 2, where the black lines represent the $\pm 3\sigma$ bound of the estimator.

3.7 Conclusion and Future Work

In this chapter, the problem of three-dimensional relative position estimation using range measurements is addressed. The first step involves deriving a sufficient condition such that the relative position states of the agents are instantaneously locally observable. Thereafter, a framework utilizing two-tag agents is developed, which exploits attitude information to satisfy the sufficient conditions for observability. Lastly, this framework is integrated with

an IMU using an MEKF and is tested in simulation and in experiments. The results show that around 40-50 cm relative positioning accuracy is achievable, using just an inexpensive IMU and range measurements between three agents.

Having addressed the problem of observability when estimating relative positions from range measurements in the absence of anchors, a few other issues remain. These are summarized as follows.

- **Performance:** The performance of the estimator can be improved by improving the IMU odometry and reducing the error in the range measurements. Currently the EKF fuses the IMU measurements, which drift significantly over a short period of time, and the UWB measurements, which are noisy and incur environment-dependent biases. Additionally, the ranging frequency is low, further degrading the performance of the estimator.

- **Magnetometer:** The use of a magnetometer is necessary in this framework to correct attitude drift. However, the magnetometer is affected by the surrounding environment as the presence of metallic objects in the vicinity of the robot affects the magnetometer readings. Therefore, the magnetometer readings are not reliable and at times degrade the performance of the estimator.

- **Scalability:** Due to the use of UWB, the number of agents that can be used in this framework is limited. This is because the UWB communication space is limited as only one pair of tags can communicate at a time, and the more agents that are added, the more congested the communication space becomes. This results in any pair of tags communicating less often as the number of tags increases.

- **Decentralization:** This chapter assumes a centralized framework, where all range measurements and all IMU measurements are available simultaneously to a central processor at every time-step. However, typically in multi-agent systems, the agents are not connected to a central processor, and the computers on-board the agents themselves are responsible for running the state-estimation algorithms. Assuming each agent having access to all the agents' sensor information is impractical due to communication constraints. To develop a practical and real-world implementable framework, the communication constraints must be addressed.

- **On-manifold:** The state estimator developed in this chapter parametrizes the state as an element of $\mathbb{R}^3 \times \mathbb{R}^3 \times SO(3)$ rather than as an element of $SE_2(3)$, which does not utilize the nice mathematical properties of on-manifold state representation discussed in Section 2.3.

- **Discontinuous ambiguities:** The presented rigidity-based observability analysis in this chapter only addresses whether or not ambiguities arise due to the graph being continuously deformable. Discontinuous ambiguities such as flex and flip ambiguities as discussed in Section 3.4.2 are not considered. In an EKF-based framework, these discontinuous ambiguities can lead to the convergence of the filter to the wrong state if the estimated state approaches the mirrored realization.

The rest of this thesis will constantly refer to these problems as the focus is now on addressing these problems to provide a practical UWB-based relative-pose estimation solution.

Chapter 4

Calibration and Uncertainty Characterization for Ultra-Wideband Two-Way-Ranging Measurements

Summary

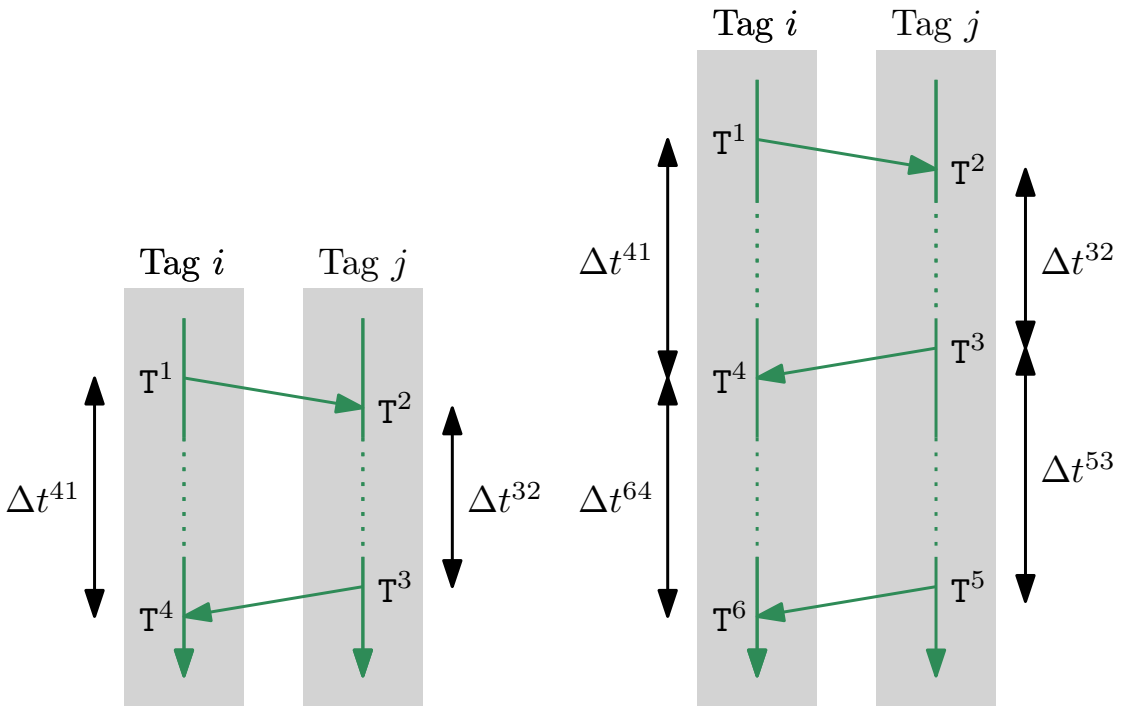
This chapter addresses the problem of performance from Section 3.7 by improving the accuracy and uncertainty characterization of the UWB range measurements to be utilized in a filtering framework. Range measurements typically suffer from a systematic error or bias that must be corrected to achieve high-accuracy localization. In this chapter, a ranging protocol is proposed alongside a robust and scalable antenna-delay calibration procedure to accurately and efficiently calibrate antenna delays for many UWB tags. Additionally, the bias and uncertainty of the measurements are modelled as a function of the received-signal power. The full calibration procedure is presented using experimental training data of 3 aerial robots fitted with 2 UWB tags each, and then evaluated on 2 test experiments. The Pozyx UWB tags from the experiments in Chapter 3 are replaced with the custom UWB boards discussed in Appendix A to allow the implementation of the proposed ranging protocol and extract the necessary information for the calibration procedure. A localization problem is then formulated on the experimental test data, and the calibrated measurements and their modelled uncertainty are fed into an extended Kalman filter (EKF). The proposed calibration is shown to yield an average of 46% improvement in localization accuracy. Lastly, this chapter is accompanied by an open-source UWB-calibration Python library, which can be found at https://github.com/decargroup/uwb_calibration.

4.1 Introduction

UWB-based ranging typically relies on measuring the ToF of radio signals from one tag to another. This requires estimating the offset between the clocks on each tag. Furthermore, the clocks often run at different rates due to physical imperfections in the individual clock's crystal oscillator, causing the offset to be time-varying. The rate of change of the clock offset is referred to as the clock *skew*. In order to negate the effect of the clock offset during ranging, different ranging protocols have been proposed, with the choice being dependent on the specific application and availability of tags [68], [15, Section 7.1.4]. As discussed in Section 2.2, a commonly used protocol is *two-way ranging* (TWR), which relies on averaging out the measured ToF between two signals to negate the clock offset. This form of TWR is referred to as *single-sided* TWR (SS-TWR), and is shown in Figure 4.1a.

Nonetheless, even after correcting for clock offsets, UWB range measurements typically suffer from a systematic error or *bias*. A significant contributor to this error is the skew between the clocks of the two ranging tags, as the different tags measure the passage of time in different units [21], [30]. This additional bias can be corrected by estimating the clock skew between the tags and embedding a skew-dependent correction factor when computing the range measurement, as proposed in [30]. However, this necessitates estimating the clock skew between all tags involved in ranging. Alternatively, [21] proposes a form of computing the range measurement utilizing *double-sided* TWR (DS-TWR), which is shown to mitigate clock-skew-dependent bias.

Another source of ranging bias stems from relative-pose-dependent antenna radiation pattern [24], where pose refers to both position and attitude. The varying signal strength can cause timestamping errors, and this effect is typically addressed using data-driven models. In [69], a simple experiment with pre-localized fixed tags or *anchors* is used to determine a relation between bias and the distance between ranging tags, while in [26], data-driven models are trained using the distance between the tags and 7 features extracted from the channel impulse response (CIR). In [40] and [70], a robot is flown around in a room with UWB anchors to learn a model of the range bias as a function of the robot's pose. The main drawback of these methods is that the learned model is dependent on the relative poses of the ranging tags, which are typically unknown in real-time without the bias-corrected measurements in the first place. Additionally, the learned models are trained and tested on the same anchor formations and are therefore not necessarily generalizable; calibration must occur for every new anchor formation. In [30], the former issue is addressed by finding a relation between the bias and the received *first-path power* (FPP) in *line-of-sight* (LOS) conditions with 2D motion.



(a) SS-TWR. (b) Proposed DS-TWR.

Figure 4.1: Timeline schematics for two tags i and j representing the different TWR ranging protocols, where T^ℓ represents the ℓ^{th} timestamp for a TWR transaction and $\Delta t^{\ell k} \triangleq \mathrm{T}^\ell - \mathrm{T}^k$.

Delays in communication between the embedded microchip and the UWB antenna are another source of bias [23]. This antenna delay is roughly the same for different UWB tags with the same physical design and is at least a few hundreds of nanoseconds [23], but can vary tenths of a nanosecond or more from tag to tag due to manufacturing inaccuracies. Given that a one-nanosecond timestamping error corresponds to 30 cm in ranging error, the need to perform antenna-delay calibration for every tag is critical. In [23], a basic TWR-based calibration procedure is suggested for calibrating antenna delays. However, the lack of motion introduces a risk of learning the aforementioned relative-pose-dependent bias as antenna delays. In [30], experiments involving a pair of tags at a time ranging with each other is used to fit what is referred to as a "pair-dependent constant". Therefore, the calibration procedure involves calibrating the relative delay between one pair at a time, which does not scale well to systems with many UWB tags.

This chapter addresses the problem of calibrating UWB tags, and the main contributions are as follows.

- An alternative DS-TWR protocol is proposed and is shown to mitigate the clock-skew-induced bias.

- A scalable antenna-delay calibration algorithm is presented that is robust to outliers and pose-dependent bias.
- The bias-versus-FPP fit presented in [30] is extended to also address the uncertainty of the measurements as a function of FPP, and DS-TWR is utilized to overcome the need to estimate the clock skew.
- The proposed antenna-delay and bias-FPP calibration are evaluated on an aerial experiment with no anchors, where all the tags are fitted to moving robots.
- The code for the full calibration procedure is attached to this chapter as an open-access online repository, which can be found at https://github.com/decargroup/uwb_calibration.

The remainder of this chapter is organized as follows. The proposed DS-TWR is discussed in Section 4.2, alongside a theoretical analysis of the clock-skew-dependent bias. In Section 4.3, a robust antenna-delay calibration algorithm is presented, followed by the bias and uncertainty calibration as a function of FPP in Section 4.4. The calibration methods presented in Sections 4.3 and 4.4 are introduced on the same experimental training data, and are then evaluated on 2 testing experiments in Section 4.5.

4.2 The Ranging Protocol

UWB ranging relies on the ToF of signals between two tags in order to compute range measurements. The simplest way to do this is using SS-TWR, shown in Figure 4.1a, where the ToF measurement can be computed as

$$t_{\mathrm{f}} = \frac{1}{2}(\Delta t^{41} - \Delta t^{32}). \tag{4.1}$$

However, different UWB tags have different clocks that are typically running at different rates, and this clock skew results in additional bias in the computed ToF measurement. In [21], an alternative DS-TWR-based ranging protocol is proposed to mitigate clock-skew-dependent bias. In this paper, the DS-TWR protocol shown in Figure 4.1b is proposed, which differs from [21] by having the responding tag instead of the initiating tag transmit the third signal. The ToF measurement can then be computed as

$$t_{\mathrm{f}} = \frac{1}{2}\left(\Delta t^{41} - \frac{\Delta t^{64}}{\Delta t^{53}}\Delta t^{32}\right). \tag{4.2}$$

This protocol is motivated by the intuitive understanding that the additional correcting factor in (4.2) transforms Δt^{32} from time units of the receiver tag's clock to time units of the

initiator tag's clock. Additionally, the proposed ranging protocol allows the initiating tag to process the range measurement by computing (4.2), without requiring additional signals for the responding tag to send Δt^{32} and Δt^{53}. Lastly, the proposed ranging protocol will be shown to be equivalent to the Cramer-Rao lower bound (CRLB) for the ToF estimation problem in Section 5.3.

4.2.1 Analytical Bias Model

To demonstrate clock-skew-dependent bias, consider in SS-TWR the clock-skew-corrupted ToF measurement,

$$\tilde{t}_{\mathrm{f}}^{\mathrm{ss}} = \frac{1}{2}\left((1+\gamma_i)(\Delta t^{41}+\eta^{41}) - (1+\gamma_j)(\Delta t^{32}+\eta^{32})\right), \tag{4.3}$$

where γ_i is the skew of Tag i's clock relative to real time, $\eta^{k\ell} = \eta^k - \eta^\ell$, and $\eta^k, \eta^\ell \sim \mathcal{N}(0, R)$ are mutually-independent timestamping white noise associated with timestamps $\mathtt{T}^k$ and $\mathtt{T}^\ell$, respectively. The ToF error is thus

$$\begin{aligned} e^{\mathrm{ss}} &\triangleq \tilde{t}_{\mathrm{f}}^{\mathrm{ss}} - t_{\mathrm{f}} \\ &= \frac{1}{2}\left(\gamma_i \Delta t^{41} + (1+\gamma_i)\eta^{41} - \gamma_j \Delta t^{32} - (1+\gamma_j)\eta^{32}\right), \end{aligned} \tag{4.4}$$

and the expected value of e^{ss} is

$$\begin{aligned} \mathbb{E}\left[e^{\mathrm{ss}}\right] &= \frac{1}{2}\left(\gamma_i \Delta t^{41} - \gamma_j \Delta t^{32}\right) \\ &\overset{(4.1)}{=} \frac{1}{2}\left(\gamma_i (2t_{\mathrm{f}} + \Delta t^{32}) - \gamma_j \Delta t^{32}\right) \\ &= \gamma_i t_{\mathrm{f}} + \frac{1}{2}\left(\gamma_i - \gamma_j\right)\Delta t^{32}. \end{aligned} \tag{4.5}$$

The first component of (4.5) is negligible as skew is in the order of tens of parts-per-million and ToF in the order of nanoseconds. However, Δt^{32} is typically in the order of hundreds of microseconds, meaning that clock-skew-dependent bias is not negligible.

Negating the second component of (4.5) is the motivation behind the proposed ranging protocol. Rewriting (4.2) as

$$t_{\mathrm{f}} = \frac{\Delta t^{41}\Delta t^{53} - \Delta t^{64}\Delta t^{32}}{2\Delta t^{53}},$$

and following the same steps as in (4.3)-(4.4), the ToF error for the proposed DS-TWR can

be derived to be approximately

$$\begin{aligned} e^{\mathrm{ds}} &\triangleq \tilde{t}_{\mathrm{f}}^{\mathrm{ds}} - t_{\mathrm{f}} \\ &\approx \gamma_i t_{\mathrm{f}} + \frac{1}{2}(1+\gamma_i)\left[\frac{\Delta t^{32}}{\Delta t^{53}}(\eta^{53} - \eta^{64}) + \eta^{41} - \eta^{32}\right], \end{aligned} \tag{4.6}$$

where $\tilde{t}_{\mathrm{f}}^{\mathrm{ds}}$ is the clock-skew-corrupted time-of-flight measurement. Deriving (4.6) relies on the assumptions that $\Delta t^{53} \gg \|\eta^{53}\|$, $\Delta t^{32} \gg \|\eta^{32}\|$. This is expected since the timestamping error is typically in the order of tenth of nanoseconds, and the delay intervals are in the order of hundreds of microseconds. The expected value of the error is therefore

$$\mathbb{E}\left[e^{\mathrm{ds}}\right] = \gamma_i t_{\mathrm{f}},$$

which suffers from less bias when compared to the error of the SS-TWR protocol.

4.3 Antenna-Delay Calibration

The delay between a chip timestamping transmission and the antenna actually transmitting the signal is referred to as the antenna transmission delay d^{t}, while the delay between an antenna receiving a signal and the chip timestamping reception is the antenna reception delay d^{r}. Looking back at Figure 4.1b, the measured timestamps are therefore

$$\tilde{\mathrm{T}}^k = \mathrm{T}^k + d^{\mathrm{t}}, \qquad k \in \{1,3,5\}, \tag{4.7}$$

$$\tilde{\mathrm{T}}^\ell = \mathrm{T}^\ell + d^{\mathrm{r}}, \qquad \ell \in \{2,4,6\}. \tag{4.8}$$

In this section, a scalable antenna-delay calibration procedure is presented that addresses the need for incorporating motion. In particular, a linear least-squares approach is presented utilizing DS-TWR, which is solved using robust least squares to accommodate for outliers [71, Section 5.4.2]. Both the antenna delays and the presented approach are environment independent, and therefore the antenna-delay calibration only needs to be performed once for new uncalibrated transceivers.

4.3.1 Least Squares Formulation

The goal of antenna-delay calibration is to find the best-fit delays based on some collected data. In order to perform this calibration procedure, the tags to be calibrated must be capable of DS-TWR. The effect of antenna delays on the DS-TWR ToF measurements is

shown by substituting (4.7) and (4.8) into (4.2), thus yielding

$$t_{\mathrm{f}} = \frac{1}{2}\Big(\Delta\tilde{t}^{41} + \underbrace{d_i^{\mathrm{t}} - d_i^{\mathrm{r}}}_{d_i} - \frac{\Delta\tilde{t}^{64}}{\Delta\tilde{t}^{53}}\big(\Delta\tilde{t}^{32} \underbrace{-d_j^{\mathrm{t}} + d_j^{\mathrm{r}}}_{-d_j}\big)\Big) \tag{4.9}$$

when Tag i initiates with Tag j, where d_i^{t} and d_i^{r} are the antenna transmission and reception delays of Tag i, respectively. In this case, transmission and reception delays can be combined into one delay variable d_i to be estimated for every tag, where $d_i = d_i^{\mathrm{t}} - d_i^{\mathrm{r}}$. This is sufficient for systems where only TWR is utilized, which is the focus of this thesis. When other ranging protocols are implemented such as time-difference-of-arrival (TDoA) or time-of-arrival (ToA), another antenna-delay calibration procedure is necessary to solve for d_i^{t} and d_i^{r} separately.

In the presence of n tags to be calibrated, let $\mathcal{P}$ denote the ordered set of tuples representing all ranging pairs of tags. Consequently, the antenna delays are calibrated by formulating a least-squares problem as

$$\hat{\mathbf{d}} = \arg\min_{\mathbf{d}\in\mathbb{R}^n} \sum_{(i,j)\in\mathcal{P}} \sum_{k=1}^{m_{ij}} g\left(e_k^{ij}(\mathbf{d})\right), \tag{4.10}$$

where the error e_k^{ij} is defined from (4.9) as a linear function of the delays as

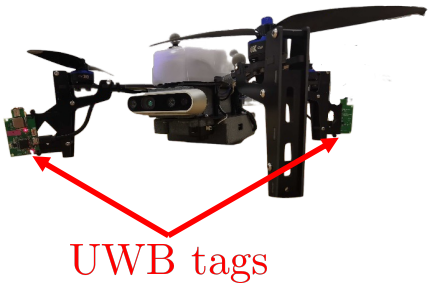
$$e_k^{ij}(\mathbf{d}) = \frac{1}{2}(d_i + K_k d_j) - t_{f,k} + \frac{1}{2}(\Delta\tilde{t}_k^{41} - K_k\Delta\tilde{t}_k^{32}),$$

m_{ij} is the number of range measurements between Tags i and j, $\mathbf{d} = \begin{bmatrix} d_1 & \cdots & d_n \end{bmatrix}^{\mathsf{T}}$, the subscript k denotes the k^{th} measurement, and $K_k \triangleq \Delta\tilde{t}_k^{64}/\Delta\tilde{t}_k^{53}$. Moreover, g is the loss function, and the choice of g is discussed in Section 4.3.2.

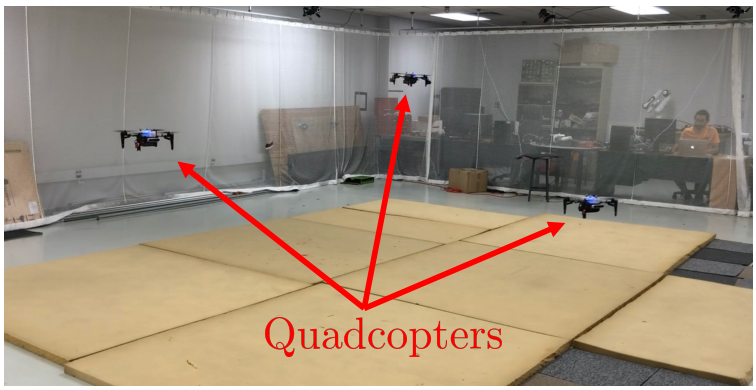
If $n = 2$, the formulated least-squares problem would have 2 unknowns and only 1 pair of ranging tags, which results in non-uniqueness of the solution. Therefore, the calibration procedure should involve at least 3 tags, yielding 3 unknowns and 3 pairs of ranging tags.

4.3.2 Experimental Results on Training Data

To evaluate the proposed antenna-delay calibration procedure experimentally, three Uvify IFO-S quadcopters are equipped with 2 UWB tags each as shown in Figure 4.2a, for a total of 6 tags. Details on the UWB modules and the quadcopters can be found in Appendix A. In order to compute the theoretical ToF t_f for any measurement to formulate a similar problem to (4.10), a motion-capture system is used to get the ground truth distances between the ranging tags. A motion-capture system is chosen for its mm-accuracy, but any other localization approach would suffice, with the accuracy of the calibration depending on the



(a) Quadcopter equipped with two UWB tags.



(b) Snapshot from training experiment.

Figure 4.2: Experimental set-up for antenna-delay and bias-FPP calibration.

accuracy of the localization algorithm. Unlike the static experiments suggested in [23], this allows a dynamic experiment where the quadcopters fly randomly in 3-dimensional space as shown in Figure 4.2b, which reduces the proneness to learning relative-pose-dependent biases as antenna delays. This dataset consists of 4 minutes of flight time and a total of 38000 range measurements. The calibration procedure can be done by fitting each drone with one tag, but two tags are used here as this set-up will be used in the remainder of this thesis as motivated in Chapter 3. The ranging schedule does not directly involve TWR measurements between any two tags on the same drone. Therefore, there are 6 unknown delays and 12 pairs of ranging tags.

Typically, problems of the form (4.10) are solved by finding $\mathbf{d}$ that minimizes the squared error (i.e., choosing g to be L2 loss), which is derived from an assumption that the underlying distribution of the noise is Gaussian. However, UWB measurements suffer from positive outliers due to multipath propagation and other sources of error, and are better modelled using Cauchy distributions [29]. It is therefore proposed for this particular application to minimize the Cauchy loss $g(x) = \log(0.5x^2 + 1)$ instead to solve (4.10) while reducing the effect of outliers [71, Section 5.4.2].

The use of the Cauchy loss is found to be useful for mitigating the effect of positive outliers as the mode of the bias distribution becomes 0, as shown in Figure 4.3. Moreover, the precision of the proposed approach is evaluated by comparing antenna-delay solutions derived from 2 datasets collected one month apart, and the delays converge to the same result within 0.03 ns of accuracy, which corresponds to less than 1 cm of ranging error.

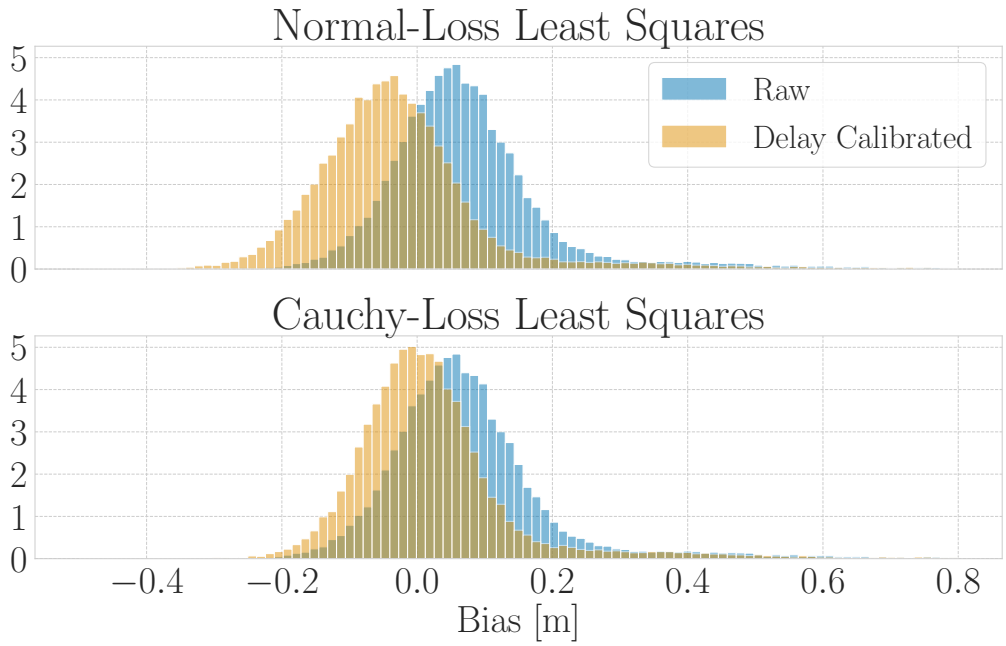


Figure 4.3: Histograms showing the effect on the ranging bias post-antenna-delay calibration. (Top) Using L2 loss. (Bottom) Using Cauchy loss.

4.3.3 Using Calibrated Tags to Calibrate New Tags

In [30], the sum of the antenna delays for every pair of tags (i, j) is calibrated as one constant c_{ij}, which for the DS-TWR protocol would be of the form $c_{ij} = d_i + Kd_j$. This approach requires that calibration is done for every pair of tags, which is tedious and not scalable. That is due to the fact that lumping the delay terms into one constant fails to utilize the constant delay terms that appear in different pairs.

By solving for the aggregate antenna delays d_i and d_j individually, this allows calibrating a new tag without collecting data between the new tag and all previously calibrated tags. In order to calibrate a new tag Tag j, only one calibrated tag Tag i is required, which then allows solving for d_j from (4.9) as

$$\hat{d}_j = \frac{2t_{\mathrm{f}} - \Delta\tilde{t}^{41} - \hat{d}_i}{K} + \Delta\tilde{t}^{32}.$$

However, it is still recommended to collect more data using a dynamic experiment with the two tags in order to improve robustness to noise and pose-dependent bias.

4.4 Power-Correlated Calibration

Another source of error in UWB-based ranging is irregularities in the antenna radiation pattern and system design elements, such as PCB-induced losses. Typically, such losses introduce biases in the measurements that are pose-dependent and that are correlated with the received signal power [24], [30]. In this section, the experiments in [30] are extended in the following ways.

1. The proposed DS-TWR is used rather than SS-TWR, which overcomes the need to estimate the skew between all pairs of tags.
2. The results are shown to hold for experiments in three-dimensional space.
3. The results are shown to hold for experiments with some non-LOS measurements due to occlusions from the quadcopters' bodies.
4. The individual measurements are used in the data-fitting process rather than averaging out measurements from a discrete number of relative poses.

The last point is particularly important as it overcomes the need to remain static during data collection, which allows the calibration procedure to be a simple experiment of robots moving randomly and covering as many relative poses as possible in a relatively short period of time. Another advantage of using all the data in the calibration process is that there is no loss of variance information through averaging out similar measurements. Consequently, the relation between the variance of the measurements and the received signal power can then be analyzed.

4.4.1 Bias Calibration

The bias calibration procedure is similar to the one presented in [30]. The reception timestamp at Tag i is usually corrupted by an unknown function $\rho_i(\cdot)$ of the received FPP p^{f}; therefore, from (4.2),

$$\begin{aligned} t_{\mathrm{f}} &= \frac{1}{2}\Big(\Delta\tilde{t}^{41} + \rho_i\left(p_4^{\mathrm{f}}\right) - \frac{\Delta\tilde{t}^{64}}{\Delta\tilde{t}^{53} + \rho_i\left(p_6^{\mathrm{f}}\right) - \rho_i\left(p_4^{\mathrm{f}}\right)}\left(\Delta\tilde{t}^{64} - \rho_j\left(p_2^{\mathrm{f}}\right)\right)\Big) \\ &\overset{(a)}{\approx} \frac{1}{2}\left(\Delta\tilde{t}^{41} - \frac{\Delta\tilde{t}^{64}}{\Delta\tilde{t}^{53}}\Delta\tilde{t}^{64}\right) + \frac{1}{2}\left(\rho_i\left(p_4^{\mathrm{f}}\right) + \rho_j\left(p_2^{\mathrm{f}}\right)\right) \\ &\triangleq \frac{1}{2}\left(\Delta\tilde{t}^{41} - \frac{\Delta\tilde{t}^{64}}{\Delta\tilde{t}^{53}}\Delta\tilde{t}^{64}\right) + f\left(\Psi\left(\frac{p_4^{\mathrm{f}} + p_2^{\mathrm{f}}}{2}\right)\right), \end{aligned} \tag{4.11}$$

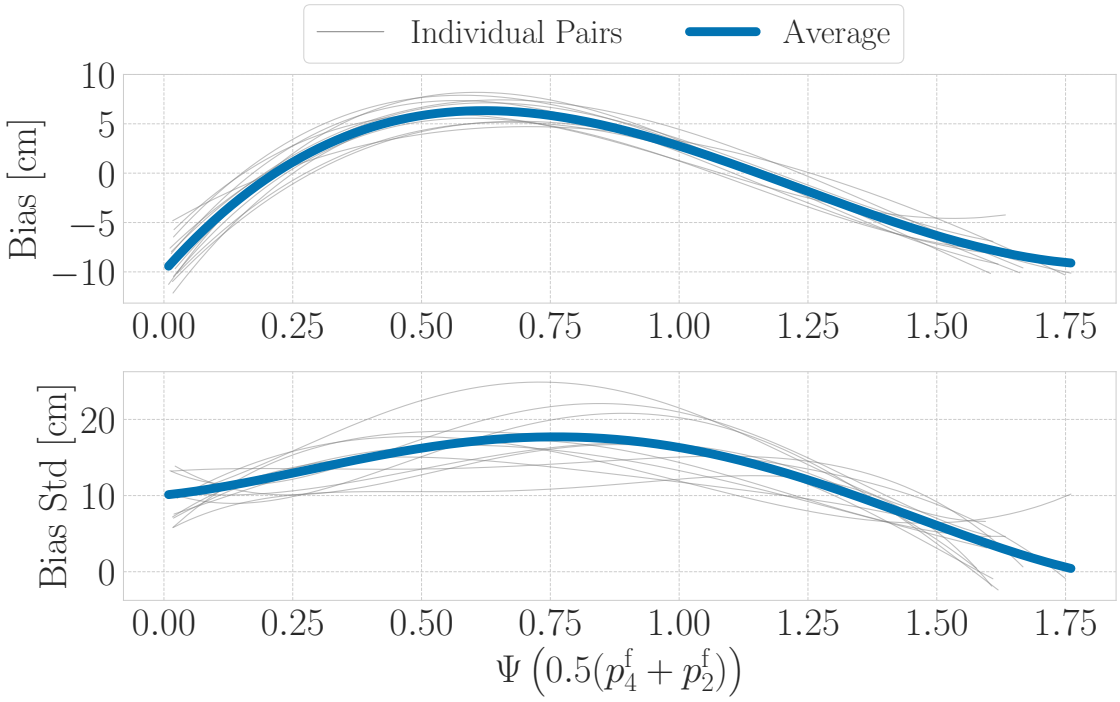


Figure 4.4: The fitted bias and standard deviation curves as a function of the lifted average FPP, using a 4-minute long training experiment.

where p_i^{f} is the FPP associated with timestamp $\mathtt{T}^i$, and $\Psi(x) \triangleq 10^{(x-\alpha)/10}$ is the lifting function suggested in [30] with α as a normalization parameter. Moreover, $f(\cdot)$ is an unknown function to be learned from data, defined based on an experimentally-motivated assumption that the effects of power-correlated bias due to the individual tags can be aggregated into one function of the average received FPP that is common to all tags of similar design. The step (a) in the derivation involves the assumptions that $\rho_i\left(p_6^{\mathrm{f}}\right) = \rho_i\left(p_4^{\mathrm{f}}\right)$ and $\frac{\Delta\tilde{t}^{64}}{\Delta\tilde{t}^{53}}\rho_j\left(p_2^{\mathrm{f}}\right) \approx \rho_j\left(p_2^{\mathrm{f}}\right)$. The former assumption is due to the fact that the motion of the robots is negligible in the time window Δt^{64} and therefore the relative-pose between the two tags is similar, while the later assumption is due to $\rho_j\left(p_2^{\mathrm{f}}\right)$ being in the order of tenths of nanoseconds, and therefore $\left(1 - \frac{\Delta\tilde{t}^{64}}{\Delta\tilde{t}^{53}}\right)\rho_j\left(p_2^{\mathrm{f}}\right) \approx 0$.

Referring back to the experiment mentioned in Section 4.3, the function f is learned by simply fitting a spline to the post-antenna-delay-calibration range bias as a function of the lifted average FPP. The result is shown at the top of Figure 4.4 for all individual pairs as well as for all the data, where the ToF bias is converted to range bias. As expected, when the antenna delay is corrected first, the bias-power curve is similar for all pairs as they all use the same antenna and PCB-board design. In this case, this additional calibration procedure can remove up to 10 cm of bias, but even though this calibration procedure is environment independent [30], this would vary for different tag designs and this process must be done separately for different antenna/board designs.

4.4.2 Variance Calibration

The variance of the measurements is also expected to vary as a function of the received-signal power, that is,

$$\mathbb{E}[(\tilde{t}_{\mathrm{f}} - t_{\mathrm{f}})^2] \triangleq \sigma\left(\Psi\left(\frac{p_4^{\mathrm{f}} + p_2^{\mathrm{f}}}{2}\right)\right)^2,$$

where σ is an unknown mapping from the lifted average FPP to the variance. Intuitively, it is expected that the receiver should be able to detect and timestamp the direct-path signal more accurately when the FPP is high as this indicates a high signal-to-noise ratio (SNR). Additionally, multipath and obstacle-attenuated signals typically have lower FPP, and therefore not a lot of variance is expected at higher received FPP. At lower received FPP, the SNR is lower, and the received measurements might have been corrupted with equally-powerful multipath and body-attenuated signals.

In order to analyze this experimentally using the training data, a similar procedure to the power-bias calibration step is proposed. The standard-deviation samples are generated by computing the standard deviation of the range bias of the measurements in a window of FPP. A spline is fitted to the standard-deviation samples, and the resulting curves are shown at the bottom of Figure 4.4. As expected, the lowest standard deviation is at the highest FPP, where the standard deviation of the range bias is as low as 2.5 cm. Additionally, the highest standard deviation of approximately 17 cm is in the mid-FPP region. This is potentially due to reflections off the ground being primarly in this region.

Even though there is a clear trend, the standard deviation curves appear to somewhat vary between different pairs. This is partly due to the training experiment being relatively short, and at some FPP values there is not enough data points to accurately compute the standard deviation. Additionally, despite the curve seeming to slope downwards at lower FPP values, it is expected that at some point as the received FPP value decreases beyond the lower detection threshold the variance will increase drastically. However, the lower detection threshold is chosen to be higher than the point where random meaningless signals would be detected, and therefore the point where the variance increases significantly does not appear in the recorded experiment.

4.5 Experimental Results on Testing Data

The proposed calibration procedure from Sections 4.3 and 4.4 is evaluated on 2 testing experiments with the same set-up as the training experiment presented in Section 4.3.2. While in the training experiment the quadcopters follow a more structured trajectory, the testing experiments involve the quadcopters flying around the 3-dimensional space randomly.

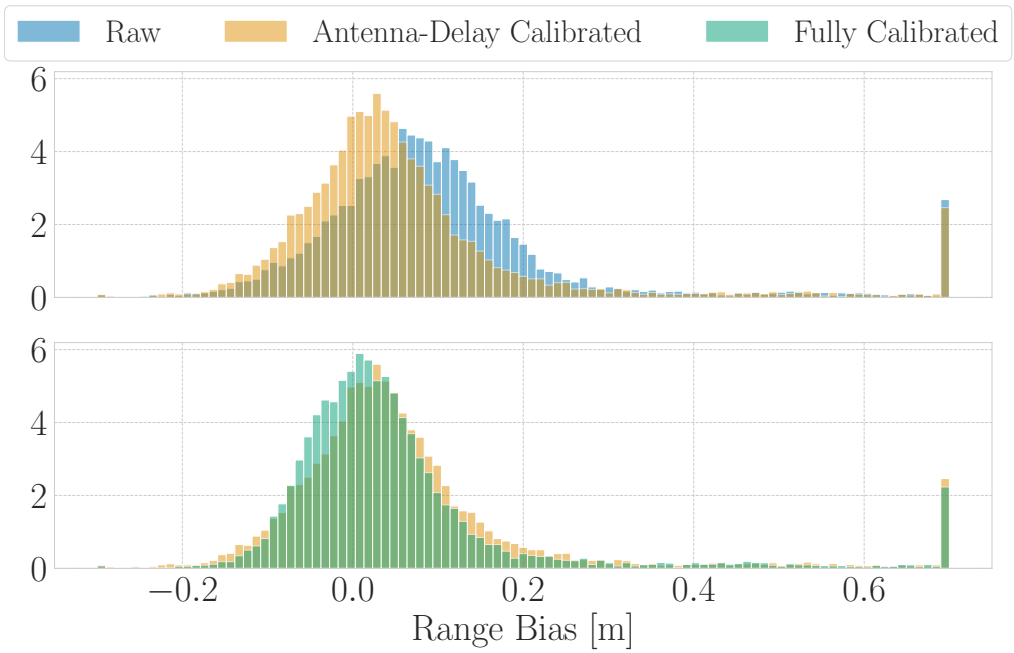


Figure 4.5: Distribution of the testing-data range bias pre- and post-calibration, with collection bins indicating the amount of outliers beyond the axes.

Each testing experiment consists of 60 seconds of flight time and 10000 range measurements between the 12 pairs.

4.5.1 Bias Correction

The distributions of the resulting biases in the testing data pre- and post-calibration are shown in Figure 4.5. The mean of the bias of the raw measurements is reduced by 36% through antenna-delay calibration only and by a further 20% by fully calibrating the measurements, bringing the mean from 11.11 cm to 5.91 cm. The standard deviation of the measurements is barely affected by antenna-delay calibration, but is reduced approximately 6% through power-correlated calibration from 18.95 cm to 17.82 cm. Both the mean and standard deviation are affected by positive outliers potentially resulting from non-LOS and multipath propagation.

In order to reject outliers, the underlying distribution must be known. Through the variance calibration procedure, the range measurements are assumed to be corrupted with zero-mean Gaussian noise with a standard deviation given by (4.12). The Gaussian-noise assumption (rather than for example Cauchy noise) is due to the nice properties of the Gaussian distribution as discussed in Section 2.1.1, which allows the use of computationally efficient filtering techniques like the KF discussed in Section 2.4. An outlier can be rejected if it does not satisfy the underlying distribution with a certain degree of confidence. For

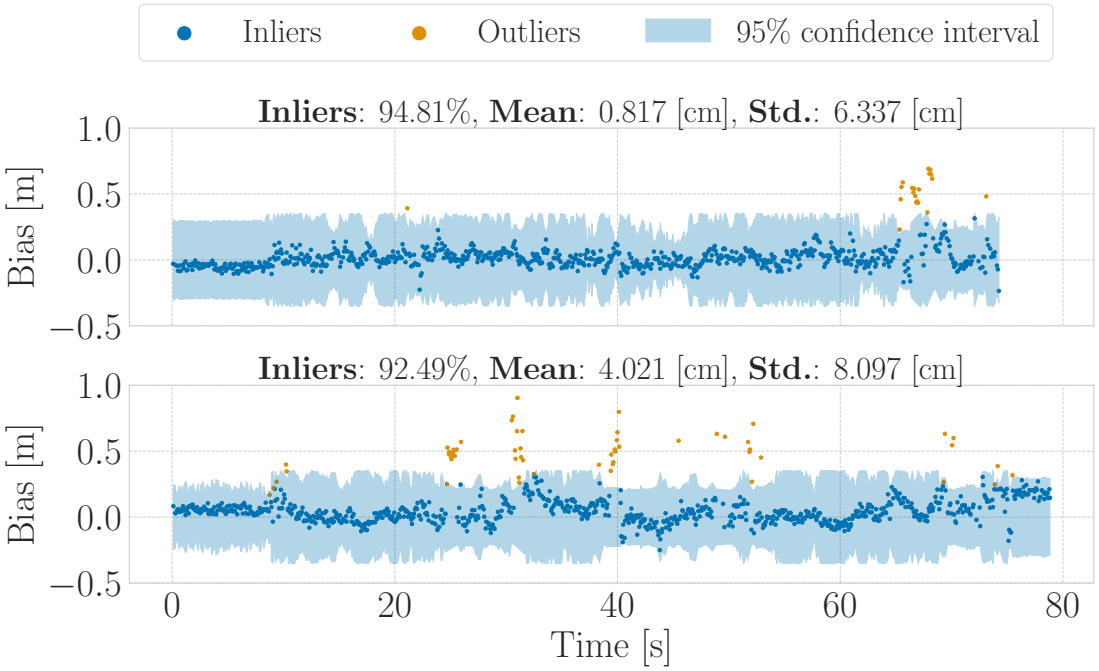


Figure 4.6: The range bias for 2 ranging pairs in one testing experiment after calibration, with some statistical information.

each individual measurement k with ground-truth-computed bias b_k and power-correlated standard deviation σ_k from (4.12), this can be done by performing the *chi-squared test* [50, Section 1.4.17]. Any measurement that does not satisfy the inequality $\frac{b_k^2}{\sigma_k^2} \leq \gamma$ indicates that it is not from the underlying distribution to a certain degree of confidence. The threshold γ depends on the chosen degree of confidence, typically 95%.

The results of this standard outlier-rejection algorithm are shown in Figure 4.6. An indication of the calibrated variance being close to the actual underlying distribution would be that exactly 5% of measurements are rejected; however, due to other factors such as non-LOS and multipath propagation, more than 5% of measurements are rejected for some pairs. Even in this non-ideal scenario with non-LOS and multipath, the mean of the measurements after outlier rejection among all pairs reduces to a maximum of roughly 4 cm and a maximum standard deviation of approximately 8 cm.

4.5.2 Position Estimator

In real-world applications, the ground truth distance between tags is usually not known but is rather estimated, and the outlier rejection method is usually done using the NIS discussed in Section 2.4.2. The NIS test is similar to the chi-squared test mentioned in Section 4.5.1, but additionally accommodates for uncertainty in the state estimates.

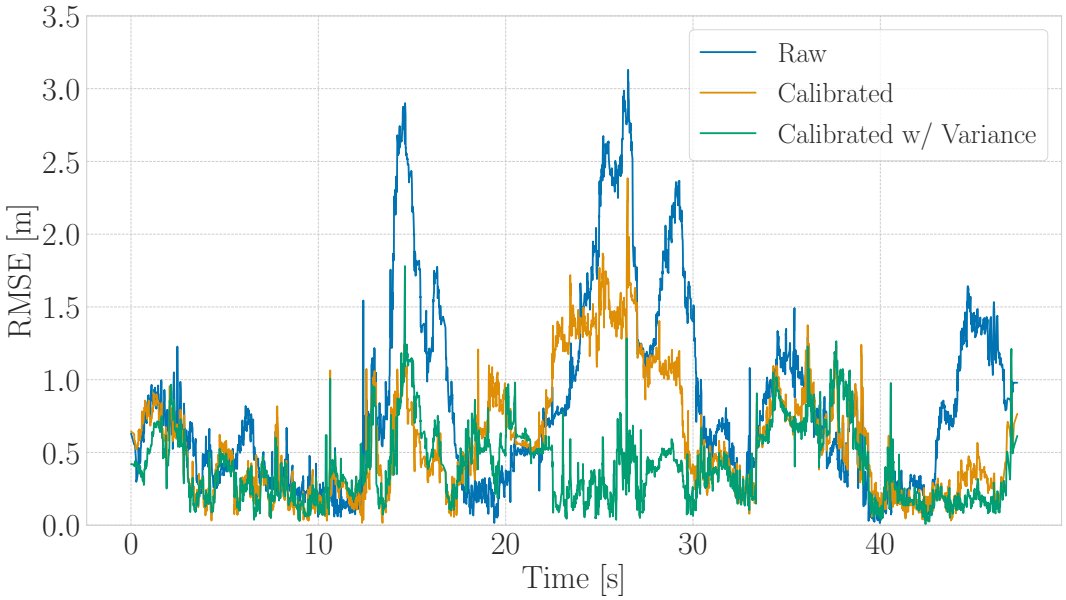


Figure 4.7: Comparison of the position-estimate RMSE using raw and calibrated data for one testing-data scenario.

To evaluate the variance calibration using the NIS test, the following simple localization problem is formulated using the testing data. Consider the problem of estimating $\mathbf{r}_a^{1w}$, the position of Robot 1 relative to some arbitrary reference point w, resolved in some inertial frame $\mathcal{F}_a$. There are two tags on the robot, and $\mathbf{r}_1^{i1}$ represents the position of Tag i relative to the robot's reference point in the robot's own body frame $\mathcal{F}_1$, which can be measured manually. Additionally, assume that the orientation of the robot given by a direction cosine matrix $\mathbf{C}_{1a} \in SO(3)$ is known, and that velocity measurements $\mathbf{v}_a^{1w}$ are available from the motion-capture system. Assuming that poses and tag positions of the neighbouring robots $n \in \{2, 3\}$ are known, an extended Kalman filter (EKF) is used to estimate $\mathbf{r}_a^{1w}$, where the measurements are modelled as

$$y = \left\| \mathbf{r}_a^{1w} + \mathbf{C}_{1a}^{\mathsf{T}} \mathbf{r}_1^{i1} - \mathbf{r}_a^{nw} - \mathbf{C}_{na}^{\mathsf{T}} \mathbf{r}_n^{jn} \right\| + \nu$$

for the range measurement between Tag i on Robot 1 and Tag j on Robot n, and $\nu \sim \mathcal{N}(0, R)$ is white Gaussian noise. The NIS test is used in the filter for outlier rejection.

The performance of the filter is shown based on the *root-mean-squared-error* (RMSE) metric in Figure 4.7 for one experimental run and summarized in Table 4.1 for 6 different scenarios using 1) the raw measurements and fixed variance, 2) the calibrated measurements and fixed variance, and 3) the calibrated measurements and the calibrated variance-power curve. The 6 scenarios represent a variation of which of the three robots is the one with an unknown position to be estimated, and doing so in two different experimental runs. The

Table 4.1: RMSE comparison for the raw and calibrated measurements in all 6 testing-data scenarios.

		RMSE [m]		
	Robot no.	**Raw**	**Calibrated**	**Calibrated w/ Variance**
Experiment 1	1	0.9245	0.5122	**0.4243**
	2	0.7577	0.5095	**0.4437**
	3	0.6731	0.3872	**0.3467**
Experiment 2	1	0.8664	**0.4879**	0.4932
	2	0.8174	0.5854	**0.3980**
	3	0.7837	0.5167	**0.4788**

choice of fixed measurement variance is decided experimentally based on what consistently yields the best performance. On average for the 6 different scenarios, the antenna-delay and bias-calibration procedures alone yield a 38% improvement in localization accuracy, while additionally utilizing the power-correlated variance calibration results in an average of 46% reduction in the RMSE, thus emphasizing the importance of calibrating UWB sensors and the added benefit of using the received FPP as an indication of the uncertainty of measurements.

4.6 Conclusion and Future Work

In this chapter, the problem of calibrating UWB bias is addressed to improve the accuracy of the range measurements, and, in turn, improving the performance of the localization algorithms. To eliminate the need for estimating the clock states, a DS-TWR-based ranging protocol is presented and shown to theoretically mitigate the effect of clock-skew-dependent bias. Furthermore, a robust and scalable antenna-delay calibration procedure is presented and trained on data from an aerial experiment. A model is then learned on the experimental data to find the relation between the remaining bias and the uncertainty of the measurements as a function of the received-signal power. The delays and models learned are then applied to two testing experiments to evaluate the calibration procedure. This framework required the replacement of the Pozyx UWB tags with the custom-built UWB tags to have the low-level access necessary to implement the proposed ranging protocol and obtain the necessary information to undertake the calibration procedure. Finally, a localization problem is then formulated using an EKF, and it is shown that an average of 46% improvement in localization accuracy can be achieved by using the corrected measurements and the modelled variance.

The calibration procedure is made open-source as a Python library, which can be found at https://github.com/decargroup/uwb_calibration. Providing this open-source code will

allow other researchers and practitioners to easily calibrate their UWB systems and improve the accuracy of their localization algorithms. This also allows for the calibration procedure to be tested on new environments, robots, and UWB tags. Future work will involve addressing how to identify non-line-of-sight (NLOS) conditions from the metrics recorded in the registers of the DWM1000 module.

Despite addressing the bias in the measurements and characterizing their uncertainty, this chapter does not address how to reduce the variance of the measurements. This turns out to be a function of Δt^{41} and Δt^{32}. These timing delays will be optimized in the next chapter in order to minimize the variance in the measurements while ensuring a sufficient ranging frequency.

Chapter 5

Reducing Two-Way Ranging Variance by Signal-Timing Optimization

Summary

The previous chapter addressed improving the accuracy of the range measurements. In doing so, a variation of the DS-TWR has been adopted due to its accuracy; however, the precision of DS-TWR has not been clearly addressed. In this chapter, an analytical model of the variance of DS-TWR is derived as a function of the user-programmed response delays, which is then compared to the *Cramer-Rao Lower Bound* (CRLB). This is then used to formulate an optimization problem over the response delays in order to maximize the information gained from range measurements. The derived analytical variance model and optimized protocol are validated experimentally with 2 ranging UWB transceivers, where 29 million range measurements are collected.

5.1 Introduction

Despite requiring an additional message transmission, the main motivation behind DS-TWR as compared to SS-TWR is to correct the clock-skew-dependent bias, which improves the accuracy of the measurements as discussed in Chapter 4. DS-TWR can also be used in the correction of other sources of error, such as the warm-up error [72]. Nonetheless, the *precision* of DS-TWR measurements as compared to SS-TWR measurements is a less commonly-addressed topic, where precision is typically measured by the variance of the range measurements. The variance of TWR measurements has been derived analytically as an approximate function of the true range [73], [74], the low-level features of the signal such as the pulse shape

[75], the surrounding environment [73], or experimentally for some fixed timing intervals [76].

The main focus of this chapter is to extend the comparison between SS-TWR and DS-TWR measurements to include the variance as a function of the timing delays in between message transmissions, as shown in Figure 4.1, which allows optimizing signal timing in DS-TWR to improve precision. Currently, the length of timing delays is arbitrarily chosen; for example, the default DS-TWR code for the commonly-used DW1000 UWB modules [7] appears to include arbitrarily-chosen timing delays without any justification. This chapter therefore presents an easily-implementable approach to setting these delays to improve the precision of the range measurements, which in the case of the DW1000 modules is as simple as changing one number in the default code.
The contributions of this chapter are as follows.

- Deriving an analytical model of the variance of SS-TWR and DS-TWR as a function of the timing of message transmissions.
- Comparing the derived DS-TWR analytical variance to the *Cramer-Rao Lower Bound* (CRLB).
- Formulating an optimization problem for DS-TWR as a function of the signal timings to maximize the information collected in one unit of time.
- Analyzing the effect of relative motion during ranging for DS-TWR.
- Validating experimentally the analytical model and the optimization procedure using static UWB transceivers.

The remainder of this chapter is organized as follows. After introducing the notation and assumptions in this paper, the analytical model of the variance and the mean squared error (MSE) of TWR measurements are derived in Section 5.2, and the former compared to the CRLB in Section 5.3. The timing-optimization problem is formulated in Section 5.4, and experimental validation is then shown in Section 5.5.

5.1.1 Notation

The notation used in this section is similar to the notation used in Section 4. The i^{th} time instance in a TWR transaction is denoted $\mathtt{T}^i \in \mathbb{R}$ as shown in Figure 4.1, and $\mathtt{T}^i_j$ denotes the i^{th} time instance as timestamped by Transceiver j. The length of time between two time instances ℓ and k is denoted $\Delta t^{k\ell} \triangleq \mathtt{T}^k - \mathtt{T}^\ell$. These can also be resolved in a Transceiver's clock, such as $\Delta t^{k\ell}_j = \mathtt{T}^k_j - \mathtt{T}^\ell_j$. The ToF between Transceivers i and j is denoted t_{f}, and

an estimate of the ToF is denoted $\hat{t}_{\mathrm{f}}$. The time-varying *clock offset* $\tau_i(t)$ of Transceiver i is defined as $\tau_i(t) \triangleq t_i(t) - t$, where $t_i(t)$ is the time t resolved in Transceiver i's clock. The *clock skew* of Transceiver i is denoted γ_i and is defined as

$$\gamma_i(t) \triangleq \dot{\tau}_i(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} \left(\tau_i(t + \Delta t) - \tau_i(t)\right).$$

5.1.2 Assumptions

It is assumed that clock skews are constant during a ranging transaction, which is a common assumption in localization applications due to the clocks' slow dynamics [21], [68]. Therefore, under a first-order approximation, the clock offset at two different time instants separated by Δt can be approximately related by

$$\tau_i(t + \Delta t) \approx \tau_i(t) + \gamma_i(t)\Delta t. \tag{5.1}$$

The IEEE 802.15.4a standard for radio-frequency systems accommodates for clock skews up to ± 20 parts-per-million (ppm) [21], [77], which is the order of the worst-case clock skew assumed in this chapter. Therefore, throughout this chapter, it is assumed that $\gamma_i \ll 1$.

Furthermore, it is assumed that $t_{\mathrm{f}} \ll \Delta t^{32}$ and $t_{\mathrm{f}} \ll \Delta t^{53}$, which is a reasonable assumption for short-range systems up to the order of tens or hundreds of meters. For example, 30 meters is equivalent to a ToF of 100 ns, while Δt^{32} and Δt^{53} are typically in the order of milliseconds or hundreds of microseconds to allow sufficient processing time in between transmitted messages. This assumption is less accurate for long-distance ranging, for example in the order of kilometres or more, which nonetheless is not common in UWB ranging.

Due to the aforementioned assumptions, approximations such as $\gamma_i \Delta t^{41} \approx \gamma_i \Delta t^{32}$ are made throughout this chapter. This follows from the term $\gamma_i t_f$ being much smaller than $\gamma_i \Delta t^{32}$, since $t_{\mathrm{f}} \ll \Delta t^{32}$ and $\gamma_i t_f$ corresponds to a value that is in the order of tens or hundreds of micrometers when multiplied by the speed of light, and can thus be neglected.

5.2 TWR Variance

5.2.1 Modelling the Timestamps

The measurement models for the timestamps recorded by Transceivers i and j in Figure 4.1 are first presented under the assumption that all transceivers are static. The noisy

timestamps recorded by Transceiver i in Figure 4.1a are modelled as

$$\mathtt{T}_i^1 = \mathtt{T}^1 + \tau_i(\mathtt{T}^1) + \eta^1, \tag{5.2}$$
$$\mathtt{T}_i^4 = \mathtt{T}^1 + 2t_{\mathrm{f}} + \Delta t^{32} + \tau_i(\mathtt{T}^4) + \eta^4, \tag{5.3}$$

where η^ℓ is random noise on the ℓ^{th} measurement. All random noise variables on timestamps are assumed to be mutually independent, zero-mean, and with the same variance σ^2.

Similarly, the noisy timestamps recorded by Transceiver j in Figure 4.1a are modelled as

$$\mathtt{T}_j^2 = \mathtt{T}^1 + t_{\mathrm{f}} + \tau_j(\mathtt{T}^2) + \eta^2, \tag{5.4}$$
$$\mathtt{T}_j^3 = \mathtt{T}^1 + t_{\mathrm{f}} + \Delta t^{32} + \tau_j(\mathtt{T}^3) + \eta^3, \tag{5.5}$$

while the additional timestamps when performing DS-TWR as in Figure 4.1b are modelled as

$$\mathtt{T}_j^5 = \mathtt{T}^1 + t_{\mathrm{f}} + \Delta t^{32} + \Delta t^{53} + \tau_j(\mathtt{T}^5) + \eta^5, \tag{5.6}$$
$$\mathtt{T}_i^6 = \mathtt{T}^1 + 2t_{\mathrm{f}} + \Delta t^{32} + \Delta t^{53} + \tau_i(\mathtt{T}^6) + \eta^6. \tag{5.7}$$

Based on the aforementioned assumptions in Section 5.1 and the relation in (5.1), the offsets in (5.2)-(5.7) can be written as a function of the clock skew, the clock offsets at $\mathtt{T}^1$, and the time delays Δt^{32}, Δt^{53}. For example,

$$\begin{aligned}\tau_i(\mathtt{T}^6) &= \tau_i(\mathtt{T}^1 + \Delta t^{61}) \\ &\approx \tau_i(\mathtt{T}^1) + \gamma_i(\mathtt{T}^1)\Delta t^{61} \\ &\approx \tau_i(\mathtt{T}^1) + \gamma_i(\mathtt{T}^1)(\Delta t^{32} + \Delta t^{53}).\end{aligned}$$

A similar process can be followed for the other offsets. The remainder of this chapter will oftentimes drop the explicit dependence on $\mathtt{T}^1$ from the notation for brevity.

5.2.2 Deriving SS-TWR Variance

A SS-TWR ToF estimate $\hat{t}_{\mathrm{f}}$ of the true ToF t_{f} can be computed from (5.2)-(5.5) as

$$\begin{aligned}\hat{t}_f^{\mathrm{ss}} &= \frac{1}{2}(\Delta t_i^{41} - \Delta t_j^{32}) \\ &= \frac{1}{2}\left(2t_{\mathrm{f}} + \Delta t^{32} + \gamma_i \Delta t^{41} + \eta^{41} - (1+\gamma_j)\Delta t^{32} - \eta^{32}\right) \\ &\approx t_{\mathrm{f}} + \frac{1}{2}\gamma_{ij}\Delta t^{32} + \frac{1}{2}(\eta^{41} - \eta^{32}),\end{aligned}$$

where $\gamma_{ij} \triangleq \gamma_i - \gamma_j$, $\eta^{k\ell} \triangleq \eta^k - \eta^\ell$, and $\gamma_i \Delta t^{41} \approx \gamma_i \Delta t^{32}$. Defining the SS-TWR ToF error as $e^{\mathrm{ss}} \triangleq \hat{t}_f^{\mathrm{ss}} - t_{\mathrm{f}}$, the expected value of the error is

$$\mathbb{E}[e^{\mathrm{ss}}] = \frac{1}{2}\gamma_{ij}\Delta t^{32}, \tag{5.8}$$

which means that $\hat{t}_f^{\mathrm{ss}}$ is in fact a biased measurement of t_{f} as is shown in Chapter 4. Meanwhile, the covariance on the measurement is

$$\mathbb{E}[(e^{\mathrm{ss}} - \mathbb{E}[e^{\mathrm{ss}}])^2] = \sigma^2. \tag{5.9}$$

5.2.3 DS-TWR Variance

The main motive behind using DS-TWR protocols rather than SS-TWR protocols is to correct the clock-skew-dependent bias in (5.8). As shown in (4.6), the DS-TWR ToF estimate from (5.2)-(5.7) can be modelled as

$$\hat{t}_f^{\mathrm{ds}} = \frac{1}{2}\left(\Delta t_i^{41} - \frac{\Delta t_i^{64}}{\Delta t_j^{53}}\Delta t_j^{32}\right) \tag{5.10}$$

$$\approx t_{\mathrm{f}} + \frac{1}{2}\left(\frac{\Delta t^{32}}{\Delta t^{53}}(\eta^{53} - \eta^{64}) + \eta^{41} - \eta^{32}\right), \tag{5.11}$$

where the approximations $\gamma_i t_{\mathrm{f}} \approx 0$ and $\gamma_i \eta \approx 0$ are used since the clock skew, time-of-flight, and timestamping noise are all small. Defining the DS-TWR ToF error as $e^{\mathrm{ds}} \triangleq \hat{t}_f^{\mathrm{ds}} - t_{\mathrm{f}}$, the expected value of the error is

$$\mathbb{E}[e^{\mathrm{ds}}] = 0,$$

meaning that unlike $\hat{t}_f^{\mathrm{ss}}$, the estimate $\hat{t}_f^{\mathrm{ds}}$ is unbiased.

Having readdressed the accuracy of the measurements for SS-TWR and DS-TWR, it might appear that DS-TWR should always be used. However, the choice of the ranging

protocol should also depend on the precision of the measurements. By manipulating (5.11), the covariance of $\hat{t}_f^{\text{ds}}$ can be found to be of the form

$$\mathbb{E}[\left(e^{\text{ds}} - \mathbb{E}[e^{\text{ds}}]\right)^2] = \sigma^2 \left(1 + \frac{\Delta t^{32}}{\Delta t^{53}} + \left(\frac{\Delta t^{32}}{\Delta t^{53}}\right)^2\right). \tag{5.12}$$

Therefore, the variance of DS-TWR measurements is greater than SS-TWR measurements, and approaches the variance of SS-TWR as $\Delta t^{32} \to 0$ and/or $\Delta t^{53} \to \infty$. The $\Delta t^{32} \to 0$ condition is due to the effect of the length of Δt^{32} on the bias, and the $\Delta t^{53} \to \infty$ condition is due to the fact that the ratio $\frac{\Delta t^{64}}{\Delta t^{53}}$ is being used to obtain a clock-skew measurement, and the longer the Δt^{53} interval is the greater the signal-to-timestamping-noise ratio.

5.2.4 Mean Squared Error of SS-TWR and DS-TWR

Knowing the mean-bias and the variance of the ToF estimates for SS-TWR and DS-TWR allows computing the mean squared error (MSE) of the estimates. The MSE of SS-TWR from (5.8) and (5.9) is

$$\begin{aligned}\mathbb{E}\left[(e^{\text{ss}})^2\right] &= \mathbb{E}\left[(e^{\text{ss}} - \mathbb{E}[e^{\text{ss}}])^2\right] + \mathbb{E}[e^{\text{ss}}]^2 \\ &= \sigma^2 + \frac{1}{4}\gamma_{ij}^2(\Delta t^{32})^2,\end{aligned} \tag{5.13}$$

and the MSE of DS-TWR is the same as (5.12) since the estimate is unbiased.

Therefore, the MSE of DS-TWR is lower than the MSE of SS-TWR when

$$\sigma^2 \left(1 + \frac{\Delta t^{32}}{\Delta t^{53}} + \left(\frac{\Delta t^{32}}{\Delta t^{53}}\right)^2\right) < \sigma^2 + \frac{1}{4}\gamma_{ij}^2(\Delta t^{32})^2,$$

which can also be written as

$$|\gamma_{ij}| > \frac{2\sigma}{\Delta t^{53}} \sqrt{\frac{\Delta t^{32} + \Delta t^{53}}{\Delta t^{32}}}. \tag{5.14}$$

The right-hand-side of (5.14) is plotted in Figure 5.1 for $\sigma = 0.0682$ ns, where the value for σ is determined experimentally in Section 5.5. Given that $|\gamma_{ij}|$ is expected to range between 0 and 40 ppm, it is very likely that the MSE of DS-TWR will be lower than that of SS-TWR, except for highly-accurate clocks with lower skew resulting in lower measurement bias.

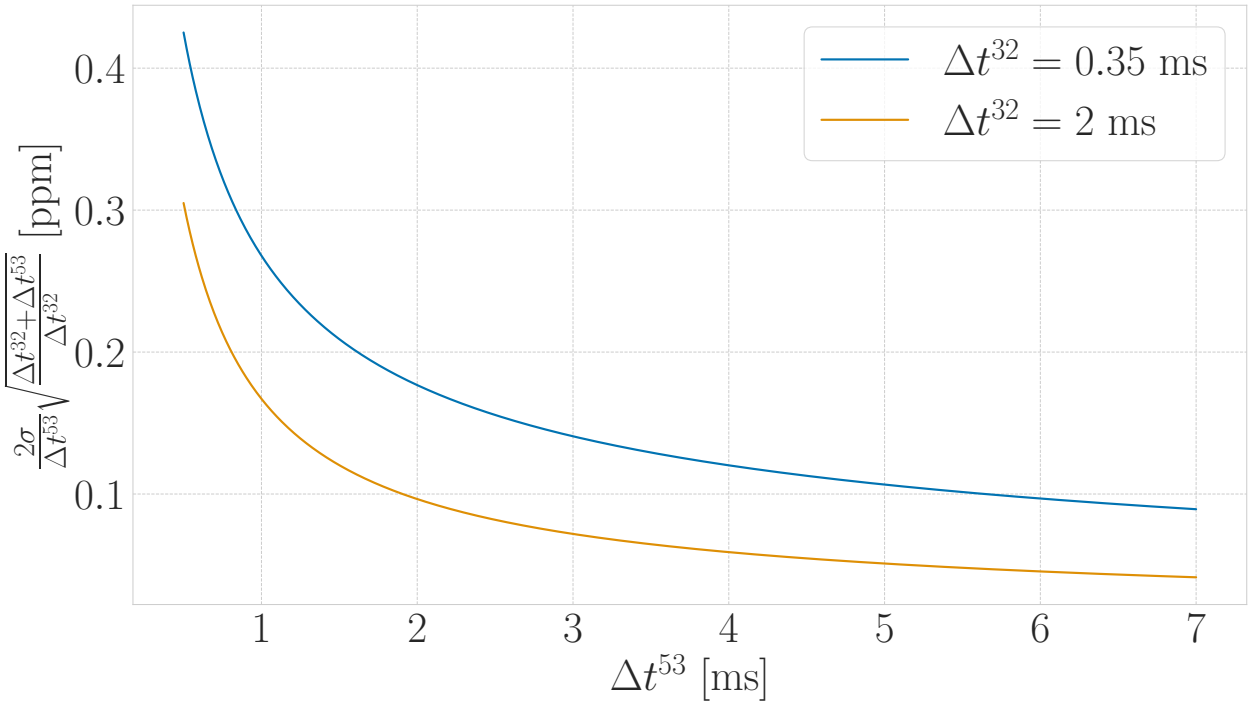


Figure 5.1: The value of the right-hand-side of (5.14) for different values of Δt^{32} and Δt^{53} when $\sigma = 0.0682$ ns. These curves represent a lower bound for the magnitude of the clock skew γ_{ij} for which the MSE of DS-TWR is lower than the MSE of SS-TWR.

5.3 Cramer-Rao Lower Bound of DS-TWR

Given that now an analytical model is available of the variance of the range estimate provided by the DS-TWR protocol (5.10), the variance of the estimator can be compared to the CRLB [78, Chapter 3].

When two transceivers are ranging with one another, time instances in global time and offsets of individual clocks remain unknown, and only the time instances in a transceiver's clocks and relative offset between the two transceivers can be estimated. Therefore, the unknown quantities to be estimated from (5.2)-(5.7) can be summarized in a state vector

$$\mathbf{x} = \begin{bmatrix} t_{\mathrm{f}} & \mathcal{T} & \tau_{ij} & \gamma_{ij} & \Delta t_j^{32} & \Delta t_j^{53} \end{bmatrix}^{\mathsf{T}},$$

where $\mathcal{T} = \mathsf{T}^1 + \tau_i(\mathsf{T}^1)$ and $\tau_{ij} \triangleq \tau_i(\mathsf{T}^1) - \tau_j(\mathsf{T}^1)$. Additionally, under the assumption that $\frac{\gamma_{ij}}{1+\gamma_j} \approx \gamma_{ij}$ since $\gamma_j \ll 1$, it can be shown that

$$\Delta t_i^{32} \approx (1+\gamma_{ij})\Delta t_j^{32},$$
$$\Delta t_i^{53} \approx (1+\gamma_{ij})\Delta t_j^{53}.$$

Therefore, the timestamp measurements (5.2)-(5.7) can be written as

$$\begin{aligned}
\mathsf{T}_i^1(\mathbf{x}) &= \mathcal{T} + \eta^1, \\
\mathsf{T}_j^2(\mathbf{x}) &= \mathcal{T} + t_\mathrm{f} - \tau_{ij} + \eta^2, \\
\mathsf{T}_j^3(\mathbf{x}) &\approx \mathcal{T} + t_\mathrm{f} - \tau_{ij} + \Delta t_j^{32} + \eta^3, \\
\mathsf{T}_i^4(\mathbf{x}) &\approx \mathcal{T} + 2t_\mathrm{f} + (1+\gamma_{ij})\Delta t_j^{32} + \eta^4, \\
\mathsf{T}_j^5(\mathbf{x}) &\approx \mathcal{T} + t_\mathrm{f} - \tau_{ij} + \Delta t_j^{32} + \Delta t_j^{53} + \eta^5, \\
\mathsf{T}_i^6(\mathbf{x}) &\approx \mathcal{T} + 2t_\mathrm{f} + (1+\gamma_{ij})(\Delta t_j^{32} + \Delta t_j^{53}) + \eta^6,
\end{aligned}$$

where the approximation $\gamma_i t_\mathrm{f} \approx 0$ has been used. The measurement vector can then be written as

$$\mathbf{y}(\mathbf{x}) = \begin{bmatrix} \mathsf{T}_i^1 & \mathsf{T}_j^2 & \mathsf{T}_j^3 & \mathsf{T}_i^4 & \mathsf{T}_j^5 & \mathsf{T}_i^6 \end{bmatrix}^\mathsf{T},$$

which is a nonlinear function of the states $\mathbf{x}$. Therefore, the measurement Jacobian can be computed as

$$\begin{aligned}
\mathbf{H} &= \left.\frac{\partial \mathbf{y}(\mathbf{x})}{\partial \mathbf{x}}\right|_{\bar{\mathbf{x}}} \\
&= \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & -1 & 0 & 0 & 0 \\
1 & 1 & -1 & 0 & 1 & 0 \\
2 & 1 & 0 & \Delta\bar{t}_j^{32} & 1+\bar{\gamma}_{ij} & 0 \\
1 & 1 & -1 & 0 & 1 & 1 \\
2 & 1 & 0 & \Delta\bar{t}_j^{32} + \Delta\bar{t}_j^{53} & 1+\bar{\gamma}_{ij} & 1+\bar{\gamma}_{ij}
\end{bmatrix},
\end{aligned}$$

where overbars denote the linearization point. Additionally, define a measurement vector covariance $\mathbf{\Sigma} \triangleq \sigma^2 \mathbf{1}_6$, where $\mathbf{1}_6$ is the 6×6 identity matrix.

The CRLB states that the covariance of any unbiased estimate $\hat{\mathbf{x}}$ of $\mathbf{x}$, given the measurements $\mathbf{y}(\mathbf{x})$ and an additive-Gaussian assumption on the measurement noise, is bounded by [78, Appendix 3C]

$$\mathbb{E}\left[(\mathbf{x}-\hat{\mathbf{x}})(\mathbf{x}-\hat{\mathbf{x}})^\mathsf{T}\right] \geq (\mathbf{H}^\mathsf{T}\mathbf{\Sigma}^{-1}\mathbf{H})^{-1}.$$

The minimum variance of the ToF estimate for the given timestamps can then be found by

extracting the first component of $(\mathbf{H}^\intercal\mathbf{\Sigma}^{-1}\mathbf{H})^{-1}$, which can be found to be

$$\sigma^2\frac{(\bar{\gamma}_{ij}^2+2\bar{\gamma}_{ij}+2)((\Delta\bar{t}_j^{32})^2+\Delta\bar{t}_j^{32}\Delta\bar{t}_j^{53}+(\Delta\bar{t}_j^{53})^2)}{2(\Delta\bar{t}_j^{53})^2}.$$

Given that $\bar{\gamma}_{ij}^2+2\bar{\gamma}_{ij}\ll 2$, this can be simplified to give exactly (5.12), thus showing that under the aforementioned approximations the DS-TWR estimator is indeed the minimum-variance unbiased estimator.

5.4 DS-TWR Timing Optimization

The *timing delays* Δt^{32} and Δt^{53} affect the variance of the range measurements, the rate of the measurements, and the ranging error due to relative motion between the transceivers. In Section 5.4.1, the choice of delays is motivated as a function of the variance and rate of the measurements while assuming no relative motion between the transceivers. This assumption is then validated in Section 5.4.2 for the DS-TWR protocol, showing that motion can indeed be neglected when choosing the timing delays.

5.4.1 Finding Optimal Timing Delays

Given (5.12), minimizing Δt^{32} within the limitations of the system is an obvious choice to reduce the measurement variance. However, it is less clear what the right choice for Δt^{53} is, as increasing this second-response delay reduces measurement variance but also reduces the rate of measurements. The choice of Δt^{53} is thus application-specific. Most commonly in estimation applications, the goal is to minimize the variance of the estimates, which is achieved by maximizing the *information* attained from measurements. Therefore, this section poses an information-maximizing (variance-minimizing) optimization problem.

The amount of information obtained in one unit of time is a function of the variance of the individual measurement and the number of measurements in that unit of time. As a result, the *optimal* delay is one that is long enough to reduce the variance of the individual measurement but short enough to ensure measurements are recorded at a sufficient rate.

The rate of the measurements is dependent on $\Delta t^{32}+\Delta t^{53}$ as well as any further processing required to retrieve the range measurements, such as reading the raw timestamps from the registers and computing the range measurement from the raw timestamps. The time taken for computational processing is defined as ρ, which is assumed to be constant for the same experimental set-up. Therefore, the time-length of one measurement is $\rho+\Delta t^{32}+\Delta t^{53}$ seconds long. The delay Δt^{32} is to be minimized as much as the hardware allows, and Δt^{53}

is to be optimized as follows. In one second, a total of $\left[\frac{1}{\rho+\Delta t^{32}+\Delta t^{53}}\right]$ measurements occur, meaning that the variance of averaging out the ToF estimates is, assuming independence, given as

$$R_{\text{avg}}(\Delta t^{53}) \triangleq [\rho + \Delta t^{32} + \Delta t^{53}] R_{\text{meas}}(\Delta t^{53}), \tag{5.15}$$

where $R_{\text{meas}}(\Delta t^{53})$ is the variance of the individual measurement given by (5.12) for some constant Δt^{32}. R_{avg} is referred to hereinafter as the *averaged uncertainty*, and can be thought of as the inverse of accumulated *information* in one second. Finding the optimal delay Δt^{53^*} is then found by solving

$$\Delta t^{53^*} = \arg \min_{\Delta t^{53} \in \mathbb{R}} R_{\text{avg}}(\Delta t^{53}). \tag{5.16}$$

The derivative of (5.15) with respect to Δt^{53} is

$$\frac{\mathrm{d}R_{\text{avg}}}{\mathrm{d}\Delta t^{53}} = \sigma^2 - (\rho + \Delta t^{32}) \frac{\Delta t^{32}}{(\Delta t^{53})^2} \sigma^2 - 2(\rho + \Delta t^{32}) \frac{(\Delta t^{32})^2}{(\Delta t^{53})^3} \sigma^2 - \frac{(\Delta t^{32})^2}{(\Delta t^{53})^2} \sigma^2,$$

and equating to 0 yields the cubic polynomial

$$0 = (\Delta t^{53})^3 - \Delta t^{32}(\rho + 2\Delta t^{32}) \Delta t^{53} - 2(\Delta t^{32})^2 (\rho + \Delta t^{32}). \tag{5.17}$$

This is a "depressed cubic equation" that can be solved analytically using Cardano's method, but the analytical solution is omitted here for conciseness. Additionally, this can be solved numerically using standard libraries (such as Bullet's cubic_roots function in C++ or NumPy's roots function in Python).

The value for T and the minimum value of Δt^{32} can be determined experimentally and are both processor and application dependent. As an example where $\rho = 7.2$ ms and $\Delta t^{32} = 0.35$ ms, the optimal delay can be found analytically to be approximately 1.9 ms using (5.17). The averaged variance R_{avg} as a function of Δt^{53} for $T = 7.2$ ms at different values of Δt^{32} is shown in Figure 5.2. As expected from (5.12), the averaged variance R_{avg} diverges as Δt^{53} approaches 0 ms.

5.4.2 Relative Motion During Ranging

A constant distance throughout ranging is commonly assumed, but this assumption introduces larger errors for longer response delays. To address this, assume the less-restrictive case of no relative acceleration between the transceivers. In this case, the three ToF mea-

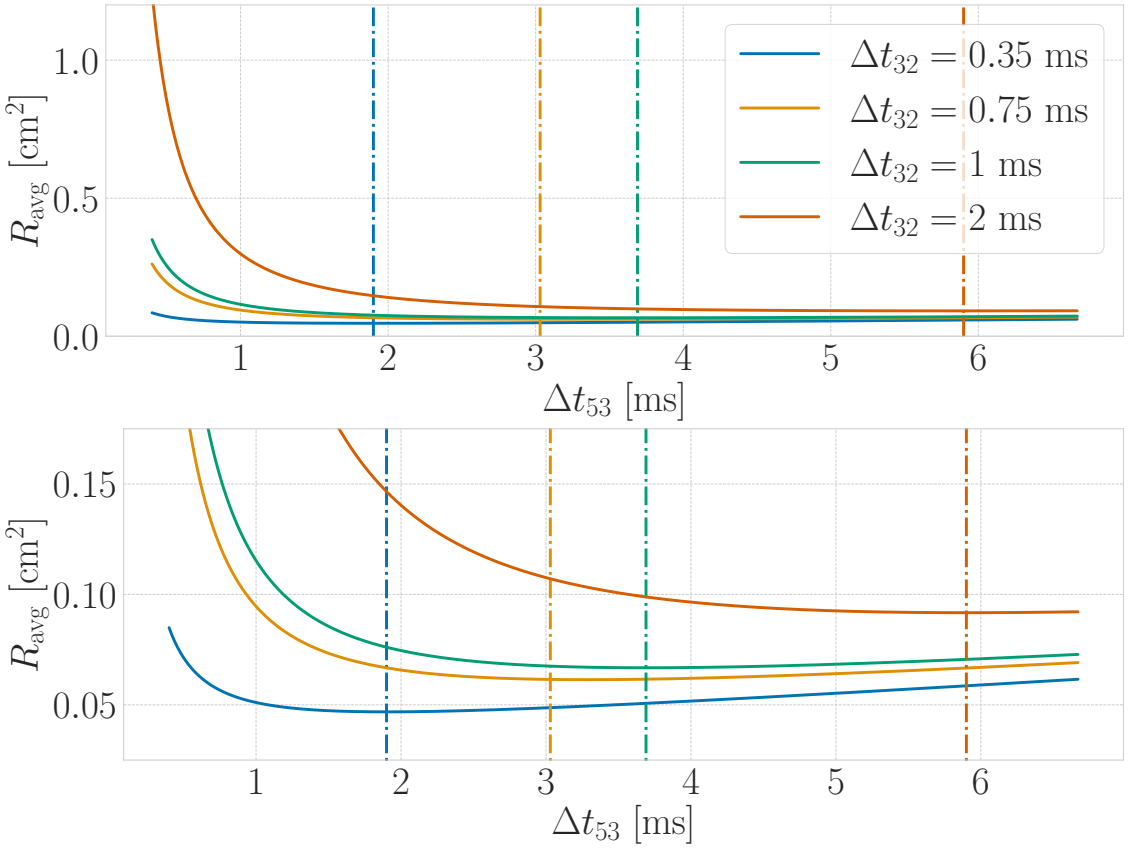


Figure 5.2: The theoretical averaged variance R_{avg} as a function of the delay Δt^{53} for 4 different values of Δt^{32}. All curves use $\rho = 7.2$ ms, which is experimentally determined for the set-up used in Section 5.5. The vertical dotted lines correspond to the analytically-evaluated minimum of the colour-matched plotted curves. The bottom plot is a close-up view of the top plot. Note that R_{avg} is converted from units of [s]^2 to [cm]^2 by multiplying with c^2 [cm^2 / s^2], where c is the speed of light, in order to visualize the variance on the range measurements directly.

surements shown in Figure 4.1b are of different distances, and are related by

$$t_{\text{f}}^2 = t_{\text{f}}^1 + \bar{v}\Delta t^{32}, \qquad t_{\text{f}}^3 = t_{\text{f}}^2 + \bar{v}\Delta t^{53},$$

where t_{f}^i is the ToF of the i^{th} message, $\bar{v} = v/c$, v is the rate of change of the distance between transceivers, and c is the speed of light. Note that motion during the intervals Δt^{32} and Δt^{53} is addressed since the intervals are in the order of milliseconds. Meanwhile, ToF is much shorter for short range measurements as discussed in Section 5.1, so motion in between time of transmission and reception is negligible.

The computed ToF measurement using the DS-TWR protocol in the absence of clock

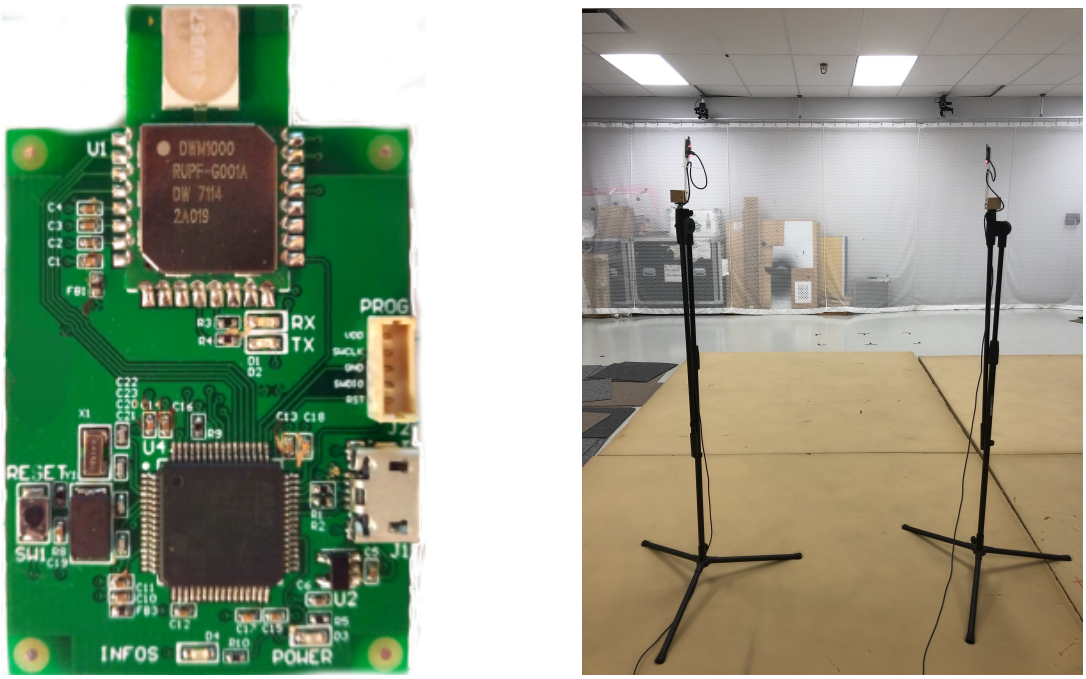


Figure 5.3: The experimental set-up. (Left) Custom-built circuit board, using the DWM1000 UWB transceiver. (Right) Two static tripods placed 1.5 metres apart, each holding a UWB transceiver.

offsets, skews, and timestamping noise is then

$$\begin{aligned}\hat{t}_{\mathrm{f}}^{\mathrm{ds}} &= \frac{1}{2}\Big(\Delta t^{41} - \frac{\Delta t^{64}}{\Delta t^{53}}\Delta t^{32}\Big) \\ &= \frac{1}{2}\Big(t_{\mathrm{f}}^{1} + \Delta t^{32} + t_{\mathrm{f}}^{2} - \frac{\Delta t^{53} + t_{\mathrm{f}}^{3} - t_{\mathrm{f}}^{2}}{\Delta t^{53}}\Delta t^{32}\Big) \\ &= \frac{1}{2}\Big(2t_{\mathrm{f}}^{1} + (1+\bar{v})\Delta t^{32} - \frac{(1+\bar{v})\Delta t^{53}}{\Delta t^{53}}\Delta t^{32}\Big) \\ &= t_{\mathrm{f}}^{1},\end{aligned}$$

meaning that the computed ToF corresponds to the distance between the transceivers at the beginning of ranging, and the error due to motion is independent from the delays Δt^{32} and Δt^{53}. Therefore, a particular feature of the DS-TWR protocol presented in Chapter 4 is that the timing optimization can be done without addressing errors due to motion.

5.5 Experimental Evaluation

To evaluate the effect of the second-response delay Δt^{53} on a real system, the following experiment is performed. Two custom-made circuit boards equipped with DWM1000 UWB transceivers [7] are fixed to two static tripods as shown in Figure 5.3. They are both con-

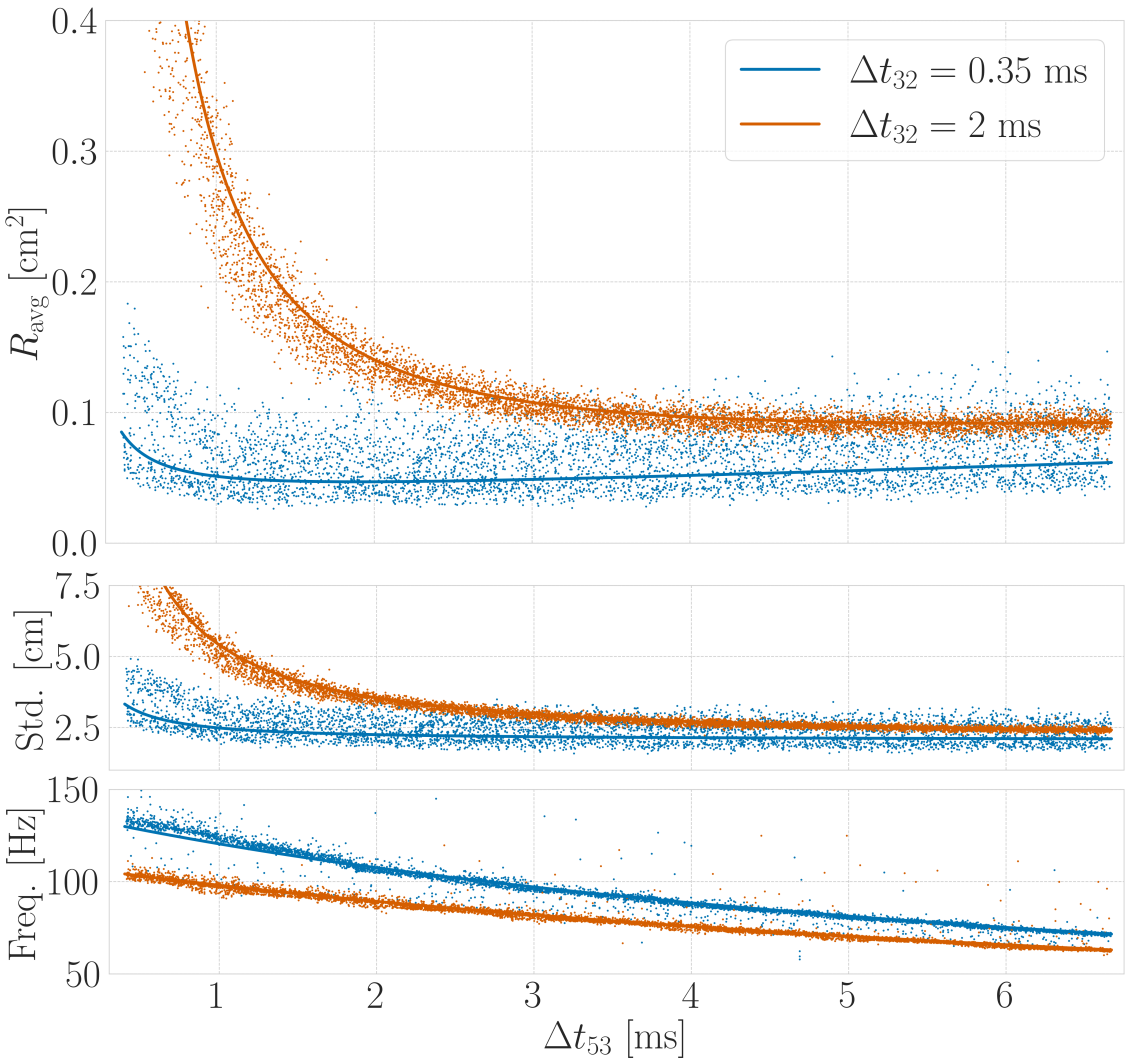


Figure 5.4: The theoretical and experimental metrics as they vary with Δt^{53}. Each point corresponds to one trial of 2500 measurements, and the solid line is the theoretical curve based on the derived analytical models and the experimentally-computed variance σ^2. The solid line matches the experimental readings. The plots show the variation of the averaged variance R_{avg} as given by (5.15), the standard deviation as given by the square root of (5.12), and the rate of the measurements as a function of Δt^{53} for two different values of Δt^{32}.

nected to a Dell XPS13 computer running Ubuntu Desktop 20.04.

First, a SS-TWR experiment is performed with 145 trials, for a total of 362500 measurements. Nothing varied in between trials, but the purpose of this experiment is to obtain the average rolling variance of SS-TWR experiments in order to get a value for σ, which is found to be $\sigma = 0.0682$ ns when averaging the variance over windows of 50 measurements. This value is used to plot the theoretical curves in Figure 5.4. The need for computing a rolling variance rather than a single value for all measurements is because of the bias of SS-TWR measurements drifting over time due to the time-varying clock skew. Nonetheless, it is worth mentioning that knowing exactly the value of σ is not necessary to perform the optimization

in (5.16), as finding the optimal delay requires solving (5.17), which is independent of σ.

With knowledge of the derived theoretical curves and the value of σ, the DS-TWR experiments are then performed to validate these values. The second-response delay Δt^{53} is varied in between many trials, and for each trial 2500 measurements are collected to compute the average variance and rate for that specific value of Δt^{53}. The results for two different values of Δt^{32} are shown in Figure 5.4, where $\rho = 7.2$ ms is found experimentally to be the time required by the computer to process a range measurement. The experiment with $\Delta t^{32} = 0.35$ ms involves 5000 trials for a total of 12.5 million measurements, while the experiment with $\Delta t^{32} = 2$ ms involves 6600 trials for a total of 16.5 million measurements.

Given that this is a static experiment, R_{avg} essentially represents the variance in the measurement obtained by averaging out all recorded measurements over a span of one second. Crucially, both experiments presented here match the theoretical expectations quite well. As Δt^{53} increases, both the standard deviation and the rate of the measurements decrease, and the optimal Δt^{53} can then be found by finding the value that minimizes R_{avg}. The experimental minimum does match the theoretical minimum, thus motivating the presented analytical optimization problem (5.16). Lastly, as expected, the experiments with a longer Δt^{32} have an order of magnitude higher standard deviation in the measurements, and in both experiments the standard deviation decreases as Δt^{53} increases.

5.6 Conclusion

This chapter extends the comparison of SS-TWR and DS-TWR to include precision by deriving an analytical model of the variance and the CRLB as a function of the signal timings of the ranging protocols. This consequently allows optimizing over the timing delays in order to minimize the variance of DS-TWR measurements, and an optimization problem is then formulated to maximize information by balancing the effect of reduced variance and reduced rate of measurements as timing delays increase. It is also shown that the effect of motion is independent of the timing delays in the proposed DS-TWR protocol. Lastly, the analytical variance model and optimization procedure are evaluated on an experimental set-up with two static ranging UWB transceivers. Future work will address finding optimal delays when the ranging protocol is customizable beyond DS-TWR, or when new TWR instances can be initiated before others are done.

This therefore concludes the thrust on improving the range measurements to improve the performance of the localization algorithms. The next chapter will address some of the remaining issues discussed in Section 3.7 by presenting a framework that allows robots to listen-in on neighbours ranging with one another. This, alongside the use of preintegration for

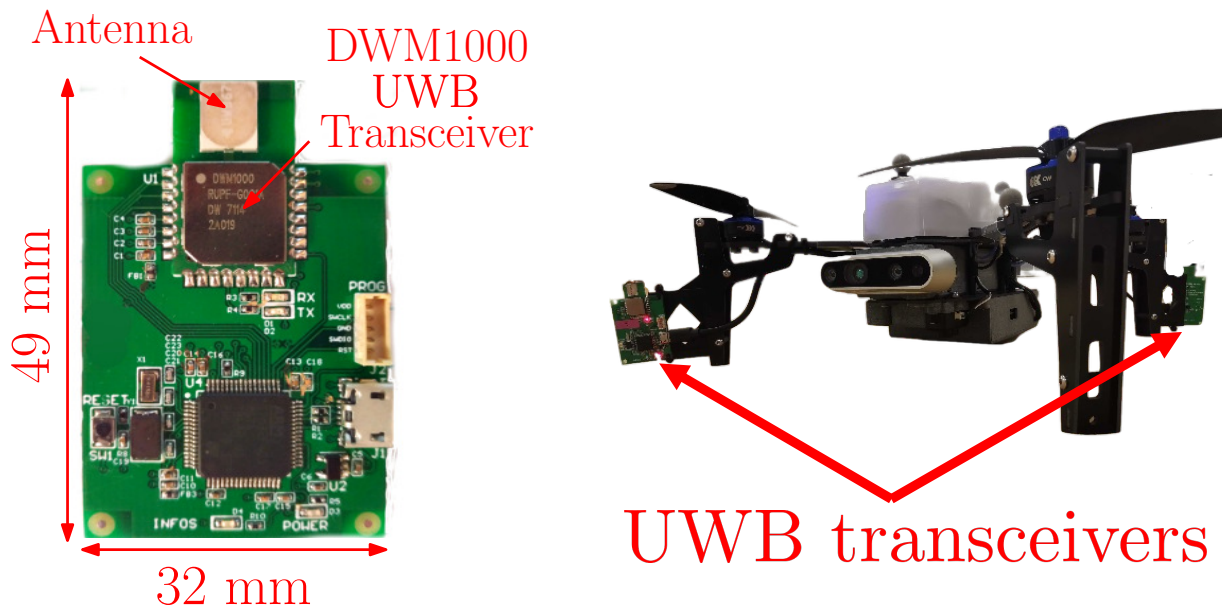
compact IMU representation, allows the deployment of a scalable and decentralized system. Additionally, a "robocentric" viewpoint will be considered, where robots estimate the relative-pose states in their own body frame to overcome the need for a magnetometer. Lastly, the relative-pose states will be represented as elements of $SE_2(3)$ to take advantage of the nice mathematical properties of on-manifold state estimation, as discussed in in Section 2.3.

Chapter 6

Multi-Robot Relative Pose Estimation and IMU Preintegration Using Passive UWB Transceivers

Summary

A major constraint associated with UWB is that only one pair of UWB transceivers can range at a time to avoid message collision, hence hindering the scalability of UWB-based localization as discussed in Section 3.7. In this chapter, a ranging protocol is proposed that allows all robots to passively listen on neighbouring communicating robots without any hierarchical restrictions on the role of the robots. This is utilized to allow each robot to obtain more range measurements and to broadcast preintegrated IMU measurements for relative extended pose state estimation directly on $SE_2(3)$. This, in turn, allows implementing decentralized algorithms that no longer require central processors similar to the one assumed in Chapter 3. Additionally, to overcome the need for a magnetometer, a "robocentric" viewpoint is considered, meaning that robots estimate relative poses in their own body frames rather than in a common local frame. Consequently, a simultaneous clock-synchronization and relative-pose estimator (CSRPE) is formulated using an on-manifold extended Kalman filter (EKF) and is evaluated in simulation using Monte-Carlo runs for up to 7 robots. The ranging protocol is implemented in C on the custom-made UWB boards fitted to 3 Uvify IFO-S quadcopters, where all the improvements in the UWB ranging accuracy and precision from Chapters 4 and 5 have been implemented. The proposed filter is evaluated in simulation and over multiple experimental trials, yielding up to 56% improvement in localization accuracy for the latter.



(a) Custom-made board fitted with a DWM1000 UWB transceiver.

(b) A Uvify IFO-S quadcopter equipped with two UWB transceivers 45 cm apart.

Figure 6.1: The experimental set-up.

6.1 Introduction

Robotic teams equipped with UWB, as in Figure 6.1, and other sensors, such as cameras and/or IMUs, have been considered for relative pose estimation, which is a prerequisite for applications such as collision avoidance and collaborative mapping and infrastructure inspection. Nonetheless, using UWB for relative pose estimation in multi-robot teams introduces a distinct set of problems, as discussed in Chapter 1. Firstly, UWB ranging and communication is not robust to message collision, thus imposing the constraint that only one pair of transceivers can communicate at a time. This is typically addressed using *time-division multiple-access* (TDMA) media-access control (MAC) protocols alongside a round-robin approach to determine which pair communicates at each time. However, the larger the team of robots, the longer the time gaps in between a robot ranging with another. Another complication with UWB ranging is the reliance on ToF measurements, which necessitates the presence of a clock at each UWB transceiver. However, in practice, these clocks run at different rates, and therefore require some synchronization mechanism. The importance of synchronization can be highlighted by the fact that 1 ns in synchronization error translates to c [m/s] $\times 10^{-9}$ [s] ≈ 30 [cm] in localization error, where c is the speed of light.

Another practical issue associated with multi-robot systems is communication constraints, which limit the amount of information that can be transmitted between robots. In filtering

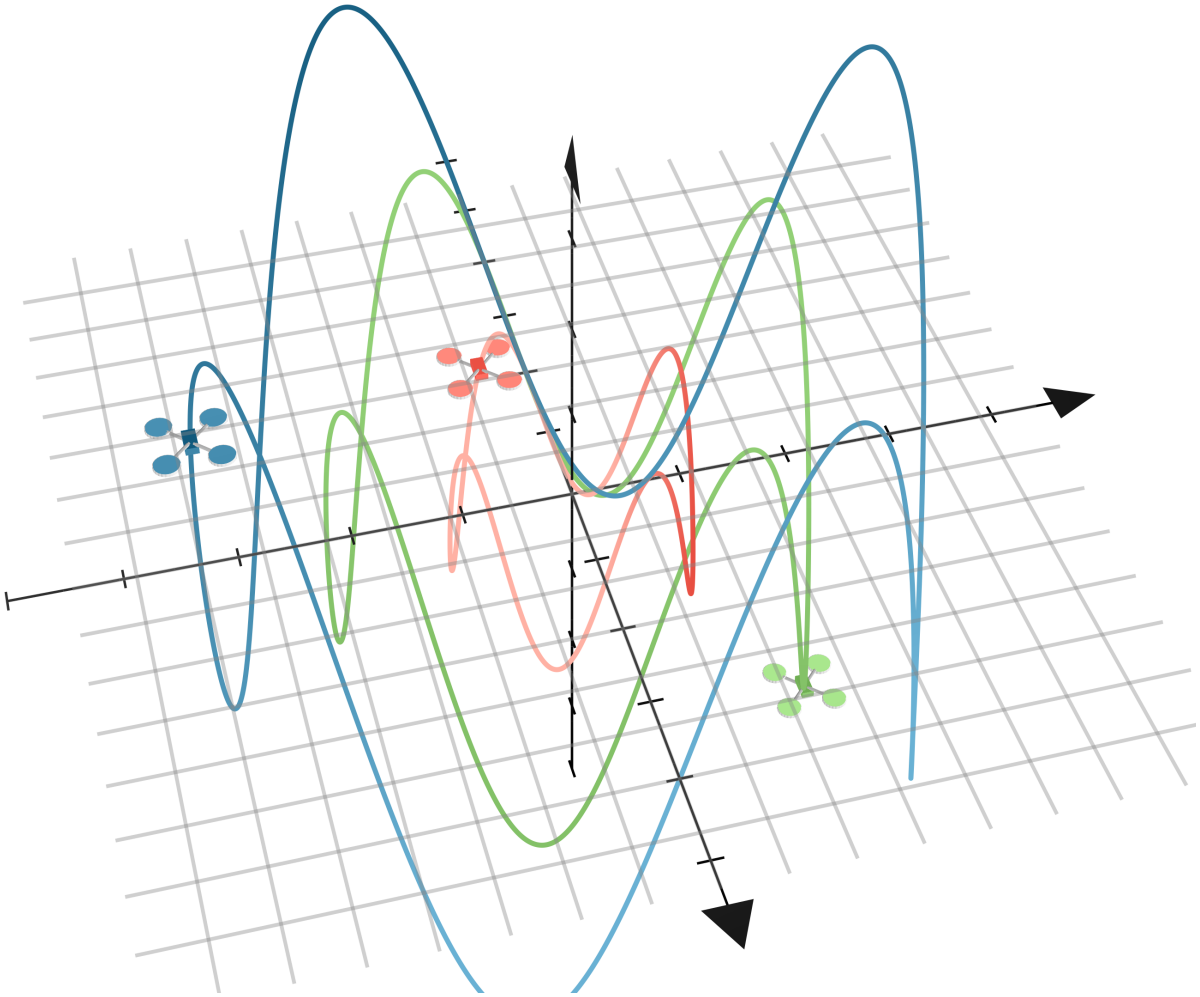


Figure 6.2: The trajectories followed by three simulated quadcopters.

applications where there are for example 3 quadcopters moving randomly in 3-dimensional space as shown in Figure 6.2, IMU measurements must be broadcasted if robots are to estimate their neighbours' relative states directly from the raw measurements. Nonetheless, IMU measurements are typically recorded at a very high frequency, and the constraint that only one pair can be communicating at a time means that communication links between robots do not always exist. Therefore, a more efficient way of sharing odometry information is required.

To achieve a practical relative pose estimation solution that is implementable on a robotic team, this chapter addresses the aforementioned constraints. The contributions of this chapter are summarized as follows.

1. A ranging protocol is introduced that extends classical ranging protocols by allowing neighbouring robots to passively listen to the measurements and timestamp receptions, with no assumptions or imposed constraints on the robots' hierarchy. The concept of

passive listening is then utilized to provide a $(1+3n)$-fold increase in the number of measurements recorded when there are a total of $n+1$ robots each equipped with two UWB transceivers. The concept of passive listening is additionally utilized for more efficient information sharing and implementing simple MAC protocols.

2. Representing the extended pose state as an element of $SE_2(3)$, an on-manifold tightly-coupled simultaneous clock-synchronization and relative-pose estimator (CSRPE) is then proposed, which allows incoporating passive listening measurements in an extended Kalman filter (EKF) to improve the relative pose estimation. This provides a means for many different robots to estimate the relative poses of their neighbours relative to themselves at a high frequency.

3. Rather than sharing high-frequency IMU readings with neighbours, the concept of preintegration [79] is developed for relative pose states on $SE_2(3)$, and is used as a means of efficient IMU data logging and communication between robots. This is additionally incorporated in the CSRPE, where the theory behind filtering with delayed inputs is developed as the preintegrated IMU measurements arrive asynchronously from neighbouring robots.

4. The proposed algorithm is evaluated in simulation using Monte-Carlo trials and in experiments using 4 trials with 3 quadcopters equipped with two UWB transceivers each. It is shown that localization accuracy improves up to 23% when compared to a centralized scenario and up to 48% when compared to the case of no passive listening.

The remainder of the chapter is organized as follows. Related work is presented in Section 6.2. The problem is formulated in Section 6.3, then the proposed ranging protocol is discussed in Section 6.4. The relative-pose process model and preintegration on $SE_2(3)$ are discussed in Sections 6.5 and 6.6, respectively. Simulation and experimental results are discussed in Sections 6.7 and 6.8, respectively, before further practical considerations are mentioned in Section 6.9.

6.1.1 Notation

The notation used in this chapter is similar to the rest of this thesis, but is reiterated here. The position of point z relative to point w is denoted $\underrightarrow{r}^{zw}$. The relative velocity and acceleration between points z and w with respect to frame i is denoted

$$\underrightarrow{v}^{zw/i} \triangleq {}^{i}\dot{\underrightarrow{r}}^{zw}, \qquad \underrightarrow{a}^{zw/i} \triangleq {}^{i}\dot{\underrightarrow{v}}^{zw/i}.$$

The rotation from a reference frame j to a reference frame i is parametrized using a rotation matrix $\mathbf{C}_{ij} \in SO(3)$. Therefore, the relationship between $\mathbf{r}_i^{zw}$ and $\mathbf{r}_j^{zw}$ is given by $\mathbf{r}_i^{zw} = \mathbf{C}_{ij}\mathbf{r}_j^{zw}$.

To obtain a range measurement, two transceivers transmit and timestamp a sequence of messages among themselves as shown between Robot 1 and Robot 2 in Figure 6.4. A time instance corresponding to the i^{th} message is denoted as $\mathtt{T}^i$ for the transmission time and $\mathtt{R}^i$ for the reception time, while a subscript j denotes the time instance as timestamped by Transceiver j. For example, $\mathtt{T}^1_{f_0}$ is the timestamp corresponding to the first message transmission as recorded by Transceiver f_0. Note that the notation used in this chapter for the UWB timestamps is different from previous chapters, in that transmission and reception time instances are denoted using different letters. The protocol example shown between Robot 1 and Robot 2 in Figure 6.4 is a modified version of the standard double-sided two-way ranging (DS-TWR [80]) protocol as presented in Chapter 4, where the message shown in red represents an "information message" used to broadcast the timestamps recorded by Robot 1.

Throughout this chapter, $\mathbf{1}$ and $\mathbf{0}$ denote identity and zero matrices of appropriate dimension. When ambiguous, a subscript will indicate the dimension of these matrices.

6.2 Related Work

The majority of UWB-based localization relies on a set of pre-localized and synchronized static transceivers, or *anchors*, to localize a mobile transceiver [14], [29], [81]. This typically relies on the anchors ranging with the mobile transceiver using standard ranging protocols such as *two-way ranging* (TWR) or *time-difference-of-arrival* (TDoA) [21], [15, Chapter 7.1.4]. More complicated ranging protocols have been proposed in [82]–[84] to allow multiple anchors to passively listen-in on messages with the mobile transceiver to localize it.

Calibrating the clocks and location of anchors is challenging, and [44], [85] propose an approach where anchors actively range with one another to synchronize and localize themselves. Meanwhile, the mobile transceiver passively listens to these signals to localize itself using the anchors' estimated clock states and positions. The work in [22], [86] extends this by applying a Kalman filter (KF) to the synchronization and localization problem. In [87], a localization cost function is proposed that is invariant to the anchors' synchronization error. Meanwhile, in [88], the synchronization approach is accurate to within a few microseconds, whereas nanosecond-level accuracy is desired for localization with cm accuracy.

Overcoming the need for a fixed infrastructure of anchors, UWB has been used more recently for teams of robots, as in Chapter 3 and in [55], [89], [90]. In [91], it is assumed that

neighbouring robots know their poses and clock states, thus essentially behaving as mobile anchors, allowing a mobile transceiver to localize itself. The use of robots with multiple transceivers is motivated in Chapter 3, and in [19] a robot equipped with 4 transceivers localizes a mobile transceiver relative to itself by having one of the 4 transceivers actively range with the target and the other 3 passively listening.

In [92], [93], a passive-listening-based ranging protocol is proposed where the network is divided into "parent robots" that actively range with one another and "child robots" that passively listen-in on these measurements. This hierarchical constraint has the limitation that parent robots cannot localize child robots and do not benefit from passive listening measurements themselves when they are not involved in a ranging transaction. Additionally, it is suggested that the child robots use the estimated position and clock states of the parent states, which in filtering applications would lead to untracked cross-correlations that would result in poor performance [5].

Furthermore, in filtering applications, the problem of communicating IMU measurements to neighbours remains unaddressed. In [32], [33], scattering theory is used to send pre-computed matrices between two robots rather than individual IMU measurements, in a manner similar to the concept of preintegration [79], [94]. However, extending this to more than two robots is challenging, particularly for preintegrated poses directly on $SE_2(3)$ [95], [96].

The motive behind using $SE_2(3)$ state representation for relative pose estimation using range measurements is due to this being inherently a nonlinear problem, which is commonly addressed using particle filtering [60], [69] to handle non-ellipsoid-shaped distributions in Cartesian coordinates, see Figure 2.4. This nonlinearity motivates the use of an on-manifold EKF, such as an EKF with states represented directly on the $SE_2(3)$ manifold, which can represent such non-ellipsoidal-shaped distributions using exponential coordinates [97].

6.3 Problem Formulation

Consider a scenario with $n+1$ robots, as shown in Figure 6.3 for $n = 3$. Throughout this chapter, the perspective of one robot is considered, denoted without loss of generality Robot 0, as any of the $n+1$ robots can be considered Robot 0. Neighbouring robots are then referred to as Robot i, $i \in \{1, \ldots, n\}$. Unlike Chapter 3, this chapter employs a "robocentric" viewpoint of the relative-pose state estimation problem, where all states are estimated relative to Robot 0 and are resolved in the body frame of that robot. The robots are assumed to be rigid bodies, so any vector can be resolved in one of the following $n+2$ reference frames:

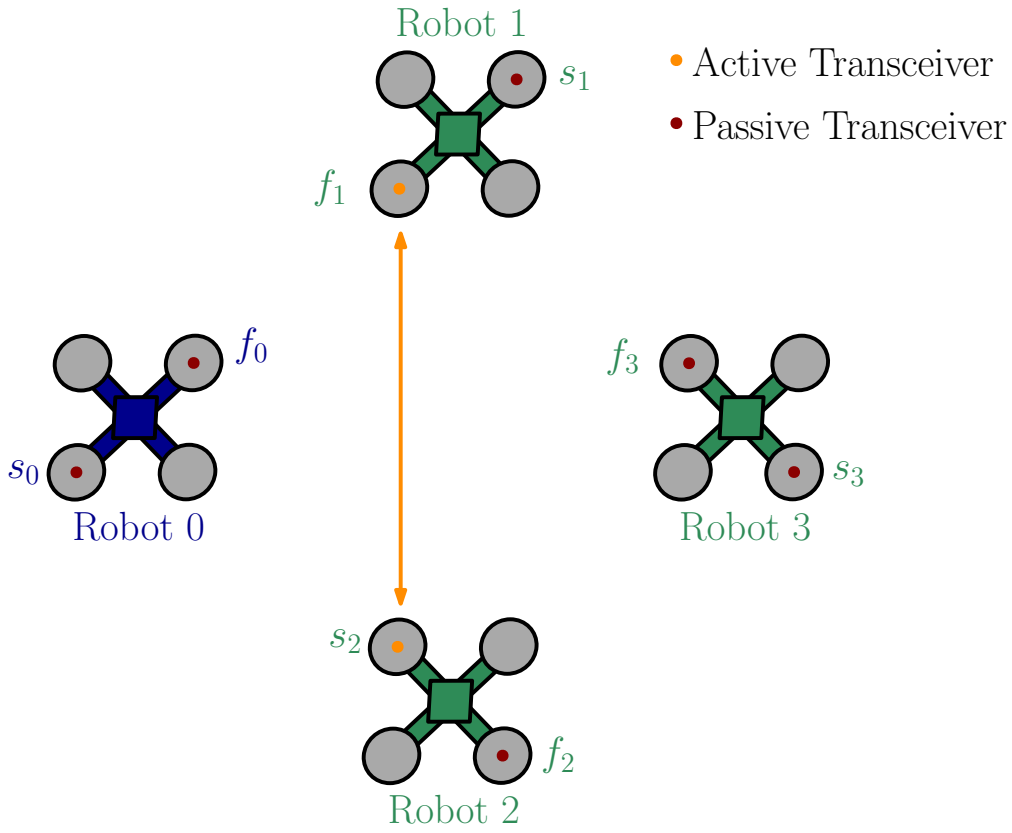


Figure 6.3: An example of a ranging transaction, where Transceivers f_1 and s_2 are actively ranging with one another and all other tags are passively listening.

- an (absolute) inertial frame denoted with a subscript a,
- Robot 0's body frame denoted with a subscript 0, or
- neighbouring Robot i's body frame denoted with a subscript i.

Each robot is equipped with an IMU at its center, consisting of a 3-axis gyroscope and accelerometer, but no magnetometer. Given the use of accelerometers, the relative pose estimation problem involves estimating the extended pose of each neighbouring robot relative to Robot 0 in Robot 0's body frame. The extended pose of Robot i is then defined as

$$\mathbf{T}_{0i} = \begin{bmatrix} \mathbf{C}_{0i} & \mathbf{v}_0^{i0/a} & \mathbf{r}_0^{i0} \\ & 1 & \\ & & 1 \end{bmatrix} \in SE_2(3), \quad i \in \{1, \ldots, n\},$$

where time dependence is omitted from the notation for conciseness. The dependence on the absolute frame a is also omitted from the notation $\mathbf{T}_{0i}$, with the convention that all extended relative pose matrices in this chapter are of this form, where the vector corresponding to the

second component in the first row is the derivative with respect to the *absolute* frame of the vector corresponding to the third component, irrespective of the fact that these vectors are resolved in frame 0.

Each robot is also equipped with 2 UWB transceivers for relative pose observability, as motivated by Chapter 3. The *first* and *second* transceivers on Robot j are denoted f_j and s_j, respectively, for $j \in \{0, \dots, n\}$. It is assumed that the vector coordinates $\mathbf{r}_j^{f_j j}$ and $\mathbf{r}_j^{s_j j}$ between the transceivers and the IMU on Robot j are known, since they can be measured by hand or more accurately using a motion capture system.

Denote the set of all transceivers as $\mathcal{C} = \{f_0, \dots, f_n, s_0, \dots, s_n\}$. Consider the state of the clock on Transceiver $i \in \mathcal{C}$ relative to real time. The evolution of the offset $\tau_i(t)$ of clock i is modelled as (2.10). A robocentric viewpoint is also maintained for the clock states, where offsets and skews of all clocks are estimated relative to the clock of Transceiver f_0 on Robot 0. The clock state of Transceiver s_0 is then

$$\mathbf{c}_{s_0 f_0} = \begin{bmatrix} \tau_{s_0 f_0} \\ \gamma_{s_0 f_0} \end{bmatrix} \triangleq \begin{bmatrix} \tau_{s_0} - \tau_{f_0} \\ \gamma_{s_0} - \gamma_{f_0} \end{bmatrix} \in \mathbb{R}^2,$$

while the clock state of neighbouring Robot i is given by

$$\boldsymbol{\mathcal{X}}_{i0}^{\mathrm{c}} \triangleq (\mathbf{c}_{f_i f_0}, \mathbf{c}_{s_i f_0}) \in \mathbb{R}^2 \times \mathbb{R}^2, \quad i \in \{1, \dots, n\},$$

where, as before, time dependence is omitted from the notation for conciseness. The full relative state estimate of Robot i is then given by

$$\boldsymbol{\mathcal{X}}_{i0} \triangleq (\mathbf{T}_{0i}, \boldsymbol{\mathcal{X}}_{i0}^{\mathrm{c}}) \in SE_2(3) \times \mathbb{R}^2 \times \mathbb{R}^2,$$

and the full state estimated by Robot 0 is

$$\boldsymbol{\mathcal{X}} \triangleq (\mathbf{c}_{s_0 f_0}, \boldsymbol{\mathcal{X}}_{10}, \dots, \boldsymbol{\mathcal{X}}_{n0}) \in \mathbb{R}^2 \times \left(SE_2(3) \times \mathbb{R}^2 \times \mathbb{R}^2\right)^n.$$

Communication constraints limit Robot 0's ability to estimate the state $\boldsymbol{\mathcal{X}}$, since to prevent message collision only one pair of transceivers can communicate at a time. As the number of transceivers increases, this can result in poor scalability due to longer wait times between successive ranging measurements by a given pair. Additionally, the rate at which transceivers communicate is typically lower than the rate at which IMU measurements are recorded at neighbouring robots, thus Robot 0 cannot collect the IMU measurements from all its neighbours without significant and impractical communication overhead. Therefore, part of the problem is to design a scalable and practical ranging protocol that accomodates

these communication constraints.

This chapter presents an on-manifold extended Kalman filter (EKF) for estimating the state $\boldsymbol{\mathcal{X}}$ using a novel ranging protocol that allows all robots to listen-in on neighbours while awaiting their turn to communicate. It is known from Chapter 3 and [18] that the relative pose states are *observable* from IMU and range measurements. In particular, the use of two transceivers per robot ensures the observability of the relative poses while overcoming the need for *persistent excitation* or constant relative motion between the robots [1]. This benefit comes at the added cost of an additional transceiver. Nonetheless, UWB transceivers are typically compact, lightweight, low-power, and inexpensive. In fact, the ones used in this paper as shown in Figure 6.1, are 32 mm $\times$ 49 mm in size and weigh approximately 8 g each. Meanwhile, assuming that the relative pose states are known since they are observable from the IMU and range measurements alone, clock-offset measurements are sufficient to ensure observability of the clock states [22].

To simplify the analysis in this paper, a complete communication graph with no packet drops or failures between the robots is assumed, which reduces the scalability of the system. Another factor impacting the scalability of the system is that the size of the state $\boldsymbol{\mathcal{X}}$ increases with n; therefore, the number of robots that can be included in Robot 0's EKF is limited by Robot 0's computational capabilities. This chapter addresses the scenario where n is limited to a few robots. The complications associated with larger systems and incomplete and dynamic communication graphs are discussed in Section 6.9.2.

6.4 Ranging Protocol

6.4.1 Overview

To address the communication constraints, a ranging protocol is proposed that involves performing DS-TWR between all pairs of transceivers not on the same robot in sequence while leveraging passive listening measurements at all other transceivers that are not actively ranging. This is shown in Figure 6.4 for an example where Transceiver f_1 is initiating a TWR transaction with transceiver s_2, and Transceivers f_0 and s_0 are passively listening. In the proposed ranging protocol, any of the $2(n+1)$ transceivers can initiate a TWR transaction with any of the $2n$ transceivers not on the same robot. In this section, the passive listening measurements are utilized in the relative-pose state estimator as a source of ranging information between the different robots. This is possible due to the tightly-coupled nature of the proposed estimator, which performs both clock synchronization and relative pose estimation, meaning that clock-offset-corrupted passive listening measurements

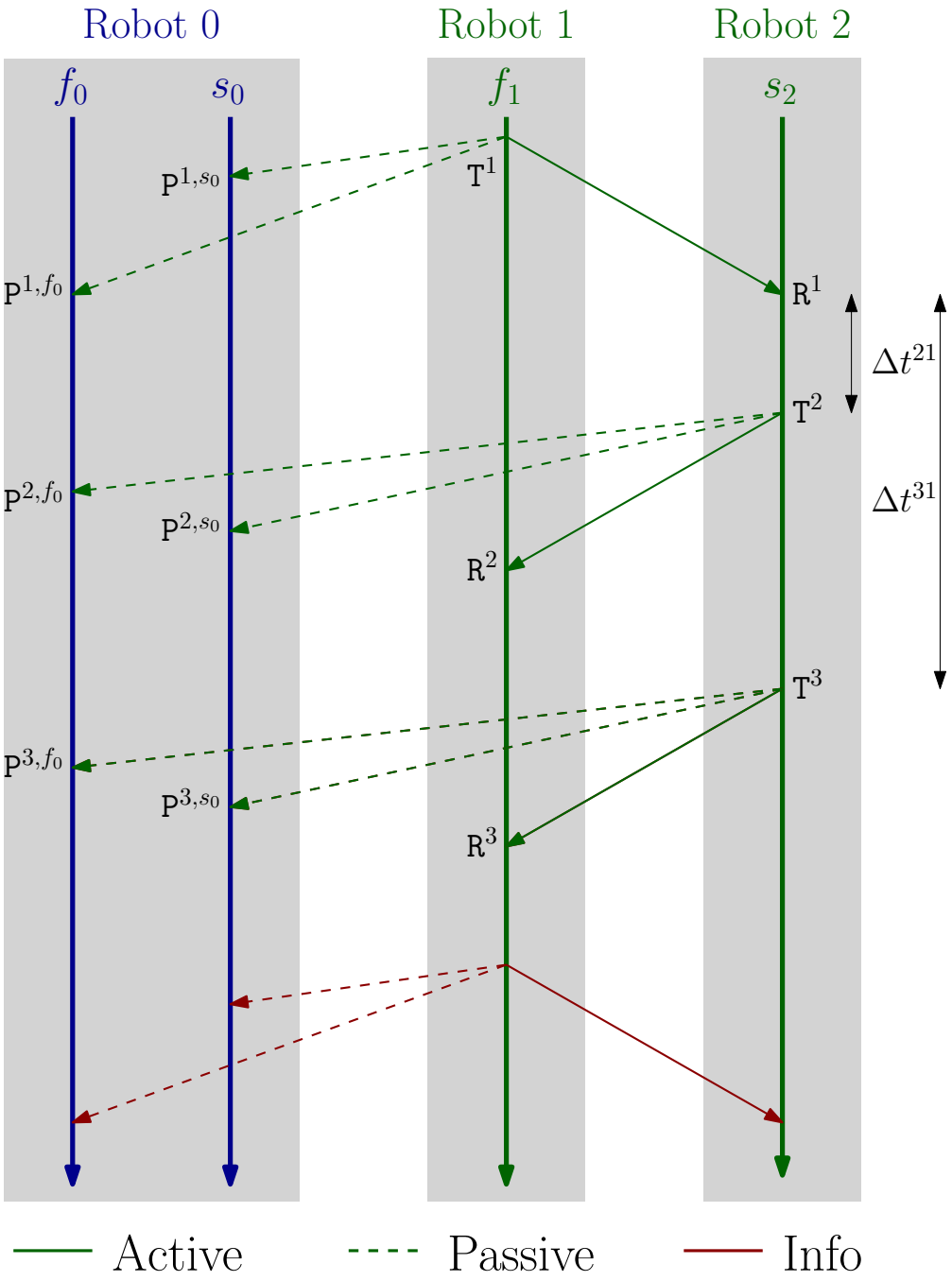


Figure 6.4: Proposed ranging protocol when Transceiver f_1 is initiating a DS-TWR ranging transaction with Transceiver s_2. This chapter proposes that all other transceivers listen in on these messages. Shown here are Transceivers f_0 and s_0 on Robot 0 passively listening, where the time instance corresponding to the i^{th} passive reception at Transceiver j is denoted $\mathrm{P}^{i,j}$. Additionally, the intervals Δt^{21} and Δt^{31} are defined in this chapter as shown in the figure.

can still be used to correct relative pose states, as cross-correlation information is available between clock states and relative pose states at all times. There are multiple advantages to passive listening in multi-robot pose estimation applications, including the availability of

more measurements for state correction, providing the robots with information-broadcasting ability, and allowing the implementation of simple MAC protocols.

The remainder of this section analyzes how the proposed ranging protocol can be used in a CSRPE. The particular scenario under study is the one shown in Figures 6.3 and 6.4, where transceivers on two neighbouring robots are the ones actively ranging. This is the most general case, and scenarios where one of the transceivers on Robot 0 is actively ranging involve similar but simpler derivations.

6.4.2 The Protocol

The ranging protocol proposed in this paper involves two transceivers actively ranging with one another, while all other transceivers passively listen-in on the messages. The actively ranging transceivers perform DS-TWR, as presented in Chapter 4. The example shown in Figure 6.4 is an example where Transceiver f_1 on Robot 1 initiates a transaction with Transceiver s_2 on Robot 2. Both robots have another transceiver, s_1 and f_2 for Robot 1 and Robot 2, respectively, which then passively listen to all the messages transmitted between the active transceivers. Additionally, all other robots have both their transceivers passively listen to all the messages. For example, Robot 0 records passive listening measurements at both Transceivers f_0 and s_0.

When the transceivers transmit and receive messages, whether actively or passively, the transceivers timestamp the time of transmission or reception. Each robot needs access to neighbouring robots' timestamp measurements in order to be able to compute range measurements from the transaction. For example, Robot 0 needs access to the timestamps recorded by Transceivers f_1 and s_2. As shown in Figure 6.4, all timestamps are made available at Robot 0 at the end of the transaction by communicating all the timestamps recorded at Robot 1 in a final information message shown in red, and the timestamps recorded at Robot 2 are communicated in the last message transmitted by Robot 2. Note, however, that passive listening measurements recorded by the other transceivers on Robot 1, Robot 2, and any other neighbouring robots are not made available to Robot 0. The ranging protocol is outlined in Algorithm 1 for the scenario shown in Figure 6.4.

When implementing the ranging protocol, a choice has to be made on the receiving robot's side (in this case, Robot 2) for the delays $\Delta t^{21} \triangleq \mathtt{T}^2 - \mathtt{R}^1$ and $\Delta t^{31} \triangleq \mathtt{T}^3 - \mathtt{R}^1$. These user-defined parameters affect the frequency and noise of the measurements, and can be chosen based on Chapter 5. Note that $\Delta t^{32} \triangleq \Delta t^{31} - \Delta t^{21}$. Additionally, it will be assumed throughout this paper that the distances between transceivers and the clock skews remain constant during one ranging transaction. These are good approximations for most robotic

Algorithm 1 The algorithm for the proposed ranging protocol when Transceiver f_1 is initiating the ranging transaction with Transceiver s_2, and Transceivers f_0, s_0, s_1, and f_2 are passively listening. Note that in this algorithm, the color blue is reserved for passive listening measurements that are available to the reference robot Robot 0, and the color teal is reserved for passive listening measurements at other robots that are not available to Robot 0.

Initiating Robot (Robot 1): The initiating robot has an active transceiver, f_1, and a passive transceiver, s_1.

1: Transmit message to s_2, and timestamp $\mathtt{T}^1$ in own clock.
2: Timestamp passive reception $\mathtt{P}^{1,s_1}$ in own clock, for message transmitted by f_1.
3: Timestamp reception $\mathtt{R}^2$ in own clock, for message transmitted by s_2.
Timestamp passive reception $\mathtt{P}^{2,s_1}$ in own clock, for message transmitted by s_2.
4: Timestamp reception $\mathtt{R}^3$ in own clock and read $\mathtt{R}^1$, $\mathtt{T}^2$, $\mathtt{T}^3$, for message transmitted by s_2.
Timestamp passive reception $\mathtt{P}^{3,s_1}$ in own clock, for message transmitted by s_2.
5: Transmit info message with $\mathtt{T}^1$, $\mathtt{R}^2$, $\mathtt{R}^3$ to s_2.

Target Robot (Robot 2): The target robot has an active transceiver, s_2, and a passive transceiver, f_2.

1: Timestamp reception $\mathtt{R}^1$ in own clock, for message transmitted by f_1.
Timestamp passive reception $\mathtt{P}^{1,f_2}$ in own clock, for message transmitted by f_1.
2: Transmit message to f_1, and timestamp $\mathtt{T}^2$ in own clock.
3: Timestamp passive reception $\mathtt{P}^{2,f_2}$ in own clock, for message transmitted by s_2.
4: Set $\mathtt{T}^3 = \mathtt{R}^1 + \Delta t^{31}$, and wait until own clock is $\mathtt{T}^3$.
5: Transmit message with $\mathtt{R}^1$, $\mathtt{T}^2$, $\mathtt{T}^3$ to f_1.
6: Timestamp passive reception $\mathtt{P}^{3,f_2}$ in own clock, for message transmitted by s_2.
7: Read $\mathtt{T}^1$, $\mathtt{R}^2$, $\mathtt{R}^3$ from info message transmitted by f_1.

Passive Robot (Robot 0): The passive robot has two passive transceivers, f_0 and s_0.

1: Timestamp passive reception $\mathtt{P}^{1,f_0}$ in own clock, for message transmitted by f_1.
Timestamp passive reception $\mathtt{P}^{1,s_0}$ in own clock, for message transmitted by f_1.
2: Timestamp passive reception $\mathtt{P}^{2,f_0}$ in own clock, for message transmitted by s_2.
Timestamp passive reception $\mathtt{P}^{2,s_0}$ in own clock, for message transmitted by s_2.
3: Timestamp passive reception $\mathtt{P}^{3,f_0}$ in own clock and read $\mathtt{R}^1$, $\mathtt{T}^2$, $\mathtt{T}^3$, for message transmitted by s_2.
Timestamp passive reception $\mathtt{P}^{3,s_0}$ in own clock, for message transmitted by s_2.
4: Read $\mathtt{T}^1$, $\mathtt{R}^2$, $\mathtt{R}^3$ from info message transmitted by f_1.

applications with typical clock rate variations [15, Chapter 7.1.4], [2].

The proposed ranging protocol has the following advantages. Given that all transceivers passively listen to neighbouring robots communicating, this proposed protocol gives robots the ability to broadcast information such as IMU measurements or estimated maps at a higher

rate as any robot can obtain information communicated between two neighbouring robots. This feature will be utilized for multi-robot preintegration in Section 6.6. Additionally, given that each robot knows which robots are currently ranging, a simple MAC protocol can be implemented to prevent message collision between multiple robots attempting to transmit messages at the same time. To do so, a user-defined sequence of ranging pairs can be made known to all robots. Each robot can then keep track of which pair in the sequence is currently ranging, and initiate a TWR transaction to a specified transceiver when it is its turn to do so. This MAC protocol is named here the *common-list protocol*.

6.4.3 Modelling Timestamp Measurements

The time instances shown in Figure 6.4 are only available to the robots as noisy timestamps and in the clocks of the transceivers rather than in the global common time. Therefore, the timestamp measurements are affected by clock offsets, clock skews, and white noise. Modelling these effects, the timestamps available at Robot 1 (hereinafter, the *initiating robot*) are of the form

$$\tilde{\mathtt{T}}^1_{f_1} = \mathtt{T}^1 + \tau_{f_1}(\mathtt{T}^1) + \eta^1_{f_1}, \tag{6.1}$$

$$\tilde{\mathtt{R}}^2_{f_1} = \mathtt{T}^1 + \frac{2}{c} d^{s_2 f_1} + \Delta t^{21} + \tau_{f_1}(\mathtt{R}^2) + \eta^2_{f_1}, \tag{6.2}$$

$$\tilde{\mathtt{R}}^3_{f_1} = \mathtt{T}^1 + \frac{2}{c} d^{s_2 f_1} + \Delta t^{31} + \tau_{f_1}(\mathtt{R}^3) + \eta^3_{f_1}, \tag{6.3}$$

where $\tilde{(\cdot)}$ here denotes a measured value, $d^{s_2 f_1}$ is the distance between Transceivers s_2 and f_1, and η^ℓ_i is the random noise on the ℓ^{th} measurement of Transceiver i. All the random noise variables on timestamps are assumed to be independent, zero-mean and with the same variance σ^2.

Similarly, the measurements available at Robot 2 (hereinafter, the *target robot*) are of the form

$$\tilde{\mathtt{R}}^1_{s_2} = \mathtt{T}^1 + \frac{1}{c} d^{s_2 f_1} + \tau_{s_2}(\mathtt{R}^1) + \eta^1_{s_2}, \tag{6.4}$$

$$\tilde{\mathtt{T}}^2_{s_2} = \mathtt{T}^1 + \frac{1}{c} d^{s_2 f_1} + \Delta t^{21} + \tau_{s_2}(\mathtt{T}^2) + \eta^2_{s_2}, \tag{6.5}$$

$$\tilde{\mathtt{T}}^3_{s_2} = \mathtt{T}^1 + \frac{1}{c} d^{s_2 f_1} + \Delta t^{31} + \tau_{s_2}(\mathtt{T}^3) + \eta^3_{s_2}. \tag{6.6}$$

The timestamp measurements (6.1)-(6.6) correspond to the DS-TWR protocol in Section 4, from which ToF pseudomeasurements can be generated. Nonetheless, additional measurements are available at Robot 0 (hereinafter, the *passive robot*) since its transceivers f_0 and

s_0 also receive the messages exchanged between the two actively ranging robots. This yields the following additional timestamp measurements at Robot 0,

$$\tilde{\mathtt{P}}_i^{1,i} = \mathtt{T}^1 + \frac{1}{c} d^{f_1 i} + \tau_i(\mathtt{P}^{1,i}) + \eta_i^1, \tag{6.7}$$

$$\tilde{\mathtt{P}}_i^{2,i} = \mathtt{T}^1 + \frac{1}{c} d^{s_2 f_1} + \frac{1}{c} d^{s_2 i} + \Delta t^{21} + \tau_i(\mathtt{P}^{2,i}) + \eta_i^2, \tag{6.8}$$

$$\tilde{\mathtt{P}}_i^{3,i} = \mathtt{T}^1 + \frac{1}{c} d^{s_2 f_1} + \frac{1}{c} d^{s_2 i} + \Delta t^{31} + \tau_i(\mathtt{P}^{3,i}) + \eta_i^3, \tag{6.9}$$

where $i \in \{f_0, s_0\}$. Similarly, each neighbouring robot not involved in the ranging transaction records its own passive listening measurements at its two transceivers. However, these are not shared with other robots as this would require each robot to take its turn transmitting a message.

In the case where Robot 0 is not involved in the ranging transaction and just listens in passively, there are 12 available timestamp measurements at Robot 0, 6 sent by neighbouring robots, and 3 passive-listening timestamps per transceiver on Robot 0. However, when one of the transceivers f_0 or s_0 is involved in the ranging transaction, only 9 timestamp measurements are available.

6.4.4 Pseudomeasurements as a Function of the State

To use the timestamp measurements (6.1)-(6.9) in the CSRPE, they must be rewritten as a function of the state being estimated. In this subection, pseudomeasurements based on the timestamps available at Robot 0 after one TWR transaction are formulated to get models that are only a function of the states being estimated, as well as the known vectors between the transceivers and the IMUs resolved in the robot's body frame.

First, notice that the distance $d^{s_2 f_1}$ between transceivers in (6.1)-(6.6) can be written as a function of the estimated states,

$$\begin{aligned} d^{s_2 f_1} &= \left\| \mathbf{r}_0^{s_2 f_1} \right\| \\ &= \left\| \mathbf{r}_0^{s_2 0} - \mathbf{r}_0^{f_1 0} \right\| \\ &= \left\| (\mathbf{C}_{02} \mathbf{r}_2^{s_2 2} + \mathbf{r}_0^{20}) - (\mathbf{C}_{01} \mathbf{r}_1^{f_1 1} + \mathbf{r}_0^{10}) \right\| \\ &= \left\| \mathbf{\Pi} \left(\mathbf{T}_{02} \tilde{\mathbf{r}}_2^{s_2 2} - \mathbf{T}_{01} \tilde{\mathbf{r}}_1^{f_1 1} \right) \right\|, \end{aligned} \tag{6.10}$$

where $\|\cdot\|$ is the Euclidean norm, $\mathbf{\Pi} = \begin{bmatrix} \mathbf{1}_3 & \mathbf{0}_{3\times 2} \end{bmatrix} \in \mathbb{R}^{3\times 5}$, and

$$\tilde{\mathbf{r}} = \begin{bmatrix} \mathbf{r}^\mathsf{T} & 0 & 1 \end{bmatrix}^\mathsf{T}.$$

To design the EKF, the linearization of (6.10) with respect to the state is shown in Appendix C.

Therefore, pseudomeasurements can be formed that are only a function of the distance between the transceivers, the clock states (relative to f_0), and the white timestamping noise. The **first pseudomeasurement** is the standard ToF measurement associated with DS-TWR as discussed in Chapter 4, which from timestamps (6.1)-(6.6) can be written as

$$\begin{aligned} y^{\text{tof}} &= \frac{1}{2}\left(\left(\tilde{\mathtt{R}}^2_{f_1} - \tilde{\mathtt{T}}^1_{f_1}\right) - \frac{\tilde{\mathtt{R}}^3_{f_1} - \tilde{\mathtt{R}}^2_{f_1}}{\tilde{\mathtt{T}}^3_{s_2} - \tilde{\mathtt{T}}^2_{s_2}}\left(\tilde{\mathtt{T}}^2_{s_2} - \tilde{\mathtt{R}}^1_{s_2}\right)\right) \\ &\approx \frac{1}{c} d^{s_2 f_1} + \frac{1}{2}\left(\eta^2_{f_1} - \eta^1_{f_1} - \eta^2_{s_2} + \eta^1_{s_2}\right). \end{aligned} \tag{6.11}$$

The relation (6.11) is obtained under the following approximations. First, clock skews γ_i are assumed constant over the duration of the transaction, where the transaction is in the order of a few milliseconds, so that during the transaction

$$\tau_i(t') - \tau_i(t) \approx \gamma_i(t' - t),$$

for any time instances t, t' and Transceiver i. Second, Δt^{21}, which like Δt^{32} is in the order of a few hundreds of microseconds, is much greater than $\frac{d}{c}$, and since clock skews are also small (in the order of a few parts-per-million [21]), then to first order $\gamma_i(\mathtt{R}^2 - \mathtt{T}^1) \approx \gamma_i \Delta t^{21}$. Third,

$$\frac{(1+\gamma_{f_1})\Delta t^{32} + \eta^3_{f_1} - \eta^2_{f_1}}{(1+\gamma_{s_2})\Delta t^{32} + \eta^3_{s_2} - \eta^2_{s_2}} \approx \frac{(1+\gamma_{f_1})}{(1+\gamma_{s_2})}$$

because the timestamping noise, in the order of a few hundred picoseconds at most, is much smaller than Δt^{32}. Finally,

$$\frac{(1+\gamma_{f_1})}{(1+\gamma_{s_2})}\left(\eta^2_{s_2} - \eta^1_{s_2}\right) \approx \eta^2_{s_2} - \eta^1_{s_2},$$

to first order, because the clock skews and timestamping noise are both small.

The **second pseudomeasurement** is a direct clock offset measurement between the

initiating and target transceivers, which from timestamps (6.1)-(6.6) can be written as

$$\begin{aligned} y^{\tau} &= \frac{1}{2}\left(\left(\tilde{\mathtt{R}}^2_{f_1}+\tilde{\mathtt{T}}^1_{f_1}\right)-\frac{\tilde{\mathtt{R}}^3_{f_1}-\tilde{\mathtt{R}}^2_{f_1}}{\tilde{\mathtt{T}}^3_{s_2}-\tilde{\mathtt{T}}^2_{s_2}}\left(\tilde{\mathtt{T}}^2_{s_2}-\tilde{\mathtt{R}}^1_{s_2}\right)-2\tilde{\mathtt{R}}^1_{s_2}\right) \\ &\approx \tau_{f_1f_0}-\tau_{s_2f_0}+\frac{1}{2}\left(\eta^2_{f_1}+\eta^1_{f_1}-\eta^2_{s_2}-\eta^1_{s_2}\right), \end{aligned} \tag{6.12}$$

using the fact that $\tau_{f_1}-\tau_{s_2}=\tau_{f_1f_0}-\tau_{s_2f_0}$. Here and in the following, clock offsets are evaluated at time $\mathtt{T}^1$, which is omitted from the notation. This model is somewhat similar to the measurement model proposed in [22], but involves an additional term to correct the effect of the clock skew on the measured offset. In fact, the form of the first two pseudomeasurements is chosen to cancel out the terms $\frac{1}{2}(1+\gamma_{f_1})\Delta t^{21}$ and $-\frac{1}{2}(1+\gamma_{s_2})\Delta t^{21}$ by multiplying the latter with $\frac{1+\gamma_{f_1}}{1+\gamma_{s_2}}$.

The **third pseudomeasurement** is associated with the first passive-listening timestamp, which is a function of the distance between the passive robot and the initiating robot, as well the clock offset between the two transceivers. Using timestamps (6.1) and (6.7) for $i\in\{f_0,s_0\}$, and $\tau_{f_0f_0}\triangleq 0$, this is written as

$$y^{\mathrm{p},1}_i=\tilde{\mathtt{P}}^{1,i}_i-\tilde{\mathtt{T}}^1_{f_1}=\frac{1}{c}d^{f_1i}+\tau_{if_0}-\tau_{f_1f_0}+\eta^1_i-\eta^1_{f_1}. \tag{6.13}$$

The **fourth pseudomeasurement** is similar to the third one, with an additional skew-correction component to model the passage of time Δt^{21} between the first and second messages in two clocks with different clock rates. Using timestamps (6.5) and (6.8) for $i\in\{f_0,s_0\}$, and $\gamma_{f_0f_0}\triangleq 0$, this is modelled as

$$\begin{aligned} y^{\mathrm{p},2}_i &= \tilde{\mathtt{P}}^{2,i}_i-\tilde{\mathtt{T}}^2_{s_2} \\ &= \frac{1}{c}d^{s_2i}+\tau_{if_0}-\tau_{s_2f_0}+(\gamma_{if_0}-\gamma_{s_2f_0})\Delta t^{21}+\eta^2_i-\eta^2_{s_2}. \end{aligned} \tag{6.14}$$

using the fact that $\gamma_i-\gamma_{s_2}=\gamma_{if_0}-\gamma_{s_2f_0}$. The exact delay Δt^{21} appearing in (6.14) is in fact unknown, as delay values are enforced by the transceivers in their own clocks. Nonetheless, to first order, the corresponding term can be replaced by

$$(\gamma_{if_0}-\gamma_{s_2f_0})\Delta t^{21}\approx(\gamma_{if_0}-\gamma_{s_2f_0})(\tilde{\mathtt{T}}^2_{s_2}-\tilde{\mathtt{R}}^1_{s_2}).$$

Lastly, the **fifth pseudomeasurement** is similar to the fourth pseudomeasurement, but modelling the evolution of the clocks over a longer time window Δt^{31}. Using timestamps

(6.6) and (6.9) for $i \in \{f_0, s_0\}$, this is modelled as

$$\begin{aligned} y_i^{\mathrm{p},3} &= \tilde{\mathtt{P}}_i^{3,i} - \tilde{\mathtt{T}}_{s_2}^3 \\ &= \frac{1}{c} d^{s_2 i} + \tau_{i f_0} - \tau_{s_2 f_0} + (\gamma_{i f_0} - \gamma_{s_2 f_0}) \Delta t^{31} + \eta_i^3 - \eta_{s_2}^3. \end{aligned} \tag{6.15}$$

As before, Δt^{31} is unknown, but to first order

$$(\gamma_{i f_0} - \gamma_{s_2 f_0}) \Delta t^{31} \approx (\gamma_{i f_0} - \gamma_{s_2 f_0})(\tilde{\mathtt{T}}_{s_2}^3 - \tilde{\mathtt{R}}_{s_2}^1).$$

Note the last three pseudomeasurements are per listening transceiver i, and therefore there are a total of 8 pseudomeasurements available at Robot 0 if it is not involved in the ranging transaction, or 5 pseudomeasurements if one of the transceivers on Robot 0 is active. The additional pseudomeasurements available at the listening transceivers results in a $(1 + 3n)$-fold increase in the total number of distinct measurements when considering a centralized approach where passive listening measurements from all robots are available, and a $(\frac{1}{2} + 2n)$-fold increase in the number of distinct measurements when considering the perspective of an individual robot that does not have access to passive listening measurements at other robots. For example, for 5 neighbouring robots, this results in a 16-fold and an 11.5-fold increase in the number of measurements, respectively. The former is purely due to passive listening measurements, while the latter is due to passive listening measurements as well as the ability to obtain direct ToF measurements between two neighbouring robots. The proof of this claim is given in Appendix B.

6.4.5 Pseudomeasurements' Covariance Matrix

Given that the pseudomeasurements are a function of the same measured timestamps, cross-correlations between the pseudomeasurements exist and must be correctly modelled in the filter. Computing the variance of the pseudomeasurements (6.11)-(6.15) is straightforward, and can be summarized as

$$\begin{aligned} \mathbb{E}\left[(y^{\mathrm{tof}} - \bar{y}^{\mathrm{tof}})^2\right] &= \sigma^2, \\ \mathbb{E}\left[(y^{\tau} - \bar{y}^{\tau})^2\right] &= \sigma^2, \\ \mathbb{E}\left[(y_i^{\mathrm{p,j}} - \bar{y}_i^{\mathrm{p,j}})^2\right] &= 2\sigma^2, \quad j \in \{1, 2, 3\}, \end{aligned}$$

where an overbar denotes a noise-free value. Meanwhile, the cross-correlation between the ToF and offset measurements can be computed as

$$\mathbb{E}\left[(y^{\text{tof}} - \bar{y}^{\text{tof}})(y^{\tau} - \bar{y}^{\tau})\right] = 0$$

as the noise values are of alternating signs. Lastly, the cross-correlations between the passive listening measurements and the ToF measurements can be shown to be

$$\begin{aligned}
\mathbb{E}\left[(y_i^{\text{p},1} - \bar{y}_i^{\text{p},1})(y^{\text{tof}} - \bar{y}^{\text{tof}})\right] &= \frac{1}{2}\sigma^2, \\
\mathbb{E}\left[(y_i^{\text{p},2} - \bar{y}_i^{\text{p},2})(y^{\text{tof}} - \bar{y}^{\text{tof}})\right] &= \frac{1}{2}\sigma^2, \\
\mathbb{E}\left[(y_i^{\text{p},3} - \bar{y}_i^{\text{p},3})(y^{\text{tof}} - \bar{y}^{\text{tof}})\right] &= 0,
\end{aligned}$$

while the cross-correlations with offset measurements are the same but with an opposite sign for the correlation with $y_i^{\text{p},2}$. Passive listening measurements of different transceivers are also correlated. Stacking all the pseudomeasurements into one column matrix gives the random measurement vector

$$\mathbf{y} = \begin{bmatrix} y^{\text{tof}} & y^{\tau} & y_{f_0}^{\text{p},1} & y_{f_0}^{\text{p},2} & y_{f_0}^{\text{p},3} & y_{s_0}^{\text{p},1} & y_{s_0}^{\text{p},2} & y_{s_0}^{\text{p},3} \end{bmatrix}^{\mathsf{T}}, \tag{6.16}$$

with mean $\bar{\mathbf{y}}$ and covariance matrix $\mathbf{R}$, where

$$\mathbf{R} = \begin{bmatrix} \sigma^2\mathbf{1}_2 & \frac{1}{2}\sigma^2\mathbf{B} & \frac{1}{2}\sigma^2\mathbf{B} \\ \frac{1}{2}\sigma^2\mathbf{B}^{\mathsf{T}} & 2\sigma^2\mathbf{1}_3 & \sigma^2\mathbf{1}_3 \\ \frac{1}{2}\sigma^2\mathbf{B}^{\mathsf{T}} & \sigma^2\mathbf{1}_3 & 2\sigma^2\mathbf{1}_3 \end{bmatrix},$$

and

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & -1 & 0 \end{bmatrix}.$$

The measurement vector $\mathbf{y}$ and its covariance $\mathbf{R}$ are used in the correction step of an on-manifold EKF, where they are fused with the process model derived in the next section.

6.5 The Process Model

To derive the process model, a Lie group referred to here as $DE_2(3)$ (D stands for *Delta*) with matrices of the form

$$\mathbf{U} = \begin{bmatrix} \mathbf{C} & \mathbf{v} & \mathbf{r} \\ & 1 & \Delta t \\ & & 1 \end{bmatrix} \in DE_2(3) \tag{6.17}$$

is introduced, where $\mathbf{C} \in SO(3)$, $\mathbf{v}, \mathbf{r} \in \mathbb{R}^3$, and $\Delta t \in \mathbb{R}$. The inverse of $\mathbf{U}$ in (6.17) is

$$\mathbf{U}^{-1} = \begin{bmatrix} \mathbf{C}^\mathsf{T} & -\mathbf{C}^\mathsf{T}\mathbf{v} & -\mathbf{C}^\mathsf{T}(\mathbf{r} - \Delta t \mathbf{v}) \\ & 1 & -\Delta t \\ & & 1 \end{bmatrix} \in DE_2(3).$$

Meanwhile, the matrix representation of the adjoint operator satisfying

$$\mathrm{Exp}(\mathrm{Ad}(\mathbf{U})\boldsymbol{\xi}) \triangleq \mathbf{U}\,\mathrm{Exp}(\boldsymbol{\xi})\mathbf{U}^{-1}, \quad \mathrm{Exp}(\boldsymbol{\xi}) \in SE_2(3)$$

is given by

$$\mathrm{Ad}(\mathbf{U}) = \begin{bmatrix} \mathbf{C} & \mathbf{0} & \mathbf{0} \\ \mathbf{v}^\times \mathbf{C} & \mathbf{C} & \mathbf{0} \\ -(\Delta t \mathbf{v} - \mathbf{r})^\times \mathbf{C} & -\Delta t \mathbf{C} & \mathbf{C} \end{bmatrix},$$

where, for $\mathbf{v} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^\mathsf{T} \in \mathbb{R}^3$,

$$\mathbf{v}^\times = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}.$$

Additionally, following the terminology in [71, Chapter 9], a *time machine* is a matrix $\mathbf{M}$ of the form

$$\mathbf{M} = \begin{bmatrix} \mathbf{1} & & \\ & 1 & \Delta t \\ & & 1 \end{bmatrix} \in \mathbb{R}^{5\times 5},$$

where $\Delta t \in \mathbb{R}$. This allows writing $\mathbf{U}$ in (6.17) as the product of two matrices,

$$\mathbf{U} = \underbrace{\begin{bmatrix} \mathbf{1} & & \\ & 1 & \Delta t \\ & & 1 \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} \mathbf{C} & \mathbf{v} & \mathbf{r} \\ & 1 & \\ & & 1 \end{bmatrix}}_{\mathbf{T} \in SE_2(3)}.$$

It can be shown that $\mathbf{M}$ is in itself an element of a Lie group closed under matrix multiplication.

This section first extends the results in [71, Chapter 9] to address relative extended pose states. The clock-state process model is then derived. These are then used alongside the ranging protocol presented in Section 6.4 in the CSRPE.

6.5.1 Deriving the Extended-Pose Process Model

Using the absolute-pose kinematic model given in Section 2.5, the on-manifold relative-pose kinematic model is first derived in continuous-time as a function of the IMU measurements. The process model for the relative attitude between Robot 0 and Robot i is

$$\dot{\mathbf{C}}_{0i} = \mathbf{C}_{0i} \left(\boldsymbol{\omega}_i^{i0}\right)^\times, \tag{6.18}$$

where $\boldsymbol{\omega}_i^{i0}$ is the angular velocity of Robot i's body frame relative to Robot 0's body frame, resolved in Robot i's body frame. However, the gyroscopes on Robots 0 and i measure $\boldsymbol{\omega}_0^{0a}$ and $\boldsymbol{\omega}_i^{ia}$, respectively. Therefore, (6.18) is rewritten as

$$\begin{aligned} \dot{\mathbf{C}}_{0i} &= \mathbf{C}_{0i} \left(\boldsymbol{\omega}_i^{ia} - \mathbf{C}_{0i}^\mathsf{T} \boldsymbol{\omega}_0^{0a}\right)^\times \\ &= -\mathbf{C}_{0i} \left(\mathbf{C}_{0i}^\mathsf{T} \boldsymbol{\omega}_0^{0a}\right)^\times + \mathbf{C}_{0i} \left(\boldsymbol{\omega}_i^{ia}\right)^\times \\ &= -\left(\boldsymbol{\omega}_0^{0a}\right)^\times \mathbf{C}_{0i} + \mathbf{C}_{0i} \left(\boldsymbol{\omega}_i^{ia}\right)^\times. \end{aligned} \tag{6.19}$$

Meanwhile, using the *transport theorem* from Section 2.5.2, the process model for the relative velocity of Robot i relative to Robot 0 is

$$\overset{0}{\underset{\rightarrow}{\dot{v}}}^{i0/a} = -\underset{\rightarrow}{\omega}^{0a} \times \underset{\rightarrow}{v}^{i0/a} + \underset{\rightarrow}{a}^{iw/a} - \underset{\rightarrow}{a}^{0w/a}, \tag{6.20}$$

where w is any point fixed to the reference frame a. Denoting the *specific forces* measured

by the accelerometers as

$$\underrightarrow{\alpha}^{0} \triangleq \underrightarrow{a}^{0w/a} - \underrightarrow{g}, \qquad \underrightarrow{\alpha}^{i} \triangleq \underrightarrow{a}^{iw/a} - \underrightarrow{g},$$

where $\underrightarrow{g}$ is the gravity vector, (6.20) can be written as

$${}^{0}\underrightarrow{\dot{v}}^{i0/a} = -\underrightarrow{\omega}^{0a} \times \underrightarrow{v}^{i0/a} + \underrightarrow{\alpha}^{i} - \underrightarrow{\alpha}^{0}. \tag{6.21}$$

Similarly, the transport theorem gives the following process model for the position of Robot i relative to Robot 0

$${}^{0}\underrightarrow{\dot{r}}^{i0} = -\underrightarrow{\omega}^{0a} \times \underrightarrow{r}^{i0} + \underrightarrow{v}^{i0/a}. \tag{6.22}$$

Lastly, resolving (6.21) and (6.22) in the body frame of Robot 0 and writing these equations as a function of the accelerometer-measured quantities $\boldsymbol{\alpha}_0^0$ and $\boldsymbol{\alpha}_i^i$ yields

$${}^{0}\dot{\mathbf{v}}_0^{i0/a} = -\left(\boldsymbol{\omega}_0^{0a}\right)^{\times} \mathbf{v}_0^{i0/a} + \mathbf{C}_{0i}\boldsymbol{\alpha}_i^i - \boldsymbol{\alpha}_0^0, \tag{6.23}$$

$${}^{0}\dot{\mathbf{r}}_0^{i0} = -\left(\boldsymbol{\omega}_0^{0a}\right)^{\times} \mathbf{r}_0^{i0} + \mathbf{v}_0^{i0/a}. \tag{6.24}$$

Combining (6.19), (6.23), and (6.24), the extended relative-pose process model for Robot i can be written compactly as

$$\begin{aligned}
\dot{\mathbf{T}}_{0i} &= \begin{bmatrix} \dot{\mathbf{C}}_{0i} & {}^{0}\dot{\mathbf{v}}_0^{i0/a} & {}^{0}\dot{\mathbf{r}}_0^{i0} \\ & 0 & \\ & & 0 \end{bmatrix} \\
&= -\begin{bmatrix} \left(\boldsymbol{\omega}_0^{0a}\right)^{\times} & \boldsymbol{\alpha}_0^0 & \\ & & 1 \\ & & 0 \end{bmatrix} \mathbf{T}_{0i} + \mathbf{T}_{0i} \begin{bmatrix} \left(\boldsymbol{\omega}_i^{ia}\right)^{\times} & \boldsymbol{\alpha}_i^i & \\ & & 1 \\ & & 0 \end{bmatrix} \\
&\triangleq -\tilde{\mathbf{U}}_0\mathbf{T}_{0i} + \mathbf{T}_{0i}\tilde{\mathbf{U}}_i,
\end{aligned} \tag{6.25}$$

with the matrices $\tilde{\mathbf{U}}_0$ and $\tilde{\mathbf{U}}_i$ containing the IMU measurements for Robot 0 and Robot i, respectively.

6.5.2 Discrete-Time Extended-Pose Process Model

In order to discretize (6.25), the common assumption is made that accelerations and angular velocities are constant between IMU measurements, which is justified by the fact that IMU

measurements typically occur at a high frequency (∼100-1000 Hz). Consequently, since (6.25) is a *differential Sylvester equation*, and setting the initial condition to be $\mathbf{T}_{0i,k}$ at time-step k, a closed-form solution exists of the form [98]

$$\mathbf{T}_{0i,k+1} = \underbrace{\exp(\tilde{\mathbf{U}}_{0,k}\Delta t)^{-1}}_{\mathbf{U}_{0,k}^{-1}}\mathbf{T}_{0i,k}\underbrace{\exp(\tilde{\mathbf{U}}_{i,k}\Delta t)}_{\mathbf{U}_{i,k}}, \tag{6.26}$$

where Δt is the time interval between the IMU measurements at time-steps k and $k+1$.

Following a similar derivation as in [71, Chapter 9], expanding the matrix exponential is shown in Appendix D to yield a closed-form matrix of the form

$$\mathbf{U}_{0,k} = \begin{bmatrix} \exp(\boldsymbol{\Omega}_{0,k}) & \Delta t\mathbf{J}_l\left(\boldsymbol{\Omega}_{0,k}\right)\boldsymbol{\alpha}_{0,k}^0 & \frac{\Delta t^2}{2}\mathbf{N}\left(\boldsymbol{\Omega}_{0,k}\right)\boldsymbol{\alpha}_{0,k}^0 \\ & 1 & \Delta t \\ & & 1 \end{bmatrix}$$

where $\boldsymbol{\Omega}_{0,k} \triangleq \left(\boldsymbol{\omega}_{0,k}^{0a}\right)^\times \Delta t$ and $\mathbf{J}_l$ is the left Jacobian of $SO(3)$. Both $\mathbf{J}_l$ and $\mathbf{N}$ are defined in Appendix D. Note that $\mathbf{U}_{0,k}$ is an element of the aforementioned Lie group $DE_2(3)$. Similarly, $\mathbf{U}_{i,k} \in DE_2(3)$ is of the same form as $\mathbf{U}_{0,k}$ with the inputs being that of neighbouring Robot i instead.

6.5.3 Linearizing the Extended-Pose Process Model

To perform uncertainty propagation computations for the extended-pose states, the process model is now linearized. Throughout this chapter, the state is perturbed on the left, as it yields simpler Jacobians. Nonetheless, a similar derivation can be done by perturbing the state on the right.

Perturbing (6.26) with respect to the state yields

$$\begin{aligned} \mathrm{Exp}(\delta\boldsymbol{\xi}_{0i,k+1})\bar{\mathbf{T}}_{0i,k+1} &= \bar{\mathbf{U}}_{0,k}^{-1}\,\mathrm{Exp}(\delta\boldsymbol{\xi}_{0i,k})\bar{\mathbf{T}}_{0i,k}\bar{\mathbf{U}}_{i,k} \\ &= \mathrm{Exp}(\mathrm{Ad}(\bar{\mathbf{U}}_{0,k}^{-1})\delta\boldsymbol{\xi}_{0i,k})\bar{\mathbf{U}}_{0,k}^{-1}\bar{\mathbf{T}}_{0i,k}\bar{\mathbf{U}}_{i,k}. \end{aligned}$$

Cancelling out nominal terms and taking the $\mathrm{Log}(\cdot)$ of both sides results in the linearized model

$$\delta\boldsymbol{\xi}_{0i,k+1} = \mathrm{Ad}(\bar{\mathbf{U}}_{0,k}^{-1})\delta\boldsymbol{\xi}_{0i,k}. \tag{6.27}$$

To perturb (6.26) with respect to the input noise, the aforementioned concept of time

machines is used. The input matrix $\mathbf{U}_{0,k}$ can be written as

$$\begin{aligned}
\mathbf{U}_{0,k} &= \mathbf{M}\begin{bmatrix} \exp(\boldsymbol{\Omega}_{0,k}) & \Delta t \mathbf{J}_l\left(\boldsymbol{\Omega}_{0,k}\right)\boldsymbol{\alpha}_{0,k}^0 & \frac{\Delta t^2}{2}\mathbf{N}\left(\boldsymbol{\Omega}_{0,k}\right)\boldsymbol{\alpha}_{0,k}^0 \\ & 1 & \\ & & 1 \end{bmatrix} \\
&= \mathbf{M}\operatorname{Exp}\left(\begin{bmatrix} \boldsymbol{\omega}_{0,k}^{0a}\Delta t \\ \boldsymbol{\alpha}_{0,k}^0\Delta t \\ \frac{\Delta t^2}{2}\mathbf{J}_l\left(\boldsymbol{\Omega}_{0,k}\right)^{-1}\mathbf{N}\left(\boldsymbol{\Omega}_{0,k}\right)\boldsymbol{\alpha}_{0,k}^0 \end{bmatrix}\right) \\
&= \mathbf{M}\operatorname{Exp}\Bigg(\underbrace{\begin{bmatrix} \Delta t\mathbf{1} & & \\ & \Delta t\mathbf{1} & \\ & \frac{\Delta t^2}{2}\mathbf{J}_l\left(\boldsymbol{\Omega}_{0,k}\right)^{-1}\mathbf{N}\left(\boldsymbol{\Omega}_{0,k}\right) \end{bmatrix}}_{\mathbf{V}_{0,k}}\underbrace{\begin{bmatrix} \boldsymbol{\omega}_{0,k}^{0a} \\ \boldsymbol{\alpha}_{0,k}^0 \end{bmatrix}}_{\mathbf{u}_{0,k}}\Bigg) \\
&\triangleq \mathbf{M}\operatorname{Exp}(\mathbf{V}_{0,k}\mathbf{u}_{0,k}),
\end{aligned} \tag{6.28}$$

where $\mathbf{u}_{0,k} \in \mathbb{R}^6$ is Robot 0's IMU measurements or *input* at time-step k. Taking the perturbation of (6.28) with respect to the input yields

$$\begin{aligned}
\mathbf{U}_{0,k} &\approx \mathbf{M}\operatorname{Exp}(\bar{\mathbf{V}}_{0,k}(\bar{\mathbf{u}}_{0,k} + \delta\mathbf{u}_{0,k})) \\
&\approx \mathbf{M}\operatorname{Exp}(\bar{\mathbf{V}}_{0,k}\bar{\mathbf{u}}_{0,k})\operatorname{Exp}(\boldsymbol{\mathcal{J}}_l(-\bar{\mathbf{V}}_{0,k}\bar{\mathbf{u}}_{0,k})\bar{\mathbf{V}}_{0,k}\delta\mathbf{u}_{0,k}) \\
&= \bar{\mathbf{U}}_{0,k}\operatorname{Exp}(\boldsymbol{\mathcal{J}}_l(-\bar{\mathbf{V}}_{0,k}\bar{\mathbf{u}}_{0,k})\bar{\mathbf{V}}_{0,k}\delta\mathbf{u}_{0,k}) \\
&\triangleq \bar{\mathbf{U}}_{0,k}\operatorname{Exp}(\mathbf{L}_{0,k}\delta\mathbf{u}_{0,k})
\end{aligned} \tag{6.29}$$

where input noise perturbations in $\mathbf{V}_{0,k}$ are neglected as the term $\frac{\Delta t^2}{2}\mathbf{J}_l\left(\boldsymbol{\Omega}_{0,k}\right)^{-1}\mathbf{N}\left(\boldsymbol{\Omega}_{0,k}\right)$ is small when the measurements are obtained using a high-rate IMU, $\mathbf{L}_{0,k} \triangleq \boldsymbol{\mathcal{J}}_l(-\bar{\mathbf{V}}_{0,k}\bar{\mathbf{u}}_{0,k})\bar{\mathbf{V}}_{0,k}$, and $\boldsymbol{\mathcal{J}}_l(\cdot)$ is the left Jacobian of $SE_2(3)$ [96, Eq. (94)]. Similarly,

$$\mathbf{U}_{i,k} = \mathbf{M}\operatorname{Exp}(\mathbf{V}_{i,k}\mathbf{u}_{i,k}) \approx \bar{\mathbf{U}}_{i,k}\operatorname{Exp}(\mathbf{L}_{i,k}\delta\mathbf{u}_{i,k}). \tag{6.30}$$

Therefore, left-perturbing the state process model (6.26) with respect to the input noise yields

$$\begin{aligned}
\operatorname{Exp}(\delta\boldsymbol{\xi}_{0i,k+1})\bar{\mathbf{T}}_{0i,k+1} &= \operatorname{Exp}(-\mathbf{L}_{0,k}\delta\mathbf{u}_{0,k})\bar{\mathbf{U}}_{0,k}^{-1}\bar{\mathbf{T}}_{0i,k}\bar{\mathbf{U}}_{i,k}\operatorname{Exp}(\mathbf{L}_{i,k}\delta\mathbf{u}_{i,k}) \\
&= \operatorname{Exp}(-\mathbf{L}_{0,k}\delta\mathbf{u}_{0,k})\operatorname{Exp}(\operatorname{Ad}(\bar{\mathbf{T}}_{0i,k+1})\mathbf{L}_{i,k}\delta\mathbf{u}_{i,k})\bar{\mathbf{T}}_{0i,k+1},
\end{aligned}$$

which can then be simplified to give

$$\delta\boldsymbol{\xi}_{0i,k+1} = -\mathbf{L}_{0,k}\delta\mathbf{u}_{0,k} + \mathrm{Ad}(\bar{\mathbf{T}}_{0i,k+1})\mathbf{L}_{i,k}\delta\mathbf{u}_{i,k}. \tag{6.31}$$

It is worth mentioning that cross-correlations develop between relative pose states for all neighbours, because the noisy IMU measurements of Robot 0 are used to propagate all the relative pose states. These cross-correlations can be tracked using the models (6.27) and (6.31).

6.5.4 Discrete-Time Clock-State Process Model

The state dynamics for every clock is modelled as in (2.10). Nonetheless, the clock states relative to real-time are unknown and unobservable. Therefore, clocks are modelled relative to clock f_0, thus giving dynamics of the form

$$\dot{\mathbf{c}}_{if_0} = \mathbf{A}\mathbf{c}_{if_0} + \begin{bmatrix} -\mathbf{1} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{f_0} \\ \mathbf{w}_i \end{bmatrix} \tag{6.32}$$

for $i \in \mathcal{C}\backslash\{f_0\}$. Discretizing (6.32) yields [8, Chapter 4.7]

$$\mathbf{c}_{if_0,k+1} = \mathbf{A}^{\mathrm{d}}\,\mathbf{c}_{if_0,k} + \mathbf{w}_{if_0,k}, \tag{6.33}$$

where

$$\mathbf{A}^{\mathrm{d}} = \exp(\mathbf{A}\Delta t) = \begin{bmatrix} 1 & \Delta t \\ & 1 \end{bmatrix},$$

$\mathbf{w}_{if_0,k} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}^{\mathrm{d}}\right)$, and

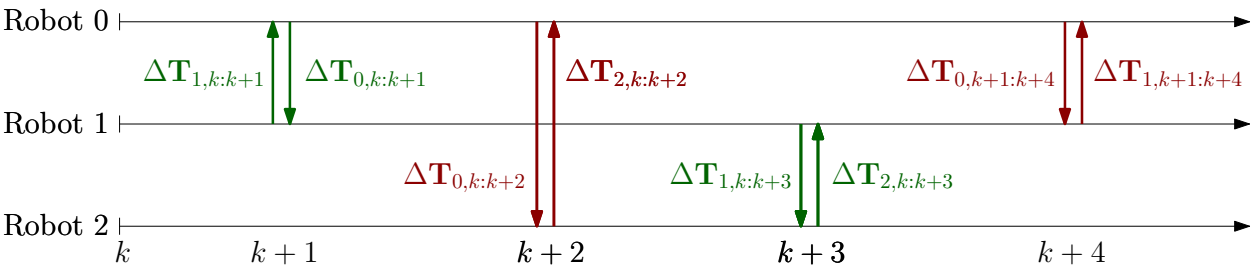
$$\mathbf{Q}^{\mathrm{d}} = 2\begin{bmatrix} \Delta t\mathcal{Q}^{\tau} + \frac{1}{3}\Delta t^3\mathcal{Q}^{\gamma} & \frac{1}{2}\Delta t^2\mathcal{Q}^{\gamma} \\ \frac{1}{2}\Delta t^2\mathcal{Q}^{\gamma} & \Delta t\mathcal{Q}^{\gamma} \end{bmatrix}.$$

Since the same noise $\mathbf{w}_{f_0}$ appears in (6.32) for all $i \in \mathcal{C}\backslash\{f_0\}$, the process noise vectors $\mathbf{w}_{if_0,k}$ in (6.33) are jointly Gaussian and correlated, and one can show that their cross-covariance is

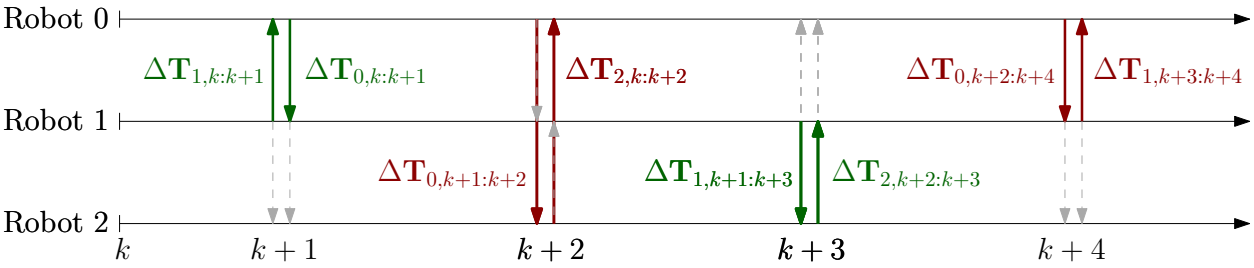
$$\mathbb{E}\left[\mathbf{w}_{if_0,k}\,\mathbf{w}_{jf_0,k}^{\mathsf{T}}\right] = \frac{1}{2}\mathbf{Q}^{\mathrm{d}}, \qquad \forall i, j \in \mathcal{C}\backslash\{f_0\}, i \neq j.$$

6.6 Relative Pose State Preintegration

6.6.1 Need for Preintegration



(a) RMI communication without passive listening.



(b) RMI communication with passive listening, where passive listening messages are shown using the grey arrows.

Figure 6.5: Communicated RMIs with and without passive listening over a window of 4 ranging transactions, where $\Delta\mathbf{T}_{i,\ell:m}$ is the RMI associated with the IMU measurements of Robot i from time-step ℓ to time-step $m-1$.

When considering Robot 0's perspective, the estimated relative-pose state is updated using (6.26) and the corresponding error covariance matrix using (6.27) and (6.31). Therefore, Robot 0 needs the IMU measurements of neighbouring robots at every time-step in order to update its estimated state of its neighbours. This is limiting since robots cannot broadcast their IMU measurements at the same rate as they are recorded due to the possibility of message collision if multiple robots attempt to broadcast at the same time. Additionally, to allow DS-TWR transactions to occur at the highest rate possible, the IMU information should ideally be transmitted using the ranging messages presented in Section 6.4.

In this section, the concept of preintegration is proposed to compactly encode the IMU measurements of a neighbouring Robot i over a window between two consecutive ranging instances using one *relative motion increment* (RMI), which is then sent over when Robot i ranges with one of its neighbours. However, as illustrated in Figure 6.5a, without passive listening the RMIs of Robot i become available to Robot 0 only when Robot 0 and Robot i communicate. Given that RMIs are computed iteratively as new IMU measurements arrive,

each robot needs to keep track of one RMI per neighbour. For example, looking at Figure 6.5a at time-step $k+3$, Robot 1 would be communicating the RMI of IMU measurements in the window k to $k+3$ to Robot 2, while also tracking a separate RMI for the window starting at $k+1$ to be sent to Robot 0 at time-step $k+4$.

On the other hand, passive listening over UWB lets two actively ranging robots broadcast their RMIs to all other robots, as shown in Figure 6.5b. This has the advantage that IMU information of neighbours becomes available faster at all robots, and the robots computing RMIs only need to track one RMI at all times since all neighbours are up-to-date with the most recently communicated RMI.

6.6.2 Relative Motion Increments

Consider the case where Robot i is an active robot only at non-adjacent time-steps ℓ and m. From (6.26), the relative pose state at time-step m can be computed from the relative pose state at time-step ℓ as

$$\mathbf{T}_{0i,m} = \left(\prod_{k=\ell}^{m-1} \mathbf{U}_{0,k}\right)^{-1} \mathbf{T}_{0i,\ell} \prod_{k=\ell}^{m-1} \mathbf{U}_{i,k}. \tag{6.34}$$

The inputs of Robot 0 are available at Robot 0 as soon as the measurements occur, therefore the first term of (6.26) can be computed directly at every time-step. On the other hand, the inputs of Robot i from time-step ℓ to $m-1$ will only be available when the robot actively shares it at time-step m. Rather than sharing the individual IMU measurements, Robot i can simply send

$$\Delta\mathbf{T}_{i,\ell:m} = \prod_{k=\ell}^{m-1} \mathbf{U}_{i,k} \in DE_2(3),$$

which is an RMI of the inputs of Robot i in the window ℓ to m. The process model representing time-propagation between non-adjacent time-steps can then be rewritten as

$$\mathbf{T}_{0i,m} = \left(\prod_{k=\ell}^{m-1} \mathbf{U}_{0,k}\right)^{-1} \mathbf{T}_{0i,\ell} \Delta\mathbf{T}_{i,\ell:m}. \tag{6.35}$$

This is a feature of the process model (6.34) being reliant on the inputs of Robot i in a separable way, meaning that the inputs of Robot i can simply be post-multiplied in (6.34). Robot i computes its RMI iteratively, starting with $\Delta\mathbf{T}_{i,\ell:\ell} = \mathbf{1}$, and updating it when a new

input measurement arrives as

$$\Delta\mathbf{T}_{i,\ell:k+1} = \Delta\mathbf{T}_{i,\ell:k}\mathbf{U}_{i,k}. \tag{6.36}$$

In order to linearize the RMI to be used in an EKF, a perturbation of the form

$$\Delta\mathbf{T}_{i,\ell:m} = \Delta\bar{\mathbf{T}}_{i,\ell:m}\,\mathrm{Exp}(\delta\mathbf{w}_{i,\ell:m})$$

is defined, where $\delta\mathbf{w}_{i,\ell m} \in \mathbb{R}^9$ is some unknown noise parameter associated with the RMI, which is a consequence of the noise associated with every input measurement. Despite $\Delta\mathbf{T}_{i,\ell:m}$ being an element of $DE_2(3)$, the above Exp is the exponential operator in $SE_2(3)$. Additionally, a right-perturbation is chosen to match the perturbation on $\mathbf{U}$ derived in (6.30), which simplifies the subsequent derivation, but a left-perturbation could also have been chosen.
Perturbing (6.36) with respect to the RMI itself then yields

$$\begin{aligned}\Delta\bar{\mathbf{T}}_{i,\ell:k+1}\,\mathrm{Exp}(\delta\mathbf{w}_{i,\ell:k+1}) &= \Delta\bar{\mathbf{T}}_{i,\ell:k}\,\mathrm{Exp}(\delta\mathbf{w}_{i,\ell:k})\bar{\mathbf{U}}_{i,k}\\ &= \Delta\bar{\mathbf{T}}_{i,\ell:k}\bar{\mathbf{U}}_{i,k}\,\mathrm{Exp}(\mathrm{Ad}(\bar{\mathbf{U}}_{i,k}^{-1})\delta\mathbf{w}_{i,\ell:k}),\end{aligned}$$

which can be simplified to give

$$\delta\mathbf{w}_{i,\ell:k+1} = \mathrm{Ad}(\bar{\mathbf{U}}_{i,k}^{-1})\delta\mathbf{w}_{i,\ell:k}. \tag{6.37}$$

Meanwhile, perturbing the RMI relative to the input noise using (6.30) yields

$$\Delta\bar{\mathbf{T}}_{i,\ell:k+1}\,\mathrm{Exp}(\delta\mathbf{w}_{i,\ell:k+1}) = \Delta\bar{\mathbf{T}}_{i,\ell:k}\bar{\mathbf{U}}_{i,k}\,\mathrm{Exp}(\mathbf{L}_{i,k}\delta\mathbf{u}_{i,k}),$$

which can also be simplified to give

$$\delta\mathbf{w}_{i,\ell:k+1} = \mathbf{L}_{i,k}\delta\mathbf{u}_{i,k}. \tag{6.38}$$

6.6.3 An Asynchronous-Input Filter

Taking advantage of the separability of the process model in the neighbour's input measurements, an asynchronous-input filter can be designed. The key idea here is to use two process models, one of the form

$$\boldsymbol{\mathcal{T}}_{0i,k+1} = \mathbf{U}_{0,k}^{-1}\mathbf{T}_{0i,k}, \qquad \boldsymbol{\mathcal{T}}_{0i,k+1} \in DE_2(3) \tag{6.39}$$

at $\ell < k < m-1$ when there is no input information from Robot i, and another of the form

$$\mathbf{T}_{0i,m} = \mathbf{U}_{0,m-1}^{-1} \boldsymbol{\mathcal{T}}_{0i,m-1} \Delta\mathbf{T}_{i,\ell:m} \tag{6.40}$$

when propagating from $k = m-1$ to m as Robot i communicates the RMI $\Delta\mathbf{T}_{i,\ell:m}$. Note that $\boldsymbol{\mathcal{T}}$ denotes an intermediate state estimate that is not an element of $SE_2(3)$. Only when the IMU measurements of the neighbouring robot are incorporated does the estimated state restore its original $SE_2(3)$ form.

Given that (6.39) is of the same form as (6.26) with $\mathbf{U}_{i,k} = \mathbf{1}$, linearization is straightforward and follows Section 6.5.3,

$$\delta\boldsymbol{\xi}_{0i,k+1} = \mathrm{Ad}(\bar{\mathbf{U}}_{0,k}^{-1})\delta\boldsymbol{\xi}_{0i,k} - \mathbf{L}_{0,k}\delta\mathbf{u}_{0,k}. \tag{6.41}$$

Similarly, (6.40) is of the same form as (6.26) with $\mathbf{U}_{i,k} = \Delta\mathbf{T}_{i,\ell:m}$, so the linearization with respect to the state is the same as (6.41), giving

$$\delta\boldsymbol{\xi}_{0i,m} = \mathrm{Ad}(\bar{\mathbf{U}}_{0,m-1}^{-1})\delta\boldsymbol{\xi}_{0i,m-1} - \mathbf{L}_{0,m-1}\delta\mathbf{u}_{0,m-1} + \mathrm{Ad}(\bar{\mathbf{T}}_{0i,m})\delta\mathbf{w}_{i,\ell:m}. \tag{6.42}$$

A summary of the proposed on-manifold EKF is shown in Algorithm 2.

Algorithm 2 Algorithm for one time-step of the proposed on-manifold EKF running on Robot 0.

The following is the pseudocode for Robot 0's EKF at time-step k. Let ℓ_p denote the last time Robot $p \in \{0, \dots, n\}$ communicated with one of its neighbours. Therefore, at time-step $k-1$, Robot 0 has the RMI $\Delta\mathbf{T}_{0,l_0:k-1}$, an intermediate estimate of neighbouring robots' relative poses, $\hat{\boldsymbol{\mathcal{T}}}_{0q,k-1}, q \in \{1, \dots, n\}$, as well as an estimate of the relative clock states. Robot 0 additionally gets an IMU measurement, allowing it to compute $\mathbf{U}_{0,k-1}$. The EKF is then as follows.

Propagate RMI using $\Delta\mathbf{T}_{0,l_0:k} = \Delta\mathbf{T}_{0,l_0:k-1}\mathbf{U}_{0,k-1}$ and its covariance using (6.37), (6.38).
if ranging with neighbour i **then**
  Communicate $\Delta\mathbf{T}_{0,l_0:k}$ and its covariance.
  Generate 5 pseudomeasurements using (6.11)-(6.15).
  Propagate the relative pose state estimates in time using

$$\begin{aligned}\check{\boldsymbol{\mathcal{T}}}_{0p,k} &= \mathbf{U}_{0,k-1}^{-1}\hat{\boldsymbol{\mathcal{T}}}_{0p,k-1}, \qquad p \in \{1, \dots, n\}, p \neq i,\\ \check{\mathbf{T}}_{0i,k} &= \mathbf{U}_{0,k-1}^{-1}\hat{\boldsymbol{\mathcal{T}}}_{0i,k-1}\Delta\mathbf{T}_{i,\ell_i:k},\end{aligned}$$

  and the clock state estimates using Section 6.5.4.
  Propagate the state-error covariances using (6.41), (6.42) and Section 6.5.4.
  Do on-manifold EKF correction step using pseudomeasurements to get $\hat{\boldsymbol{\mathcal{T}}}_{0p,k}$ & $\hat{\mathbf{T}}_{0i,k}$.
  Initiate a new RMI $\Delta\mathbf{T}_{0,k:k} = \mathbf{1}$ with covariance $\mathbf{0}$.
else if neighbours i and j are ranging **then**
  Generate 8 pseudomeasurements using (6.11)-(6.15).
  Propagate the relative pose state estimates in time using

$$\begin{aligned}\check{\boldsymbol{\mathcal{T}}}_{0p,k} &= \mathbf{U}_{0,k-1}^{-1}\hat{\boldsymbol{\mathcal{T}}}_{0p,k-1}, \quad p \in \{1, \dots, n\}, p \neq i, j,\\ \check{\mathbf{T}}_{0i,k} &= \mathbf{U}_{0,k-1}^{-1}\hat{\boldsymbol{\mathcal{T}}}_{0i,k-1}\Delta\mathbf{T}_{i,\ell_i:k},\\ \check{\mathbf{T}}_{0j,k} &= \mathbf{U}_{0,k-1}^{-1}\hat{\boldsymbol{\mathcal{T}}}_{0j,k-1}\Delta\mathbf{T}_{j,\ell_j:k},\end{aligned}$$

  and the clock state estimates using Section 6.5.4.
  Propagate the state-error covariances using (6.41), (6.42) and Section 6.5.4.
  Do on-manifold EKF correction step using pseudomeasurements to get $\hat{\boldsymbol{\mathcal{T}}}_{0p,k}$, $\hat{\mathbf{T}}_{0i,k}$, & $\hat{\mathbf{T}}_{0j,k}$.
else if no one is ranging **then**
  Propagate the relative pose state estimates in time using

$$\check{\boldsymbol{\mathcal{T}}}_{0p,k} = \mathbf{U}_{0,k-1}^{-1}\hat{\boldsymbol{\mathcal{T}}}_{0p,k-1}, \qquad p \in \{1, \dots, n\},$$

  and the clock state estimates using Section 6.5.4.
  Propagate the state-error covariances using (6.41) and Section 6.5.4.
end if

6.6.4 Equivalence to the No Communication Constraint Case

In the absence of communication constraint, each robot would have access to all of its neighbours' IMU measurements at all times. As explained in Section 6.6.1, this is not possible, so that preintegration is needed. It is shown in (6.35) that the state can be propagated using RMIs in a manner equivalent to the case with no communication constraint. In this subsection, it is shown that computing the uncertainty propagation for the state is also equivalent in both cases, despite the Jacobians used being different. This is in fact a consequence of the structure of the Jacobians when perturbing the state from the left.

No Communication Constraints

When there are no communication constraints and IMU measurements of neighbours are available at all times, the models shown in Section 6.5 can be used to propagate the state. The covariance of the state is propagated using (6.27) and (6.31), which for two nonadjacent timestamps ℓ and m would be written as

$$\begin{aligned}\delta\boldsymbol{\xi}_{0i,m} = \operatorname{Ad}(\Delta\mathbf{T}_{0,\ell:m})^{-1}\delta\boldsymbol{\xi}_{0i,\ell} - \sum_{k=\ell}^{m-1}\operatorname{Ad}(\Delta\mathbf{T}_{0,k+1:m})^{-1}\mathbf{L}_{0,k}\delta\mathbf{u}_{0,k} \\ + \sum_{k=\ell}^{m-1}\operatorname{Ad}(\Delta\mathbf{T}_{0,k+1:m}^{-1}\bar{\mathbf{T}}_{0i,k+1})\mathbf{L}_{i,k}\delta\mathbf{u}_{i,k}.\end{aligned} \tag{6.43}$$

With Preintegration

First, the uncertainty of the RMI can be computed using (6.37) and (6.38) as

$$\delta\mathbf{w}_{i,\ell:m} = \sum_{k=\ell}^{m-1}\operatorname{Ad}(\Delta\mathbf{T}_{i,k+1:m})^{-1}\mathbf{L}_{i,k}\delta\mathbf{u}_{i,k}.$$

Note that the RMI gets communicated at time-step m, so from time-step ℓ to $m-1$ the state propagation occurs only with the IMU measurements of Robot 0 as shown in (6.39). The uncertainty propagation from timestamp ℓ to $m-1$ then follows as per (6.41), which can be written as

$$\delta\boldsymbol{\xi}_{0i,m-1} = \operatorname{Ad}(\Delta\mathbf{T}_{0,\ell:m-1})^{-1}\delta\boldsymbol{\xi}_{0i,\ell} - \sum_{k=\ell}^{m-2}\operatorname{Ad}(\Delta\mathbf{T}_{0,k+1:m-1})^{-1}\mathbf{L}_{0,k}\delta\mathbf{u}_{0,k}.$$

Meanwhile, propagating the uncertainty from timestamp $m-1$ to m using the RMI as shown

in (6.40) then follows as per (6.42) to give

$$\begin{aligned}\delta\boldsymbol{\xi}_{0i,m} &= \mathrm{Ad}(\Delta\mathbf{T}_{0,\ell:m})^{-1}\delta\boldsymbol{\xi}_{0i,\ell} - \sum_{k=\ell}^{m-1}\mathrm{Ad}(\Delta\mathbf{T}_{0,k+1:m})^{-1}\mathbf{L}_{0,k}\delta\mathbf{u}_{0,k} \\ &\qquad + \mathrm{Ad}(\Delta\mathbf{T}_{0,\ell:m}^{-1}\bar{\mathbf{T}}_{0i,\ell}\Delta\mathbf{T}_{i,\ell:m})\delta\mathbf{w}_{i,\ell:m} \\ &= \mathrm{Ad}(\Delta\mathbf{T}_{0,\ell:m})^{-1}\delta\boldsymbol{\xi}_{0i,\ell} - \sum_{k=\ell}^{m-1}\mathrm{Ad}(\Delta\mathbf{T}_{0,k+1:m})^{-1}\mathbf{L}_{0,k}\delta\mathbf{u}_{0,k} \\ &\qquad + \sum_{k=\ell}^{m-1}\mathrm{Ad}(\Delta\mathbf{T}_{0,k+1:m}^{-1}\Delta\mathbf{T}_{0,\ell:k+1}^{-1}\bar{\mathbf{T}}_{0i,\ell}\Delta\mathbf{T}_{i,\ell:k+1})\mathbf{L}_{i,k}\delta\mathbf{u}_{i,k},\end{aligned}$$

which, using (6.35), simplifies to be exactly equal to (6.43).

6.6.5 Communication Requirements

The proposed multi-robot preintegration approach provides an alternative efficient way of communicating odometry information as compared to communicating the individual IMU measurements. When sending IMU measurements, no covariance information is required as the covariance matrix is typically a fixed value that can be assumed common among all robots if they all share the same kind of IMU. Meanwhile, when sending an RMI, the components of a corresponding 9×9 positive-definite symmetric matrix representing its computed uncertainty must also be sent, as this is not constant but rather a function of the individual inputs.

Each IMU measurement consists of 6 single-precision floats, 3 for the gyroscope and 3 for the accelerometer readings, for a total of 24 bytes. Meanwhile, each RMI can be represented using 10 single-precision floats and the corresponding covariance matrix using the upper triangular part of the 9×9 matrix, which requires communicating an additional 45 single-precision floats. Therefore, sending one RMI and its covariance matrix requires over 220 bytes of information. Therefore, unless an RMI replaces more than 9 IMU measurements, it is sometimes more efficient to communicate the raw IMU measurements. Nonetheless, using the proposed multi-robot preintegration framework has the following advantages (in addition to the discussion in Section 6.6.1).

- It overcomes the need for variable amount of communication, as the RMI and its covariance matrix are of fixed length but a varying number of IMU readings might be accumulated in between two instances of a robot ranging. This consequently eases implementation and provides a more reliable system.

Table 6.1: Simulation parameters based on the ICM-20689 IMU and the DWM1000 UWB transceiver.

Specification	**Value**
Accelerometer std. dev. [m/s^2]	0.023
Gyroscope std. dev. [rad/s]	0.0066
IMU rate [Hz]	250
UWB timestamping std. dev. [ns]	0.33
UWB rate [Hz]	125
Clock offset PSD [ns^2/Hz]	0.4
Clock skew PSD [ppb^2/Hz]	640

- It provides robustness to loss of communication, as a robot re-establishing communication with its neighbours after a few seconds would not be able to send over all the accumulated IMU information.
- It reduces the amount of processing required at neighbours, as the input matrices $\mathbf{U}_{i,k}$ are pre-multiplied at Robot i on behalf of all its neighbours.
- It overcomes the need to know the noise distribution of the neighbours' IMUs, which would be useful if not all robots have the same IMU.
- It allows easy implementation with IMU-bias estimators and approaches that dynamically tune the covariance of the IMU measurements, without needing to send the bias estimates or the tuned covariances over UWB.

Additionally, UWB protocols by default allow 128 bytes of information to be sent per message transmission [77], for a total of 256 bytes per transceiver in each TWR instance. Given that each transceiver only needs to send 2 bytes of frame-control data per signal (thus 4 bytes of frame-control data in total) [77] and a total of 3 single-precision timestamps (thus 12 bytes of timestamps), there is enough room for the 220 bytes required to send an RMI. Note that if more information is required, some modules such as DWM1000 allow up to 1024 bytes of data per message transmission [7].

6.7 Simulation Results

To evaluate the benefits of using passive listening on the estimation accuracy of relative pose states, the clock dynamics and quadcopter kinematics have been simulated. The clocks' evolution is modelled relative to a "global time" using the simulating computer's own clock, while the absolute-state quadcopter kinematics are simulated relative to some inertial frame. Noisy

IMU and timestamp measurements are then modelled and fed into the CSRPE algorithm to estimate the relative clock and pose states.
To evaluate the proposed approach, 3 datasets are simulated.

1. **S1**: A single run with 4 quadcopters,
2. **S2**: 100 Monte-Carlo trials with 3 to 7 quadcopters, and
3. **S3**: 500 Monte-Carlo trials with 4 quadcopters.

The trajectory of the quadcopters in the case of a system with 3 quadcopters is shown in Figure 6.2, and the simulation parameters are shown in Table 6.1. The simulated trajectories are 60 seconds long and each quadcopter covers a distance between 60 m and 218 m, with a maximum speed of 5.5 m/s. The maximum and mean angular velocities are 1 rad/s and 0.3 rad/s, respectively. Following a periodic sequence, each pair of transceivers performs in turn a ranging transaction, except for pairs of transceivers on the same robot. The proposed algorithm is then tested on each dataset and compared to two scenarios.

1. **Centralized**: A hypothetical centralized scenario where each robot has access to range measurements between neighbours in the absence of passive listening. This differs from the proposed framework in that the pseudomeasurements associated with passive listening do not exist, and that this is practically impossible without passive listening or some other communication media. This serves as the benchmark on what is the best achievable estimator using existing methods.
2. **No passive listening**: A decentralized approach but in the absence of passive listening, meaning that robots do not have access to the passive listening pseudomeasurements nor range measurements between neighbours. This serves as the benchmark on what is currently a practically implementable solution without requiring a central processor or additional communication media.

The evaluation is based on the following three criteria.

1. Accuracy: The accuracy of the proposed algorithm as compared to the case with no passive listening is quantified using error plots and the *root-mean-squared-error* (RMSE), which for the pose estimation error $\mathbf{e}_k = \mathrm{Log}(\hat{\mathbf{T}}_k \mathbf{T}_k^{-1})$ is computed as

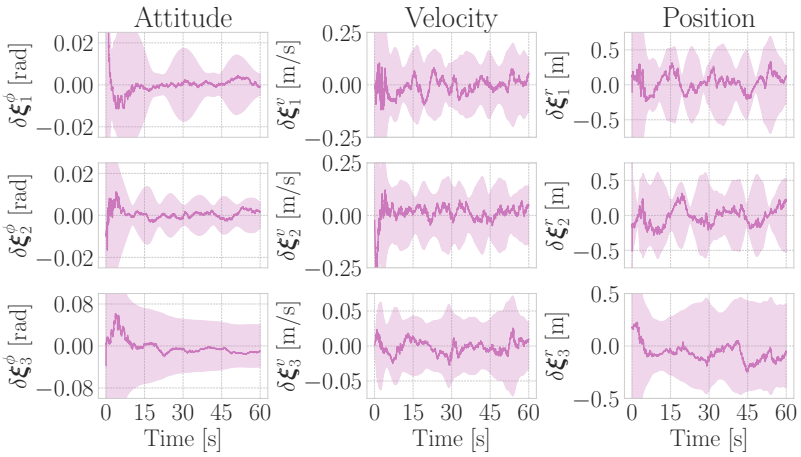
$$\mathrm{RMSE} \triangleq \sqrt{\frac{1}{N+1} \sum_{k=0}^{N} \mathbf{e}_k^{\mathsf{T}} \mathbf{e}_k}$$

for $N+1$ time-steps.

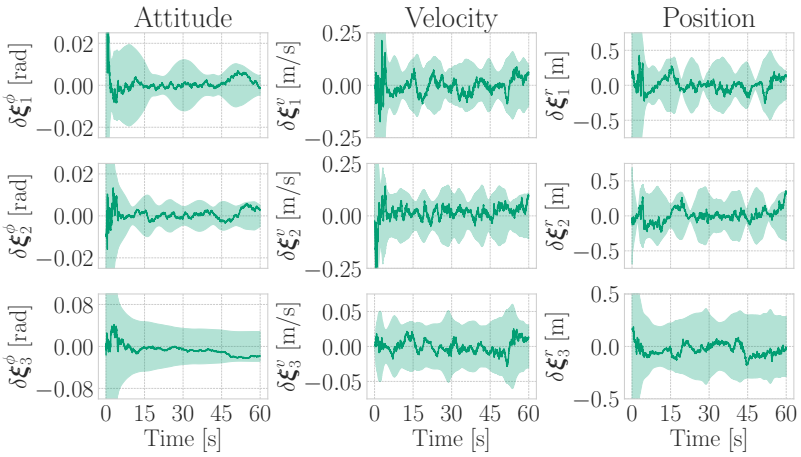
2. Precision: The precision of the proposed algorithm is quantified using $\pm 3\sigma$-bound regions about the estimate, which represent a 99.73% confidence bound under a Gaussian distribution assumption.

3. Consistency: A consistent estimator is an estimator with a modelled precision that reflects the true precision of its estimate. In more specific terms, a consistent estimator outputs a covariance matrix on its estimate that is representative of the true uncertainty of that estimate, as formally defined in Section 2.4.1. Consistency is evaluated using the *normalized-estimation-error-squared* (NEES) test [50, Section 5.4]. Consistency and the NEES test are discussed in further detail in Section 2.4.1.

6.7.1 Estimation Accuracy and Precision

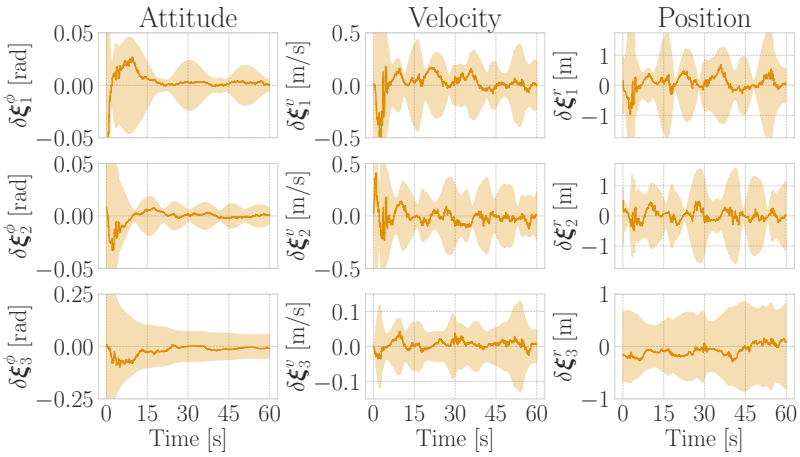


(a) Centralized.

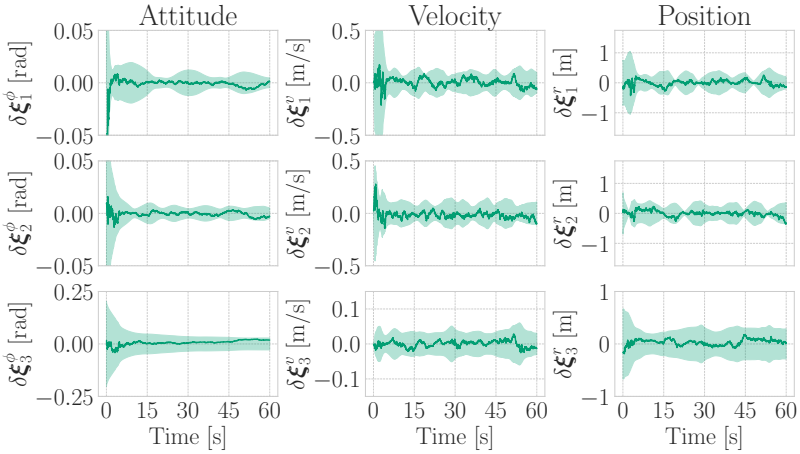


(b) With passive listening.

Figure 6.6: Error plots and $\pm 3\sigma$ bounds (shaded region) for Robot 0's estimate of Robot 1's relative pose for Simulation S1, comparing the centralized and proposed approaches.



(a) No passive listening.

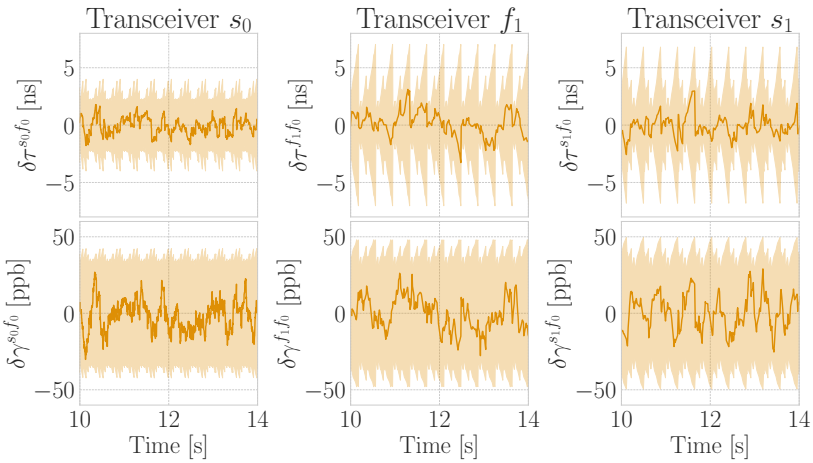


(b) With passive listening.

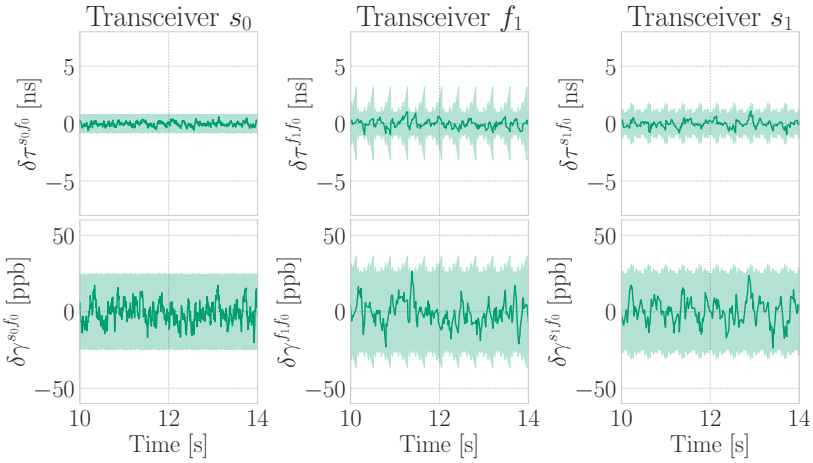
Figure 6.7: Error plots and $\pm 3\sigma$ bounds (shaded region) for Robot 0's estimate of Robot 1's relative pose for Simulation S1, comparing the decentralized no-passive-listening and proposed approaches. The right figure is a zoomed-out version of Figure 6.6b.

Table 6.2: The average RMSE (aRMSE) for all trials of Robot 0's estimate of neighbouring robots' relative pose for Simulation S2. The percentage change is $\frac{\text{Proposed}-\text{Comparison}}{\text{Comparison}}$, where the Comparison is either Centralized or No Passive.

	Position aRMSE averaged over all Robots [m]			**Percentage change [%]**	
Number of Robots	**Centralized**	**No Passive**	**Proposed**	**Centralized**	**No Passive**
3	0.277	0.486	**0.263**	-5.05	-45.88
4	0.231	0.574	**0.222**	-3.90	-61.32
5	0.220	0.662	**0.211**	-4.09	-68.13
6	0.220	0.737	**0.199**	-9.55	-73.00
7	0.186	0.917	**0.165**	-11.29	-82.01



(a) No passive listening.



(b) With passive listening.

Figure 6.8: Error plots and $\pm 3\sigma$ bounds (shaded region) for Robot 0's estimate of the clock states of Transceivers s_0, f_1, and s_1 relative to Transceiver f_0 for Simulation S1. These plots are zoomed in to a window of 4 seconds to show clearly the cycle of expanding and contracting uncertainty in the clock estimates as the transceiver alternates between active ranging and passive listening.

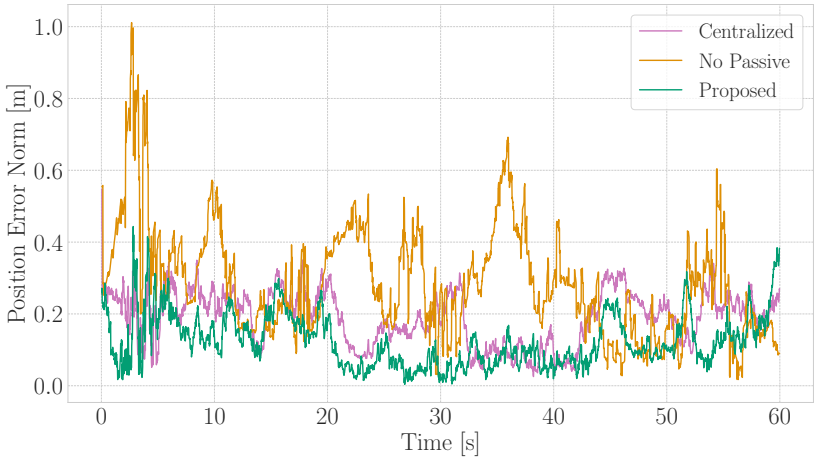
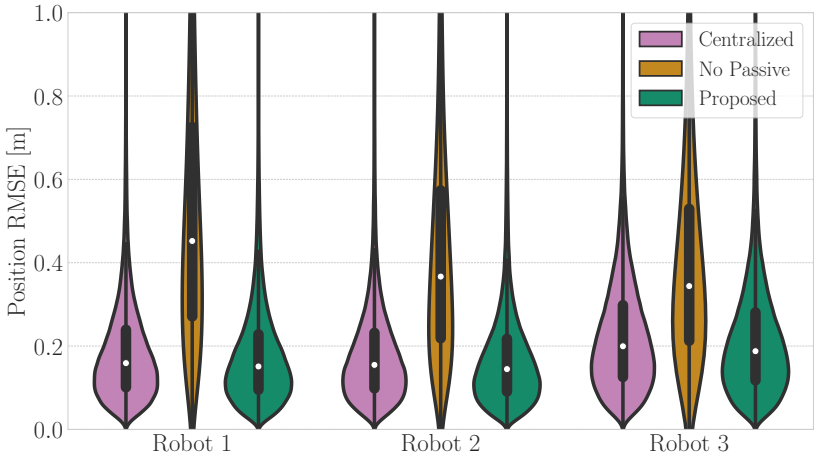
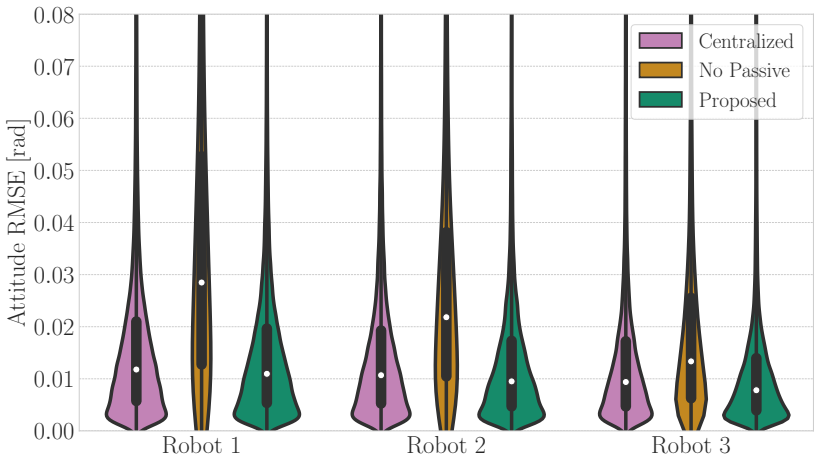


Figure 6.9: The error norm for Robot 0's estimate of Robot 1's relative pose for Simulation S1.



(a) Position RMSE.



(b) Attitude RMSE.

Figure 6.10: Violin and box plots showing the distribution of the position and attitude RMSEs for Simulation S3. The envelope shows the relative frequency of RMSE values. The box plot shows the median as a white dot, while the first and third quartile of the data are represented using the lower and upper bound of the thick black bar, respectively.

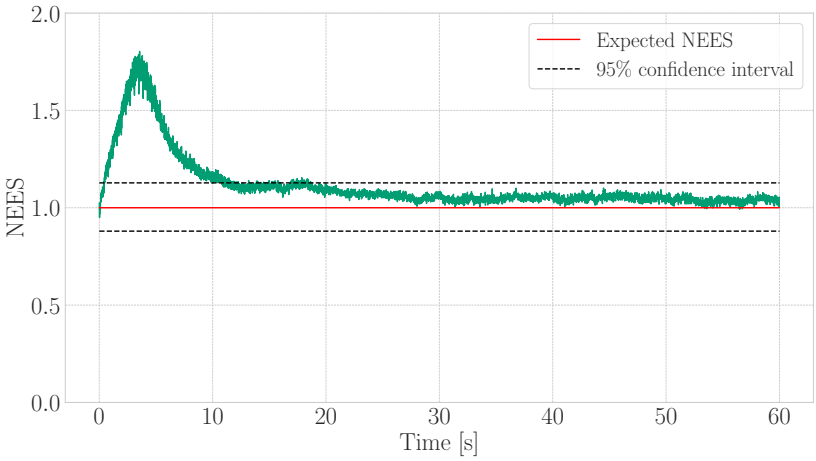


Figure 6.11: 500-trial NEES plot for the proposed estimator on Simulation S3.

The error plots for the relative pose estimate of Robot 1 relative to Robot 0 in Simulation S1 are shown in Figures 6.6 and 6.7. Passive listening reduces the positioning RMSE by 29.4% from 0.204 m to 0.144 m as compared to the centralized approach, and by 55.96% from 0.327 m to 0.144 m when compared to the case of no passive listening. Additionally, passive listening produces at almost every time-step a position error with smaller norm, as shown in Figure 6.9. The proposed estimator is also significantly more confident in its estimate, as shown by the covariance bounds in Figures 6.6 and 6.7.

This improvement in localization performance can be attributed to more measurements and stronger cross-correlation between the different states when passive listening measurements are available. As shown in Figure 6.8, passive listening results in the clock state of a transceiver not drifting significantly in between instances where this transceiver is ranging. This brings down the clock offset RMSE of Transceiver f_1 for example by 59.31% from 1.155 ns to 0.470 ns when compared to the case with no passive listening.

The improvement in performance can also be seen as the number of robots is increased, as shown in Table 6.2 for the Simulation S2. Because only one pair of transceivers can communicate at a time, in the absence of passive listening the rate at which each transceiver participates in a ranging transaction decreases with the number of transceivers, and as a result the overall localization performance degrades. With passive listening on the other hand, adding robots does not result in longer periods without measurements and the measurement rate per robot remains the same. In fact, it turns out that adding robots in the presence of passive listening produces better performance due to spatial variations in the range-measurement sources [99]. This is also the case for the centralized estimator.

To provide further insight into the contribution of passive listening measurements on the behaviour of the estimator, the distribution of the RMSEs of the position and attitude estimates of all robots in Simulation S3 are visualized in Figure 6.10. Not only does the proposed approach significantly outperform the no passive listening approach, but it matches the centralized approach, which is typically the best possible solution under an assumption of the availability of a central processor. In fact, the proposed framework slightly outperforms the standard centralized approach due to the availability of additional pseudomeasurements.

6.7.2 Consistency

Given that the estimator is an EKF, consistency cannot be guaranteed due to linearization and discretization errors. Nonetheless, the proposed on-manifold framework can characterize banana-shaped error distributions that result from range measurements as shown in Figure 2.4 more efficiently. Consequently, the error distribution appears to be well-characterized by

the estimator as shown in Figures 6.7 and 6.8, as the error trajectory typically lies within the $\pm 3\sigma$ bounds.

A better evaluation of the consistency of the estimator is a NEES test, which is performed over the 500 trials of Simulation S3 and is shown in Figure 6.11. During the first few seconds when the quadcopters are taking off from the ground, their geometry and low speeds result in a weakly-observable system [18], which results in overconfidence of the estimator as linearization-based filters can correct in unobservable directions [100], [101]. Nonetheless, the estimator then converges towards consistency, although it is never perfectly consistent due to linearization and discretization errors, which is a feature of EKFs. This can be solved by slightly inflating the associated covariance matrices used in the filter.

6.8 Experimental Results

The proposed approach is tested on multiple experimental trials. The ranging protocol discussed in Section 6.4 is implemented in C on custom-made boards fitted with DWM1000 UWB transceivers [7]. Two boards are then fitted to Uvify IFO-S quadcopters approximately 45 cm apart. The experimental set-up is shown in Figure 6.1. Three of these quadcopters are then used for the experimental results shown in this section, with multiple approximately-75-second-long trajectories similar to the one shown in Figure 6.12 in a roughly 5 m $\times$ 5 m area. The quadcopters in the experimental trajectories each cover a distance between 20 m and 35 m, with a maximum speed of 3.75 m/s. The maximum and mean angular velocities are 2.12 rad/s and 0.3 rad/s, respectively. In order to analyze the error in the pose estimates of the robots, a 12-camera Vicon motion-capture system is used to record the ground-truth pose of each quadcopter.

To enable the 6 transceivers to take turn ranging with one another, the common-list protocol discussed in Section 6.4 is implemented using *robot operating system* (ROS). This allows each robot to range with its neighbours at a rate of 90 Hz, and collect passive listening measurements at a rate of 150 Hz. These UWB measurements are corrected for antenna delays and power-induced biases using Chapter 4, before fusing them with the onboard IMU and height measurements in the proposed EKF. An ICM-20689 IMU is used with characteristics similar to the simulated ones given in Table 6.1, and the height measurements are obtained from a downward-facing camera. The height measurement error is assumed Gaussian with 5 cm of standard deviation. To reject outliers in the range and passive-listening measurements, the *normalized-innovation-squared* (NIS) test is used in the filter [50, Section 5.4].

Note that before flight, all transceivers are allowed to range with one another to initial-

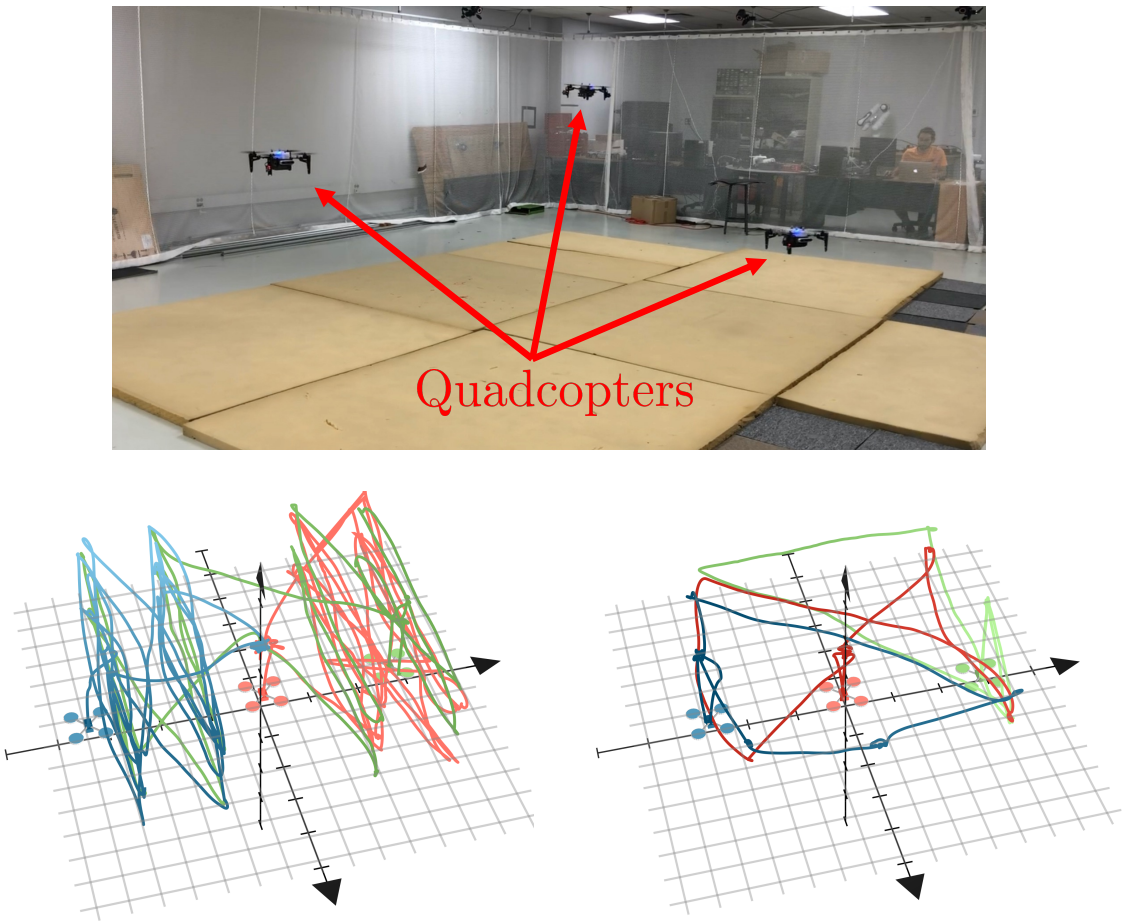
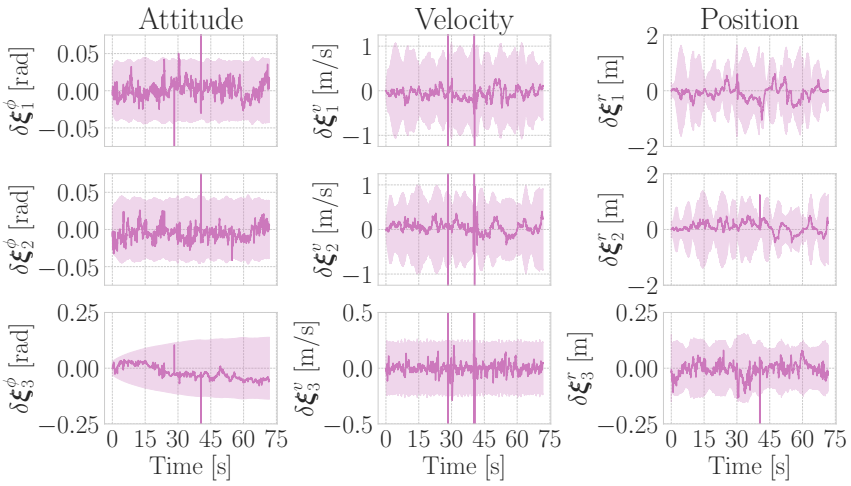


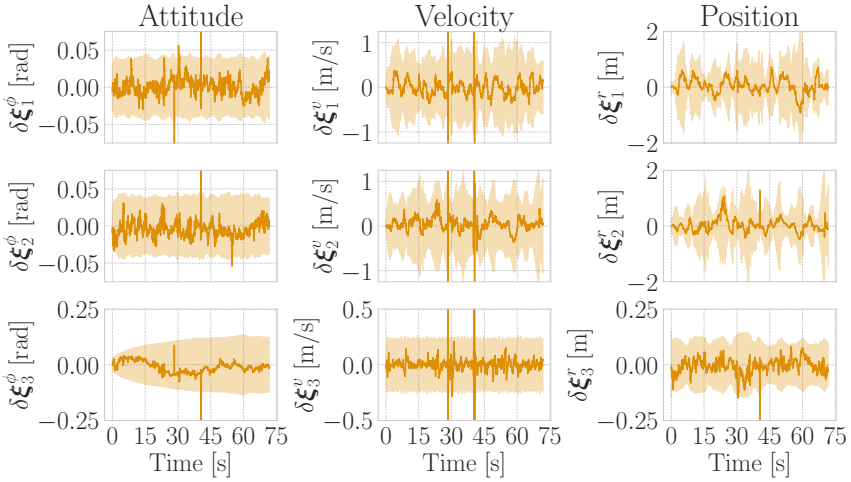
Figure 6.12: (Top) 3 quadcopters in the experimental space. (Bottom Left) The experimental trajectory for Trial 1, where each colour represents the trajectory of a different quadcopter and the grid represents a roughly 5 m $\times$ 5 m area. (Bottom Right) The experimental trajectory for Trial 2.

ize the relative clock offset states using the second pseudomeasurement from Section 4.2, alongside a pseudomeasurement $y^{\gamma} = \frac{\tilde{\mathrm{R}}^3 - \tilde{\mathrm{R}}^2}{\tilde{\mathrm{T}}^3 - \tilde{\mathrm{T}}^2} - 1 \approx \gamma_{f_1 s_2}$ that is not used in the filter. Meanwhile, the IMU biases are initialized using the motion capture system and are then assumed constant throughout the experiment, which is sufficient for the duration of the experiments presented here. Addressing IMU biases for longer experiments is presented in Section 6.9.1.

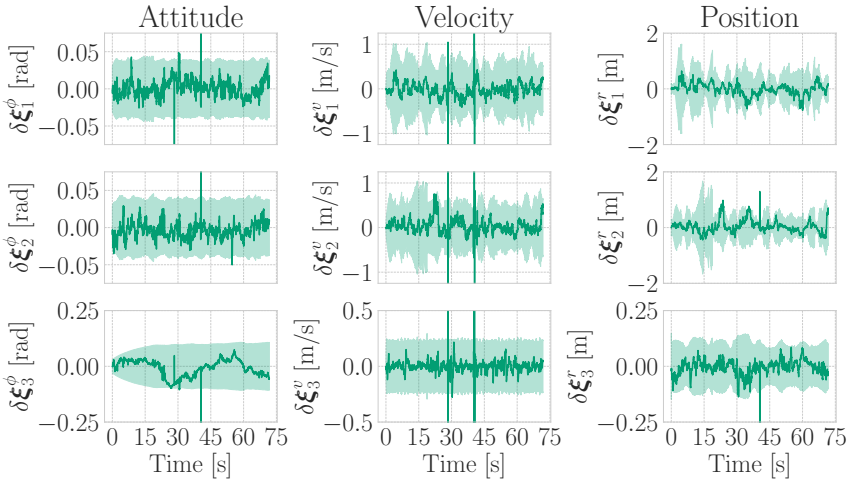
The pose-error plots for one of the trials are shown in Figure 6.13 with and without fusing passive listening measurements, and the RMSE comparison for 4 different trials with varying motion are shown in Table 6.3. Even though both scenarios result in error trajectories that fall within the error bounds, it is clear that with the additional passive listening measurements available to the robot at 150 Hz, the relative position estimates in particular become significantly less uncertain. Additionally, these error plots correspond to the first row in Table 6.3, showing that the improvement in the confidence of the estimator is additionally



(a) Centralized.



(b) No passive listening.



(c) With passive listening.

Figure 6.13: Error plots and $\pm 3\sigma$ bounds (shaded region) for Robot 0's estimate of Robot 1's relative pose for experimental trial 1.

accompanied with a 12.56% and 21.97% reduction in the RMSE as compared to the centralized and no passive listening position RMSE, respectively. This reduction in RMSE goes up to 23.28% and 48.20%, respectively, for one of the runs when passive listening measurements are utilized.

Table 6.3: The RMSE of Robot 0's estimate of neighbouring robots' relative pose for multiple experimental trials. The percentage change is $\frac{\text{Proposed}-\text{Comparison}}{\text{Comparison}}$, where the Comparison is either Centralized or No Passive.

	Position RMSE averaged over all Robots [m]			**Percentage change [%]**	
	Centralized	**No Passive**	**Proposed**	**Centralized**	**No Passive**
Trial 1	0.390	0.437	**0.341**	-12.56	-21.97
Trial 2	0.614	0.954	**0.576**	-6.19	-39.62
Trial 3	0.462	0.593	**0.443**	-4.11	-25.30
Trial 4	0.580	0.859	**0.445**	-23.28	-48.20

6.9 Further Practical Considerations

6.9.1 IMU Biases

The IMU measurements typically suffer from time-varying biases, which must be estimated as part of the state for long-term navigation. It can be shown that, when modelling the evolution of biases as a random walk, some form of the IMU biases can be incorporated into the process model while still maintaining the differential Sylvester equation form (6.25). To do so, each Robot i estimates its own gyroscope bias $\boldsymbol{\beta}_i^{\text{gyr},i}$ in its own body frame, and uses this estimate to correct the IMU measurements and inflating the covariance when constructing the RMI. Additionally, each robot estimates a relative accelerometer bias to every neighbour in the robot's own body frame, which does not affect the computed RMI. For example, Robot 0's estimate of Robot i's relative accelerometer bias is defined as

$$\boldsymbol{\beta}_0^{\text{acc},0i} \triangleq \boldsymbol{\beta}_0^{\text{acc},0} - \mathbf{C}_{0i}\boldsymbol{\beta}_i^{\text{acc},i},$$

where $\boldsymbol{\beta}_i^{\text{acc},i}$ is Robot i's accelerometer bias. The interested reader can refer to Appendix E for derivation of the pose and bias process models, corresponding linearization, preintegration, and simulation and experimental results.

6.9.2 Incomplete and Dynamic Communication Graphs

The proposed framework has been evaluated under the assumptions of a full communication graph, no packet drop, and no communication failures. Nonetheless, these are all real-world problems that must be addressed before implementing the proposed framework. This is beyond the scope of this paper; nonetheless, a brief discussion regarding these issues is provided in this section.

The ranging protocol and the proposed estimator do not require a full communication graph and the lack of communication failures. However, the common-list MAC protocol does. The common-list MAC protocol is a very simple approach made possible only due to passive listening, and is ideal for small teams of robots that will always be within communication range with one another, thus allowing a full communication graph. Whenever a ranging transaction between a pair of transceivers fails, it is reattempted multiple times until a timeout is triggered, after which the ranging pair and all other robots who have not heard a message for the duration of the timeout move onto the next entry in the list. The protocol can handle a robot's communication failure by having each robot eliminate an element in the list when it fails more than κ times, where κ is a user-defined threshold.

When extending to larger teams, it is not possible to assume that all robots are within communication range of one another, thus invalidating the full-communication-graph assumption. Additionally, robots might fall in and out of range with one another over time, thus necessitating an incomplete dynamic communication graph model. In such scenarios, the common-list protocol is no longer simple, as robots need to know what other robots out of communication range are doing. Therefore, such systems may benefit from other MAC protocols such as token passing, which is still possible with the proposed ranging protocol and estimator as they are independent of the choice of the MAC protocol. The benefits of passive listening thus still stand, not just due to additional measurements, but because it also allows each robot to maintain a list of neighbours within communication range.

Another implication of incomplete graphs is that each robot only estimates relative poses for the subset of robots that lie within its communication range. This is useful as it reduces the dimensionality of the onboard estimator, since each robot only estimates the relative states of $m < n$ neighbouring robots. However, having dynamic graphs due to robots falling in and out of the communication range of the robot mean that the robots must initialize the states of neighbours when they appear and marginalize out the states of neighbours that have not been within the communication range for an extended period of time. The initialization can potentially be done by listening to a window of measurements from the new neighbour and formulating a least-squares problem.

6.10 Conclusion and Future Work

In this chapter, the problem of relative extended pose estimation has been addressed for a team of robots, each equipped with UWB transceivers. To allow the implementation of scalable algorithms, a novel ranging protocol is proposed that allows neighbouring robots to passively listen-in on the measurements without any underlying assumption on the hierarchy of the communication. This is then utilized to implement a simple MAC protocol and an efficient means for sharing preintegrated IMU information, such that each robot can have access to its neighbours' IMU measurements to fuse in their own estimator, in a decentralized manner. The proposed estimator fuses the preintegrated IMU information with the UWB measurements in a filter that estimates both the clock states of the transceivers and the relative poses of the robots, without any reliance on magnetometers. The relative poses and the preintegration are formulated directly on $SE_2(3)$ to compactly represent the state while utilizing the nice mathematical properties of on-manifold state estimation. This is then all evaluated in simulation using different numbers of robots and Monte-Carlo trials, and in experiments using multiple trials of 3 Uvify IFO-S quadcopters, each equipped with 2 UWB transceivers. This required the implementation of the passive listening ranging protocol on the custom-made UWB transceivers. The proposed framework is then shown to improve the localization performance significantly when compared to the case of no passive listening measurements.

This chapter can be extended to address complications that arise in wireless communications, such as packet drop. When a packet drop occurs, neighbours miss an RMI, which is required to propagate their estimates forward, and therefore this must be addressed in a real-world application, potentially by providing a means for robots to request a missed RMI from their neighbours. Future work will additionally consider more efficient scheduling schemes where only a subset of the transceivers range with one another in pairs while the remaining transceivers are always passively listening. Other potential extensions of this chapter include addressing dynamic graphs and incomplete graphs, where not all transceivers can range with one another. Although more scalable than the simple framework in Chapter 3, the size of the team is still limited under the proposed framework in this chapter as the dimensionality of the state each robots estimates grows with the size of the team. To address this, a potential extension of this chapter for larger teams would be to have each robot only estimate the relative states of $m < n$ neighbouring robots.

Another limitation of this chapter is the lack of collaboration between robots. Given the proposed framework, different robots are estimating similar states, and by sharing these state estimates, each robot could fuse their neighbours' state estimates in the robot's own

state estimator to potentially achieve better performance. Nonetheless, care must be given to the fact that under such an approach, cross-correlations develop between the estimators of different robots, and not properly modelling these cross-correlations can lead to inconsistent results, as will be shown in the next chapter. Filters feeding into other estimators is referred to as "loosely-coupled filtering" or "cascaded filtering", and deriving a general cascaded-filtering framework will be the emphasis of the next chapter.

Chapter 7

Cascaded Filtering Using the Sigma Point Transformation

Summary

It is often convenient to separate a state estimation task into smaller "local" tasks, where each local estimator estimates a subset of the overall system state. This is particularly prevalent in multi-robot systems, such as the framework in the previous chapter, where each robot might be estimating the relative poses of a few but not necessarily all robots in the team. However, neglecting cross-covariance terms between state estimates can result in over-confident estimates, which can ultimately degrade the accuracy of the estimator. Common cascaded filtering techniques focus on the problem of modelling cross-covariances when the local estimators share a common state vector. This chapter introduces a novel cascaded and decentralized filtering approach that approximates the cross-covariances when the local estimators consider distinct state vectors. The proposed estimator is validated in simulations and in experiments on a three-dimensional pose estimation problem. The proposed approach is compared to a naive cascaded filtering approach that neglects cross-covariance terms, a sigma point-based Covariance Intersection filter, and a full-state centralized filter. In both simulations and experiments, the proposed filter outperforms the naive and the Covariance Intersection filters, while performing comparatively to the centralized full-state filter.

7.1 Introduction

State estimation is an integral part of technologies that use measured data to make decisions. For complex systems, augmenting all the required states into one monolithic state

estimator can become laborious, inconvenient, or even infeasible due to computational or bandwidth limitations. Therefore, the ability to split the state estimation task into many different estimators is desirable [102]–[104]. For example, multi-robot systems would benefit from each robot having a local estimator, and large complex systems would ideally have a set of interconnected and specialized estimators, also known as cascaded estimators. Such modularity allows the independent design, analysis, tuning, debugging, and testing of each state estimator, in addition to providing architectural clarity. More efficient computing capability is also possible as smaller state vectors can be considered in a parallel framework.

To be able to use cascaded filtering approaches, the problem of modelling the cross-covariance between the estimates of the different filters must be addressed [36], [105]–[107]. Consider an attitude and heading reference system (AHRS) providing an attitude estimate to a position estimator, where the AHRS is the *feeding filter* and the position estimator is the *receiving filter*, as depicted in Figure 7.1. Any error in the AHRS results in an error in the position estimator, because sensors such as an accelerometer rely on attitude information when used for position estimation. Therefore, the estimation error of the attitude estimate and the position estimate are correlated. Assuming the attitude estimation error is uncorrelated with the position estimation error is equivalent to neglecting the fact that the attitude information is not entirely new, thus resulting in an overconfident estimate [34], [108]. In this chapter, a state estimator is said to be *consistent* if its calculated error covariance does not underestimate the true error covariance, which is defined formally as (2.36) based on [36]. The approach of neglecting cross-covariances is common [102]–[104], [109], but can lead to divergence of the state estimate as a consequence of the filter inconsistency [36], [105], [106].

To accommodate cross-covariances, a master filter is proposed in [106], [107] named the *Federated filter*, that takes as input the output of all the local, sensor-specific filters, and fuses the estimates using a least squares algorithm. This is extended in [110] to nonlinear systems, by designing the master filter to be an unscented Kalman filter (UKF) [43, Section 5.6]. A more recent modification of the Federated filter is an optimal sample-based fusion algorithm discussed in [111]. Another popular fusion algorithm is *Covariance Intersection* (CI), introduced in [36], where a convex combination of the means and covariances of different estimates of the same unknown produces a consistent fused mean and covariance. This involves the selection of weighing parameters, and in [112], the optimal choice of weights is addressed in relation to state estimation of dynamical systems. A major limitation of all the discussed fusion algorithms is the focus on scenarios where the local filters share a set of common states. In [37], a method for computing pose estimates for a decentralized multi-robot system is presented. However, it is assumed that relative pose measurements

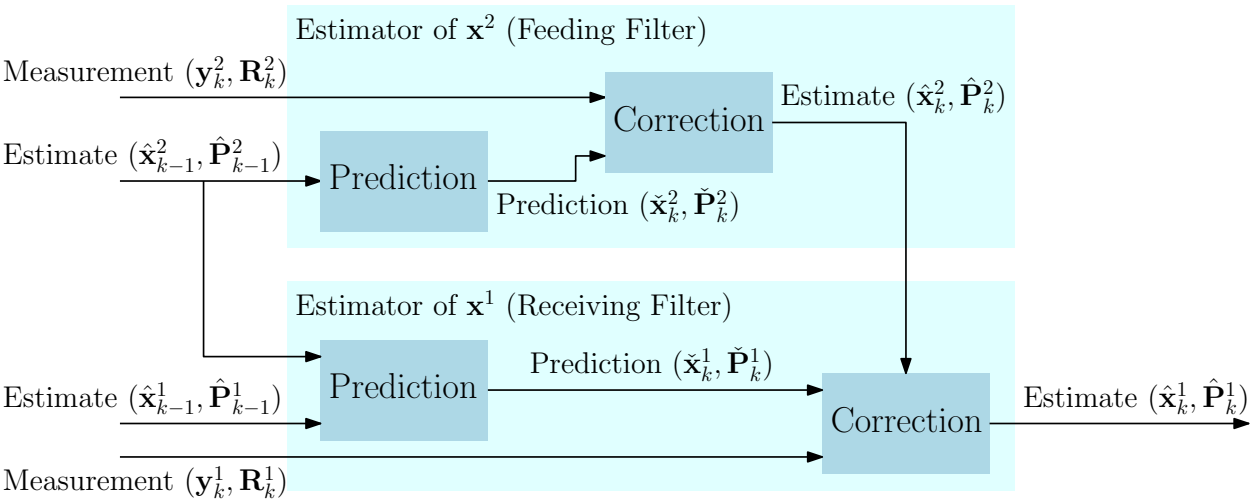


Figure 7.1: A block diagram of the architecture of a cascaded filter at time-step k. The estimator of $\mathbf{x}^2$ is independent from the estimator of $\mathbf{x}^1$, while the estimator of $\mathbf{x}^1$ uses $\hat{\mathbf{x}}_{k-1}^2$ and $\hat{\mathbf{x}}_k^2$ as a measurement.

between the robots are available to reconstruct a common shared state. Lastly, in [40], two approaches are introduced, one based on the CI method and another that hinges on solving a linear matrix inequality problem.

The presentation of a probabilistic approximation of the propagated cross-covariance terms when the feeding and receiving filters do not estimate the same exact set of states is a novel contribution of this chapter. In particular, a novel cascaded and decentralized state estimation approach that approximates the cross-covariance terms using a sigma point transformation is introduced. The proposed approximation is general to any loosely-coupled system of estimators, not just multi-robot systems. Another contribution of this chapter is demonstrating improved performance and consistency in simulation and experiments when compared to both an estimator that neglects cross-covariances and a sigma point-based Covariance Intersection (SPCI) estimator. Rather than addressing complex multi-robot systems, the simulation and experiments consider the more general problem of pose estimation but in a decoupled and decentralized manner.

The remainder of this chapter is organized as follows. The cascaded filtering problem is formulated in Section 7.2, where a probabilistic analysis of the approximate prior and posterior distributions of the receiving filter is also given. The proposed algorithm is presented in Section 7.3, and is then evaluated in simulations in Section 7.4 and in experiments in Section 7.5.

7.1.1 Notation

In this chapter, $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is used to denote a Gaussian random variable $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. The notation $\check{(\cdot)}$ and $\hat{(\cdot)}$ denotes the predicted and measurement-corrected state estimates, respectively. The superscripts $(\cdot)^i$ are indices, not exponents. Lastly, $\boldsymbol{\Sigma}_{\mathbf{ab}}$ is used to denote the cross-covariance matrix between vectors $\mathbf{a}$ and $\mathbf{b}$, and the following notation is used for special cross-covariance matrices,

$$\check{\mathbf{P}}_k^i \triangleq \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^i \check{\mathbf{x}}_k^i}, \quad \hat{\mathbf{P}}_k^i \triangleq \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^i \hat{\mathbf{x}}_k^i}, \quad \check{\mathbf{P}}_{p,q}^{1,2} \triangleq \boldsymbol{\Sigma}_{\check{\mathbf{x}}_p^1 \hat{\mathbf{x}}_q^2}, \quad \hat{\mathbf{P}}_{p,q}^{1,2} \triangleq \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_p^1 \hat{\mathbf{x}}_q^2}.$$

7.2 Cascaded Filtering

Consider two discrete-time processes evolving through

$$\mathbf{x}_k^1 = \mathbf{f}^1\left(\mathbf{x}_{k-1}^1, \mathbf{x}_{k-1}^2, \mathbf{w}_{k-1}^1\right), \tag{7.1}$$
$$\mathbf{x}_k^2 = \mathbf{f}^2\left(\mathbf{x}_{k-1}^2, \mathbf{w}_{k-1}^2\right), \tag{7.2}$$

where $\mathbf{x}_k^1 \in \mathbb{R}^{n_1}$ and $\mathbf{x}_k^2 \in \mathbb{R}^{n_2}$ are distinct state vectors, and $\mathbf{w}_{k-1}^i \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_{k-1}^i\right)$, $\mathbf{w}_{k-1}^i \in \mathbb{R}^{\ell_i}$ represents the process noise associated with the evolution of $\mathbf{x}_k^i$. Additionally, consider two measurement signals modelled as

$$\mathbf{y}_k^1 = \mathbf{g}^1\left(\mathbf{x}_k^1, \mathbf{x}_k^2, \boldsymbol{\nu}_k^1\right), \tag{7.3}$$
$$\mathbf{y}_k^2 = \mathbf{g}^2\left(\mathbf{x}_k^2, \boldsymbol{\nu}_k^2\right), \tag{7.4}$$

where $\mathbf{y}_k^1 \in \mathbb{R}^{m_1}$ and $\mathbf{y}_k^2 \in \mathbb{R}^{m_2}$ are distinct measurement vectors, and $\boldsymbol{\nu}_k^i \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_{k-1}^i\right) \in \mathbb{R}^{h_i}$ represents the measurement noise associated with $\mathbf{y}_k^i$. All process and measurement noise are assumed to be mutually independent.

In the process models (7.1) and (7.2), no inputs are considered to simplify the derivation of the proposed framework. However, this can be extended to systems with known inputs, since inputs in the receiving filter are dealt with in the standard way, and the approximation to be discussed in Section 7.3.3 still holds when the feeding filter has inputs.

The standard, centralized, full-state filtering approach involves designing a filter with the augmented state vector $\mathbf{x}_k = \begin{bmatrix} (\mathbf{x}_k^1)^\mathsf{T} & (\mathbf{x}_k^2)^\mathsf{T} \end{bmatrix}^\mathsf{T} \in \mathbb{R}^{n_1+n_2}$, using knowledge of (7.1)-(7.4). Meanwhile, the cascaded filtering approach separates this problem into two filters, as shown in Figure 7.1. The feeding filter outputs an approximate *a posteriori* distribution of

$$\mathbf{x}_k^2 \big| \mathcal{I}_k^2 \sim \mathcal{N}\left(\hat{\mathbf{x}}_k^2, \hat{\mathbf{P}}_k^2\right), \tag{7.5}$$

where $\mathcal{I}_k^i = \{\check{\mathbf{x}}_0^i, \mathbf{y}_{0:k}^i\}$ is an "information" set containing a prior and measurements, $a|b$ denotes a random variable a conditioned on b, $\check{\mathbf{x}}_0^2 \in \mathbb{R}^{n_2}$ is the initial prediction of $\mathbf{x}_k^2$, $\hat{\mathbf{x}}_k^2 \in \mathbb{R}^{n_2}$ is its estimate at time-step k based on (7.2) and (7.4), and $\hat{\mathbf{P}}_k^2 \in \mathbb{R}^{n_2 \times n_2}$ is the associated covariance matrix. The receiving filter uses this output alongside (7.1) and (7.3) to estimate the *a posteriori* distribution of

$$\mathbf{x}_k^1 \big| \check{\mathbf{x}}_0^1, \mathbf{y}_{0:k}^1, \check{\mathbf{x}}_0^2, \mathbf{y}_{0:k}^2 \equiv \mathbf{x}_k^1 | \mathcal{I}_k^1, \mathcal{I}_k^2 \tag{7.6}$$

without access to $\mathcal{I}_k^2$. Naive cascaded filtering [102]–[104], [109] considers the estimate $\hat{\mathbf{x}}_k^2$ to be a measurement in the receiving filter. This breaks a requirement for consistency of the Kalman filter and its descendants, which is the conditional independence of the measurement from the state and measurement histories [43, Property 4.2].

7.2.1 Receiving Filter Approximate Probability Distributions

When the *a posteriori* distribution of (7.5) is known, the goal is to find a consistent estimator of (7.6) without access to the feeding filter's process model (7.2), measurement model (7.4), or corresponding inputs and measurements. Such a solution allows the design of an $\mathbf{x}^1$ estimator with minimal knowledge of the inner workings of the $\mathbf{x}^2$ estimator.

The propagation of $\mathbf{x}_k^1$ is dependent on $\mathbf{x}_{k-1}^2$, and mutual independence cannot be assumed. The conditional joint distribution of $\mathbf{x}_k^1$ and $\mathbf{x}_{k-1}^2$ is therefore assumed to be

$$\begin{bmatrix} \mathbf{x}_k^1 \\ \mathbf{x}_{k-1}^2 \end{bmatrix} \Big| \, \mathcal{I}_{k-1}^1, \mathcal{I}_{k-1}^2 \sim \mathcal{N}\left(\begin{bmatrix} \check{\mathbf{x}}_k^1 \\ \hat{\mathbf{x}}_{k-1}^2 \end{bmatrix}, \begin{bmatrix} \check{\mathbf{P}}_k^1 & \check{\mathbf{P}}_{k,k-1}^{1,2} \\ \left(\check{\mathbf{P}}_{k,k-1}^{1,2}\right)^\mathsf{T} & \hat{\mathbf{P}}_{k-1}^2 \end{bmatrix} \right), \tag{7.7}$$

where $\check{\mathbf{x}}_k^1 = \mathbf{f}^1\left(\hat{\mathbf{x}}_{k-1}^1, \hat{\mathbf{x}}_{k-1}^2, \mathbf{0}\right)$. This block partitioning follows from an assumption that the conditional joint distribution is Gaussian. Using [43, Lemma A.2], (7.7) gives

$$\begin{aligned} \mathbf{x}_k^1 | \mathcal{I}_{k-1}^1, \mathcal{I}_{k-1}^2, \mathbf{x}_{k-1}^2 \sim \mathcal{N}\Big(& \check{\mathbf{x}}_k^1 + \check{\mathbf{P}}_{k,k-1}^{1,2} \left(\hat{\mathbf{P}}_{k-1}^2\right)^{-1} \left(\mathbf{x}_{k-1}^2 - \hat{\mathbf{x}}_{k-1}^2\right), \\ & \check{\mathbf{P}}_k^1 - \check{\mathbf{P}}_{k,k-1}^{1,2} \left(\hat{\mathbf{P}}_{k-1}^2\right)^{-1} \left(\check{\mathbf{P}}_{k,k-1}^{1,2}\right)^\mathsf{T} \Big). \end{aligned} \tag{7.8}$$

and since $\mathcal{I}_{k-1}^2$ and $\mathbf{x}_{k-1}^2$ are available only through $\hat{\mathbf{x}}_{k-1}^2\left(\mathcal{I}_{k-1}^2\right)$ and $\hat{\mathbf{P}}_{k-1}^2\left(\mathcal{I}_{k-1}^2\right)$ in the receiving filter, $\mathbf{x}_{k-1}^2$ in particular is replaced in (7.8) with $\hat{\mathbf{x}}_{k-1}^2$ to obtain

$$\mathbf{x}_k^1 | \mathcal{I}_{k-1}^1, \hat{\mathbf{x}}_{k-1}^2, \hat{\mathbf{P}}_{k-1}^2 \sim \mathcal{N}\Big(\check{\mathbf{x}}_k^1, \check{\mathbf{P}}_k^1 - \check{\mathbf{P}}_{k,k-1}^{1,2} \left(\hat{\mathbf{P}}_{k-1}^2\right)^{-1} \left(\check{\mathbf{P}}_{k,k-1}^{1,2}\right)^\mathsf{T} \Big). \tag{7.9}$$

In (7.9), $\hat{\mathbf{P}}_{k-1}^2$ is known and $\check{\mathbf{P}}_k^1$ can be obtained using common filtering techniques. However,

$\check{\mathbf{P}}^{1,2}_{k,k-1}$ is unknown and is required to consistently update the estimate of $\mathbf{x}^1_k$.

Similarly, an approximation of the distribution of (7.6) is derived in Appendix F yielding the correction step equations

$$\hat{\mathbf{x}}^1_k = \check{\mathbf{x}}^1_k + \mathbf{K}^1_k \left(\mathbf{y}^1_k - \check{\mathbf{y}}^1_k\right) - \mathbf{K}^{1,2}_k \mathbf{K}^2_k \left(\mathbf{y}^1_k - \check{\mathbf{y}}^1_k\right), \tag{7.10}$$

$$\hat{\mathbf{P}}^1_k = \check{\mathbf{P}}^1_k - \mathbf{K}^1_k \boldsymbol{\Sigma}^{\mathsf{T}}_{\check{\mathbf{x}}^1_k \check{\mathbf{y}}^1_k} - \mathbf{K}^{1,2}_k \left(\left(\check{\mathbf{P}}^{1,2}_{k,k}\right)^{\mathsf{T}} - \mathbf{K}^2_k \boldsymbol{\Sigma}^{\mathsf{T}}_{\check{\mathbf{x}}^1_k \check{\mathbf{y}}^1_k}\right), \tag{7.11}$$

$$\hat{\mathbf{P}}^{1,2}_{k,k} = \check{\mathbf{P}}^{1,2}_{k,k} - \mathbf{K}^1_k \boldsymbol{\Sigma}^{\mathsf{T}}_{\hat{\mathbf{x}}^2_k \check{\mathbf{y}}^1_k}, \tag{7.12}$$

where $\check{\mathbf{y}}^1_k = \mathbf{g}^1\left(\check{\mathbf{x}}^1_k, \hat{\mathbf{x}}^2_k, \mathbf{0}\right)$ is the predicted measurement, and

$$\mathbf{K}^1_k = \boldsymbol{\Sigma}_{\check{\mathbf{x}}^1_k \check{\mathbf{y}}^1_k} \boldsymbol{\Sigma}^{-1}_{\check{\mathbf{y}}^1_k \check{\mathbf{y}}^1_k}, \qquad \mathbf{K}^2_k = \boldsymbol{\Sigma}_{\hat{\mathbf{x}}^2_k \check{\mathbf{y}}^1_k} \boldsymbol{\Sigma}^{-1}_{\check{\mathbf{y}}^1_k \check{\mathbf{y}}^1_k}, \tag{7.13}$$

$$\mathbf{K}^{1,2}_k = \left(\check{\mathbf{P}}^{1,2}_{k,k} - \mathbf{K}^1_k \boldsymbol{\Sigma}^{\mathsf{T}}_{\mathbf{x}^2_k \mathbf{y}^1_k}\right) \left(\hat{\mathbf{P}}^2_k - \mathbf{K}^2_k \boldsymbol{\Sigma}^{\mathsf{T}}_{\mathbf{x}^2_k \mathbf{y}^1_k}\right)^{-1}. \tag{7.14}$$

The form of the update equation for the cross-covariance matrix $\hat{\mathbf{P}}^{1,2}_{k,k}$ is from the joint distribution of $\mathbf{x}^1$ and $\mathbf{x}^2$, given in the second step of Appendix F. As with the prediction step, naive cascaded filtering techniques neglect the cross-covariance components $\boldsymbol{\Sigma}_{\hat{\mathbf{x}}^2_k \check{\mathbf{y}}^1_k}$ and $\check{\mathbf{P}}^{1,2}_{k,k}$, which appear in (7.10)-(7.14). Note that (7.10) and (7.11) are similar to the standard Kalman filter, except that this new form of the receiving filter also corrects the feeding filter's state $\mathbf{x}^2$ locally before correcting the state $\mathbf{x}^1$. The updated state of the feeding filter is never communicated back to the feeding filter, resulting in a loss of performance as compared to a full estimator.

7.3 Proposed Cascaded Receiving Filter

The proposed framework requires passing Gaussian distributions through nonlinear process and measurement models, which is a well-documented problem with a plethora of possible solutions. In this chapter, the sigma point transformation from Section 2.1.2 is used as it is a concise, general approach that is suitable for handling nonlinearities. Another possible solution is a linearization-based one, which is further discussed in Appendix H.

In the prediction step, $\hat{\mathbf{P}}^{1,2}_{k-1,k-1}$ is used to generate sigma points, which are then propagated forward in time to approximate $\check{\mathbf{P}}^{1,2}_{k,k-1}$. The exact change in the cross-covariance matrix going from $\check{\mathbf{P}}^{1,2}_{k,k-1}$ to $\check{\mathbf{P}}^{1,2}_{k,k}$ will be derived in Section 7.3.3, and an approximation is then proposed. In the correction step, $\check{\mathbf{P}}^{1,2}_{k,k}$ is then used to generate sigma points, which allows computing $\hat{\mathbf{P}}^{1,2}_{k,k}$.

7.3.1 Prediction Step

For prediction, the goal is to propagate the estimate $\hat{\mathbf{x}}^1_{k-1}$ forward in time to get a predicted state $\check{\mathbf{x}}^1_k$. To do so, the process model (7.1) and the output of the feeding estimator, $\hat{\mathbf{x}}^2_{k-1}$ and $\hat{\mathbf{P}}^2_{k-1}$, are used. Initially, $\hat{\mathbf{P}}^{1,2}_{0,0}$ is set to be the zero matrix, unless there is prior knowledge of a correlation between the initial estimates of the local estimators.

The state vectors $\mathbf{x}^1_{k-1}$ and $\mathbf{x}^2_{k-1}$ and the process noise $\mathbf{w}^1_{k-1}$ are augmented into a vector $\mathbf{v}$, with

$$\hat{\mathbf{v}}_{k-1} = \begin{bmatrix} \hat{\mathbf{x}}^1_{k-1} \\ \hat{\mathbf{x}}^2_{k-1} \\ \mathbf{0} \end{bmatrix}, \tag{7.15}$$

$$\hat{\mathbf{P}}_{\mathbf{v}_{k-1}} = \begin{bmatrix} \hat{\mathbf{P}}^1_{k-1} & \hat{\mathbf{P}}^{1,2}_{k-1,k-1} & \mathbf{0} \\ \left(\hat{\mathbf{P}}^{1,2}_{k-1,k-1}\right)^{\mathsf{T}} & \hat{\mathbf{P}}^2_{k-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}^1_{k-1} \end{bmatrix}. \tag{7.16}$$

Let $L \triangleq \dim\left(\hat{\mathbf{v}}_{k-1}\right)$, and $\mathbf{L}\mathbf{L}^{\mathsf{T}} \triangleq \hat{\mathbf{P}}_{\mathbf{v}_{k-1}}$. Using the spherical cubature rule [43], define the $2L$ sigma points $\mathbf{s}$ to be

$$\mathbf{s}^i_{k-1} \triangleq \hat{\mathbf{v}}_{k-1} + \sqrt{L}\,\mathrm{col}_i(\mathbf{L}), \quad \mathbf{s}^{i+L}_{k-1} \triangleq \hat{\mathbf{v}}_{k-1} - \sqrt{L}\,\mathrm{col}_i(\mathbf{L}).$$

By unstacking each sigma point into

$$\mathbf{s}^i_{k-1} = \begin{bmatrix} \left(\hat{\mathbf{x}}^1_{k-1,i}\right)^{\mathsf{T}} & \left(\hat{\mathbf{x}}^2_{k-1,i}\right)^{\mathsf{T}} & \left(\mathbf{w}^1_{k-1,i}\right)^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}, \tag{7.17}$$

the sigma points are propagated through (7.1),

$$\check{\mathbf{x}}^1_{k,i} = \mathbf{f}^1\left(\hat{\mathbf{x}}^1_{k-1,i}, \hat{\mathbf{x}}^2_{k-1,i}, \mathbf{w}^1_{k-1,i}\right). \tag{7.18}$$

Hence, the statistics of the propagated sigma points are

$$\check{\mathbf{x}}^1_k = \frac{1}{2L}\sum_{i=1}^{2L} \check{\mathbf{x}}^1_{k,i}, \quad \check{\mathbf{P}}^1_k = \frac{1}{2L}\sum_{i=1}^{2L} \left(\check{\mathbf{x}}^1_{k,i} - \check{\mathbf{x}}^1_k\right)\left(\check{\mathbf{x}}^1_{k,i} - \check{\mathbf{x}}^1_k\right)^{\mathsf{T}},$$

$$\check{\mathbf{P}}^{1,2}_{k,k-1} = \frac{1}{2L}\sum_{i=1}^{2L} \left(\check{\mathbf{x}}^1_{k,i} - \check{\mathbf{x}}^1_k\right)\left(\hat{\mathbf{x}}^2_{k-1,i} - \hat{\mathbf{x}}^2_{k-1}\right)^{\mathsf{T}}. \tag{7.19}$$

7.3.2 Correction Step

For the correction step, the goal is to correct the predicted state $\check{\mathbf{x}}_k^1$ using the measurement $\mathbf{y}_k^1$, to obtain $\hat{\mathbf{x}}_k^1$. As $\mathbf{y}_k^1$ is a function of $\mathbf{x}_k^2$, the output of the feeding estimator $\hat{\mathbf{x}}_k^2$ and $\hat{\mathbf{P}}_k^2$ is required. As before, the state vectors $\mathbf{x}_k^1$ and $\mathbf{x}_k^2$ and the measurement noise $\boldsymbol{\nu}_k^1$ are augmented into a vector $\mathbf{u}$, with

$$\hat{\mathbf{u}}_k = \begin{bmatrix} \check{\mathbf{x}}_k^1 \\ \hat{\mathbf{x}}_k^2 \\ \mathbf{0} \end{bmatrix}, \quad \hat{\mathbf{P}}_{\mathbf{u}_k} = \begin{bmatrix} \check{\mathbf{P}}_k^1 & \check{\mathbf{P}}_{k,k}^{1,2} & \mathbf{0} \\ \left(\check{\mathbf{P}}_{k,k}^{1,2}\right)^\mathsf{T} & \hat{\mathbf{P}}_k^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_k^1 \end{bmatrix}. \tag{7.20}$$

The problem of finding $\check{\mathbf{P}}_{k,k}^{1,2}$ in (7.20) from (7.19) is addressed in Section 7.3.3. Let $M \triangleq \dim(\hat{\mathbf{u}}_k)$, and $\mathbf{M}\mathbf{M}^\mathsf{T} \triangleq \hat{\mathbf{P}}_{\mathbf{u}_k}$. Consequently, using the spherical cubature rule, define the $2M$ sigma points $\mathbf{p}$ to be

$$\mathbf{p}_k^i \triangleq \hat{\mathbf{u}}_k + \sqrt{M}\,\mathrm{col}_i(\mathbf{M}), \quad \mathbf{p}_k^{i+L} \triangleq \hat{\mathbf{u}}_k - \sqrt{M}\,\mathrm{col}_i(\mathbf{M}).$$

By unstacking each sigma point into

$$\mathbf{p}_k^i = \begin{bmatrix} \left(\check{\mathbf{x}}_{k,i}^1\right)^\mathsf{T} & \left(\hat{\mathbf{x}}_{k,i}^2\right)^\mathsf{T} & \left(\boldsymbol{\nu}_{k,i}^1\right)^\mathsf{T} \end{bmatrix}^\mathsf{T}, \tag{7.21}$$

the nonlinear transformation of the sigma points as per the measurement model is

$$\check{\mathbf{y}}_{k,i}^1 = \mathbf{g}^1\left(\check{\mathbf{x}}_{k,i}^1, \hat{\mathbf{x}}_{k,i}^2, \boldsymbol{\nu}_{k,i}^1\right). \tag{7.22}$$

Hence, the statistics of the propagated sigma points are

$$\check{\mathbf{y}}_k^1 = \frac{1}{2M}\sum_{i=1}^{2M} \check{\mathbf{y}}_{k,i}^1, \tag{7.23}$$

$$\boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1\check{\mathbf{y}}_k^1} = \frac{1}{2M}\sum_{i=1}^{2M} \left(\check{\mathbf{x}}_{k,i}^1 - \check{\mathbf{x}}_k^1\right)\left(\check{\mathbf{y}}_{k,i}^1 - \check{\mathbf{y}}_k^1\right)^\mathsf{T}, \tag{7.24}$$

$$\boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2\check{\mathbf{y}}_k^1} = \frac{1}{2M}\sum_{i=1}^{2M} \left(\hat{\mathbf{x}}_{k,i}^2 - \hat{\mathbf{x}}_k^2\right)\left(\check{\mathbf{y}}_{k,i}^1 - \check{\mathbf{y}}_k^1\right)^\mathsf{T}, \tag{7.25}$$

$$\boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1\check{\mathbf{y}}_k^1} = \frac{1}{2M}\sum_{i=1}^{2M} \left(\check{\mathbf{y}}_{k,i}^1 - \check{\mathbf{y}}_k^1\right)\left(\check{\mathbf{y}}_{k,i}^1 - \check{\mathbf{y}}_k^1\right)^\mathsf{T}. \tag{7.26}$$

The filter equations given in (7.10)-(7.14) can then be used to obtain the approximated distribution of the corrected state $\hat{\mathbf{x}}_k^1$ and its cross-covariance matrix with the state estimate $\hat{\mathbf{x}}_k^2$.

7.3.3 Approximating the Effect of the Feeding Filter on the Cross-Covariance

The cross-covariance matrix $\check{\mathbf{P}}^{1,2}_{k,k-1}$ is approximated using sigma points in the prediction step. However, $\check{\mathbf{P}}^{1,2}_{k,k}$ is needed to generate the sigma points for the correction step. If knowledge of the process and measurement models of the feeding filter is available, $\check{\mathbf{P}}^{1,2}_{k,k-1}$ can be propagated to $\check{\mathbf{P}}^{1,2}_{k,k}$ analytically. To elucidate this, consider the linear system

$$\mathbf{x}^1_k = \mathbf{A}^1_{k-1}\mathbf{x}^1_{k-1} + \mathbf{B}^1_{k-1}\mathbf{x}^2_{k-1} + \mathbf{w}^1_{k-1}, \tag{7.27}$$

$$\mathbf{x}^2_k = \mathbf{A}^2_{k-1}\mathbf{x}^2_{k-1} + \mathbf{w}^2_{k-1}, \tag{7.28}$$

$$\mathbf{y}^1_k = \mathbf{C}^1_k\mathbf{x}^1_k + \mathbf{D}^1_{k-1}\mathbf{x}^2_k + \boldsymbol{\nu}^1_k, \quad \mathbf{y}^2_k = \mathbf{C}^2_k\mathbf{x}^2_k + \boldsymbol{\nu}^2_k, \tag{7.29}$$

where the state vectors and noise parameters follow the same notation as in (7.1)-(7.4). Moreover, consider for the moment that each of the process and measurement models in (7.27)-(7.29) are known. Let $\mathbf{G}^2_k \in \mathbb{R}^{n_2 \times m_2}$ denote the Kalman gain of the estimator of $\mathbf{x}^2_k$ as per Section 2.4, then

$$\begin{aligned}
\check{\mathbf{P}}^{1,2}_{k,k} &= \mathbb{E}\left[\left(\mathbf{x}^1_k - \check{\mathbf{x}}^1_k\right)\left(\mathbf{x}^2_k - \hat{\mathbf{x}}^2_k\right)^{\mathsf{T}} \,\middle|\, \mathcal{I}^1_{k-1}, \mathcal{I}^2_k\right] \\
&= \mathbb{E}\Big[\left(\mathbf{x}^1_k - \check{\mathbf{x}}^1_k\right)\left(\mathbf{A}^2_{k-1}\mathbf{x}^2_{k-1} + \mathbf{w}^2_{k-1} - \mathbf{A}^2_{k-1}\hat{\mathbf{x}}^2_{k-1}\right. \\
&\qquad - \mathbf{G}^2_k\mathbf{C}^2_k\left(\mathbf{A}^2_{k-1}\mathbf{x}^2_{k-1} + \mathbf{w}^2_{k-1}\right) - \mathbf{G}^2_k\boldsymbol{\nu}^2_k \\
&\qquad \left. + \mathbf{G}^2_k\mathbf{C}^2_k\mathbf{A}^2_{k-1}\hat{\mathbf{x}}^2_{k-1}\right)^{\mathsf{T}} \,\Big|\, \mathcal{I}^1_{k-1}, \mathcal{I}^2_k\Big] \\
&= \mathbb{E}\left[\left(\mathbf{x}^1_k - \check{\mathbf{x}}^1_k\right)\left(\mathbf{x}^2_{k-1} - \hat{\mathbf{x}}^2_{k-1}\right)^{\mathsf{T}} \,\middle|\, \mathcal{I}^1_{k-1}, \mathcal{I}^2_k\right] \times \left(\mathbf{A}^2_{k-1} - \mathbf{G}^2_k\mathbf{C}^2_k\mathbf{A}^2_{k-1}\right)^{\mathsf{T}} \\
&= \check{\mathbf{P}}^{1,2}_{k,k-1}\boldsymbol{\Psi}_k,
\end{aligned} \tag{7.30}$$

where the assumptions $\mathbb{E}\left[\left(\mathbf{x}^1_k - \check{\mathbf{x}}^1_k\right)\left(\mathbf{w}^2_{k-1}\right)^{\mathsf{T}} \,\middle|\, \mathcal{I}^1_{k-1}, \mathcal{I}^2_k\right] = \mathbf{0}$ and $\mathbb{E}\left[\left(\mathbf{x}^1_k - \check{\mathbf{x}}^1_k\right)\left(\boldsymbol{\nu}^2_k\right)^{\mathsf{T}} \,\middle|\, \mathcal{I}^1_{k-1}, \mathcal{I}^2_k\right] = \mathbf{0}$ have been used, and $\boldsymbol{\Psi}_k \triangleq \left(\left(\mathbf{1} - \mathbf{G}^2_k\mathbf{C}^2_k\right)\mathbf{A}^2_{k-1}\right)^{\mathsf{T}}$. Since, however, the estimator of $\mathbf{x}^1$ does not have access to the process and measurement models of the feeding filter, computing (7.30) exactly is not possible. If cooperation from the feeding filter is at all possible, the most accurate solution is for the feeding filter to share the matrix $\boldsymbol{\Psi}_k$ at every time-step, in addition to the estimated distribution of $\mathbf{x}^2$. When this is not possible, however, an alternative is to approximate $\boldsymbol{\Psi}_k$. Using (2.33), $\boldsymbol{\Psi}_k$ can be rewritten as

$$\boldsymbol{\Psi}_k = \left(\hat{\mathbf{P}}^2_k\left(\check{\mathbf{P}}^2_k\right)^{-1}\mathbf{A}^2_{k-1}\right)^{\mathsf{T}}. \tag{7.31}$$

Once the feeding filter reaches a steady state, $\hat{\mathbf{P}}_k^2 \left(\check{\mathbf{P}}_k^2\right)^{-1} \approx \mathbf{1}$, yielding the approximation $\hat{\mathbf{\Psi}}_k = \left(\mathbf{A}_{k-1}^2\right)^\mathsf{T}$. Hence, with knowledge of what $\hat{\mathbf{\Psi}}$ should be if access to the process model (7.2) is available, the user can form an educated guess of the Jacobian $\mathbf{A}^2$. Since the state vector $\mathbf{x}^2$ of the feeding filter is known, the user can reconstruct an approximate (i.e., lower fidelity) process model for the states of the feeding filter, and use the Jacobian of this approximate process model as an approximation of the true Jacobian of the process model (7.2) used in the feeding filter.

An implementation nuisance with over-estimating $\check{\mathbf{P}}_{k,k}^{1,2}$ using this approximation is that $\hat{\mathbf{P}}_{\mathbf{u}_k}$ in (7.20) can become indefinite. Therefore, at every iteration, the definiteness of $\hat{\mathbf{P}}_{\mathbf{u}_k}$ must be checked, and the cross-covariances are to be deflated using a scalar parameter until the definiteness test passes.

7.4 Simulation Results

The main criteria for evaluating the proposed cascaded filter is its consistency, which will be done using the normalized estimation error squared (NEES) test on Monte Carlo runs. As discussed in Section 2.4.1, the NEES test involves computing a chi-squared statistic $\bar{\epsilon}$ using the error trajectory and corresponding covariance of multiple trials. If the statistic is below a certain threshold, the hypothesis that the estimator is consistent cannot be rejected with 95% confidence [50]. The $\pm 3\sigma$ bound test is also considered, and the root-mean-squared-error (RMSE) is computed to evaluate the performance of the estimators.

7.4.1 Linear System

To elucidate the benefits of the proposed approach, consider first a toy problem with two discrete-time linear time-invariant processes evolving through

$$x_k^1 = x_{k-1}^1 - x_{k-1}^2 + w_{k-1}^1, \quad w_{k-1}^1 \sim \mathcal{N}\left(0, q^1\right), \tag{7.32}$$

$$x_k^2 = x_{k-1}^2 + w_{k-1}^2, \quad w_{k-1}^2 \sim \mathcal{N}\left(0, q^2\right), \tag{7.33}$$

where $x_k^1 \in \mathbb{R}$ and $x_k^2 \in \mathbb{R}$ are distinct states. Additionally, consider two measurements modelled as

$$y_k^1 = x_k^1 + x_k^2 + \nu_k^1, \quad \nu_k^1 \sim \mathcal{N}\left(0, r^1\right), \tag{7.34}$$

$$y_k^2 = x_k^2 + \nu_k^2, \quad \nu_k^2 \sim \mathcal{N}\left(0, r^2\right), \tag{7.35}$$

where $y_k^1 \in \mathbb{R}$ and $y_k^2 \in \mathbb{R}$ are distinct measurements.

As before, assume at every time-step k the estimates $\hat{x}^1_{k-1}$, $\hat{x}^2_{k-1}$, $\hat{x}^2_k$, the covariances $\hat{P}^1_{k-1}$, $\hat{P}^2_{k-1}$, $\hat{P}^2_k$, and the cross-covariance $\hat{P}^{1,2}_{k-1,k-1}$ are known, and the goal is to find the estimate $\hat{x}^1_k$ and its corresponding covariance $\hat{P}^1_k$ using the known values, the process model (7.32), and the measurement model (7.34). Using (7.33), Ψ is set to equal 1, with the definiteness check in place as per Section 7.3.3.

Deriving the update equations for x^1 yields

$$\begin{aligned}
&\check{x}^1_k = \hat{x}^1_{k-1} - \hat{x}^2_{k-1}, \qquad \hat{x}^1_k = \check{x}^1_k + K(y^1_k - \check{y}^1_k),\\
&\check{P}^1_k = \hat{P}^1_{k-1} + 2\hat{P}^{1,2}_{k-1,k-1} + \hat{P}^2_{k-1} + q^1,\\
&\hat{P}^1_k = (1-K)^2\check{P}^1_k - 2(1-K)K\check{P}^{1,2}_{k,k} + K^2(\hat{P}^2_k + r^1),
\end{aligned}$$

where the optimal gain K is

$$K = \left(\check{P}^1_k + \check{P}^{1,2}_{k,k}\right)\left(\check{P}^1_k + 2\check{P}^{1,2}_{k,k} + \hat{P}^2_k + r^1\right)^{-1}. \tag{7.36}$$

The proposed cascaded filter approximates the cross-covariance terms $\hat{P}^{1,2}_{k-1,k-1}$ and $\check{P}^{1,2}_{k,k}$, and is evaluated against the naive approach in [109], to reiterate the importance of modelling cross-covariances.

To evaluate consistency, 1000 Monte Carlo trials with varying initial conditions and noise realizations are performed. A naive estimator that assumes the estimate $\hat{x}^2_k$ is independent from the estimate $\hat{x}^1_k$ thinks it has access to more information than it actually does, resulting in an overconfident estimator. Therefore, the naive estimator fails both the $\pm 3\sigma$ test shown in Figure 7.2 and the NEES test in Figure 7.3, while the proposed estimator that accommodates for cross-covariances passes both consistency checks.

The full estimator is clearly the best performer as it can perfectly calculate the cross-covariances and update all states using all measurements. The average RMSE of the proposed framework is 45% worse than the full estimator, while the naive estimator is 90% worse. Therefore, this highlights the importance of addressing cross-covariances to prevent the reuse of old information in the filter.

7.4.2 Nonlinear System

Consider a rigid body navigating 3D space with an onboard inertial measurement unit (IMU) and an ultra-wideband (UWB) tag, where the position of the UWB tag p relative to the IMU z resolved in the body frame $\mathcal{F}_b$ is known, and is denoted $\mathbf{r}^{pz}_b \in \mathbb{R}^3$. Additionally, let there be a $\rho \in \mathbb{N}_{>3}$ number of UWB anchors scattered within ranging distance of the UWB tag, thus providing a noisy position measurement of $\mathbf{r}^{p_k w}_a \in \mathbb{R}^3$, which is the position of the UWB tag

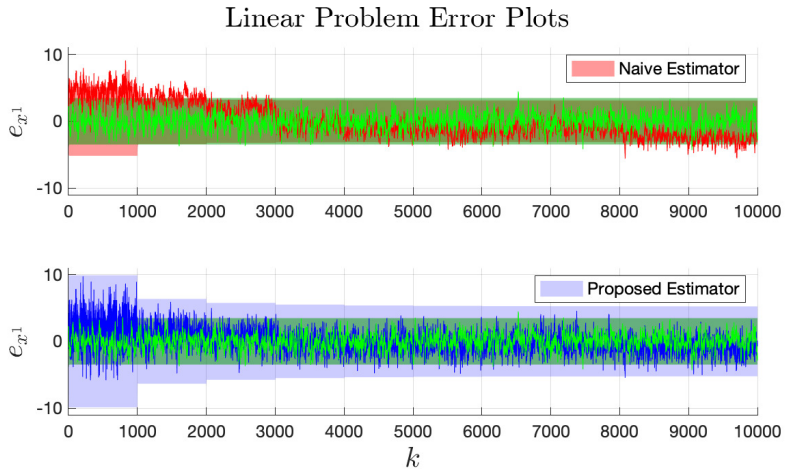


Figure 7.2: Single run of the naive and proposed estimators on the linear system. Shaded regions correspond to $\pm 3\sigma$ bounds. Red, blue, and green correspond to naive, proposed, and full estimators, respectively. After $k = 1000$, the $\pm 3\sigma$ bound of the naive and full estimator almost fully overlap.

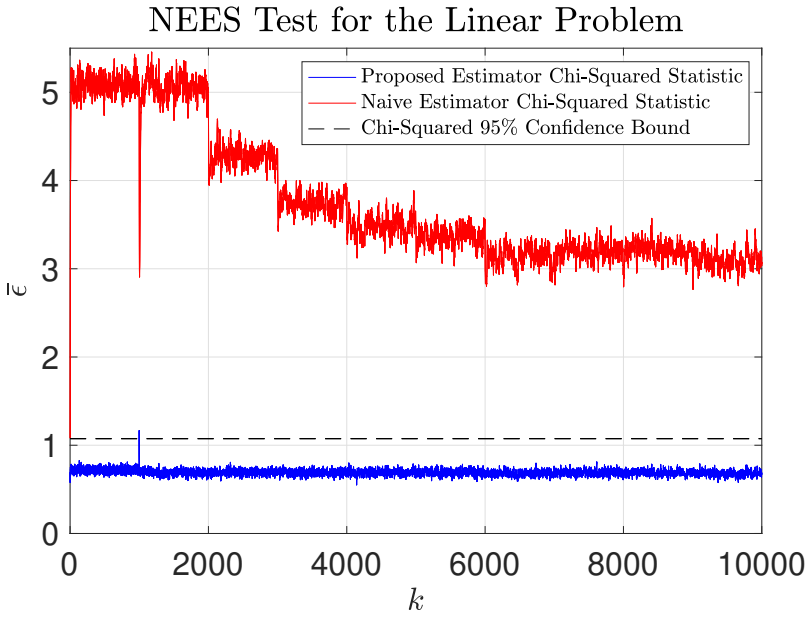


Figure 7.3: The NEES test for 1000 Monte Carlo trials on the linear system, showing the consistency evaluation of the cascaded estimators.

relative to some arbitrary point w at time-step k, in the absolute frame $\mathcal{F}_a$. The simulation parameters of the system are shown in Table 7.1, and the set-up in Figure 7.4.

Using the gyroscope, the magnetometer, and the accelerometer, an AHRS is designed using the invariant extended Kalman filter (IEKF) [47] framework, where accelerometer aiding follows a thresholding rule similar to the one considered in [8]. The output of the AHRS is a direction cosine matrix (DCM) estimate $\hat{\mathbf{C}}_{ab} \in SO(3)$ that gives the relation

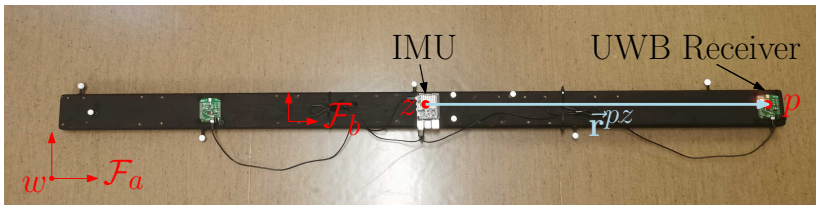


Figure 7.4: The prototype used in simulation and for collection of experimental data. The UWB receiver is used to obtain position measurements from a system of UWB anchors.

Table 7.1: Simulation parameters for the nonlinear system.

	Specification	**Value**
Std. Deviation	Accel. (m/s^2)	0.10
	Gyro. (rad/s)	0.0032
	Mag. (μF)	2.00
	Pos. meas. (m)	0.22
	Initial pos. (m)	0.45
	Initial vel. (m/s)	0.45
	Initial att. (rad)	0.22
Rate	IMU (Hz)	100
	Pos. meas. (Hz)	50

$\mathbf{r}_a = \hat{\mathbf{C}}_{ab}\mathbf{r}_{\hat{b}}$, where $\mathbf{r}_a$ and $\mathbf{r}_{\hat{b}}$ are the same arbitrary vector resolved in $\mathcal{F}_a$, the absolute frame, and $\mathcal{F}_{\hat{b}}$, the estimated body frame, respectively. The corresponding covariance matrix $\hat{\mathbf{P}}^{\text{AHRS}} \in \mathbb{R}^{3\times3}$ is also made available by the AHRS.

The position estimator has access to the position measurement $\mathbf{y} \in \mathbb{R}^3$ and the accelerometer measurement $\mathbf{u}_b \in \mathbb{R}^3$ resolved in the body frame $\mathcal{F}_b$. The state vector of the position estimator is

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{r}_a^{zw}(t) \\ \mathbf{v}_a^{zw}(t) \end{bmatrix}, \tag{7.37}$$

where $\mathbf{r}_a^{zw}(t) \in \mathbb{R}^3$ is the position of the IMU relative to the arbitrary point w resolved in $\mathcal{F}_a$, and $\mathbf{v}_a^{zw}(t) \in \mathbb{R}^3$ is the velocity of the IMU relative to the point w with respect to $\mathcal{F}_a$, resolved in $\mathcal{F}_a$. The corresponding process model is

$$\dot{\mathbf{r}}_a^{zw}(t) = \mathbf{v}_a^{zw}(t), \tag{7.38}$$

$$\dot{\mathbf{v}}_a^{zw}(t) = \mathbf{C}_{ab}(t)\left(\mathbf{u}_b(t) - \mathbf{w}_b(t)\right) + \mathbf{g}_a, \tag{7.39}$$

where $\mathbf{w}_b \in \mathbb{R}^3$ denotes white Gaussian process noise, and $\mathbf{g}_a \in \mathbb{R}^3$ is the gravitational acceleration vector resolved in $\mathcal{F}_a$. The discrete-time measurement model is

$$\mathbf{y}_k = \mathbf{r}_a^{z_k w} + \mathbf{C}_{ab_k}\mathbf{r}_b^{p_k z} + \boldsymbol{\nu}_{a_k}, \tag{7.40}$$

where $\boldsymbol{\nu}_a \in \mathbb{R}^3$ denotes white Gaussian measurement noise. This set-up has inputs in both feeding and receiving filters.

Note that due to the presence of a moment-arm between the IMU and the position sensor, and since the accelerometer measurements are obtained in the body frame $\mathcal{F}_b$, cross-covariances develop between the states of the AHRS and the position estimator. Through

knowledge of (7.38)-(7.40), $\mathbf{r}_b^{pz}$, and the output of the AHRS, the goal is to design a consistent position estimator. Four approaches are considered. They are

1. an AHRS and a **naive** sigma point Kalman filter, where cross-covariances are neglected,
2. an AHRS and the **proposed** sigma point-based filter, where $\hat{\boldsymbol{\Psi}}$ is the identity matrix as per Section 7.3.3,
3. an AHRS and a sigma point-based Covariance Intersection (**SPCI**) filter, and
4. a **full** sigma point Kalman filter that augments the state vector $\mathbf{x}(t)$ with the attitude states $\mathbf{C}_{ab}(t)$.

The SPCI filter used is a sigma point extension of [40], and is designed in a similar way to the proposed framework. The main difference is that instead of approximating the cross-covariances to compute (7.16) and (7.20), the estimated covariance matrices $\check{\mathbf{P}}^1$, $\hat{\mathbf{P}}^1$, and $\hat{\mathbf{P}}^2$ are inflated as per the CI approach, and the cross-covariances are assumed to be zero. This involves tuning an additional scalar parameter. More details on this approach are given in Appendix G.

For the full sigma point Kalman filter, the geodesic L_2-mean discussed in [113] is used for computing means and covariances on $SO(3)$, which are required for sigma point transformations with attitude states. The results of the full sigma point Kalman filter are used as the baseline best possible performance, but in practice would be computationally heavy, inflexible, and even impossible for larger systems.

The computational overhead of the proposed framework is not significantly different from a naive SPKF or the SPCI filter, as all the considered cascaded estimators use sigma point transformations and a similar set of equations. Additionally, a more computationally efficient linearization-based approach is possible as shown in Appendix H.

To evaluate the 4 estimators, 500 Monte Carlo trials are performed, each 60 seconds long with varying initial conditions and noise realizations. A summary of the position RMSEs is given in Figure 7.5 and Table 7.2. The proposed estimator clearly outperforms both the SPCI and naive estimators, even though all 3 estimators share the same AHRS. As expected, the proposed cascaded filter also passes the consistency test, as shown in Figure 7.6.

The proposed framework provides a 35.9% worse position estimate than the full estimator even though the provided attitude estimates are 61.1% worse. The worse attitude estimates are due to the UWB measurements correcting the attitude states in the full estimator, but not in cascaded architectures as discussed in Section 7.2. However, the loss of performance is compensated by the modularity, computational gain, and flexibility of the proposed cascaded filter.

Table 7.2: RMSE of the estimators averaged 500 trials.

	Average RMSE	**% Diff. to Full**
Full - Position (m)	0.0487	-
Proposed - Position (m)	**0.0662**	**35.9%**
SPCI - Position (m)	0.0862	77.0%
Naive - Position (m)	0.1733	256%
Full - Attitude (rad)	0.0190	-
AHRS - Attitude (rad)	0.0306	61.1%

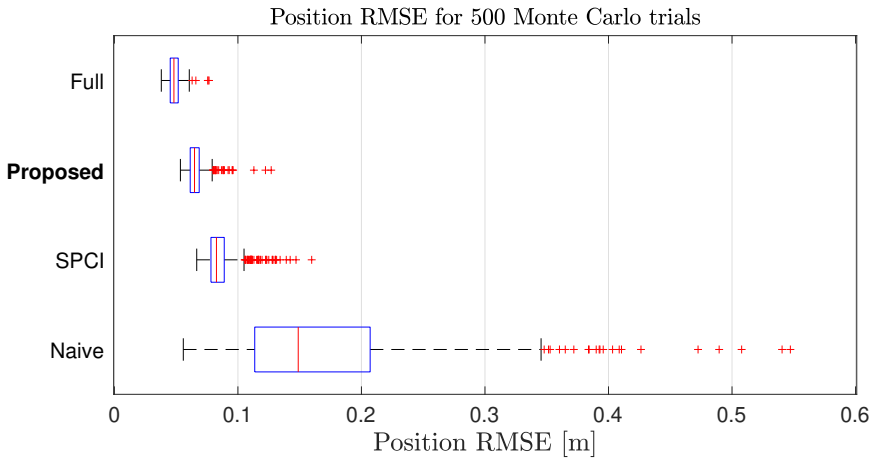


Figure 7.5: A box plot showing the median RMSE, outliers, and variation of different estimators over 500 Monte Carlo trials.

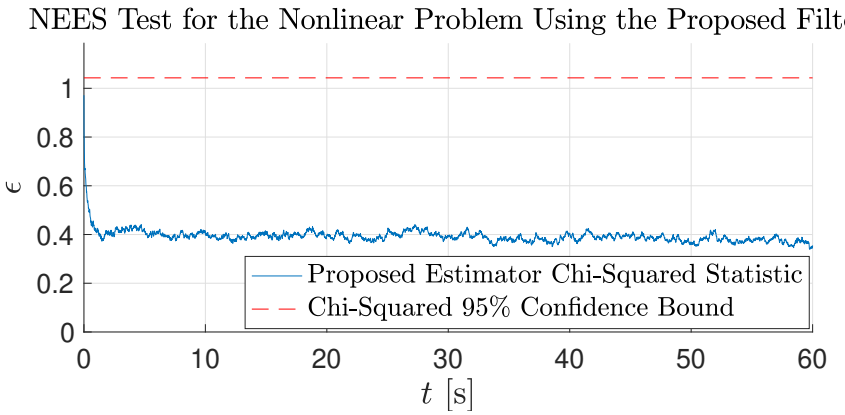


Figure 7.6: The NEES test results for the 500 Monte Carlo trials, showing the consistency of the proposed estimator.

7.5 Experimental Results

Experimental data is collected for the nonlinear system discussed in Section 7.4.2 using the prototype shown in Figure 7.4. The IMU data is collected using a Raspberry Pi Sense HAT

at 240 Hz, and the position measurements are collected at 16 Hz using the Pozyx Creator Kit, which is a UWB-based positioning system. Five UWB anchors communicate with a UWB tag placed on the body 84 cm away from the IMU. This is complemented with ground truth data collected using an OptiTrack optical motion capture system at 120 Hz.

Three datasets are tested using the four estimators discussed in Section 7.4.2. Each run involves moving the rigid body randomly in a volume of approximately 5 m $\times$ 5 m $\times$ 2 m while recording the IMU, UWB, and ground truth data. The main difference between the three datasets is the pace at which the robot is moved around and rotated.

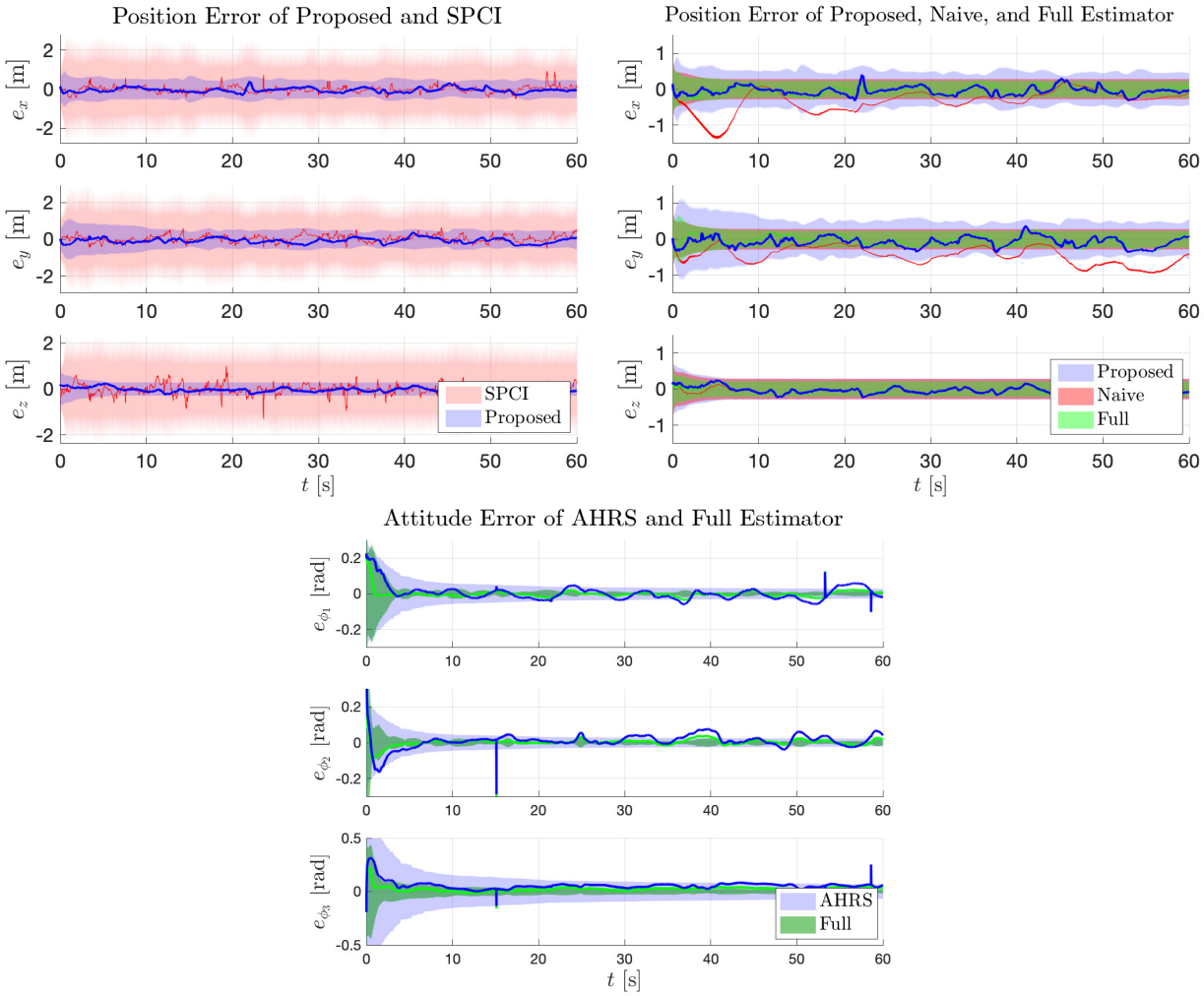


Figure 7.7: The error trajectories associated with the slow-pace experimental run for the 4 position estimators, and the AHRS. The AHRS is shared by all 3 cascaded estimators. The shaded regions correspond to the $\pm 3\sigma$ bounds, and the colour of each error trajectory and covariance region are the same.

The $\pm 3\sigma$ bound error plots for the 4 estimators on the slow pace experiment are shown in Figure 7.7, and an RMSE-based comparison on the three datasets is shown in Table 7.3. As expected, the naive cascaded filter performs poorly, while the performance of the

proposed cascaded filter beats the SPCI approach and is comparable to the full estimator. Not only does the proposed framework achieve a lower position RMSE, but it is also less conservative as compared to the SPCI. This is further displayed in Figure 7.8, where the KL divergence [114, Chapter 9] measure shows that the estimated distribution of the proposed estimator is closer to that of the full estimator, which is the best available estimate of the true distribution.

Table 7.3: RMSE of the different estimators on 3 experimental runs.

	RMSE		
	Slow	**Medium**	**Fast**
Full - Position (m)	0.2187	0.19613	0.17518
Proposed - Position (m)	**0.23138**	**0.24818**	**0.28261**
SPCI - Position (m)	0.26690	0.32152	0.32045
Naive - Position (m)	0.69616	0.50807	1.3655
Full - Attitude (rad)	0.04400	0.03225	0.02896
AHRS - Attitude (rad)	0.08934	0.07669	0.13757

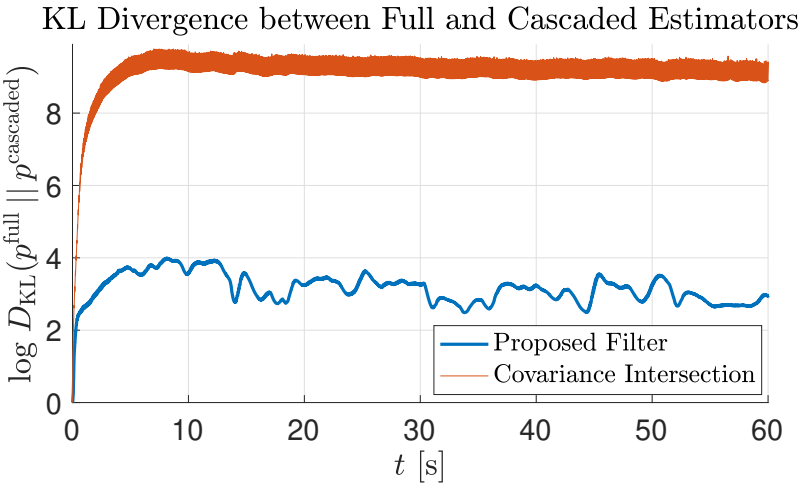


Figure 7.8: The logarithm of the KL divergence measure associated with the slow-pace experimental run. The KL divergence is computed between the estimated distribution of each cascaded estimator and the full estimator.

Due to imperfectly calibrated hardware, further unmodelled phenomena such as indoor magnetic perturbations, and nonlinearities of the system, the AHRS in the slow-pace experiment takes over 15 seconds to achieve near steady-state conditions, as shown in Figure 7.7. Additionally, the AHRS performs poorly particularly in the fast pace experiment, as shown in Table 7.3. However, the proposed framework still achieves a performance comparable to the full estimator, which shows that the approximation discussed in Section 7.3.3 is reasonable, even for such complex systems. Additionally, the worse attitude estimates as compared

to the simulation runs means that the cross-covariances between the AHRS and the position estimator are more significant, which is why the naive estimator performs so poorly. This emphasizes the significance of the proposed estimator, particularly when estimation error is persistent in the feeding filter.

7.6 Conclusion and Future Work

In this chapter, the importance of modelling cross-covariances in cascaded filtering is addressed, and a novel approach is introduced using the sigma point transformation. The main contribution is the probabilistic approximation of the propagation of cross-covariance terms when the local filters do not estimate a common set of states. The proposed approach is compared to the naive filtering approach and a Covariance Intersection filter approach both in simulations and in experiments. Superior, consistent results are achieved for both a linear and a nonlinear problem. Comparable performance to a full non-cascaded estimator is also achieved, but with all the advantages of cascaded filtering such as improved flexibility and reduced computational complexity.

The proposed cascaded filtering algorithm is tested on general problems typically encountered in robotics. Future work will involve the implementation of this algorithm in the multi-robot framework presented in Chapter 6. This will involve having each robot broadcast their state estimate, and thus allowing other robots to fuse this state estimate in a consistent manner using the probabilistic approximation presented in this framework. Additionally, given the generalizability of the approach presented in this chapter, future work will involve open-sourcing the code to allow the robotics community to implement and test this approach on many different loosely-coupled filtering systems.

Having addressed a potential approach to allowing collaboration among robots, the next chapter deviates slightly from the general theme of this thesis. The goal of the next chapter is to further emphasize the generalizability of the proposed algorithms in this thesis, and to do so, the focus will now shift to an application of UWB-based navigation made possible by simultaneously designing the ranging protocol and the state estimator. Rather than considering a team of robots, a single robot in an environment with spaced-out anchors at unknown locations will be considered, and the goal is to teach the robot a trajectory that can be repeated autonomously many times.

Chapter 8

Ultra-Wideband Teach and Repeat

Summary

Autonomously retracing a manually-taught path is desirable for many applications, and Teach and Repeat (T&R) algorithms present an approach that is suitable for long-range autonomy. In this chapter, the concepts of passive listening and simultaneous clock synchronization and pose estimation is adopted from Chapter 6 to propose an UWB ranging-based T&R solution for vehicles with limited resources. By fixing single UWB transceivers at distant and unknown locations in an indoor environment, a robot with 3 UWB transceivers builds a locally-consistent map during a *teach pass* by fusing range measurements under a custom ranging protocol with an on-board IMU and height measurements. The robot then uses information from the teach pass to retrace the same trajectory autonomously. The proposed ranging protocol and T&R algorithm are validated in simulation, where it is shown that the robot can successfully retrace the taught trajectory despite estimating a globally-inconsistent map of the environment.

8.1 Introduction

UWB-based positioning typically involves having a few static transceivers, or *anchors*, at fixed known locations communicating with one or more mobile transceivers, or *tags*, in order to obtain range measurements between them. This then allows positioning of the robot through trilateration or filtering [14], [22], [85]. Alternatively, to overcome the need for a fixed infrastructure, the most common approach for indoor navigation is simultaneous localization and mapping (SLAM). Most SLAM algorithms are vision- [115], [116] or LIDAR-based [117], which requires computationally-heavy processing and large storage to process

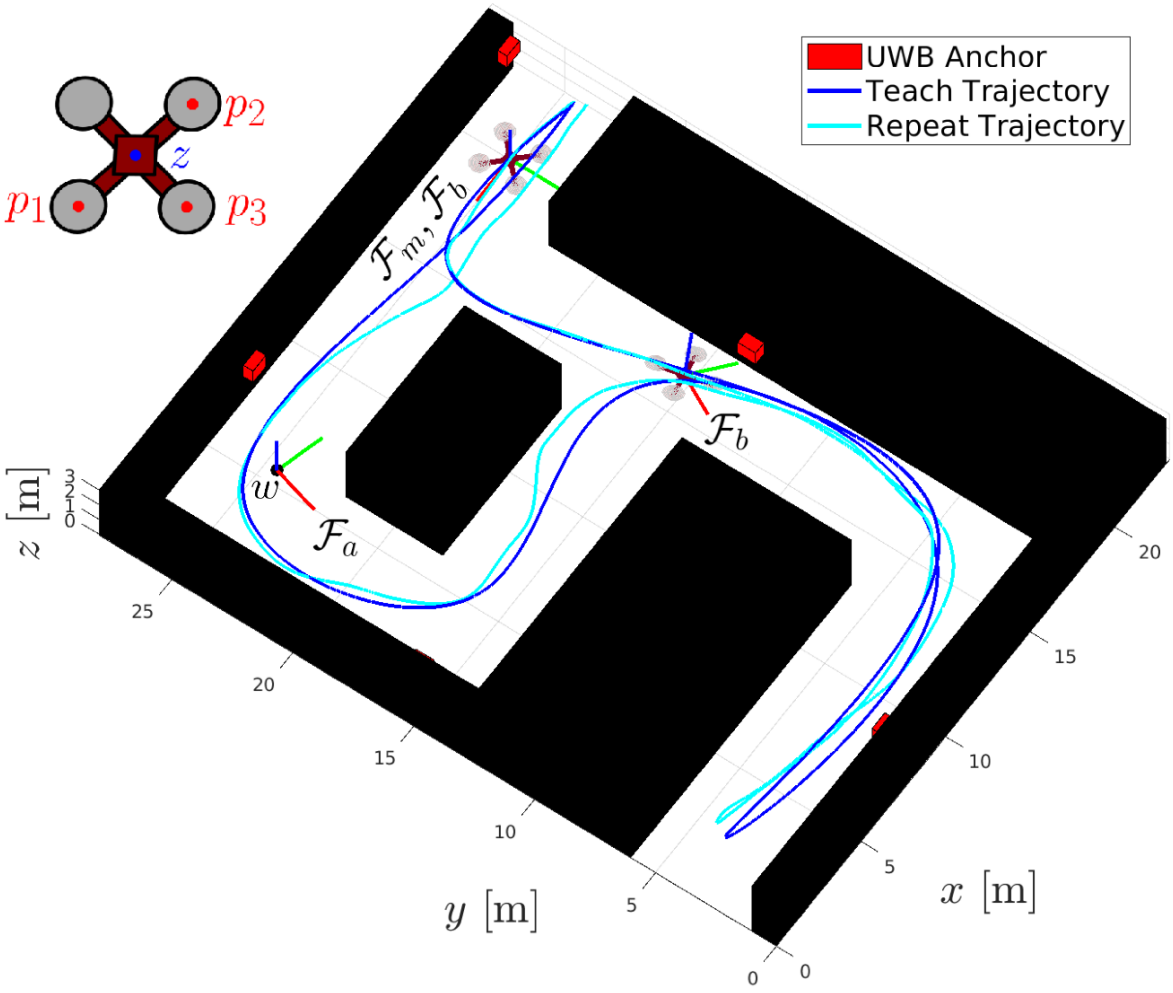


Figure 8.1: A 3D indoor environment with spaced-out UWB anchors fixed at unknown positions. The robot is equipped with 3 UWB ranging tags, and successfully retraces the teach trajectory autonomously in the repeat pass.

and store images or point clouds. Additionally, SLAM is oftentimes restricted to small areas, as global map consistency is only achieved through complex data association and loop closure algorithms that are difficult for larger maps and not always possible [118]. Range-based SLAM has also been proposed mainly for fixed-infrastructure applications [53], [54], while [119] proposes dropping beacons from a quadcopter and utilizing the associated range measurements in a SLAM framework.

A more recent approach to overcoming the pitfalls of SLAM is T&R, where a robot learns a locally-consistent map of the environment during a *teach pass*, and then retraces

the trajectory it took during a *repeat pass* using the learnt map [120]. As long as a robot behaves the same way every time relative to its local features, the robot is able to retrace the same trajectory [120], [121]. This has been studied using stereo cameras [120] and LIDAR [122] on ground vehicles, and also with monocular cameras [123] and aerial vehicles [124]. As with all vision-based algorithms, difficulties arise when the environment changes due to weather or other external factors. This requires relying on colour-constant imaging [125], using multiple experiences simultaneously [126], and/or deep learning approaches for image association and localization [127], [128].

The contributions of this chapter are threefold and revolve around combining the advantages of using a T&R framework with those of using UWB-based range measurements for localization.

- Firstly, a ranging protocol is proposed to allow a 3-tag agent to synchronize the clocks of its tags and receive 3 range measurements from a fixed anchor with only one two-way ranging (TWR) transaction. This therefore allows necessary additional information for localization [1] without compromising the frequency at which range measurements are recorded.

- Secondly, a novel UWB-based T&R framework is presented for a vehicle moving in an environment with spaced-out fixed anchors at unknown locations over a large area. The robot also utilizes an inertial measurement unit (IMU) and height measurements. As is standard with T&R, the robot is manually controlled over a trajectory that is then autonomously retraced.

- Lastly, the presented framework is tested in simulation using the environment shown in Figure 8.1 and is shown to achieve sub-metre tracking performance.

The rest of this chapter is structured as follows. The problem is formulated in Section 8.2, and the ranging protocol is presented in Section 8.3. The proposed teach and repeat passes are covered in Sections 8.4 and 8.5, respectively. The results are then discussed in Section 8.6, and some concluding remarks are given in Section 8.7.

8.1.1 Notation

Anchor i is denoted s_i, and the set of all static anchors is denoted $\mathcal{S}$. Tag i on the robot is denoted p_i, and the IMU is assumed to be at point z, as shown in Figure 8.1. The subscript k is used throughout the chapter to denote the k^{th} time-step as before, while a subscript $i : i + j$ denotes all the variables at time-steps $k \in \{i, \dots, i + j\}$. Meanwhile, K is used to denote the total number of time-steps in each of the teach and repeat passes. When

applicable, the subscript $0:K$ is omitted for conciseness when addressing the full trajectory. The superscripts $\mathfrak{t}$, $\mathfrak{r}$, and $\mathfrak{i}$ are used to allocate variables to the teach pass, the repeat pass, and the initialization phase of the repeat pass, respectively.

The reference frames used in this text are shown in Figure 8.1. A local reference frame is denoted $\mathcal{F}_a$, while the body-fixed reference frame is denoted $\mathcal{F}_b$. The map frame $\mathcal{F}_m$ is fixed to the body frame at $k = 0$, and differs from $\mathcal{F}_a$ only through a rotation about the z-axis and a translation about the x- and y-axes. The vector notations and timestamp notation suggested in Section 6.1.1 will also be used in this chapter. Meanwhile, $\tilde{\mathbf{r}}_q^{ij}$ is used to represent a position in $\mathbb{R}^2$, and the heading $\tilde{\theta}_{\ell q}$ is associated with the DCM $\tilde{\mathbf{C}}_{\ell q} \in SO(2)$, an element of the 2D special orthogonal group $SO(2)$.

Additionally, $(\cdot)^\times$ denotes the skew-symmetric cross-product matrix operator in $\mathbb{R}^3$, and $\|\cdot\|_{\mathbf{\Sigma}} = \sqrt{(\cdot)^\mathsf{T}\mathbf{\Sigma}^{-1}(\cdot)}$ denotes the weighted Euclidean norm. Lastly, $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ and $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ represent process and measurement Gaussian white noise, respectively. The random variables and their covariance matrices always share the same subscripts and superscripts $\big($e.g., $\mathbf{w}_b^{\mathrm{acc,t}} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_b^{\mathrm{acc,t}})\big)$.

8.2 Problem Formulation

Consider a robot with 3 UWB tags in an environment with n spaced-out and fixed UWB anchors at unknown locations $\mathbf{r}_a^{s_i w}$, $i \in \{1, \ldots, n\}$, as shown in Figure 8.1. Initially, the robot is manually driven on a trajectory denoted $\boldsymbol{\pi}^\mathfrak{t}$ that has approximately the same starting and ending point on a flat ground. As the robot moves, it records measurements from an IMU, a height sensor, and range measurements with any anchor that lies within communication range. The robot is tasked with estimating the trajectory it is following such that it can be autonomously repeated in the future. Additionally, the robot maps the anchors as they are detected by estimating their position. These two tasks consequently fall under what is termed the teach pass, as the robot learns during this process the trajectory it must follow and a map of the environment it encounters, which is just a map of the UWB anchor positions.

Despite having access to only the estimated teach trajectory $\hat{\boldsymbol{\pi}}^\mathfrak{t}$ and the estimated anchor positions, the robot then attempts to repeat the teach trajectory $\boldsymbol{\pi}^\mathfrak{t}$ autonomously. This is termed the repeat pass. The robot first estimates its pose relative to the initial pose in the teach pass, since the robot at the end of the teach trajectory does not necessarily land at the same exact position and with the same heading. The repeat pass then involves using a state estimator and a trajectory-tracking controller to retrace the teach trajectory. A summary of the major processing blocks is given in Figure 8.2.

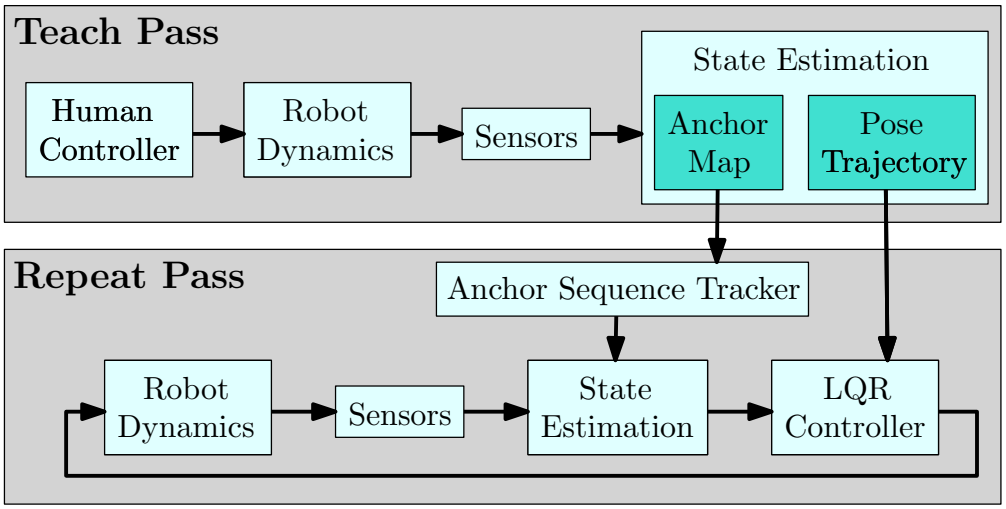


Figure 8.2: A high-level overview of the major processing blocks of UWB T&R.

As is common with SLAM approaches, the robot estimates the trajectory relative to its initial pose. Therefore, the trajectory states to be estimated in the teach or repeat pass are

$$\boldsymbol{\pi} = \begin{bmatrix} (\mathbf{r}_m^{zw})^\mathsf{T} & (\mathbf{v}_m^{zw/a})^\mathsf{T} & (\boldsymbol{\phi}_{mb})^\mathsf{T} \end{bmatrix}^\mathsf{T}. \tag{8.1}$$

In the teach pass, the estimated trajectory is stored alongside the estimated anchor positions, while in the repeat pass the estimated state is input to the controller.
The continuous-time kinematics of the trajectory in the presence of IMU biases are given by

$$ {}^a\dot{\mathbf{r}}_m^{zw} = \mathbf{v}_m^{zw/a}, \tag{8.2}$$

$$ {}^a\dot{\mathbf{v}}_m^{zw/a} = \mathbf{C}_{mb}\left(\mathbf{u}_b^{\mathrm{acc}} + \boldsymbol{\beta}_b^{\mathrm{acc}} + \mathbf{w}_b^{\mathrm{acc}}\right) + \mathbf{g}_m, \tag{8.3}$$

$$ \dot{\mathbf{C}}_{mb} = \mathbf{C}_{mb}\left(\mathbf{u}_b^{\mathrm{gyr}} + \boldsymbol{\beta}_b^{\mathrm{gyr}} + \mathbf{w}_b^{\mathrm{gyr}}\right)^\times, \tag{8.4}$$

where $\mathbf{u}_b^{\mathrm{acc}}$, $\boldsymbol{\beta}_b^{\mathrm{acc}}$, and $\mathbf{w}_b^{\mathrm{acc}}$ denote the accelerometer measurement, bias, and Gaussian white noise, respectively, and $\mathbf{u}_b^{\mathrm{gyr}}$, $\boldsymbol{\beta}_b^{\mathrm{gyr}}$, and $\mathbf{w}_b^{\mathrm{gyr}}$ denote the gyroscope measurement, bias, and Gaussian white noise, respectively. Meanwhile, $\mathbf{g}_m$ is the gravity vector resolved in the map frame. Assuming that the robot lies on a flat ground at the beginning of the teach pass and that $\mathbf{g}_a = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}$, then $\mathbf{g}_m = \mathbf{g}_a$.

In addition to the trajectory states, bias states must be estimated to correct the IMU measurements, and clock states must be estimated to correct the UWB range measurements. The accelerometer and gyroscope suffer from slowly-changing biases that can degrade the estimation performance if not properly addressed. It is possible to estimate the absolute IMU biases in this chapter due to the presence of static anchors. The evolution of the biases

is modelled as a random walk,

$$\dot{\boldsymbol{\beta}}_b^{\text{acc}} = \mathbf{w}_b^{\beta,\text{acc}}, \qquad \dot{\boldsymbol{\beta}}_b^{\text{gyr}} = \mathbf{w}_b^{\beta,\text{gyr}}. \tag{8.5}$$

The robot has 3 UWB tags, Tags p_0, p_1, and p_2. It is assumed that the positions of the tags relative to the IMU at point z are known in the body frame of the robot; in other words, the quantities $\mathbf{r}_b^{p_1 z}$, $\mathbf{r}_b^{p_2 z}$, and $\mathbf{r}_b^{p_3 z}$ are measured and known beforehand. The communicating UWB tags rely on ToF measurements to calculate distance measurements, but suffer from having unsynchronized clocks. To overcome this issue, a ranging protocol similar to the one proposed in Section 6.4 is utilized. This involves a direct DS-TWR transaction between Tag p_0 on the robot and any anchors within ranging distance of the robot. Meanwhile, Tags p_1 and p_2 passively listen-in on the transmitted DS-TWR messages, as further discussed in Section 8.3. Therefore, p_1 and p_2 must estimate their clock offset and skew relative to Tag p_0, denoted by $\tau_{p_i p_0}$ and $\gamma_{p_i p_0}$, respectively for $i \in \{1, 2\}$. The process model of the clock states is modelled as in 2.2.

Both the biases and clock states are estimated during the teach pass and the repeat pass, but neither are recorded as part of the trajectory or fed into the controller. The sole purpose of estimating these states is to correct the IMU and range measurements to mitigate their effect on the performance of the estimators. The state being estimated at time-step k is then

$$\mathbf{x}_k = \begin{bmatrix} \boldsymbol{\pi}_k^{\mathsf{T}} & (\boldsymbol{\beta}_k^{\text{acc}})^{\mathsf{T}} & (\boldsymbol{\beta}_k^{\text{gyr}})^{\mathsf{T}} & (\mathbf{x}_k^{\text{UWB}})^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}},$$

where $\mathbf{x}_k^{\text{UWB}} = [\tau_{p_1 p_0,k} \ \tau_{p_2 p_0,k} \ \gamma_{p_1 p_0,k} \ \gamma_{p_2 p_0,k}]^{\mathsf{T}}$. In order to estimate $\mathbf{x}_k$, two exteroceptive sensors are available. Firstly, height measurements are modelled as

$$y^{\text{h}} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{r}_m^{zw} + \eta^{\text{h}}. \tag{8.6}$$

Meanwhile, range measurements and the ranging protocol associated with the UWB tags are discussed in Section 8.3.

8.3 Ranging Protocol

In order to ensure observability of the anchor positions relative to the robot, three tags are placed on the robot [1]. When Anchor s_i is within communication range of the robot, the proposed ranging protocol is shown in Figure 8.3. This involves Tag p_0 performing the DS-TWR protocol presented in Chapter 4 with Anchor s_i, while Tags p_1 and p_2 listen-in on all signals in a manner similar to Section 6.4. Therefore, the 12 recorded timestamps can be

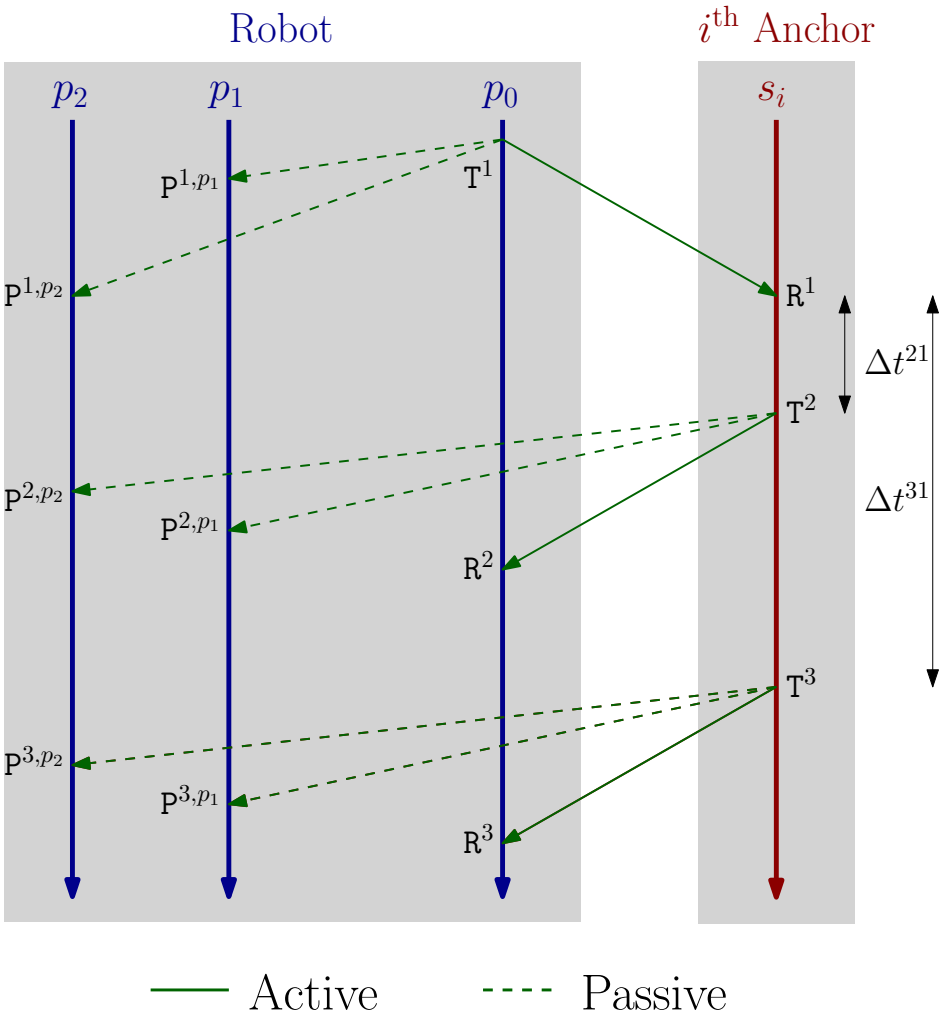


Figure 8.3: A schematic of the time instances where timestamps are recorded by the three tags on the robot while Tag p_0 performs DS-TWR with Anchor s_i. Shown here are also Transceivers p_1 and p_2 on the Robot passively listening, where the time instance corresponding to the i^{th} passive reception at Transceiver j is denoted $\mathtt{P}^{i,j}$. Additionally, the intervals Δt^{21} and Δt^{31} are defined in this chapter as shown in the figure.

modelled as a function of the distances and clock states between the tags and the anchors. The timestamps recorded by Tag p_0 are

$$\tilde{\mathtt{T}}^1_{p_0} = \mathtt{T}^1 + \tau_{p_0}(\mathtt{T}^1) + \eta^1_{p_0}, \tag{8.7}$$

$$\tilde{\mathtt{R}}^2_{p_0} = \mathtt{T}^1 + \frac{2}{c} d^{s_i p_0} + \Delta t^{21} + \tau_{p_0}(\mathtt{R}^2) + \eta^2_{p_0}, \tag{8.8}$$

$$\tilde{\mathtt{R}}^3_{p_0} = \mathtt{T}^1 + \frac{2}{c} d^{s_i p_0} + \Delta t^{31} + \tau_{p_0}(\mathtt{R}^3) + \eta^3_{p_0}, \tag{8.9}$$

while the timestamps recorded by Anchor s_i are

$$\tilde{\mathtt{R}}^1_{s_i} = \mathtt{T}^1 + \frac{1}{c}d^{s_i p_0} + \tau_{s_i}(\mathtt{R}^1) + \eta^1_{s_i}, \tag{8.10}$$

$$\tilde{\mathtt{T}}^2_{s_i} = \mathtt{T}^1 + \frac{1}{c}d^{s_i p_0} + \Delta t^{21} + \tau_{s_i}(\mathtt{T}^2) + \eta^2_{s_i}, \tag{8.11}$$

$$\tilde{\mathtt{T}}^3_{s_i} = \mathtt{T}^1 + \frac{1}{c}d^{s_i p_0} + \Delta t^{31} + \tau_{s_i}(\mathtt{T}^3) + \eta^3_{s_i}. \tag{8.12}$$

Additional measurements are also available at the robot due to its passively-listening tags. This yields the following additional timestamp measurements,

$$\tilde{\mathtt{P}}^{1,j}_j = \mathtt{T}^1 + \frac{1}{c}d^{p_0 j} + \tau_j(\mathtt{P}^{1,j}) + \eta^1_j, \tag{8.13}$$

$$\tilde{\mathtt{P}}^{2,j}_j = \mathtt{T}^1 + \frac{1}{c}d^{s_i p_0} + \frac{1}{c}d^{s_i j} + \Delta t^{21} + \tau_j(\mathtt{P}^{2,j}) + \eta^2_j, \tag{8.14}$$

$$\tilde{\mathtt{P}}^{3,j}_j = \mathtt{T}^1 + \frac{1}{c}d^{s_i p_0} + \frac{1}{c}d^{s_i j} + \Delta t^{31} + \tau_j(\mathtt{P}^{3,j}) + \eta^3_j, \tag{8.15}$$

where $j \in \{p_1, p_2\}$.

In a manner similar to Section 6.4, and following the same assumptions highlighted there, the following pseudomeasurements are then proposed to be used in the state estimator. Given that only clock states between the tags on the robot are being estimated, the pseudomeasurements are chosen such that any clock states associated with the anchors are cancelled out. Firstly, the standard DS-TWR range measurements from timestamps (8.7)-(8.12) can be written as

$$\begin{aligned} y^{\text{tof}} &= \frac{1}{2}\left(\left(\tilde{\mathtt{R}}^2_{p_0} - \tilde{\mathtt{T}}^1_{p_0}\right) - \frac{\tilde{\mathtt{R}}^3_{p_0} - \tilde{\mathtt{R}}^2_{p_0}}{\tilde{\mathtt{T}}^3_{s_i} - \tilde{\mathtt{T}}^2_{s_i}}\left(\tilde{\mathtt{T}}^2_{s_i} - \tilde{\mathtt{R}}^1_{s_i}\right)\right) \\ &\approx \frac{1}{c}d^{s_i p_0} + \frac{1}{2}\left(\eta^2_{p_0} - \eta^1_{p_0} - \eta^2_{s_i} + \eta^1_{s_i}\right). \end{aligned} \tag{8.16}$$

Meanwhile, from the passive-listening measurements, the following three pseudomeasurements can be generated per passively-listening tag,

$$y^{\text{p},1}_j = \tilde{\mathtt{P}}^{1,j}_j - \tilde{\mathtt{T}}^1_{p_0} = \frac{1}{c}d^{p_0 j} + \tau_{j p_0} + \eta^1_j - \eta^1_{p_0}. \tag{8.17}$$

$$\begin{aligned} y^{\text{p},2}_j &= \tilde{\mathtt{P}}^{2,j}_j - \tilde{\mathtt{T}}^2_{s_i} \\ &= \frac{1}{c}d^{s_i j} - \frac{1}{c}d^{s_i p_0} + \tau_{j p_0} + \gamma_{j p_0}\Delta t^{21} + \eta^2_j - \eta^2_{p_0} \end{aligned} \tag{8.18}$$

$$\begin{aligned} y^{\text{p},3}_j &= \tilde{\mathtt{P}}^{3,j}_j - \tilde{\mathtt{T}}^3_{s_i} \\ &= \frac{1}{c}d^{s_i j} - \frac{1}{c}d^{s_i p_0} + \tau_{j p_0} + \gamma_{j p_0}\Delta t^{31} + \eta^3_j - \eta^3_{p_0}, \end{aligned} \tag{8.19}$$

where $j \in \{p_1, p_2\}$. As discussed in Section 6.4, the unknown values Δt^{21} and Δt^{31} are approximated to the first order as

$$\gamma_{jp_0}\Delta t^{21} \approx \gamma_{jp_0}(\tilde{\mathtt{T}}^2_{s_i} - \tilde{\mathtt{R}}^1_{s_i}),$$
$$\gamma_{jp_0}\Delta t^{31} \approx \gamma_{jp_0}(\tilde{\mathtt{T}}^3_{s_i} - \tilde{\mathtt{R}}^1_{s_i}).$$

Note that the pseudomeasurements (8.16), (8.18), and (8.19) are a function of distances to the anchors, while (8.17) is independent of the anchor positions. The distance quantities can be written as a function of the known quantities and the states and anchor positions that are being estimated as

$$d^{s_i\ell} = \left\| \mathbf{r}^{zw}_m - \mathbf{r}^{s_i w}_m + \mathbf{C}_{mb}\mathbf{r}^{\ell z}_b \right\|, \tag{8.20}$$
$$d^{p_0 j} = \left\| \mathbf{r}^{p_0 z}_b - \mathbf{r}^{jz}_b \right\|, \tag{8.21}$$

for $\ell \in \{p_0, p_1, p_2\}$ and $j \in \{p_1, p_2\}$.

Additionally note that none of the pseudomeasurements are a function of the clock states of the anchors, which are not being estimated as each anchor is not always within communication range of the robot. Nonetheless, the choice of the pseudomeasurements results in the same noise random variables showing up in (8.16)-(8.19); therefore, the errors in these pseudomeasurements are correlated, which must be properly modelled when designing state estimators in a manner similar to Section 6.4.5.

8.4 Teach Pass

Given that the positions of the anchors are unknown, the anchors must be localized when encountered in the teach pass. Characteristic of T&R algorithms is the lack of loop closure enforcements. When the robot encounters an anchor $i \in \mathcal{S}$ for the first time, it estimates and stores in memory the anchor position. However, when the robot encounters the same anchor again in the future, the previously estimated position is not used, as this constitutes a loop closure that requires correcting all the states estimated between the two instances where the same anchor is detected. In a T&R scenario, the robot disregards the fact that this anchor was previously encountered and re-localizes the anchor position, since global consistency of the map is not necessary.

The robot tracks the sequence of initialized anchors and each anchor's ID and estimated position. Letting the ℓ^{th} encountered anchor be Anchor i, the initialization result is encoded in an ordered triple $\left(\hat{\mathbf{r}}^{\ell w}_m, i, \ell\right)$. The estimated anchor map is then represented as a set of

ordered triples

$$\mathcal{M} = \{(\hat{\mathbf{r}}_m^{\ell w}, i, \ell) \mid \hat{\mathbf{r}}_m^{\ell w} \in \mathbb{R}^3, i \in \mathcal{S}, \ell \in \mathbb{Z}^{\geq 0}\}.$$

where $\mathbb{Z}^{\geq 0}$ denotes the set of non-negative integers.

During the teach pass, the robot keeps track of the set of *currently active anchors* $\mathcal{C} \subset \mathcal{S}$, the anchors that lie within communication range of the robot. This allows the robot to use recently-localized anchors to correct its state estimate. When an active Anchor i is no longer within communication range, Anchor i is dropped from the set of active anchors such that when Anchor i is detected again, it is re-initialized. The teach pass sequence tracker is summarized in Algorithm 3.

Algorithm 3 Teach pass anchor sequence tracker.

The *id* is the ID of the communicating anchor, $\mathcal{C}$ is the current set of active anchors, and $\mathcal{M}$ is the anchor map. For all elements $(\mathbf{r}, i, \ell) \in \mathcal{M}$, the term "most recent" corresponds to the element with the largest quantity ℓ among the elements with $i = id$.

function TEACHSEQUENCETRACKER(*anchor*, $\mathcal{C}$, $\mathcal{M}$)
  if *anchor* $\in \mathcal{C}$ **then**
    $\hat{\mathbf{r}}_m^{\ell w}$ = GETMOSTRECENTPOSITION(*anchor*, $\mathcal{M}$)
  else
    $\hat{\mathbf{r}}_m^{\ell w}$ = INITIALIZEANCHOR(*anchor*)
    $n_{\mathcal{M}}$ = number of elements in $\mathcal{M}$
    $\mathcal{M} = \mathcal{M} \cup \{\hat{\mathbf{r}}_m^{\ell w}, anchor, n_{\mathcal{M}}\}$
    $\mathcal{C} = \mathcal{C} \cup \{anchor\}$
  end if
  for $j \in \mathcal{C}$ **do**
    if j no longer within communication range **then**
      $\mathcal{C} = \mathcal{C} \backslash \{j\}$
    end if
  end for
  return $\hat{\mathbf{r}}_m^{\ell w}$, $\mathcal{C}$, $\mathcal{M}$
end function

8.4.1 Anchor Localization

Assume that at time-step k Anchor $i \notin \mathcal{C}_k$ is detected as the ℓ^{th} component of the sequence of detected anchors. In order to localize the anchor, an optimization problem is formulated using the predicted state $\check{\mathbf{x}}_k^{\mathsf{t}}$, its uncertainty $\check{\mathbf{P}}_k^{\mathsf{t}}$, and the interoceptive and exteroceptive measurements recorded over a short time window from k to $k + \lambda$, where λ is the length of the window. The goal is to estimate the anchor position $\hat{\mathbf{r}}_m^{\ell w}$ to be added to the anchor map and the state estimates $\hat{\mathbf{x}}_{k:k+\lambda}^{\mathsf{t}}$ to add $\hat{\boldsymbol{\pi}}_{k:k+\lambda}^{\mathsf{t}}$ to the estimated teach pose trajectory.

To formulate a discrete-time optimization problem over the states, the process models

(8.2)-(8.5), alongside the clocks' process model from (2.10), are concatenated and discretized to yield a system process model of the form

$$\mathbf{x}_{k+1}^{\mathrm{t}} = \mathbf{f}_k\left(\mathbf{x}_k^{\mathrm{t}}, \mathbf{u}_k^{\mathrm{t}}, \mathbf{w}_k^{\mathrm{t}}\right), \tag{8.22}$$

where $\mathbf{u}_k^{\mathrm{t}}$ represents the interoceptive measurements. Similarly, the height measurements (8.6) and UWB pseudomeasurements (8.16)-(8.19) at time-step k are concatenated and represented using the discrete-time measurement model

$$\mathbf{y}_k^{\mathrm{t}} = \mathbf{g}_k\left(\mathbf{x}_k^{\mathrm{t}}, \mathbf{r}_m^{s_i w}, \boldsymbol{\eta}_k^{\mathrm{t}}\right), \qquad i \in \mathcal{C}_k. \tag{8.23}$$

The optimization problem is then formulated as a standard discrete-time batch problem with Gaussian errors, thus the *maximum a posteriori* (MAP) estimate is the solution to

$$\begin{aligned}\left(\hat{\mathbf{x}}_{k:k+\lambda}^{\mathrm{t}}, \hat{\mathbf{r}}_m^{\ell w}\right) = \underset{\left(\mathbf{x}_{k:k+\lambda}^{\mathrm{t}}, \mathbf{r}_m^{\ell w}\right)}{\arg\min} & \left\|\mathbf{x}_k^{\mathrm{t}} - \check{\mathbf{x}}_k^{\mathrm{t}}\right\|_{\check{\mathbf{P}}_k^{\mathrm{t}}}^2 + \sum_{j=k}^{k+\lambda}\left\|\mathbf{y}_j^{\mathrm{t}} - \mathbf{g}_j\left(\mathbf{x}_j^{\mathrm{t}}, \mathbf{r}_m^{\ell w}, \mathbf{0}\right)\right\|_{\mathbf{R}_j^{\mathrm{t}}}^2 \\ & + \sum_{j=k}^{k+\lambda-1}\left\|\mathbf{x}_{j+1}^{\mathrm{t}} - \mathbf{f}_j\left(\mathbf{x}_j^{\mathrm{t}}, \mathbf{u}_j^{\mathrm{t}}, \mathbf{0}\right)\right\|_{\mathbf{Q}_j^{\mathrm{t}}}^2 + R_{\mathrm{h}}^{-1}\left(h - [0\ 0\ 1]\,\mathbf{r}_m^{\ell w}\right)^2.\end{aligned}$$

The last error term corresponds to a prior on the vertical distance of the anchor to the floor being h. This prevents the estimate from converging to the local minimum associated with the *flip ambiguities*, as discussed in Section 3.4.2, that would result in the anchor being initialized below ground level. The covariance R_{h} can be tuned to tailor to the user's confidence in the height h.

The initial iterate of the states $\left(\hat{\mathbf{x}}_{k:k+\lambda}^{\mathrm{t}}\right)^{(0)}$ is obtained by dead-reckoning, while the initial iterate of the anchor position $\left(\hat{\mathbf{r}}_m^{\ell m}\right)^{(0)}$ is obtained by assuming a height of h and analytically solving for the other two components using the range measurement at time-step k. The iterates are updated using Gauss-Newton or Levenberg-Marquardt until convergence. Once the optimizer converges, the states $\hat{\boldsymbol{\pi}}_{k:k+\lambda}^{\mathrm{t}}$ are added to the estimated pose trajectory, and the current state of the state estimator is updated to be $\hat{\mathbf{x}}_{k+\lambda}^{\mathrm{t}}$. Meanwhile, Anchor i is added to the current set of active anchors $\mathcal{C}_{k+\lambda}$, and the ordered triple $(\hat{\mathbf{r}}_m^{\ell w}, i, \ell)$ is added to the anchor map $\mathcal{M}$.

8.4.2 Trajectory Estimation

The state estimator is a basic EKF that provides an estimate $\mathbf{x}_k^{\mathrm{t}} \sim \mathcal{N}\left(\hat{\mathbf{x}}_k^{\mathrm{t}}, \hat{\mathbf{P}}_k^{\mathrm{t}}\right)$ at every time-step $k \in \{0, \ldots, K\}$ to be stored as the trajectory to be tracked in the repeat pass. If a

new anchor is detected at time-step k, the state estimator does not correct the estimates at time-step k, but the predicted state $\mathbf{x}_k^{\mathrm{t}} \sim \mathcal{N}\left(\check{\mathbf{x}}_k^{\mathrm{t}}, \check{\mathbf{P}}_k^{\mathrm{t}}\right)$ is input into the anchor initializer. The state estimator then resumes predicting and correcting at time-step $k+\lambda$ with state estimate $\mathbf{x}_{k+\lambda}^{\mathrm{t}} \sim \mathcal{N}\left(\hat{\mathbf{x}}_{k+\lambda}^{\mathrm{t}}, \hat{\mathbf{P}}_{k+\lambda}^{\mathrm{t}}\right)$, which comes from solving the optimization problem formulated in Section 8.4.1. Additionally, when an encountered anchor is initialized and is active, the estimated anchor position is used in the correction step as if it is the true anchor position, which leads to globally-inconsistent but locally-consistent maps.

8.5 Repeat Pass

8.5.1 State Initialization

At the end of the teach pass, the robot is assumed to land on the floor close to its original take-off location, such that the initial pose of the repeat trajectory is similar to that of the teach trajectory. However, it is unlikely that the robot lands at the same exact position and that the robot is oriented the same way it was when it first took off. Under a flat-floor assumption, the robot would have the same height, pitch and roll, but the 2D position and heading are not necessarily the same. Therefore, before attempting to retrace the teach pass trajectory, the robot must estimate its initial 2D position, denoted $\tilde{\mathbf{r}}_m^{zw,\mathrm{i}} = \mathbf{D}\mathbf{r}_{m,0}^{zw,\mathrm{r}} \in \mathbb{R}^2$, where $\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, as well as its heading, denoted using a DCM $\tilde{\mathbf{C}}_{mb}^{\mathrm{i}} \in SO(2)$. During the repeat pass's state initialization, the robot remains static, and therefore the process model associated with the 2D position and heading is simply

$$ {}^a\dot{\tilde{\mathbf{r}}}_m^{zw,\mathrm{i}} = \mathbf{0}, \quad \dot{\tilde{\mathbf{C}}}_{mb}^{\mathrm{i}} = \mathbf{0}. $$

As before, the robot still needs to estimate its biases and clock states. The process model of the biases and clock states are as given in (8.5) and (2.10), respectively. Therefore, the state vector to be initialized is of the form

$$ \tilde{\mathbf{x}}^{\mathrm{i}} = \left[(\tilde{\boldsymbol{\pi}}^{\mathrm{i}})^{\mathsf{T}} \;\; (\boldsymbol{\beta}^{\mathrm{acc,i}})^{\mathsf{T}} \;\; (\boldsymbol{\beta}^{\mathrm{gyr,i}})^{\mathsf{T}} \;\; (\mathbf{x}^{\mathrm{UWB,i}})^{\mathsf{T}} \right]^{\mathsf{T}}, $$

where $\tilde{\boldsymbol{\pi}}^{\mathrm{i}} = \begin{bmatrix} \left(\tilde{\mathbf{r}}_m^{zw,\mathrm{i}}\right)^{\mathsf{T}} & \tilde{\theta}_{mb}^{\mathrm{i}} \end{bmatrix}^{\mathsf{T}}$.

The anchor-independent range measurements are of the same form as given in (8.17). However, the anchor-dependent measurements (8.16), (8.18), and (8.19) are a function of the robot's pose, and therefore should be written explicitly as a function of the 2D states by isolating the components of the third dimension. This is done by simply rewriting the

distance model (8.20) as

$$d^{s_i\ell} = \sqrt{n_k^\ell + m_k^\ell} \tag{8.24}$$

for $\ell \in \{p_0, p_1, p_2\}$, where

$$n^\ell = \left\| \tilde{\mathbf{r}}_m^{zw,\mathrm{i}} - \mathbf{D}\mathbf{r}_m^{s_i w} + \tilde{\mathbf{C}}_{mb}^{\mathrm{i}} \mathbf{D}\mathbf{r}_b^{p_\ell z} \right\|^2, \tag{8.25}$$

$$m^\ell = \left\| \mathbf{E}\mathbf{r}_m^{p_\ell z} - \mathbf{E}\mathbf{r}_m^{s_i w} \right\|^2, \tag{8.26}$$

$\mathbf{E} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$, and $\mathbf{E}\mathbf{r}_m^{p_\ell z} = \mathbf{E}\mathbf{r}_b^{p_\ell z}$ at time-step $k = 0$ since $\mathcal{F}_m$ is fixed to the initial body frame of the teach pass. Additionaly, note that $\mathbf{r}_m^{s_i w}$ is the estimated position of the first anchor in the teach pass; in other words, $\mathbf{r}_m^{s_i w}$ comes from the element in the anchor map $\mathcal{M}$ with $\ell = 0$.

The height measurements are not useful when the robot is static on the floor, but the accelerometer and gyroscope measurements can be used as bias measurements as any non-zero measurement is a direct consequence of only gravity, noise, and biases. In fact, in the static case, the accelerometer and gyroscope measurements are modelled as

$$\mathbf{y}_b^{\mathrm{acc,i}} = -\boldsymbol{\beta}_b^{\mathrm{acc,i}} - \mathbf{g}_b - \mathbf{w}_b^{\mathrm{acc,i}}, \quad \mathbf{y}_b^{\mathrm{gyr,i}} = -\boldsymbol{\beta}_b^{\mathrm{gyr,i}} - \mathbf{w}_b^{\mathrm{gyr,i}} \tag{8.27}$$

where again assuming that the robot lies on a flat ground and that $\mathbf{g}_a = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}$, thus $\mathbf{g}_b = \mathbf{g}_a$.

In order to obtain a reliable estimate of the initial pose of the robot, a batch estimation problem similar to the one presented in Section 8.4.1 is formulated. As before, a concatenated and discretized process model and measurement model of the range, accelerometer, and gyroscope measurements of the form

$$\tilde{\mathbf{x}}_{k+1}^{\mathrm{i}} = \tilde{\mathbf{f}}_k \left(\tilde{\mathbf{x}}_k^{\mathrm{i}}, \tilde{\mathbf{w}}_k^{\mathrm{i}} \right), \qquad \tilde{\mathbf{y}}_k^{\mathrm{i}} = \tilde{\mathbf{g}}_k \left(\tilde{\mathbf{x}}_k^{\mathrm{i}}, \mathbf{r}_m^{s_i w}, \tilde{\boldsymbol{\eta}}_k^{\mathrm{i}} \right) \tag{8.28}$$

are assumed, where $k \in \{0, \ldots, L\}$ and L is the size of the initialization window. The initialization problem is then formulated as

$$\hat{\tilde{\mathbf{x}}}_{0:L}^{\mathrm{i}} = \arg\min_{\tilde{\mathbf{x}}_{0:L}^{\mathrm{i}}} \left\| \tilde{\mathbf{x}}_0^{\mathrm{i}} - \check{\tilde{\mathbf{x}}}_0^{\mathrm{i}} \right\|_{\check{\mathbf{P}}_0^{\mathrm{i}}}^2 + \sum_{j=0}^{L-1} \left\| \tilde{\mathbf{x}}_{j+1}^{\mathrm{i}} - \tilde{\mathbf{f}}_j \left(\tilde{\mathbf{x}}_j^{\mathrm{i}}, \mathbf{0} \right) \right\|_{\tilde{\mathbf{Q}}_j^{\mathrm{i}}}^2 + \sum_{j=0}^{L} \left\| \tilde{\mathbf{y}}_j^{\mathrm{i}} - \tilde{\mathbf{g}}_j \left(\tilde{\mathbf{x}}_j^{\mathrm{i}}, \hat{\mathbf{r}}_m^{s_i w}, \mathbf{0} \right) \right\|_{\tilde{\mathbf{R}}_j^{\mathrm{i}}}^2.$$

This relies on an assumption that exactly one anchor is within communication range at the beginning of the trajectory, and the position of the robot is initialized relative to that

anchor's estimated position $\hat{\mathbf{r}}_m^{s_i w}$. This, however, can be easily extended to include multiple anchors.

As before, Gauss-Newton or Levenberg-Marquardt can be used to solve this problem. The estimate $\hat{\check{\mathbf{x}}}_L$ is then used to initialize the repeat pass's EKF. In particular,

$$\check{\mathbf{x}}_0^{\mathrm{r}} = \left[(\check{\boldsymbol{\pi}}_0^{\mathrm{r}})^{\mathsf{T}} \ (\hat{\boldsymbol{\beta}}_L^{\mathrm{acc,i}})^{\mathsf{T}} \ (\hat{\boldsymbol{\beta}}_L^{\mathrm{gyr,i}})^{\mathsf{T}} \ (\hat{\mathbf{x}}_L^{\mathrm{UWB,i}})^{\mathsf{T}}\right],$$

where $\check{\boldsymbol{\pi}}_0^{\mathrm{r}} = \begin{bmatrix} \left(\hat{\check{\mathbf{r}}}_m^{zw,\mathrm{i}}\right)^{\mathsf{T}} & 0_{\mathrm{h}} & \mathbf{0}_{\mathrm{v}}^{\mathsf{T}} & \mathbf{0}_{\mathrm{rp}}^{\mathsf{T}} & \hat{\check{\theta}}_{mb}^{\mathrm{i}} \end{bmatrix}^{\mathsf{T}}$. The components $0_{\mathrm{h}} \in \mathbb{R}$, $\mathbf{0}_{\mathrm{v}} \in \mathbb{R}^3$, and $\mathbf{0}_{\mathrm{rp}} \in \mathbb{R}^2$ correspond to an estimate of zero height, velocity, and roll and pitch, respectively.

8.5.2 State Estimation

The state estimator in the repeat step is an EKF with the same model assumptions as the EKF used in the teach step. The repeat step EKF provides an estimate $\mathbf{x}_k^{\mathrm{r}} \sim \mathcal{N}\left(\hat{\mathbf{x}}_k^{\mathrm{r}}, \hat{\mathbf{P}}_k^{\mathrm{r}}\right)$ at every time-step $k \in \{0, \ldots, K\}$. The main differences between the teach and repeat estimators are that the state estimate and its uncertainty are initialized using the algorithm discussed in Section 8.5.1 for the repeat estimator, and that the anchor position estimates from the teach step are assumed to be the true positions of the anchors in the repeat step.

A key component of the repeat pass is the anchor sequence tracker that allows the robot to match each anchor with the corresponding feature from the anchor map. This requires the robot to keep track of active anchors in a manner similar to the teach pass estimator, as presented in Section 8.4. The tracker removes anchors that are no longer within communication range from the set of active anchors, and increments through the anchor map as new previously-inactive anchors are detected.

8.5.3 LQR Controller

The trajectory-tracking controller implemented is based on the finite-horizon LQR controller presented in [129]. The trajectory to be tracked is the teach pass estimated pose trajectory $\hat{\boldsymbol{\pi}}^{\mathrm{t}}$, represented at any time-step k as an element of a matrix Lie group $\hat{\mathbf{X}}_k^{\mathrm{t}} \in SE_2(3)$. Similarly, the repeat pass estimated pose $\hat{\boldsymbol{\pi}}_k^{\mathrm{r}}$ at time-step k is represented as $\hat{\mathbf{X}}_k^{\mathrm{r}} \in SE_2(3)$. The left-invariant tracking error is then $\delta\mathbf{X}_k = \left(\hat{\mathbf{X}}_k^{\mathrm{t}}\right)^{-1}\hat{\mathbf{X}}_k^{\mathrm{r}}$.

The control inputs at any time-step k are the thrust f_b and the angular velocity $\boldsymbol{\omega}_b^{bw}$.

The dynamics are modelled as

$$ {}^{a}\dot{\mathbf{v}}_m^{zw/a} = \mathbf{C}_{mb} \begin{bmatrix} 0 & 0 & f_b \end{bmatrix}^{\mathsf{T}} + \mathbf{g}_m + \mathbf{w}_m^f, \tag{8.29} $$

$$ \dot{\mathbf{C}}_{mb} = \mathbf{C}_{mb} \left(\boldsymbol{w}_b^{bw} + \mathbf{w}_b^{\omega} \right)^{\times}, \tag{8.30} $$

and the command inputs are computed similarly to [129].

Even though the estimated teach trajectory drifts from the true trajectory, the anchors are localized based on the drifted trajectory. Since the estimated anchor positions are utilized in the repeat pass estimator, the estimated repeat pass trajectory drifts in the same direction. By controlling the robot to retrace the estimated teach pass trajectory through knowledge of only the repeat pass estimator's state, the robot is therefore tracing the true teach trajectory. However, when there are no anchors within communication range, the repeat pass estimator's state might drift in a different direction than the teach pass estimator. During such intervals, the repeat pass estimator must not be trusted by the controller. Therefore, whenever no anchors are within range, the controller is temporarily turned off and the reference inputs from the teach pass are implemented in a feedforward fashion.

8.6 Results

The proposed UWB T&R algorithm is evaluated in simulation in the environment presented in Figure 8.1 and with a 3-tag quadcopter. All sensor measurements are corrupted with Gaussian white noise with characteristics corresponding to standard low-cost sensors. The results of the teach and repeat passes are shown in Figures 8.1 and 8.4. The teach pass estimator drifts with time, consequently also resulting in poor anchor localization. Nonetheless, the map is "locally consistent" in the T&R sense, meaning that locally the estimated trajectory relative to the estimated anchor position is representative of the true trajectory relative to the true anchor positions. Therefore, by tracking the estimated teach trajectory and using the estimated anchors in the repeat estimator, the LQR controller is able to to track the true teach trajectory as shown in Figure 8.1. This is despite the robot having a poor and globally inconsistent estimate of the true trajectory.

The trajectory-tracking controller computes its tracking error at any time-step by comparing the teach estimator's pose and the repeat estimator's pose at that time-step. Therefore, the controller attempts to get the estimated repeat trajectory as close as possible to the estimated teach trajectory. Nonetheless, the controller does not know what its true tracking error is, which is the error between the true repeat trajectory and the true teach trajectory. This error shown in Figure 8.5 for the position and heading is the metric in which the pro-

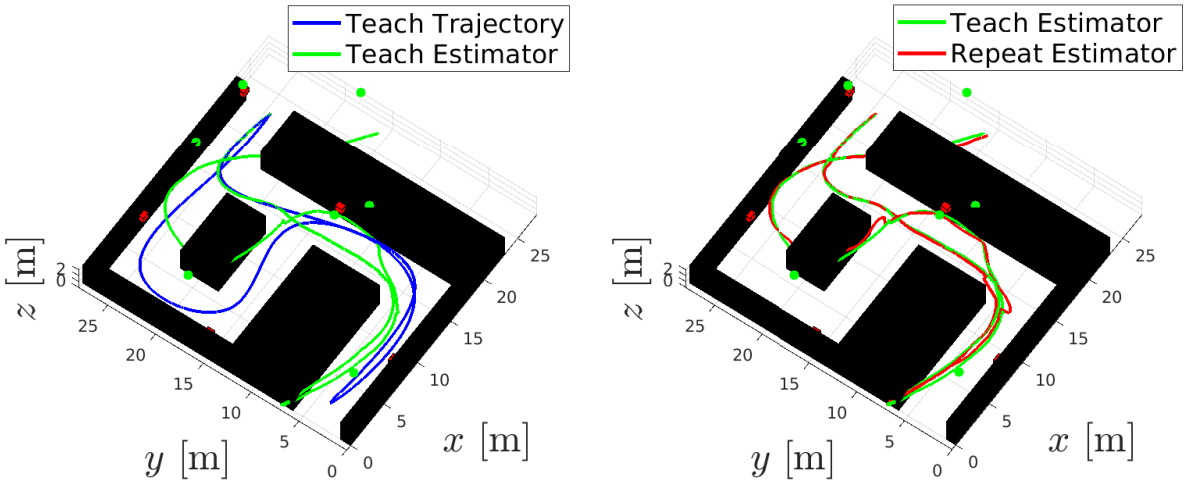


Figure 8.4: (Left) The true and estimated trajectories during the teach pass, and the true and estimated anchor positions as represented by the red and green blocks, respectively. There are more green blocks than red blocks as an anchor is initialized every time it is encountered. (Right) The estimated teach and repeat trajectories.

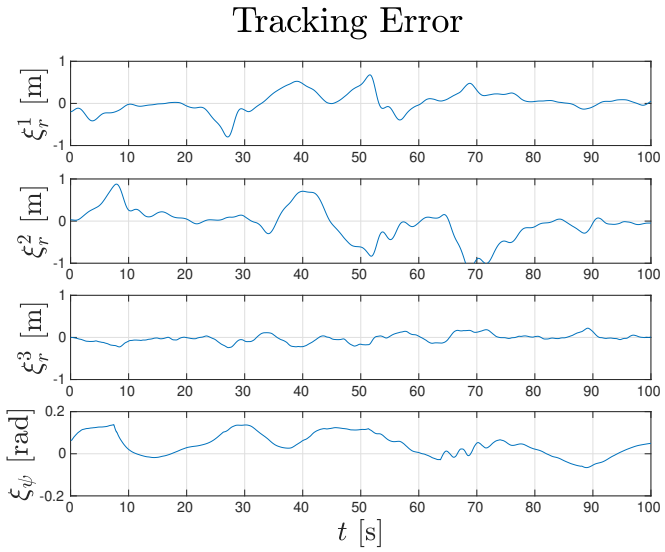


Figure 8.5: The position and heading error of the true repeat pass trajectory relative to the true teach pass trajectory.

posed algorithm is evaluated. As shown for this particular example, the robot manages to autonomously retrace the teach trajectory to within a metre and approximately 10 degrees in heading.

To statistically evaluate the performance of the proposed framework in simulation, 1000 Monte Carlo trials are performed. All trials share the same environment and the same teach trajectory, but the noise realizations, biases, clock states, and initial repeat pass's pose are distinct. The repeat position tracking root-mean-squared-error (RMSE) given by

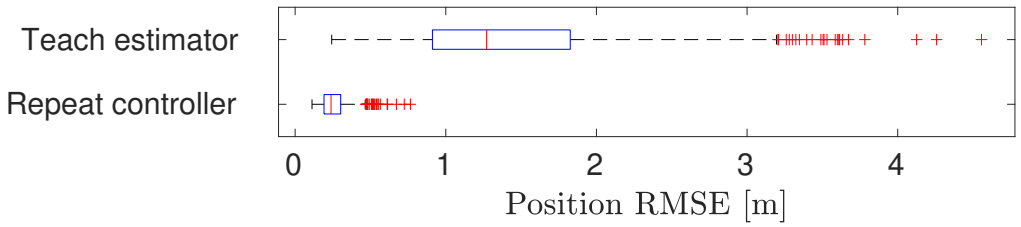


Figure 8.6: A box plot for the RMSE on 1000 Monte Carlo simulation trials. Although the teach estimator performs poorly, the robot manages to retrace the teach trajectory to within 1-metre accuracy for all runs.

$\mathbf{e}^{\mathrm{RMSE},\mathfrak{r}} = \sqrt{\frac{1}{N}\sum_{k=1}^{K}\left\|\mathbf{r}_m^{zw,\mathfrak{r}} - \mathbf{r}_m^{zw,\mathfrak{t}}\right\|^2}$, and the position estimation RMSE given by $\mathbf{e}^{\mathrm{RMSE},\mathfrak{t}} = \sqrt{\frac{1}{N}\sum_{k=1}^{K}\left\|\hat{\mathbf{r}}_m^{zw,\mathfrak{t}} - \mathbf{r}_m^{zw,\mathfrak{t}}\right\|^2}$ are computed for each trial. The results are summarized in Figure 8.6. Despite the estimated trajectory available to the robot being poor with an average RMSE of 1.44 m, the tracking RMSE is always below 1 m, with an average of 0.26 m.

8.7 Conclusion

UWB-based T&R, as opposed to vision-based T&R, is presented as a novel application of the algorithms presented in this thesis. A custom ranging protocol allows a 3-tag robot to range with the UWB anchors without compromising ranging frequency, and the measurements are fused with IMU and height measurements to compute a locally-consistent map and an estimated teach trajectory. To retrace the trajectory autonomously, the robot first estimates its initial pose in the repeat pass by remaining static, then fuses all on-board sensors with the map information from the teach pass as it navigates the same trajectory. The proposed algorithm is shown in simulation to be capable of retracing the trajectory with sub-metre tracking error.

Future work involves validating the algorithm in real-world experiments on ground and/or aerial vehicles, and comparing the tracking performance and computational cost to a SLAM-based solution. Additionally, as part of the process of further validating this T&R algorithm, another goal is to open-source the corresponding code to allow implementation by the robotics community on many different systems. Lastly, this T&R application is one of many potential novel applications enabled by the ability to simultaneously design ranging protocols and state estimation algorithms, and future work aims to explore other novel single-robot and multi-robot applications.

Chapter 9

Concluding Remarks

This thesis studies practical localization approaches for robots utilizing UWB radio. The goal of this thesis is to simultaneously design ranging protocols and state estimation algorithms to implement practical, scalable, and accurate localization algorithms. To the Author's knowledge, and at the time of writing this thesis, no other methods exist that combine custom ranging protocols, on-manifold state estimation, and IMU preintegration for efficient communication and odometry sharing.

The journey of the thesis can be summarized as follows. Firstly, in Chapter 3, the observability of the relative positions between robots in 3D from just range measurements is studied, and a sufficient condition for observability is then derived. This motivated the use of multi-tag robots throughout the remainder of the thesis.

The second thrust of this thesis in Chapters 4 and 5 looks into improving the quality of the UWB measurements themselves. The former chapter looks at the accuracy of the measurements, and consequently a new ranging protocol is proposed and a data-driven calibration procedure for removing any systematic error is presented and evaluated in experiments. Meanwhile, Chapter 5 addresses the precision of the measurements, where an optimization problem is proposed to maximize the amount of information attained in one unit of time.

The next chapter, Chapter 6, addresses many of the limitations of existing UWB-based localization solutions by utilizing the concept of passive listening. By allowing robots to listen-in on neighbours, more robots can be added to the team without reducing the rate at which each robot gets measurements. Additionally, the proposed communication scheme allows a simple MAC protocol and allows robots to broadcast IMU information, which benefits from preintegration to allow for efficient data transfer and storage. All these benefits are then utilized in a simultaneous clock-synchronization and relative-pose estimator, where the relative pose states are represented and manipulated directly on the $SE_2(3)$ manifold.

In order to allow the robots to share their state estimates with one another, the problem of loosely-coupled or cascaded filtering is then addressed in Chapter 7. A probabilistic approximation for the cross-covariances that develop is then proposed and evaluated on the common robotics problem of an attitude-and-heading reference system (AHRS) feeding into a position estimator.

Lastly, Chapter 8 showcases the novel applications made possible through simultaneous ranging-protocol and state-estimator design. A novel UWB-based teach-and-repeat (T&R) framework is presented, where a robot manages to autonomously retrace a taught trajectory in an environment with spaced-out static anchors at unknown locations.

Nonetheless, the presented work in this thesis is not perfect, and there are a few potential areas for improvement as will be discussed next. Nonetheless, the Author hopes that this thesis will serve as a basis for future work in the field of UWB-based localization, and that the work presented here will be a stepping stone towards the goal of enabling robots to operate in a fully autonomous-manner in unknown environments.

9.1 Future Work

The work presented in this thesis mainly addresses the use of UWB for inter-robot ranging and communication. Nonetheless, robots need to be able to interact with their environment, which requires the use of other sensors such as cameras or LIDARs. The goal of this thesis is to build the foundation for a practical and scalable UWB-based localization system, and the next step is to integrate other sensors to allow robots to interact with their environment. Additionally, this external information will be required to estimate IMU biases, which in the case of inter-robot ranging is not possible as the biases are unobservable, and only relative IMU biases can be estimated as shown in Appendix E.

Another potential area for improvement is the quality of the odometry. Dead-reckoning IMU alone is prone to significant drift in a short period of time, which necessitates the use of some form of correction. The problem with UWB alone is that the range measurements do not provide a lot of information to correct the drift, meaning that IMU + UWB only is not sufficient for highly-accurate localization that would be sufficient for closed-loop control. Future work should look at potentially improving the quality of the odometry by imposing constraints on the motion model of the robots to characterize physical limitations, potentially through data-driven approaches.

The discontinuous ambiguities discussed in Section 3.4.2 also present another challenge for UWB-based localization, where the filter sometimes converges to the local minimum at the ambiguity. To address this, multi-modal estimation tools, such as the Gaussian-sum

filter, can be utilized to allow the filter to converge to the global minimum.

The majority of the algorithms presented in this thesis are tested in simple lab settings, whereas real-life scenarios involve more complex environments that introduce more sources of error in the UWB range measurements due to non-line-of-sight (NLOS). Future work will look at the effect of NLOS on the performance of the proposed algorithms, and potentially incorporating NLOS into the calibration procedure presented in Chapter 4.

Lastly, to allow the community to benefit from the hardware used in this thesis, a dataset of the collected measurements and the ground-truth trajectories will be made publicly available. The hardware will also be used to evaluate the T&R framework, which will be made open-source for the community to use.

Appendix A

Hardware

A.1 Ultra-wideband

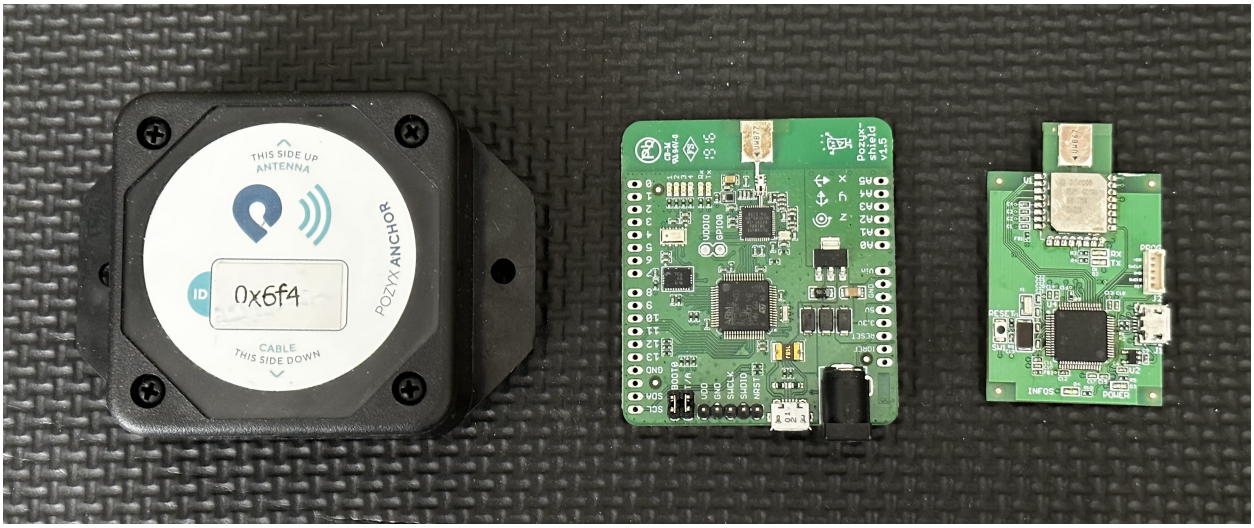


Figure A.1: The different UWB transceivers used. (Left) Pozyx anchor. (Middle) Pozyx tag. (Right) Custom-built tag.

The experiments in this thesis involved the use of two different versions of UWB transceivers; an off-the-shelf version supplied by Pozyx [130], and a custom-built version. Both of these transceivers are based on the DWM1000 module [7] supplied by Qorvo (formerly Decawave), and are shown in Figure A.1. The DWM1000 is a UWB transceiver that operates in the 3.5 GHz to 6.5 GHz frequency range, and is capable of ranging with a precision of 10 cm. The DWM1000 is also capable of data transfer at a rate of 6.8 Mbps.

A.1.1 Pozyx

At the beginning of this project, a ready-to-use off-the-shelf UWB transceiver was required to test the relative position estimation algorithms. The choice of the Pozyx system was motivated by the fact that it satisfied that requirement, and the system was used in the experiments in Chapters 3 and 7. The Pozyx system consists of two different types of UWB transceivers; anchors and tags. The anchors are fixed to the environment, and are used as reference points for the tags. The tags are attached to the robots, and are used to estimate their relative positions. In the presence of 4 or more anchors, the Pozyx system can be used to directly output position measurements, as is used in Chapter 7. The Pozyx system is shown in Figure A.1.

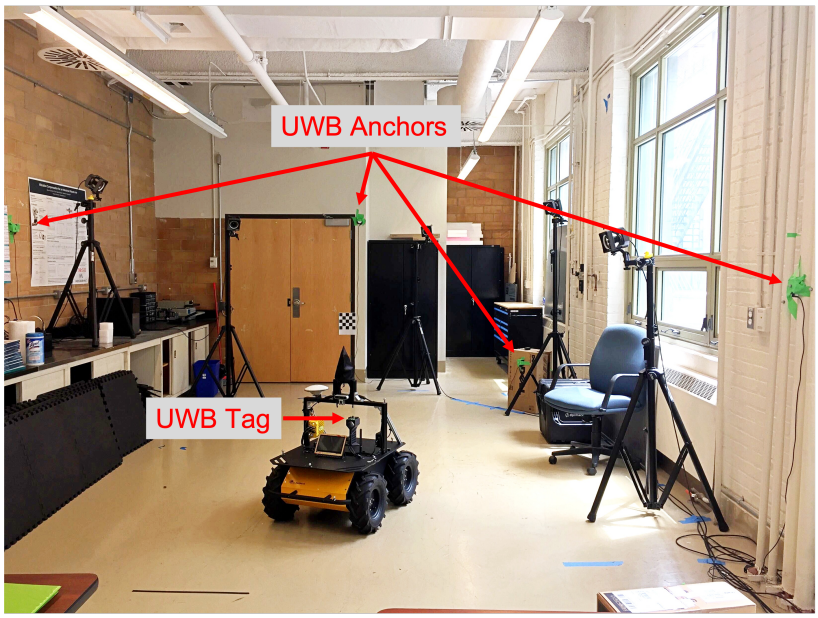
Nonetheless, the performance obtained from the relative position estimators is not great, and this is largely due to inaccuracies in the UWB measurements. To characterize the performance of the Pozyx system, 5 Pozyx anchors are placed around a room as shown in Figure A.2, and a robot with 1 Pozyx tag moves around randomly while collecting range measurements. Clearly, the mean and mode of the measurements are positively biased and the measurements have a lot of variance, which is typical of uncalibrated UWB systems when used indoors.

This motivated the idea of digging into the UWB calibration procedure and tying the ranging protocols with the localization algorithms through, for example, clock state estimation and passive listening as discussed in Chapter 6. Given that off-the-shelf UWB systems are typically not fully customizable, a custom-built UWB system is used.

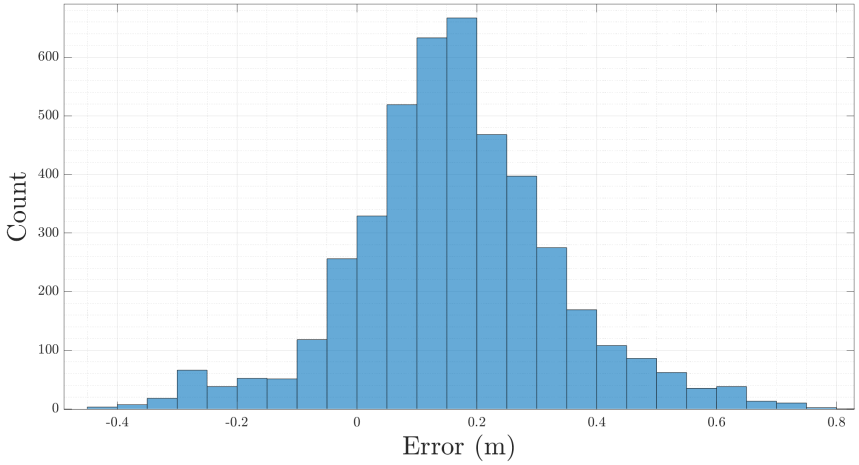
A.1.2 Custom-built PCBs

Collaborators at the *Mobile Robotics and Autonomous Systems Laboratory* (MRASL) at Polytechnique Montreal have built a DWM1000-based UWB transceiver, shown in Figure A.3. This transceiver is compact and lightweight, measuring at 32 mm $\times$ 49 mm, and is much smaller than the Pozyx transceivers as shown in Figure A.1. These transceivers also weigh 8 grams each.

The main advantage of using these over the Pozyx system is complete customizability and accessibility of the DWM1000 microprocessor's registers. The firmware for these transceivers has been written from scratch in C, which allowed for the implementation of the calibration procedures and ranging protocols discussed in Chapters 4, 5, and 6. This in fact yielded much better range measurements as shown in Chapter 4. The firmware is also written in a way that allows for easy integration with the *robot operating system* (ROS), which is used for integrating the UWB transceivers with the robots. Note that all algorithms presented



(a) The setup for the experiment. There is 1 UWB tag on the ground robot, which moves around randomly in an environment with 5 UWB anchors (1 is not visible in the figure).



(b) The distribution of the bias in the measurements.

Figure A.2: Testing the accuracy of the Pozyx system.

in thesis are not dependent on the specific hardware used, and can be implemented on any open-access UWB transceiver.

A.1.3 Media-Access Control

Media-access control (MAC) is the term used to address algorithms used to control when each pair of transceivers range with one another while preventing other transceivers from ranging during this window to prevent message collision. This is typically done using time-division

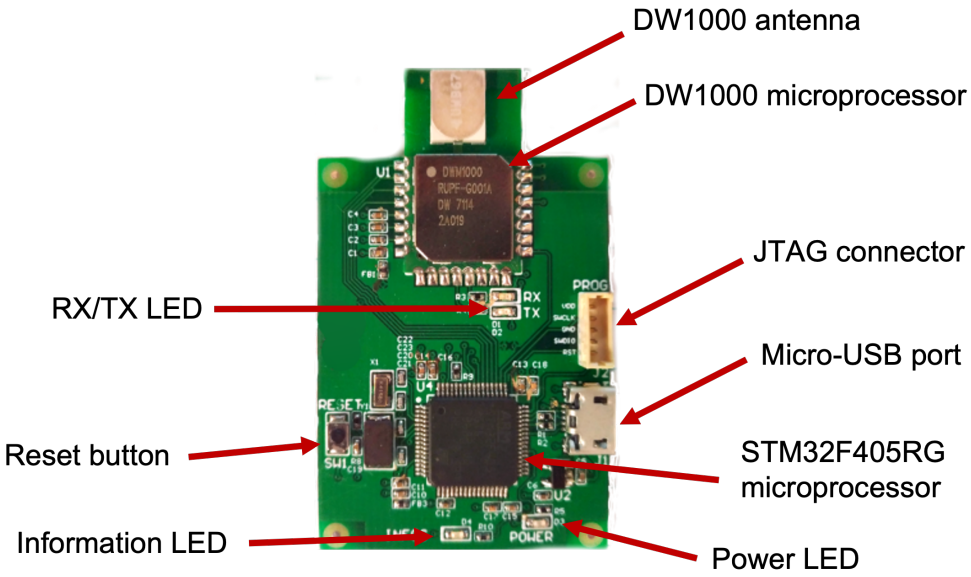


Figure A.3: The custom-built UWB transceiver, with labels identifying the different components.

multiple-access (TDMA) or the token-passing protocol [31, Chapter 3.3]. The token-passing protocol is implemented on both the Pozyx and custom-built transceivers, and is used in the experiments in Chapters 3 and 7. This involves the target transceiver in one TWR transaction being the next initiating transceiver, and therefore prevents any two transceivers initiating a transaction at the same time. The maximum ranging frequency that was attained with the Pozyx system is 16 Hz, while a ranging frequency of 240 Hz was achieved with the custom-built modules.

Passive listening, as presented in Chapter 6, allows an alternative algorithm, where a user-defined sequence of ranging pairs can be made known to all robots. Each robot can then keep track of which pair in the sequence is currently ranging, and initiate a TWR transaction to a specified transceiver when it is its turn to do so. This MAC protocol is named the common-list protocol. The experiments in Chapter 6 use the common-list protocol.

A.1.4 Radiation Pattern

A source of errors in UWB ranging is the fact that the UWB antennae used in practice are not perfectly omni-directional. To see this, an experiment is conducted where one tag is placed on a tripod, and another tag is placed on a robot rotating in place as shown in Figure A.4. The tripod is placed at different angles and different distances with respect to the rotating robot, and the received signal power is recorded. The results are shown

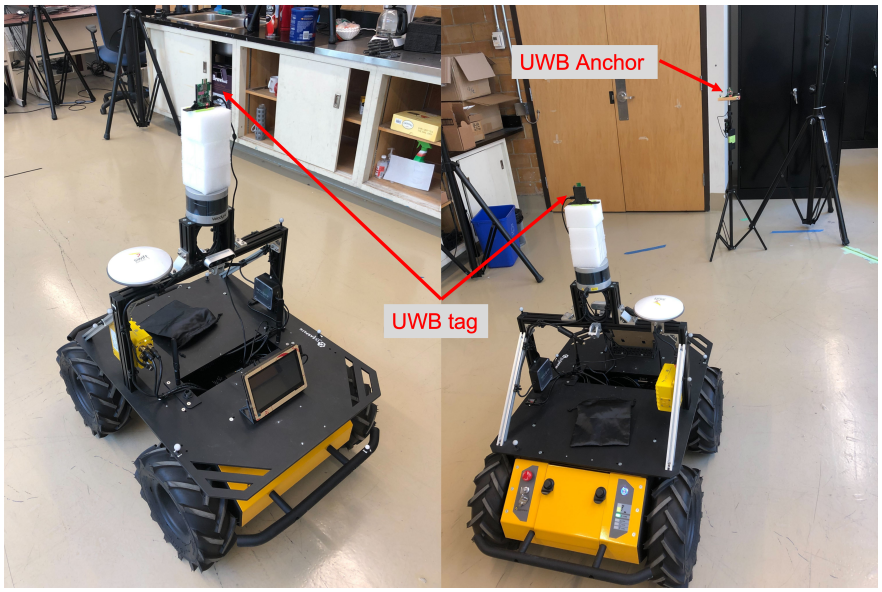


Figure A.4: The experimental set-up with one tag on a robot rotating in place and another tag on a static tripod.

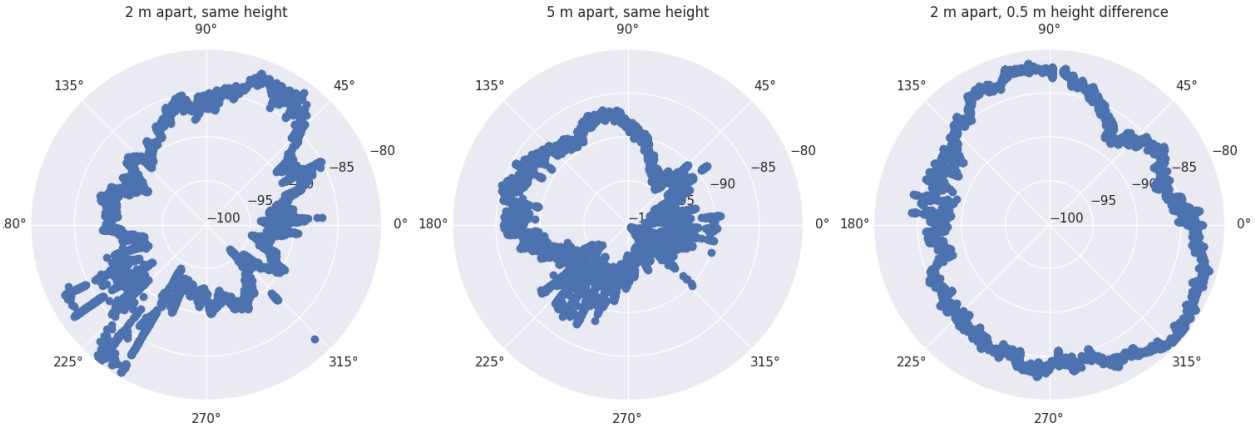


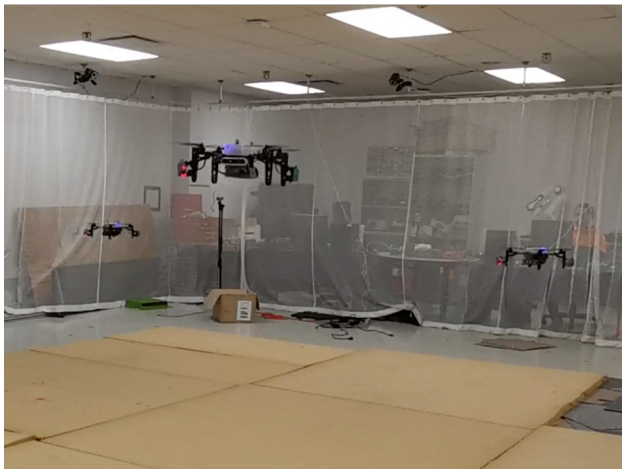
Figure A.5: The radiation pattern when the tripod is placed at different angles and different distances with respect to the rotating robot. The radial axes refers to the received signal power in dBm, and the angular axes refers to the heading of the robot with respect to the tripod.

in Figure A.5. As expected, the received signal power is a function of the relative pose between the transceivers, which motivates the use of the received signal power to correct the pose-dependent bias in the UWB measurements as discussed in Chapter 4.

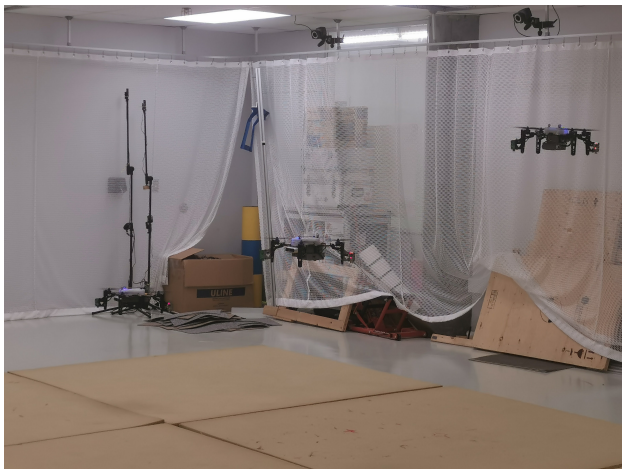
A.2 Uvify IFO-S Quadcopter

The experiments in Chapters 3 and 7 use hand-held sensor rigs equipped with UWB transceivers to simulate moving robots. Nonetheless, this is problematic as the human body is an obstacle

(a) The Uvify IFO-S quadcopter with 2 custom-made UWB transceivers.



(b) Three Uvify IFO-S in a triangle formation.



(c) Three Uvify IFO-S in a line formation.

Figure A.6: The quadcopters used in this thesis.

that interferes with the messages being transmitted between the transceivers. Additionally, these hand-held rigs do not represent real-world scenarios, where IMU measurements for example suffer from additional sources of error due to vibrations on the quadcopter.

This motivated the use of the ready-to-use off-the-shelf Uvify IFO-S quadcopter [131], which comes equipped with a Jetson Nano computer running Ubuntu 18.04 and a Pixhawk 4 autopilot. Two custom-made UWB transceivers are fitted to the quadcopter as shown in Figure A.6a approximately 42 cm apart, and hours of data are collected in a 5 m $\times$ 5 m $\times$ 2 m space with 3 quadcopters flying around as shown in Figure A.6. The data collected is then used to test the calibration and relative pose estimation algorithms in Chapters 4 and 6.

Appendix B

Fold Increase in Measurements

When there are $n+1$ robots and 2 transceivers per robot, the total number of transceivers is $n_t = 2(n+1)$. Therefore, the number of ranging pairs with transceivers on distinct robots is

$$n_{\mathrm{p}} = \frac{2(n+1)(2(n+1)-1)}{2} - (n+1) = 2n(n+1).$$

The number of direct measurements between all robots is then $2n_{\mathrm{p}}$ (one range and one offset measurement per pair), while the number of passive listening measurements recorded at all robots is $n_{\mathrm{p}}(3(n_{\mathrm{t}}-2)) = 6nn_{\mathrm{p}}$. Therefore, the fold increase in measurements is

$$\frac{2n_{\mathrm{p}} + 6nn_{\mathrm{p}}}{2n_{\mathrm{p}}} = 1 + 3n$$

when considering a centralized approach where passive listening measurements from all robots are available.

A similar analysis can be done from the perspective of one robot that does not have access to passive listening measurements recorded at neighbouring robots. Without passive listening it can be shown that the robot only gets $8n$ distinct measurements, while with listening-in on neighbouring robots' messages the robot gets $2n_{\mathrm{p}} - 8n$ new measurements from the direct measurements between the neighbours and $12n^2$ new passive listening measurements. This can be shown to be a $(\frac{1}{2}+2n)$-fold increase in the number of measurements from the individual robot's perspective.

Appendix C

Linearizing the Range Measurement Model

Consider as in (6.10) an expression of the form

$$d = \left\| \Big(\boldsymbol{\Pi} \big(\mathbf{T}_2 \tilde{\mathbf{r}}_2 - \mathbf{T}_1 \tilde{\mathbf{r}}_1 \big) \Big) \right\|, \tag{C.1}$$

where $\mathbf{T}_1, \mathbf{T}_2 \in SE(3)$ and $\mathbf{r}_1, \mathbf{r}_2 \in \mathbb{R}^5$. Squaring both sides and perturbing the measurement and the pose states yields

$$(\bar{d} + \delta d)^2 = \Big(\boldsymbol{\Pi} \big(\operatorname{Exp}(\delta \boldsymbol{\xi}_2) \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2 - \operatorname{Exp}(\delta \boldsymbol{\xi}_1) \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1 \big) \Big)^{\mathsf{T}} \Big(\cdot \Big),$$

which, using (2.21), can be expanded to give

$$\begin{aligned} \bar{d}^2 + 2\bar{d}\delta d \approx\ & (\boldsymbol{\Pi} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2)^{\mathsf{T}} \boldsymbol{\Pi} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2 + (\boldsymbol{\Pi} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1)^{\mathsf{T}} \boldsymbol{\Pi} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1 \\ & - (\boldsymbol{\Pi} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2)^{\mathsf{T}} \boldsymbol{\Pi} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1 - (\boldsymbol{\Pi} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1)^{\mathsf{T}} \boldsymbol{\Pi} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2 \\ & - (\boldsymbol{\Pi} \delta \boldsymbol{\xi}_2^{\wedge} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2)^{\mathsf{T}} \boldsymbol{\Pi} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1 - (\boldsymbol{\Pi} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2)^{\mathsf{T}} \boldsymbol{\Pi} \delta \boldsymbol{\xi}_1^{\wedge} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1 \\ & - (\boldsymbol{\Pi} \delta \boldsymbol{\xi}_1^{\wedge} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1)^{\mathsf{T}} \boldsymbol{\Pi} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2 - (\boldsymbol{\Pi} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1)^{\mathsf{T}} \boldsymbol{\Pi} \delta \boldsymbol{\xi}_2^{\wedge} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2, \end{aligned}$$

where higher order terms have been neglected. Cancelling out the nominal terms on both sides, using the fact that each term is scalar, and recalling (2.20),

$$\begin{aligned} 2\bar{d}\delta d &= -2(\boldsymbol{\Pi} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2)^{\mathsf{T}} \boldsymbol{\Pi} \delta \boldsymbol{\xi}_1^{\wedge} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1 - 2(\boldsymbol{\Pi} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1)^{\mathsf{T}} \boldsymbol{\Pi} \delta \boldsymbol{\xi}_2^{\wedge} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2 \\ &= -2(\boldsymbol{\Pi}^{\mathsf{T}} \boldsymbol{\Pi} \bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2)^{\mathsf{T}} (\bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1)^{\odot} \delta \boldsymbol{\xi}_1 - 2(\boldsymbol{\Pi}^{\mathsf{T}} \boldsymbol{\Pi} \bar{\mathbf{T}}_1 \tilde{\mathbf{r}}_1)^{\mathsf{T}} (\bar{\mathbf{T}}_2 \tilde{\mathbf{r}}_2)^{\odot} \delta \boldsymbol{\xi}_2. \end{aligned}$$

Therefore, the linearized model for (C.1) is

$$\delta d = -\frac{1}{\bar{d}}(\mathbf{\Pi}^{\mathsf{T}}\mathbf{\Pi}\bar{\mathbf{T}}_2\tilde{\mathbf{r}}_2)^{\mathsf{T}}(\bar{\mathbf{T}}_1\tilde{\mathbf{r}}_1)^{\odot}\delta\boldsymbol{\xi}_1 - \frac{1}{\bar{d}}(\mathbf{\Pi}^{\mathsf{T}}\mathbf{\Pi}\bar{\mathbf{T}}_1\tilde{\mathbf{r}}_1)^{\mathsf{T}}(\bar{\mathbf{T}}_2\tilde{\mathbf{r}}_2)^{\odot}\delta\boldsymbol{\xi}_2.$$

Appendix D

Discretizing the Input Matrix

The matrices $\tilde{\mathbf{U}}_{0,k}$ and $\tilde{\mathbf{U}}_{i,k}$ in (6.26) are of the general form

$$\tilde{\mathbf{U}} = \begin{bmatrix} \mathbf{u}^\wedge & \mathbf{e}_4 \\ \mathbf{0}_{1\times 4} & 0 \end{bmatrix}, \tag{D.1}$$

where $\mathbf{u} = \begin{bmatrix} \boldsymbol{\omega}^\mathsf{T} & \boldsymbol{\alpha}^\mathsf{T} \end{bmatrix}^\mathsf{T}$, $(\cdot)^\wedge$ is the wedge operator in $SE(3)$, and $\mathbf{e}_4 = \begin{bmatrix} \mathbf{0}_{1\times 3} & 1 \end{bmatrix}^\mathsf{T}$. Consequently,

$$\begin{aligned} \mathbf{U} = \exp(\tilde{\mathbf{U}}\Delta t) &= \sum_{\ell=0}^{\infty} \frac{1}{\ell!} \left(\tilde{\mathbf{U}}\Delta t\right)^\ell \\ &= \mathbf{1} + \begin{bmatrix} \mathbf{u}^\wedge & \mathbf{e}_4 \\ \mathbf{0} & 0 \end{bmatrix} \Delta t + \frac{1}{2!} \begin{bmatrix} (\mathbf{u}^\wedge)^2 & \mathbf{u}^\wedge\mathbf{e}_4 \\ \mathbf{0} & 0 \end{bmatrix} (\Delta t)^2 + \frac{1}{3!} \begin{bmatrix} (\mathbf{u}^\wedge)^3 & (\mathbf{u}^\wedge)^2\mathbf{e}_4 \\ \mathbf{0} & 0 \end{bmatrix} (\Delta t)^3 + \ldots \\ &= \begin{bmatrix} \sum_{\ell=0}^{\infty} \frac{1}{\ell!}(\mathbf{u}^\wedge\Delta t)^\ell & \sum_{\ell=0}^{\infty} \frac{1}{(\ell+1)!}(\mathbf{u}^\wedge\Delta t)^\ell\mathbf{e}_4\Delta t \\ \mathbf{0} & 1 \end{bmatrix}. \end{aligned} \tag{D.2}$$

Note that $\sum_{\ell=0}^{\infty} \frac{1}{\ell!}(\mathbf{u}^\wedge\Delta t)^\ell = \mathrm{Exp}(\mathbf{u}\Delta t)$, where Exp is the $SE(3)$ exponential operator, giving

$$\sum_{\ell=0}^{\infty} \frac{1}{\ell!}(\mathbf{u}^\wedge\Delta t)^\ell = \begin{bmatrix} \mathrm{Exp}(\boldsymbol{\omega}\Delta t) & \Delta t\mathbf{J}_l(\boldsymbol{\omega}\Delta t)\boldsymbol{\alpha} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{D.3}$$

and

$$\begin{aligned} \mathbf{J}_l(\boldsymbol{\psi}) &= \sum_{\ell=0}^{\infty} \frac{1}{(\ell+1)!} \left(\phi\boldsymbol{\phi}^\times\right)^\ell \\ &= \frac{\sin\phi}{\phi}\mathbf{1} + \left(1 - \frac{\sin\phi}{\phi}\right)\boldsymbol{\phi}\boldsymbol{\phi}^\mathsf{T} + \frac{1-\cos\phi}{\phi}\boldsymbol{\phi}^\times \end{aligned}$$

is the left Jacobian of $SO(3)$, where $\phi = |\boldsymbol{\psi}|$ and $\boldsymbol{\phi} = \boldsymbol{\psi}/\phi$. Meanwhile,

$$\begin{aligned}
&\sum_{\ell=0}^{\infty} \frac{1}{(\ell+1)!}(\mathbf{u}^{\wedge}\Delta t)^{\ell} \\
&= \mathbf{1} + \frac{1}{2!}\begin{bmatrix} \boldsymbol{\omega}^{\times} & \boldsymbol{\alpha} \\ \mathbf{0} & 0 \end{bmatrix}\Delta t + \frac{1}{3!}\begin{bmatrix} (\boldsymbol{\omega}^{\times})^2 & \boldsymbol{\omega}^{\times}\boldsymbol{\alpha} \\ \mathbf{0} & 0 \end{bmatrix}(\Delta t)^2 + \frac{1}{4!}\begin{bmatrix} (\boldsymbol{\omega}^{\times})^3 & (\boldsymbol{\omega}^{\times})^2\boldsymbol{\alpha} \\ \mathbf{0} & 0 \end{bmatrix}(\Delta t)^3 + \dots \\
&= \begin{bmatrix} \sum_{\ell=0}^{\infty} \frac{1}{(\ell+1)!}(\boldsymbol{\omega}^{\times}\Delta t)^{\ell} & \sum_{\ell=0}^{\infty} \frac{1}{(\ell+2)!}(\boldsymbol{\omega}^{\times}\Delta t)^{\ell}\boldsymbol{\alpha}\Delta t \\ \mathbf{0} & 1 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{J}_l(\boldsymbol{\omega}\Delta t) & \frac{\Delta t}{2}\mathbf{N}_l(\boldsymbol{\omega}\Delta t)\boldsymbol{\alpha} \\ \mathbf{0} & 1 \end{bmatrix},
\end{aligned} \tag{D.4}$$

where

$$\begin{aligned}
\mathbf{N}(\boldsymbol{\psi}) &= 2\sum_{\ell=0}^{\infty} \frac{1}{(\ell+2)!}\left(\phi\boldsymbol{\phi}^{\times}\right)^{\ell} \\
&= \boldsymbol{\phi}\boldsymbol{\phi}^{\mathsf{T}} + 2\left(\frac{1}{\phi} - \frac{\sin\phi}{\phi^2}\right)\boldsymbol{\phi}^{\times} + 2\frac{\cos\phi - 1}{\phi^2}\boldsymbol{\phi}^{\times}\boldsymbol{\phi}^{\times}.
\end{aligned}$$

Substituting (D.3) and (D.4) back into (D.2) gives

$$\mathbf{U} = \begin{bmatrix} \mathrm{Exp}(\boldsymbol{\omega}\Delta t) & \Delta t\mathbf{J}_l(\boldsymbol{\omega}\Delta t)\boldsymbol{\alpha} & \frac{\Delta t^2}{2}\mathbf{N}_l(\boldsymbol{\omega}\Delta t)\boldsymbol{\alpha} \\ & 1 & \Delta t \\ & & 1 \end{bmatrix}. \tag{D.5}$$

Appendix E

Addressing IMU Biases

The need to estimate IMU biases is particularly important for applications that involve long-term navigation. As compared to the framework presented in this thesis, most robotic applications involve additional exteroceptive sensors such as GPS, a magnetometer, or a camera that provide measurements relative to static environmental quantities, which allow individual robots to estimate their own IMU biases using standard methods [8, Ch. 10], [132], [133]. In a multi-robot scenario, these bias estimates can be used by each robot to correct its own IMU measurement before adding the measurement to the RMI. This is a loosely-coupled solution that overcomes the need for each robot to share its IMU biases with its neighbours.

In this thesis, only range measurements are available that provide constraints among two moving bodies, hence estimating the biases is trickier as there is no static reference. Nonetheless, even though the proposed framework will mostly be used alongside additional exteroceptive sensors such as a camera to allow for real-world applications, it is indeed important to address the issue of IMU biases in the context of the framework presented in this thesis, and particularly Chapter 6, to allow long-term navigation without relying on additional sensors. To do so, this appendix presents how gyroscope biases and relative accelerometer biases can be estimated while maintaining the differential Sylvester equation form of the process model as shown in (6.25), under a few assumptions.

The remainder of this appendix is based on [134], and is organized as follows. Section E.1 presents the pose process model with IMU biases, and Section E.2 presents the bias process model. In Section E.3, preintegration of the IMU measurements. The simulation and experimental results are shown in Sections E.4 and E.5, respectively.

E.1 Pose Process Model with Bias

The IMU biases affect the process model presented in Section 6.5, but the ranging protocol and the form of the measurement model from Section 6.4 remain unchanged. The relative attitude process model is

$$\dot{\mathbf{C}}_{0i} = -\left(\boldsymbol{\omega}_0^{0a} - \boldsymbol{\beta}_0^{\text{gyr},0}\right)^{\times} \mathbf{C}_{0i} + \mathbf{C}_{0i}\left(\boldsymbol{\omega}_i^{ia} - \boldsymbol{\beta}_i^{\text{gyr,i}}\right)^{\times} \tag{E.1}$$

in the presence of bias, where $\boldsymbol{\beta}_i^{\text{gyr,i}}$ is the bias of the gyroscope of Robot i resolved in the Robot i's body frame. Similarly, the relative velocity process model is

$$\begin{aligned}
{}^0\dot{\mathbf{v}}_0^{i0/a} &= -\left(\boldsymbol{\omega}_0^{0a} - \boldsymbol{\beta}_0^{\text{gyr},0}\right)^{\times} \mathbf{v}_0^{i0/a} + \mathbf{C}_{0i}\left(\boldsymbol{\alpha}_i^{i} - \boldsymbol{\beta}_i^{\text{acc,i}}\right) - \left(\boldsymbol{\alpha}_0^{0} - \boldsymbol{\beta}_0^{\text{acc},0}\right) && \text{(E.2)}\\
&= -\left(\boldsymbol{\omega}_0^{0a} - \boldsymbol{\beta}_0^{\text{gyr},0}\right)^{\times} \mathbf{v}_0^{i0/a} + \mathbf{C}_{0i}\boldsymbol{\alpha}_i^{i} - \boldsymbol{\alpha}_0^{0} \underbrace{- \mathbf{C}_{0i}\boldsymbol{\beta}_i^{\text{acc,i}} + \boldsymbol{\beta}_0^{\text{acc},0}}_{\boldsymbol{\beta}_0^{\text{acc},0i}} && \text{(E.3)}\\
&= -\left(\boldsymbol{\omega}_0^{0a} - \boldsymbol{\beta}_0^{\text{gyr},0}\right)^{\times} \mathbf{v}_0^{i0/a} + \mathbf{C}_{0i}\boldsymbol{\alpha}_i^{i} - \boldsymbol{\alpha}_0^{0} + \boldsymbol{\beta}_0^{\text{acc},0i}, && \text{(E.4)}
\end{aligned}$$

where $\boldsymbol{\beta}_i^{\text{acc,i}}$ is the bias of the accelerometer of Robot i resolved in the Robot i's body frame, and $\boldsymbol{\beta}_0^{\text{acc,0i}} \triangleq \boldsymbol{\beta}_0^{\text{acc},0} - \mathbf{C}_{0i}\boldsymbol{\beta}_i^{\text{acc,i}}$ is the relative accelerometer bias of Robot 0 relative to Robot i, resolved in Robot 0's body frame. Lastly, the relative position process model is

$${}^0\dot{\mathbf{r}}_0^{i0} = -\left(\boldsymbol{\omega}_0^{0a} - \boldsymbol{\beta}_0^{\text{gyr},0}\right)^{\times} \mathbf{r}_0^{i0} + \mathbf{v}_0^{i0/a}. \tag{E.5}$$

Note that the choice of estimating the relative accelerometer bias $\boldsymbol{\beta}_0^{\text{acc,0i}}$ and the absolute gyroscope bias $\boldsymbol{\beta}_0^{\text{gyr},0}$ is made to ensure that the process model remains of the form of a differential Sylvester equation,

$$\begin{aligned}
\dot{\mathbf{T}}_{0i} &= \begin{bmatrix} \dot{\mathbf{C}}_{0i} & {}^0\dot{\mathbf{v}}_0^{i0/a} & {}^0\dot{\mathbf{r}}_0^{i0} \\ & 0 & \\ & & 0 \end{bmatrix}\\
&= -\left(\begin{bmatrix} \left(\boldsymbol{\omega}_0^{0a}\right)^{\times} & \boldsymbol{\alpha}_0^{0} & \\ & & 1 \\ & & 0 \end{bmatrix} - \begin{bmatrix} \left(\boldsymbol{\beta}_0^{\text{gyr},0}\right)^{\times} & \boldsymbol{\beta}_0^{\text{acc},0i} & \\ & & 0 \\ & & 0 \end{bmatrix}\right)\mathbf{T}_{0i}\\
&\qquad + \mathbf{T}_{0i}\left(\begin{bmatrix} \left(\boldsymbol{\omega}_i^{ia}\right)^{\times} & \boldsymbol{\alpha}_i^{i} & \\ & & 1 \\ & & 0 \end{bmatrix} - \begin{bmatrix} \left(\boldsymbol{\beta}_i^{\text{gyr,i}}\right)^{\times} & \mathbf{0} & \\ & & 0 \\ & & 0 \end{bmatrix}\right)\\
&\triangleq -(\tilde{\mathbf{U}}_0 - \tilde{\mathbf{B}}_0)\mathbf{T}_{0i} + \mathbf{T}_{0i}(\tilde{\mathbf{U}}_i - \tilde{\mathbf{B}}_i). && \text{(E.6)}
\end{aligned}$$

This is of a similar form as the process model (6.25), and has a closed-form solution of the form

$$\mathbf{T}_{0i,k+1} = \underbrace{\exp((\tilde{\mathbf{U}}_{0,k} - \tilde{\mathbf{B}}_{0,k})\Delta t)^{-1}}_{\mathbf{B}_{0,k}^{-1}} \mathbf{T}_{0i,k} \underbrace{\exp((\tilde{\mathbf{U}}_{i,k} - \tilde{\mathbf{B}}_{i,k})\Delta t)}_{\mathbf{B}_{i,k}} \tag{E.7}$$

for an initial condition $\mathbf{T}_{0i,k}$.

The next step is then to linearize the discrete-time process model, in a manner similar to Section 6.5.3. Defining $\boldsymbol{\Omega}_{0,k}^{B} \triangleq \left(\boldsymbol{\omega}_{0,k}^{0a} - \boldsymbol{\beta}_{0,k}^{\text{gyr},0}\right)\Delta t$ yields

$$\begin{aligned}
\mathbf{B}_{0,k} &= \begin{bmatrix} \operatorname{Exp}(\boldsymbol{\Omega}_{0,k}^{B}) & \Delta t \mathbf{J}_l\left(\boldsymbol{\Omega}_{0,k}^{\text{B}}\right)\left(\boldsymbol{\alpha}_{0,k}^{0} - \boldsymbol{\beta}_{0,k}^{\text{acc},0i}\right) & \frac{\Delta t^2}{2}\mathbf{N}\left(\boldsymbol{\Omega}_{0,k}^{\text{B}}\right)\left(\boldsymbol{\alpha}_{0,k}^{0} - \boldsymbol{\beta}_{0,k}^{\text{acc},0i}\right) \\ & 1 & \Delta t \\ & & 1 \end{bmatrix} \\
&= \mathbf{M}\operatorname{Exp}\left(\underbrace{\begin{bmatrix} \Delta t\mathbf{1} & & \\ & \Delta t\mathbf{1} & \\ & & \frac{\Delta t^2}{2}\mathbf{J}_l\left(\boldsymbol{\Omega}_{0,k}^{\text{B}}\right)^{-1}\mathbf{N}\left(\boldsymbol{\Omega}_{0,k}^{\text{B}}\right) \end{bmatrix}}_{\mathbf{V}_{0,k}^{\text{B}}} \left(\underbrace{\begin{bmatrix} \boldsymbol{\omega}_{0,k}^{0a} \\ \boldsymbol{\alpha}_{0,k}^{0} \end{bmatrix}}_{\mathbf{u}_{0,k}} - \underbrace{\begin{bmatrix} \boldsymbol{\beta}_{0,k}^{\text{gyr},0} \\ \boldsymbol{\beta}_{0,k}^{\text{acc},0i} \end{bmatrix}}_{\boldsymbol{\beta}_{0,k}}\right)\right) \\
&\triangleq \mathbf{M}\operatorname{Exp}\left(\mathbf{V}_{0,k}^{\text{B}}(\mathbf{u}_{0,k} - \boldsymbol{\beta}_{0,k})\right).
\end{aligned}$$

Perturbing this with respect to the input $\mathbf{u}_{0,k}$ and the bias $\boldsymbol{\beta}_{0,k}$ yields

$$\mathbf{B}_{0,k} = \bar{\mathbf{B}}_{0,k}\operatorname{Exp}\left(\mathbf{L}_{0,k}^{\text{B}}(\delta\mathbf{u}_{0,k} - \delta\boldsymbol{\beta}_{0,k})\right),$$

where $\mathbf{L}_{0,k}^{\text{B}} \triangleq \boldsymbol{\mathcal{J}}_l(-\bar{\mathbf{V}}_{0,k}^{\text{B}}(\bar{\mathbf{u}}_{0,k} - \bar{\boldsymbol{\beta}}_{0,k}))\bar{\mathbf{V}}_{0,k}^{\text{B}}$, and $\boldsymbol{\mathcal{J}}_l(\cdot)$ is the left Jacobian of $SE_2(3)$. Similarly, defining $\boldsymbol{\Omega}_{i,k}^{B} \triangleq \left(\boldsymbol{\omega}_{i,k}^{ia} - \boldsymbol{\beta}_{i,k}^{\text{gyr,i}}\right)\Delta t$,

$$\begin{aligned}
\mathbf{B}_{i,k} &= \begin{bmatrix} \operatorname{Exp}(\boldsymbol{\Omega}_{i,k}^{B}) & \Delta t \mathbf{J}_l\left(\boldsymbol{\Omega}_{i,k}^{\text{B}}\right)\boldsymbol{\alpha}_{i,k}^{i} & \frac{\Delta t^2}{2}\mathbf{N}\left(\boldsymbol{\Omega}_{i,k}^{\text{B}}\right)\boldsymbol{\alpha}_{i,k}^{i} \\ & 1 & \Delta t \\ & & 1 \end{bmatrix} \\
&= \mathbf{M}\operatorname{Exp}\left(\underbrace{\begin{bmatrix} \Delta t\mathbf{1} & & \\ & \Delta t\mathbf{1} & \\ & & \frac{\Delta t^2}{2}\mathbf{J}_l\left(\boldsymbol{\Omega}_{i,k}^{\text{B}}\right)^{-1}\mathbf{N}\left(\boldsymbol{\Omega}_{i,k}^{\text{B}}\right) \end{bmatrix}}_{\mathbf{V}_{i,k}^{\text{B}}} \left(\underbrace{\begin{bmatrix} \boldsymbol{\omega}_{i,k}^{ia} \\ \boldsymbol{\alpha}_{i,k}^{i} \end{bmatrix}}_{\mathbf{u}_{i,k}} - \underbrace{\begin{bmatrix} \mathbf{1} \\ \mathbf{0} \end{bmatrix}}_{\mathbf{E}}\boldsymbol{\beta}_{i,k}^{\text{gyr,i}}\right)\right) \\
&\triangleq \mathbf{M}\operatorname{Exp}\left(\mathbf{V}_{i,k}^{\text{B}}(\mathbf{u}_{i,k} - \mathbf{E}\boldsymbol{\beta}_{i,k}^{\text{gyr,i}})\right).
\end{aligned}$$

Perturbing this with respect to the input $\mathbf{u}_{i,k}$ and the bias $\boldsymbol{\beta}_{i,k}^{\text{gyr,i}}$ yields

$$\mathbf{B}_{i,k} = \bar{\mathbf{B}}_{i,k} \operatorname{Exp}\left(\mathbf{L}_{i,k}^{\text{B}}(\delta\mathbf{u}_{i,k} - \mathbf{E}\delta\boldsymbol{\beta}_{i,k}^{\text{gyr,i}})\right), \tag{E.8}$$

where $\mathbf{L}_{i,k}^{\text{B}} \triangleq \boldsymbol{\mathcal{J}}_l(-\bar{\mathbf{V}}_{i,k}^{\text{B}}(\bar{\mathbf{u}}_{i,k} - \bar{\boldsymbol{\beta}}_{i,k}^{\text{gyr,i}}))\bar{\mathbf{V}}_{i,k}^{\text{B}}$.

E.2 Bias Process Model

Having derived and linearized the pose process model, the focus now shifts to the bias process model. The bias states being estimated are the gyroscope biases $\boldsymbol{\beta}_0^{\text{gyr},0}$ and $\boldsymbol{\beta}_i^{\text{gyr,i}}$, and the relative accelerometer bias $\boldsymbol{\beta}_0^{\text{acc},0\text{i}}$. The evolution of IMU biases is oftentimes modelled as a random walk [79], [133]. Therefore, the process model for the gyroscope biases is given by

$$\boldsymbol{\beta}_{0,k+1}^{\text{gyr},0} = \boldsymbol{\beta}_{0,k}^{\text{gyr},0} + \Delta t\mathbf{w}_{0,k}^{\text{gyr},0}, \tag{E.9}$$
$$\boldsymbol{\beta}_{i,k+1}^{\text{gyr,i}} = \boldsymbol{\beta}_{i,k}^{\text{gyr,i}} + \Delta t\mathbf{w}_{i,k}^{\text{gyr,i}}. \tag{E.10}$$

The relative accelerometer bias is more involved. The evolution of the individual accelerometer biases of the robots are also modelled as random walks,

$$\boldsymbol{\beta}_{0,k+1}^{\text{acc},0} = \boldsymbol{\beta}_{0,k}^{\text{acc},0} + \Delta t\mathbf{w}_{0,k}^{\text{acc},0}, \tag{E.11}$$
$$\boldsymbol{\beta}_{i,k+1}^{\text{acc,i}} = \boldsymbol{\beta}_{i,k}^{\text{acc,i}} + \Delta t\mathbf{w}_{i,k}^{\text{acc,i}}. \tag{E.12}$$

The evolution of the relative accelerometer bias is a function of the individual accelerometer biases of the robots and the relative pose between the robots, and is given by

$$\boldsymbol{\beta}_{0,k+1}^{\text{acc},0\text{i}} = \boldsymbol{\beta}_{0,k+1}^{\text{acc},0} - \boldsymbol{\Pi}\mathbf{T}_{0i,k+1}\boldsymbol{\Pi}^{\mathsf{T}}\boldsymbol{\beta}_{i,k+1}^{\text{acc,i}}, \tag{E.13}$$

where $\boldsymbol{\Pi} \triangleq \begin{bmatrix} \mathbf{1}_3 & \mathbf{0}_{3\times 2} \end{bmatrix} \in \mathbb{R}^{3\times 5}$. Using (E.7), (E.11), and (E.12), the relation in (E.13) can be written as

$$\boldsymbol{\beta}_{0,k+1}^{\text{acc},0\text{i}} = \boldsymbol{\beta}_{0,k}^{\text{acc},0} - \boldsymbol{\Pi}\mathbf{B}_{0,k}^{-1}\mathbf{T}_{0i,k}\mathbf{B}_{i,k}\boldsymbol{\Pi}^{\mathsf{T}}\boldsymbol{\beta}_{i,k}^{\text{acc,i}} + \Delta t\mathbf{w}_{0,k}^{\text{acc},0} - \boldsymbol{\Pi}\mathbf{B}_{0,k}^{-1}\mathbf{T}_{0i,k}\mathbf{B}_{i,k}\boldsymbol{\Pi}^{\mathsf{T}}\Delta t\mathbf{w}_{i,k}^{\text{acc,i}} \tag{E.14}$$
$$\approx \boldsymbol{\beta}_{0,k}^{\text{acc},0\text{i}} + \Delta t\mathbf{w}_{0,k}^{\text{acc},0} - \boldsymbol{\Pi}\mathbf{B}_{0,k}^{-1}\mathbf{T}_{0i,k}\mathbf{B}_{i,k}\boldsymbol{\Pi}^{\mathsf{T}}\Delta t\mathbf{w}_{i,k}^{\text{acc,i}}, \tag{E.15}$$

where the lattermost approximation is dependent on an assumption that Δt is sufficiently small. As $\Delta t \to 0$, it can be shown that $\mathbf{B}_{0,k} \to \mathbf{1}$ and $\mathbf{B}_{i,k} \to \mathbf{1}$, meaning that the first two components in (E.14) are approximately of the same form as the right-hand side in (E.13) and can be combined into the relative acceleromer bias term $\boldsymbol{\beta}_{0,k}^{\text{acc},0\text{i}}$. Perturbing (E.9), (E.10),

and (E.15) is then straightforward.

E.3 Preintegration

The preintegration of the IMU measurements is also affected by the presence of IMU biases, but is quite similar to the preintegration shown in Section 6.6. In the proposed preintegration framework, each robot constructs its RMI by correcting the gyroscope measurements using its own gyroscope bias estimate and inflating the uncertainty associated with the RMI based on the uncertainty of the bias estimate. Nonetheless, each robot leaves the accelerometer measurements uncorrected when constructing the RMI.

To derive this, first note that

$$\mathbf{T}_{0i,m} = \left(\prod_{k=\ell}^{m-1} \mathbf{B}_{0,k}\right)^{-1} \mathbf{T}_{0i,\ell} \prod_{k=\ell}^{m-1} \mathbf{B}_{i,k}, \tag{E.16}$$

meaning that the RMI constructed by Robot i is of the form

$$\Delta\mathbf{T}^{\mathrm{B}}_{i,\ell:m} = \prod_{k=\ell}^{m-1} \mathbf{B}_{i,k} \in DE_2(3).$$

Therefore, (E.16) can be written as

$$\mathbf{T}_{0i,m} = \left(\prod_{k=\ell}^{m-1} \mathbf{B}_{0,k}\right)^{-1} \mathbf{T}_{0i,\ell} \Delta\mathbf{T}^{\mathrm{B}}_{i,\ell:m}, \tag{E.17}$$

which differs from the RMI in Section 6.6 in that the gyroscope measurements are corrected using the Robot i's estimate of its own gyroscope bias. Consequently, the RMI can be updated iteratively as

$$\Delta\mathbf{T}^{\mathrm{B}}_{i,\ell:k+1} = \Delta\mathbf{T}^{\mathrm{B}}_{i,\ell:k} \mathbf{B}_{i,k}. \tag{E.18}$$

As in Section 6.6, a perturbation of the form

$$\Delta\mathbf{T}^{\mathrm{B}}_{i,\ell:m} = \Delta\bar{\mathbf{T}}^{\mathrm{B}}_{i,\ell:m} \operatorname{Exp}(\delta\mathbf{w}^{\mathrm{B}}_{i,\ell:m})$$

is defined for the RMI, and using (E.8), the perturbation of the RMI is given by

$$\delta\mathbf{w}^{\mathrm{B}}_{i,\ell:k+1} = \operatorname{Ad}(\bar{\mathbf{B}}^{-1}_{i,k})\delta\mathbf{w}^{\mathrm{B}}_{i,\ell:k} + \mathbf{L}^{\mathrm{B}}_{i,k}\delta\mathbf{u}_{i,k} - \mathbf{L}^{\mathrm{B}}_{i,k}\mathbf{E}\delta\boldsymbol{\beta}^{\mathrm{gyr},i}_{i,k}, \tag{E.19}$$

where the last term reflects the increased uncertainty of the RMI associated with the uncer-

tainty in the gyroscope bias estimate.

Lastly, the asynchronous-input filter shown in Section 6.6.3 can now be formulated for bias-modelling applications. At time-steps where there is no communication with the neighbour, the pose and bias process models are given by

$$\boldsymbol{\mathcal{T}}_{0i,k+1} = \mathbf{B}_{0,k}^{-1}\mathbf{T}_{0i,k}, \qquad \boldsymbol{\mathcal{T}}_{0i,k+1} \in DE_2(3), \tag{E.20}$$

$$\boldsymbol{\beta}_{0,k+1}^{\mathrm{gyr},0} = \boldsymbol{\beta}_{0,k}^{\mathrm{gyr},0} + \Delta t \mathbf{w}_{0,k}^{\mathrm{gyr},0}, \tag{E.21}$$

$$\boldsymbol{\beta}_{0,k+1}^{\mathrm{acc},0\mathrm{i}} = \boldsymbol{\beta}_{0,k}^{\mathrm{acc},0\mathrm{i}} + \Delta t \mathbf{w}_{0,k}^{\mathrm{acc},0} - \boldsymbol{\Pi}\mathbf{B}_{0,k}^{-1}\mathbf{T}_{0i,k}\boldsymbol{\Pi}^{\mathsf{T}}\Delta t \mathbf{w}_{i,k}^{\mathrm{acc},\mathrm{i}}. \tag{E.22}$$

Meanwhile, at time-steps when Robot i sends the RMI $\Delta\mathbf{T}_{i,\ell:m}^{\mathrm{B}}$, the pose and bias process models are given by

$$\mathbf{T}_{0i,m} = \mathbf{B}_{0,m-1}^{-1}\boldsymbol{\mathcal{T}}_{0i,m-1}\Delta\mathbf{T}_{i,\ell:m}^{\mathrm{B}}, \tag{E.23}$$

$$\boldsymbol{\beta}_{0,m}^{\mathrm{gyr},0} = \boldsymbol{\beta}_{0,m-1}^{\mathrm{gyr},0} + \Delta t \mathbf{w}_{0,m-1}^{\mathrm{gyr},0}, \tag{E.24}$$

$$\boldsymbol{\beta}_{0,m}^{\mathrm{acc},0\mathrm{i}} = \boldsymbol{\beta}_{0,m-1}^{\mathrm{acc},0\mathrm{i}} + \Delta t \mathbf{w}_{0,m-1}^{\mathrm{acc},0} - \boldsymbol{\Pi}\mathbf{B}_{0,m-1}^{-1}\boldsymbol{\mathcal{T}}_{0i,m-1}\Delta\mathbf{T}_{i,\ell:m}^{\mathrm{B}}\boldsymbol{\Pi}^{\mathsf{T}}\Delta t \mathbf{w}_{i,m-1}^{\mathrm{acc},\mathrm{i}} \tag{E.25}$$

These equations can then be perturbed in a manner similar to Section 6.6.3.

E.4 Simulation Results

Table E.1: Bias simulation parameters. Other simulation parameters remain unchanged from Table 6.1.

Specification	Value
Accelerometer bias random walk std. dev. [m/s^2]	1.58×10^{-3}
Gyroscope bias random walk std. dev. [rad/s]	2.5×10^{-5}

In order to validate the proposed framework in the presence of IMU biases, the same simulation runs as in Section 6.7 are repeated, but with the addition of IMU biases. The bias simulation parameters are given in Table E.1. Given that neighbours use their own gyroscope bias estimates to correct their gyroscope measurement before constructing the RMI, additional noise is added to the gyroscope bias true state of neighbours to simulate uncertain gyroscope bias estimates. This is then used to correct the gyroscope measurements and to inflate the RMI.

The results for Simulation S1 are shown in Figure E.1, where it can be seen that the gyroscope and relative accelerometer biases estimated by the reference robot do converge

Table E.2: The RMSE of Robot 0's estimate of neighbouring robots' relative pose for multiple experimental trials, without offline bias correction.

	Robot 1		**Robot 2**	
	Without Bias Estimation RMSE [m]	**With Bias Estimation RMSE [m]**	**Without Bias Estimation RMSE [m]**	**With Bias Estimation RMSE [m]**
Trial 1	0.785	**0.404**	0.723	**0.410**
Trial 2	1.232	**0.828**	0.902	**0.615**
Trial 3	0.916	**0.548**	0.649	**0.413**
Trial 4	1.282	**0.753**	0.853	**0.638**

to the true values. Additionally, Simulation S3 is run to assess the consistency of the proposed estimator in the presence of IMU biases. The NEES plot for this simulation is shown in Figure E.2, where it can be seen that the NEES values display a similar behaviour to Figure 6.11, starting with weak observability and then converging towards consistency.

E.5 Experimental Results

The experimental results are also rerun with IMU bias estimation to validate the approach proposed in this document. These results differ from the results presented in Section 6.8 in that the IMU biases are no longer initialized using the motion capture system, except for the gyroscope biases of neighbouring robots to simulate neighbours running their own estimator. The proposed framework presented in Chapter 6 is then compared to the one presented here with IMU bias estimation, and it is shown that estimating biases in the absence of bias initialization does indeed improve performance, as shown in Figure E.5 for Trial 1 and in Table E.2 for all trials. Note that the bias error plots are not shown as the true IMU bias is unknown. The performance with bias estimation is also comparable to the performance of the estimator in Section 6.8 with bias initialization, but is in fact typically worse probably due to the transient of the bias estimates before convergence, thus resulting in more uncertain pose estimates during the earlier stages. It is expected that for longer trajectories the performance of the estimator with bias estimation will be better than the estimator in Chapter 6 with bias initialization, as the initial bias estimates becomes less accurate with the progress of time.

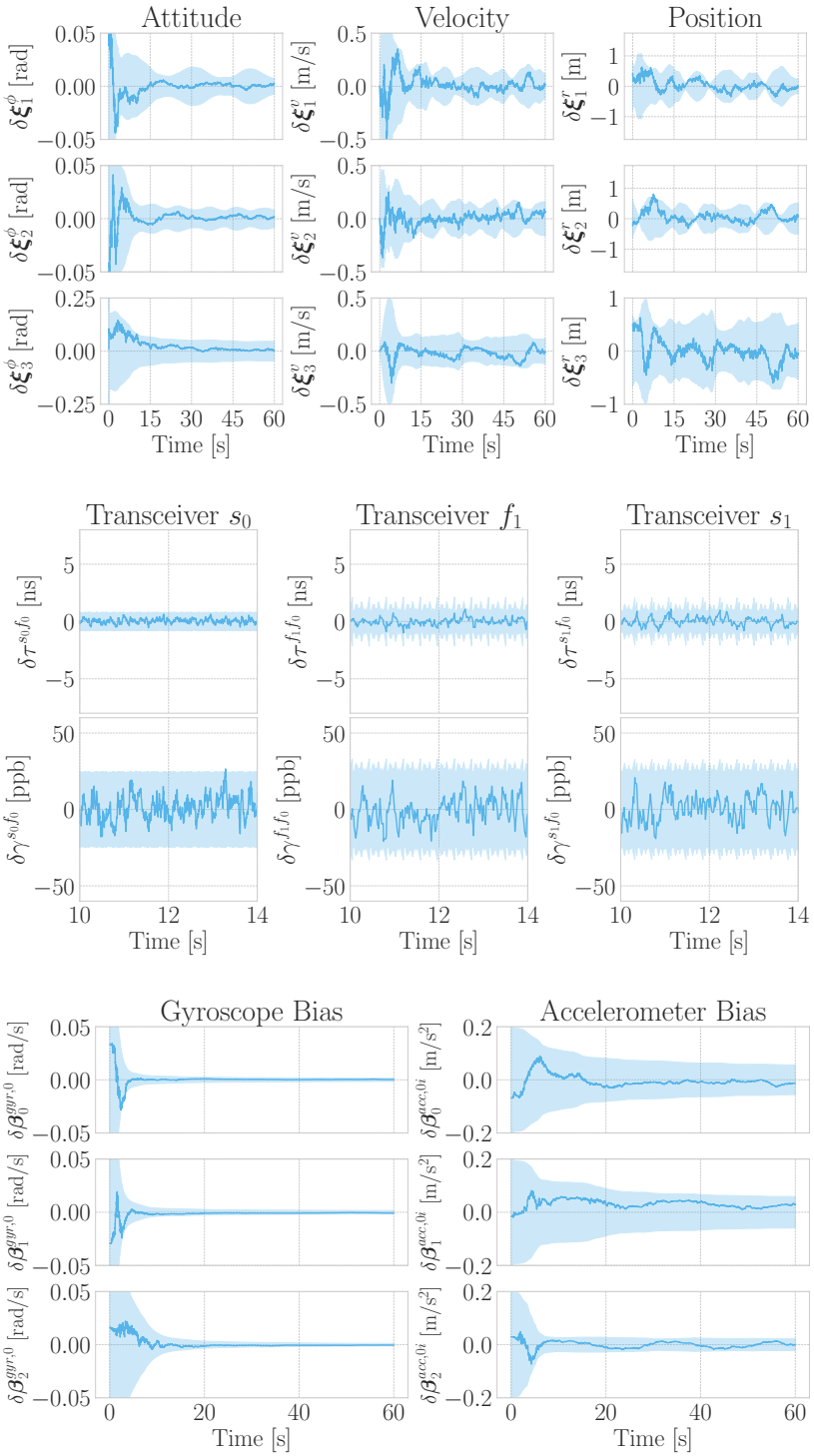


Figure E.1: Error plots and $\pm 3\sigma$ bounds (shaded region) for Robot 0's estimate of Robot 1's relative pose, its own gyroscope bias, and Robot 1's relative accelerometer bias for Simulation S1.

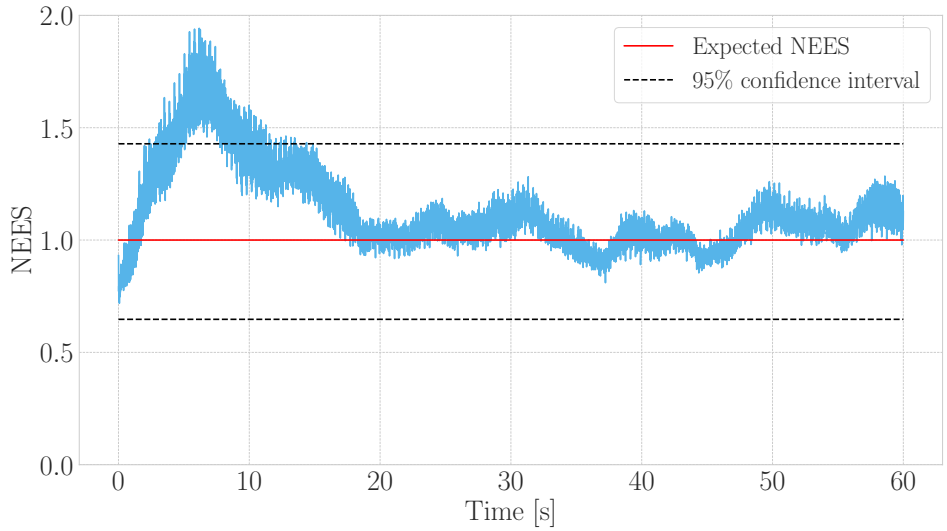


Figure E.2: 50-trial NEES plot for the proposed estimator on Simulation S3 in the presence of bias.

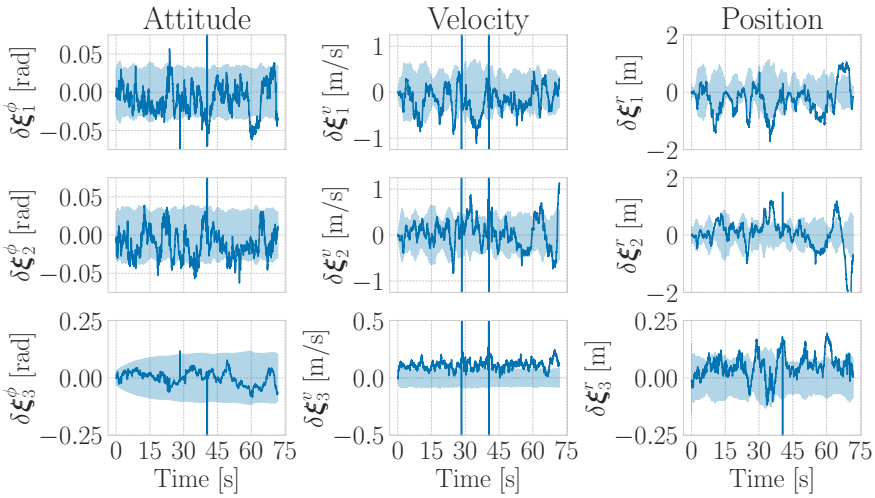


Figure E.3: Without bias estimation.

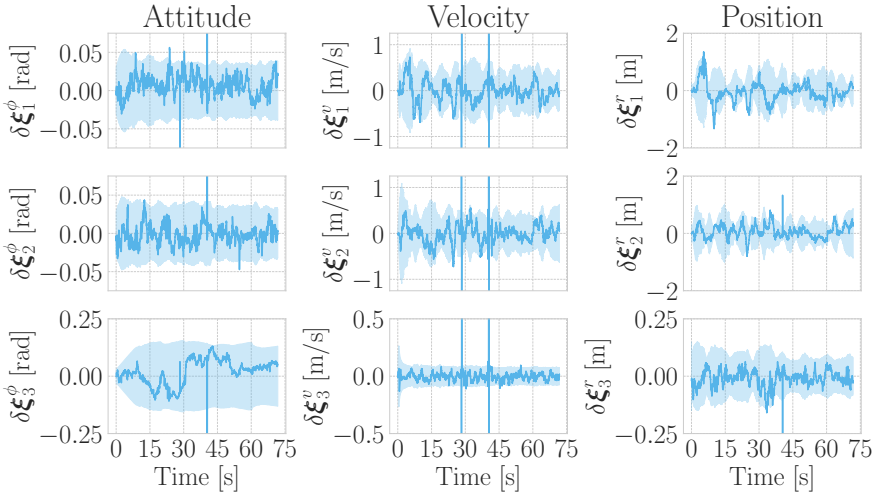


Figure E.4: With bias estimation.

Figure E.5: Error plots and $\pm 3\sigma$ bounds (shaded region) for Robot 0's estimate of Robot 1's relative pose for experimental trial 1, without offline bias correction.

Appendix F

Deriving the Measurement Update

The conditional joint distribution of $\mathbf{x}_k^1, \mathbf{x}_k^2, \mathbf{y}_k^1 \big| \mathcal{I}_{k-1}^1, \mathcal{I}_k^2$ is assumed to be Gaussian according to

$$\mathcal{N}\left(\begin{bmatrix} \check{\mathbf{x}}_k^1 \\ \hat{\mathbf{x}}_k^2 \\ \check{\mathbf{y}}_k^1 \end{bmatrix}, \begin{bmatrix} \check{\mathbf{P}}_k^1 & \check{\mathbf{P}}_{k,k}^{1,2} & \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} \\ \left(\check{\mathbf{P}}_{k,k}^{1,2}\right)^\mathsf{T} & \hat{\mathbf{P}}_k^2 & \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1} \\ \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1}^\mathsf{T} & \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1}^\mathsf{T} & \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1} \end{bmatrix} \right).$$

Using [43, Lemma A.2] to condition on $\mathbf{y}_k^1$ as well, the conditional joint distribution of $\mathbf{x}_k^1, \mathbf{x}_k^2 \big| \mathcal{I}_k^1, \mathcal{I}_k^2$ is given by

$$\mathcal{N}\left(\begin{bmatrix} \check{\mathbf{x}}_k^1 + \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \left(\mathbf{y}_k^1 - \check{\mathbf{y}}_k^1\right) \\ \hat{\mathbf{x}}_k^2 + \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \left(\mathbf{y}_k^1 - \check{\mathbf{y}}_k^1\right) \end{bmatrix}, \begin{bmatrix} \check{\mathbf{P}}_k^1 - \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1}^\mathsf{T} & \check{\mathbf{P}}_{k,k}^{1,2} - \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1}^\mathsf{T} \\ \left(\check{\mathbf{P}}_{k,k}^{1,2}\right)^\mathsf{T} - \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1}^\mathsf{T} & \hat{\mathbf{P}}_k^2 - \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1}^\mathsf{T} \end{bmatrix} \right).$$

Using [43, Lemma A.2] again, and by replacing the conditioning on $\mathcal{I}_k^2$ using $\hat{\mathbf{x}}_k^2\left(\mathcal{I}_k^2\right)$ and $\hat{\mathbf{P}}_k^2\left(\mathcal{I}_k^2\right)$ as was done in Section 7.2, the distribution of $\mathbf{x}_k^1 | \mathcal{I}_k^1, \hat{\mathbf{x}}_k^2, \hat{\mathbf{P}}_k^2$ is given by

$$\begin{aligned}
&\mathcal{N}\Big(\check{\mathbf{x}}_k^1 + \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \Big(\mathbf{y}_k^1 - \check{\mathbf{y}}_k^1\Big) \\
&\quad + \Big(\check{\mathbf{P}}_{k,k}^{1,2} - \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1}^\mathsf{T}\Big)\Big(\hat{\mathbf{P}}_k^2 - \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1}^\mathsf{T}\Big)^{-1} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \Big(\mathbf{y}_k^1 - \check{\mathbf{y}}_k^1\Big), \\
&\check{\mathbf{P}}_k^1 - \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1}^\mathsf{T} \\
&\quad - \Big(\check{\mathbf{P}}_{k,k}^{1,2} - \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1}^\mathsf{T}\Big)\Big(\hat{\mathbf{P}}_k^2 - \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1}^\mathsf{T}\Big)^{-1} \Big(\left(\check{\mathbf{P}}_{k,k}^{1,2}\right)^\mathsf{T} - \boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1} \boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}^{-1} \boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1}^\mathsf{T}\Big)\Big).
\end{aligned}$$

Therefore, the filter equations using a Bayesian approach are given by (7.10)-(7.14).

Appendix G

Sigma Point-Based Covariance Intersection

Consider the problem of fusing two estimates

$$\mathbf{x}^1 \sim \mathcal{N}\left(\hat{\mathbf{x}}^1, \hat{\mathbf{P}}^1\right), \tag{G.1}$$

$$\mathbf{x}^2 \sim \mathcal{N}\left(\hat{\mathbf{x}}^2, \hat{\mathbf{P}}^2\right), \tag{G.2}$$

of the same state vector $\mathbf{x}$, where the cross-covariance matrix

$$\mathbf{P}^{1,2} = \mathbb{E}\left[\left(\mathbf{x} - \hat{\mathbf{x}}^1\right)\left(\mathbf{x} - \hat{\mathbf{x}}^2\right)^\mathsf{T}\right] \tag{G.3}$$

is unknown. One consistent way to obtain a state estimate $\hat{\mathbf{x}}$ with a covariance matrix $\hat{\mathbf{P}}$ by fusing $\mathbf{x}^1$ and $\mathbf{x}^2$ is the Covariance Intersection (CI) method [36], [135]. The core of this approach is to disregard any cross-covariances by inflating the joint covariance matrices $\hat{\mathbf{P}}^1$ and $\hat{\mathbf{P}}^2$. Therefore, the assumed joint distribution between $\mathbf{x}^1$ and $\mathbf{x}^2$ is given by

$$\begin{bmatrix} \mathbf{x}^1 \\ \mathbf{x}^2 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \hat{\mathbf{x}}^1 \\ \hat{\mathbf{x}}^2 \end{bmatrix}, \begin{bmatrix} \frac{1}{w}\hat{\mathbf{P}}^1 & \mathbf{0} \\ \mathbf{0} & \frac{1}{1-w}\hat{\mathbf{P}}^2 \end{bmatrix}\right), \tag{G.4}$$

where $w \in (0, 1)$ is a weighting parameter.

In Sections 7.4 and 7.5, a sigma point Covariance Intersection (SPCI) approach is used as an alternative, less accurate solution to the proposed framework. The proposed framework attempts to approximate the cross-covariance matrices, while CI inflates the joint covariances matrices to avoid doing so. Therefore, designing a SPCI approach is similar to the proposed framework. The state vectors $\mathbf{x}^1_{k-1}$ and $\mathbf{x}^2_{k-1}$ and the process noise $\mathbf{w}^1_{k-1}$ are augmented into

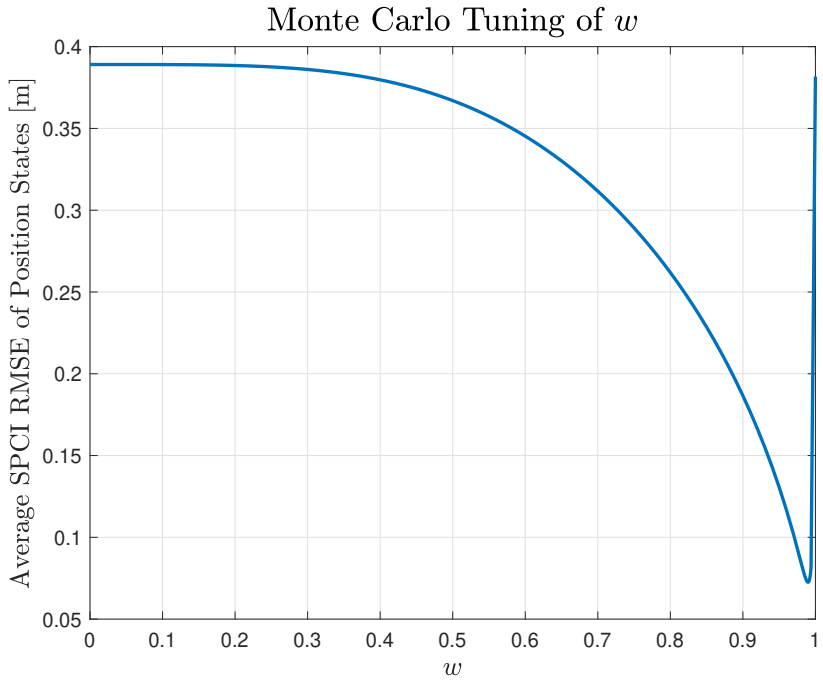


Figure G.1: The RSME of the SPCI filter averaged over 50 different trajectories for different values of w, showing best performance at $w = 0.99$.

one vector as in (7.15). The difference, however, is in the covariance matrix, which now is of the form

$$\hat{\mathbf{P}}_{\mathbf{v}_{k-1}} = \begin{bmatrix} \frac{1}{w}\hat{\mathbf{P}}^1_{k-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{1-w}\hat{\mathbf{P}}^2_{k-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}^1_{k-1} \end{bmatrix} \tag{G.5}$$

instead of (7.16). Similarly, for the correction step, the new augmented state vector is of the form (7.20) and the new covariance matrix is of the form

$$\hat{\mathbf{P}}_{\mathbf{u}_k} = \begin{bmatrix} \frac{1}{w}\check{\mathbf{P}}^1_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{1-w}\check{\mathbf{P}}^2_k & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}^1_k \end{bmatrix}. \tag{G.6}$$

One of the disadvantages of the Covariance Intersection method is that it requires tuning of an additional scalar parameter, w. Usually, when the two vectors being fused are different estimates of the same state vector, the weighting factors are the result of an optimization approach that minimizes the trace or some other metric of the fused covariance matrix. However, in the problem of cascaded filtering when state vectors being estimated are distinct, the different units of the diagonal elements of the fused covariance matrix would mean this

approach yields biased results. Another approach involves choosing the weighting parameters and the Kalman gain to minimize the trace of the posterior covariance matrix, as in [40]. However, it is unclear how this approach could be utilized in a sigma point-based approach. Therefore, the value of the weighting parameter w is tuned using Monte Carlo trials as shown in Figure G.1, leading to the choice of $w = 0.99$.

Appendix H

Linearization-Based Approach

H.1 Proposed Linearization-Based Cascaded Receiving Filter

In this section, a linearization-based approach is derived using the proposed framework. Deriving a linearized model from a nonlinear system is standard, using a first-order Taylor series approximation. Therefore, rather than considering a nonlinear system and addressing its linearized counterpart, a discrete-time linear system of the form

$$\begin{aligned}
\mathbf{x}_k^1 &= \mathbf{A}_{k-1}^1 \mathbf{x}_{k-1}^1 + \mathbf{B}_{k-1}^1 \mathbf{x}_{k-1}^2 + \mathbf{L}_{k-1}^1 \mathbf{w}_{k-1}^1, \\
\mathbf{y}_k^1 &= \mathbf{C}_k^1 \mathbf{x}_k^1 + \mathbf{D}_k^1 \mathbf{x}_k^2 + \mathbf{M}_k^1 \boldsymbol{\nu}_k^1, \\
\mathbf{x}_k^2 &= \mathbf{A}_{k-1}^2 \mathbf{x}_{k-1}^2 + \mathbf{L}_{k-1}^2 \mathbf{w}_{k-1}^2, \\
\mathbf{y}_k^2 &= \mathbf{C}_k^2 \mathbf{x}_k^2 + \mathbf{M}_k^2 \boldsymbol{\nu}_k^2,
\end{aligned}$$

will be considered for conciseness of notation, where

$$\begin{aligned}
\mathbf{w}_{k-1}^1 &\sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_{k-1}^1\right), \quad \boldsymbol{\nu}_k^1 \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_k^1\right), \\
\mathbf{w}_{k-2}^1 &\sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_{k-1}^2\right), \quad \boldsymbol{\nu}_k^2 \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_k^2\right).
\end{aligned}$$

In what follows, the notation $\mathcal{I}_k = \mathcal{I}_k^1 \cup \mathcal{I}_k^2$ is used.

H.1.1 Prediction Step

When propagating from time-step $k-1$ to k, the state estimates $\hat{\mathbf{x}}_{k-1}^1$, $\hat{\mathbf{x}}_{k-1}^2$ and the covariance matrices $\hat{\mathbf{P}}_{k-1}^1$, $\hat{\mathbf{P}}_{k-1}^2$, $\hat{\mathbf{P}}_{k-1,k-1}^{1,2}$ are known. Therefore, the prediction step is straightforward,

and is given by

$$
\begin{aligned}
\check{\mathbf{x}}_k^1 &= \mathbf{A}_{k-1}^1 \hat{\mathbf{x}}_{k-1}^1 + \mathbf{B}_{k-1}^1 \hat{\mathbf{x}}_{k-1}^2, \\
\check{\mathbf{P}}_k^1 &= \mathbb{E}\left[\left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right) \left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right)^{\mathsf{T}} \,\middle|\, \mathcal{I}_{k-1} \right] \\
&= \mathbb{E}\Big[\left(\mathbf{A}_{k-1}^1 \mathbf{x}_{k-1}^1 + \mathbf{B}_{k-1}^1 \mathbf{x}_{k-1}^2 + \mathbf{L}_{k-1}^1 \mathbf{w}_{k-1}^1 - \mathbf{A}_{k-1}^1 \hat{\mathbf{x}}_{k-1}^1 + \mathbf{B}_{k-1}^1 \hat{\mathbf{x}}_{k-1}^2\right) (\cdot)^{\mathsf{T}} \,\Big|\, \mathcal{I}_{k-1} \Big] \\
&= \mathbb{E}\Big[\left(\mathbf{A}_{k-1}^1 \left(\mathbf{x}_{k-1}^1 - \hat{\mathbf{x}}_{k-1}^1\right) + \mathbf{B}_{k-1}^1 \left(\mathbf{x}_{k-1}^2 - \hat{\mathbf{x}}_{k-1}^2\right) + \mathbf{L}_{k-1}^1 \mathbf{w}_{k-1}^1\right) (\cdot)^{\mathsf{T}} \,\Big|\, \mathcal{I}_{k-1} \Big] \\
&= \mathbf{A}_{k-1}^1 \hat{\mathbf{P}}_{k-1}^1 \left(\mathbf{A}_{k-1}^1\right)^{\mathsf{T}} + \mathbf{A}_{k-1}^1 \hat{\mathbf{P}}_{k-1,k-1}^{1,2} \left(\mathbf{B}_{k-1}^1\right)^{\mathsf{T}} + \mathbf{B}_{k-1}^1 \left(\hat{\mathbf{P}}_{k-1,k-1}^{1,2}\right)^{\mathsf{T}} \left(\mathbf{A}_{k-1}^1\right)^{\mathsf{T}} \\
&\quad + \mathbf{B}_{k-1}^1 \hat{\mathbf{P}}_{k-1}^2 \left(\mathbf{B}_{k-1}^1\right)^{\mathsf{T}} + \mathbf{L}_{k-1}^1 \mathbf{Q}_{k-1}^1 \left(\mathbf{L}_{k-1}^1\right)^{\mathsf{T}}, \\
\check{\mathbf{P}}_{k,k-1}^{1,2} &= \mathbb{E}\left[\left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right) \left(\mathbf{x}_{k-1}^2 - \hat{\mathbf{x}}_{k-1}^2\right)^{\mathsf{T}} \,\middle|\, \mathcal{I}_{k-1} \right] \\
&= \mathbb{E}\Big[\left(\mathbf{A}_{k-1}^1 \mathbf{x}_{k-1}^1 + \mathbf{B}_{k-1}^1 \mathbf{x}_{k-1}^2 + \mathbf{L}_{k-1}^1 \mathbf{w}_{k-1}^1 - \mathbf{A}_{k-1}^1 \hat{\mathbf{x}}_{k-1}^1 + \mathbf{B}_{k-1}^1 \hat{\mathbf{x}}_{k-1}^2\right) \left(\mathbf{x}_{k-1}^2 - \hat{\mathbf{x}}_{k-1}^2\right)^{\mathsf{T}} \,\Big|\, \mathcal{I}_{k-1} \Big] \\
&= \mathbb{E}\Big[\left(\mathbf{A}_{k-1}^1 \left(\mathbf{x}_{k-1}^1 - \hat{\mathbf{x}}_{k-1}^1\right) + \mathbf{B}_{k-1}^1 \left(\mathbf{x}_{k-1}^2 - \hat{\mathbf{x}}_{k-1}^2\right) + \mathbf{L}_{k-1}^1 \mathbf{w}_{k-1}^1\right) \left(\mathbf{x}_{k-1}^2 - \hat{\mathbf{x}}_{k-1}^2\right)^{\mathsf{T}} \,\Big|\, \mathcal{I}_{k-1} \Big] \\
&= \mathbf{A}_{k-1}^1 \hat{\mathbf{P}}_{k-1,k-1}^{1,2} + \mathbf{B}_{k-1}^1 \hat{\mathbf{P}}_{k-1}^2,
\end{aligned}
$$

where (expression)$(\cdot)^{\mathsf{T}}$ is used to denote (expression)(expression)$^{\mathsf{T}}$, and the assumptions

$$
\begin{aligned}
\mathbb{E}\Big[\left(\mathbf{x}_{k-1}^1 - \hat{\mathbf{x}}_{k-1}^1\right) \left(\mathbf{w}_{k-1}^1\right)^{\mathsf{T}} \,\Big|\, \mathcal{I}_{k-1} \Big] &= \mathbf{0}, \\
\mathbb{E}\Big[\left(\mathbf{x}_{k-1}^2 - \hat{\mathbf{x}}_{k-1}^2\right) \left(\mathbf{w}_{k-1}^1\right)^{\mathsf{T}} \,\Big|\, \mathcal{I}_{k-1} \Big] &= \mathbf{0}
\end{aligned}
$$

are made. Finding $\check{\mathbf{P}}_{k,k}^{1,2}$ from $\check{\mathbf{P}}_{k,k-1}^{1,2}$ then follows the approximation given in Section 7.3.3.

H.1.2 Correction Step

When correcting the predicted state at time-step k using measurements $\mathbf{y}_k^1$, the state estimates $\check{\mathbf{x}}_k^1$, $\hat{\mathbf{x}}_k^2$ and the covariance matrices $\check{\mathbf{P}}_k^1$, $\hat{\mathbf{P}}_k^2$, $\check{\mathbf{P}}_{k,k}^{1,2}$ are known. The posterior distribution of the states of the receiving filter, conditioned on the states of the feeding filter is derived in Appendix F and Section 7.2. Based on this, a filter of the form (7.10)-(7.14) is given. The analytical covariance terms $\boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1}$, $\boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2 \check{\mathbf{y}}_k^1}$, $\boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1 \check{\mathbf{y}}_k^1}$ for the linear system are then derived to be

$$
\begin{aligned}
\boldsymbol{\Sigma}_{\check{\mathbf{x}}_k^1 \check{\mathbf{y}}_k^1} &= \mathbb{E}\left[\left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right) \left(\mathbf{y}_k^1 - \check{\mathbf{y}}_k^1\right)^{\mathsf{T}} \,\middle|\, \mathcal{I}_k \right] \\
&= \mathbb{E}\Big[\left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right) \left(\mathbf{C}_k^1 \mathbf{x}_k^1 + \mathbf{D}_k^1 \mathbf{x}_k^2 + \mathbf{M}\boldsymbol{\nu}_k^1 - \mathbf{C}_k^1 \check{\mathbf{x}}_k^1 - \mathbf{D}_k^1 \hat{\mathbf{x}}_k^2 \,\middle|\, \mathcal{I}_k\right)^{\mathsf{T}} \Big] \\
&= \mathbb{E}\Big[\left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right) \left(\mathbf{C}_k^1 \left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right) + \mathbf{D}_k^1 \left(\mathbf{x}_k^2 - \hat{\mathbf{x}}_k^2\right) + \mathbf{M}\boldsymbol{\nu}_k^1\right)^{\mathsf{T}} \,\Big|\, \mathcal{I}_k \Big] \\
&= \check{\mathbf{P}}_k^1 \left(\mathbf{C}_k^1\right)^{\mathsf{T}} + \check{\mathbf{P}}_{k,k}^{1,2} \left(\mathbf{D}_k^1\right)^{\mathsf{T}},
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\Sigma}_{\hat{\mathbf{x}}_k^2\check{\mathbf{y}}_k^1} &= \mathbb{E}\left[\left(\mathbf{x}_k^2 - \hat{\mathbf{x}}_k^2\right)\left(\mathbf{y}_k^1 - \check{\mathbf{y}}_k^1\right)^{\mathsf{T}} \mid \mathcal{I}_k\right] \\
&= \mathbb{E}\Big[\left(\mathbf{x}_k^2 - \hat{\mathbf{x}}_k^2\right)\left(\mathbf{C}_k^1\mathbf{x}_k^1 + \mathbf{D}_k^1\mathbf{x}_k^2 + \mathbf{M}\boldsymbol{\nu}_k^1 - \mathbf{C}_k^1\check{\mathbf{x}}_k^1 - \mathbf{D}_k^1\hat{\mathbf{x}}_k^2\right)^{\mathsf{T}} \mid \mathcal{I}_k\Big] \\
&= \mathbb{E}\Big[\left(\mathbf{x}_k^2 - \hat{\mathbf{x}}_k^2\right)\left(\mathbf{C}_k^1\left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right) + \mathbf{D}_k^1\left(\mathbf{x}_k^2 - \hat{\mathbf{x}}_k^2\right) + \mathbf{M}\boldsymbol{\nu}_k^1\right)^{\mathsf{T}} \mid \mathcal{I}_k\Big] \\
&= \left(\check{\mathbf{P}}_{k,k}^{1,2}\right)^{\mathsf{T}}\left(\mathbf{C}_k^1\right)^{\mathsf{T}} + \hat{\mathbf{P}}_k^2\left(\mathbf{D}_k^1\right)^{\mathsf{T}}, \\
\boldsymbol{\Sigma}_{\check{\mathbf{y}}_k^1\check{\mathbf{y}}_k^1} &= \mathbb{E}\left[\left(\mathbf{y}_k^1 - \check{\mathbf{y}}_k^1\right)\left(\mathbf{y}_k^1 - \check{\mathbf{y}}_k^1\right)^{\mathsf{T}} \mid \mathcal{I}_k\right] \\
&= \mathbb{E}\Big[\left(\mathbf{C}_k^1\mathbf{x}_k^1 + \mathbf{D}_k^1\mathbf{x}_k^2 + \mathbf{M}\boldsymbol{\nu}_k^1 - \mathbf{C}_k^1\check{\mathbf{x}}_k^1 - \mathbf{D}_k^1\hat{\mathbf{x}}_k^2\right)(\cdot)^{\mathsf{T}} \mid \mathcal{I}_k\Big] \\
&= \mathbb{E}\Big[\left(\mathbf{C}_k^1\left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right) + \mathbf{D}_k^1\left(\mathbf{x}_k^2 - \hat{\mathbf{x}}_k^2\right) + \mathbf{M}\boldsymbol{\nu}_k^1\right)(\cdot)^{\mathsf{T}} \mid \mathcal{I}_k\Big] \\
&= \mathbf{C}_k^1\check{\mathbf{P}}_k^1\left(\mathbf{C}_k^1\right)^{\mathsf{T}} + \mathbf{C}_k^1\check{\mathbf{P}}_{k,k}^{1,2}\left(\mathbf{D}_k^1\right)^{\mathsf{T}} + \mathbf{D}_k^1\left(\check{\mathbf{P}}_{k,k}^{1,2}\right)^{\mathsf{T}}\left(\mathbf{C}_k^1\right)^{\mathsf{T}} + \mathbf{D}_k^1\hat{\mathbf{P}}_k^2\left(\mathbf{D}_k^1\right)^{\mathsf{T}} + \mathbf{M}_k^1\mathbf{R}_k^1\left(\mathbf{M}_k^1\right)^{\mathsf{T}},
\end{aligned}
$$

where (expression)$(\cdot)^{\mathsf{T}}$ is used to denote (expression)(expression)$^{\mathsf{T}}$, and the assumptions

$$
\begin{aligned}
\mathbb{E}\Big[\left(\mathbf{x}_k^1 - \check{\mathbf{x}}_k^1\right)\left(\boldsymbol{\nu}_k^1\right)^{\mathsf{T}} \mid \mathcal{I}_k\Big] &= \mathbf{0}, \\
\mathbb{E}\Big[\left(\mathbf{x}_k^2 - \hat{\mathbf{x}}_k^2\right)\left(\boldsymbol{\nu}_k^1\right)^{\mathsf{T}} \mid \mathcal{I}_k\Big] &= \mathbf{0}
\end{aligned}
$$

are made.

H.2 Simulation Results

The nonlinear problem addressed in Section 7.4 will also be used here to evaluate the linearization-based approach. To evaluate the different approaches, 500 Monte Carlo trials with varying initial conditions and noise realizations are performed as in Section 7.4. A summary of the results are given in Figure H.1 and Table H.1. The proposed sigma point estimator achieves the best performance and beats both the linearization-based approach and the SPCI. The linearization-based approach on average performs better than the SPCI estimator. However, the presence of more significant outliers is possibly due to linearization errors, while the SPCI approach considered here uses sigma points and does not suffer from linearization errors. Even then, the SPCI is still on average outperformed by the significantly less computationally complex linearization-based approach. A linearization-based CI approach is possible, but is not considered as further approximations associated with linearization are expected to provide worse results anyway.

A NEES test with 5% significance level is also performed to evaluate the consistency of the proposed estimators. As seen in Figure H.2, and as per the theory validating the NEES test, the hypothesis that the estimator is consistent cannot be rejected with 95% confidence.

Table H.1: RMSE of the estimators averaged 500 trials.

	Average RMSE	**% Diff. to Full**
Full - Position (m)	0.0487	-
Proposed: SP - Position (m)	**0.0662**	**35.9%**
Proposed: Lin. - Position (m)	**0.0729**	**49.7%**
SPCI - Position (m)	0.0862	77.0%
Naive - Position (m)	0.1733	256%
Full - Attitude (rad)	0.0190	-
AHRS - Attitude (rad)	0.0306	61.1%

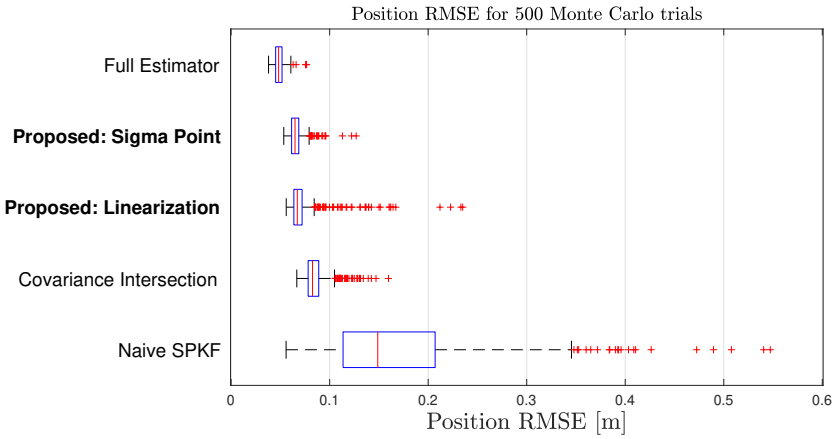


Figure H.1: A box plot showing the median RMSE, outliers, and variation of different estimators over 500 Monte Carlo trials.

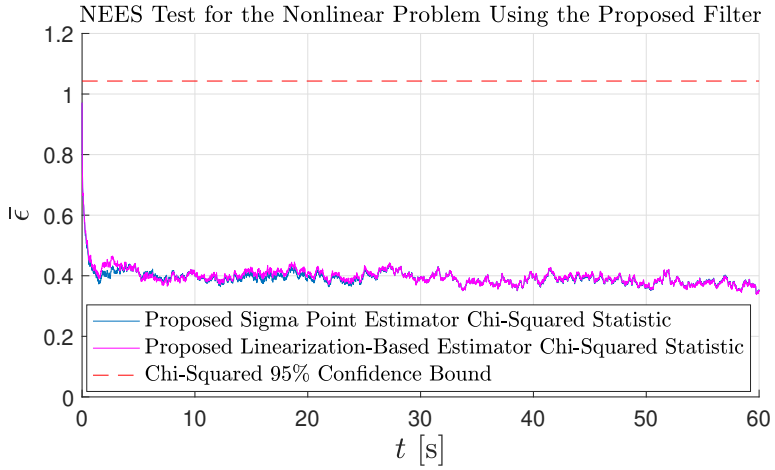


Figure H.2: The NEES test results for the 500 Monte Carlo trials, showing the consistency of both proposed estimators.

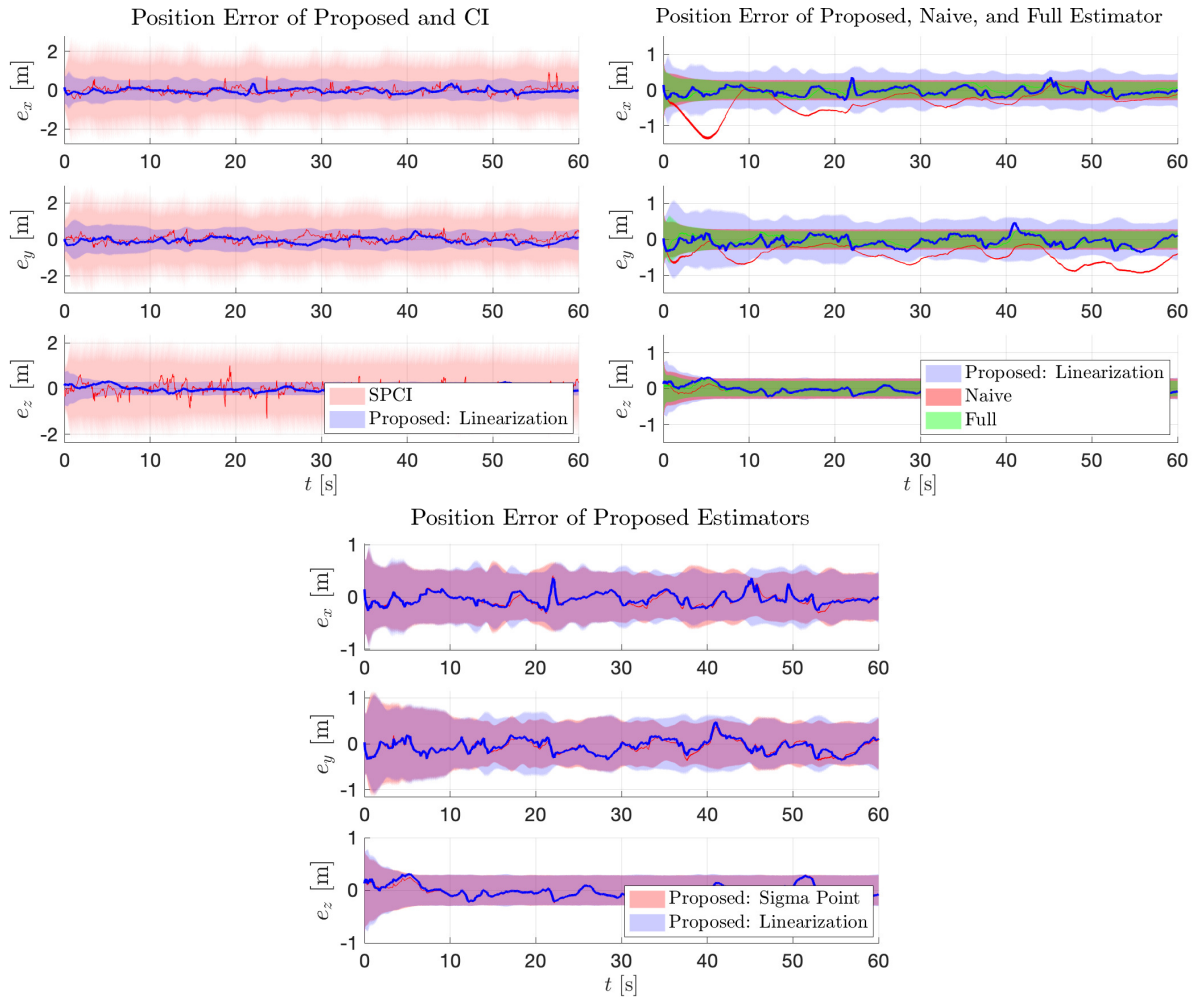


Figure H.3: The position errors associated with the experimental run at a slow pace. The shaded region corresponds to the $\pm 3\sigma$ bound, and the colour of each error trajectory and covariance region are the same.

H.3 Experimental Results

The linearization-based approach is also tested on the slow-paced experimental run. The error trajectory plots and corresponding $\pm 3\sigma$ bounds are shown in Figure H.3. The proposed linearization-based approach behaves similarly to the proposed sigma point approach, and as is the case with the sigma point approach, it remains mostly within the bounds while being less over-conservative than the SPCI approach. This can also be seen in Figure H.4, where the KL divergence [114, Chapter 9] measure shows that the estimated distribution of the linearization-based approach is much closer to the estimated distribution of the full estimator when compared to the SPCI, which is the best available estimate of the true distribution.

The position RMSE of the proposed linearization-based approach is 0.23494 m, compared to the proposed sigma point-based estimator's of 0.23138 m and the full estimator's of 0.21870

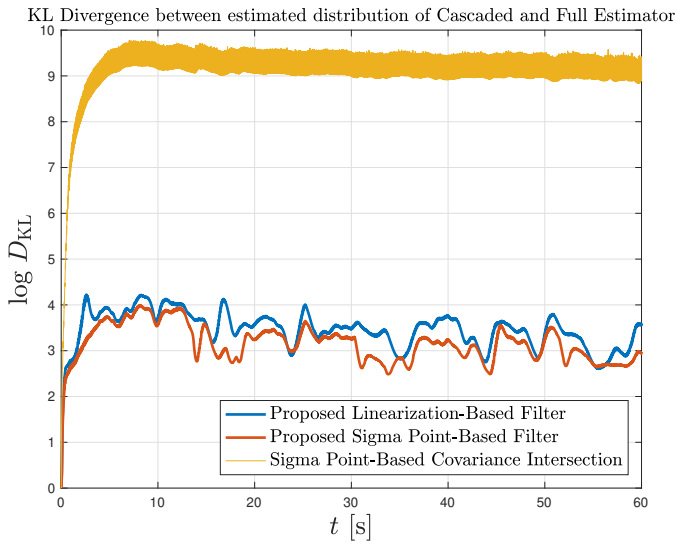


Figure H.4: The logarithm of the KL divergence measure associated with the experimental run at a slow pace. The KL divergence is computed between each cascaded estimator and the full estimator.

m. Both proposed approaches are similar to the performance of the full estimator, while the linearization-based approach is slightly worse as linearization errors contribute to a slightly less accurate estimated distribution, as shown in Figure H.4.

Bibliography

[1] M. Shalaby, C. C. Cossette, J. R. Forbes, and J. Le Ny, "Relative Position Estimation in Multi-Agent Systems Using Attitude-Coupled Range Measurements," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 3, pp. 4955–4961, 2021.

[2] M. A. Shalaby, C. C. Cossette, J. R. Forbes, and J. Le Ny, "Calibration and uncertainty characterization for ultra-wideband two-way-ranging measurements," in *IEEE International Conference on Robotics and Automation (ICRA)*, London, UK, 2023.

[3] M. A. Shalaby, C. Champagne Cossette, J. R. Forbes, and J. Le Ny, "Reducing Two-way Ranging Variance by Signal-Timing Optimization," *arXiv: 2211.00538 [eess.SP]*, 2022.

[4] M. A. Shalaby, C. C. Cossette, J. R. Forbes, and J. Le Ny, "Multi-Robot Relative Pose Estimation and IMU Preintegration Using Passive UWB Transceivers," *arXiv: 2304.03837v2 [cs.ro]*, 2023.

[5] M. Shalaby, C. C. Cossette, J. Le Ny, and J. R. Forbes, "Cascaded Filtering Using the Sigma Point Transformation," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 3, pp. 4758–4765, 2021.

[6] M. A. Shalaby, C. C. Cossette, J. L. Ny, and J. R. Forbes, "Ultra-Wideband Teach and Repeat," *arXiv: 202.01134 [cs.ro]*, 2022.

[7] Qorvo, *Dw1000*, https://www.qorvo.com/products/p/DW1000.

[8] J. Farrell, *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill Education, 2008.

[9] P. D. Groves, "Shadow matching: A new GNSS positioning technique for urban canyons," *Journal of Navigation*, vol. 64, no. 3, pp. 417–430, 2011.

[10] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2017, ISSN: 23318422.

[11] H. Wang, C. Wang, and L. Xie, "Intensity-slam: Intensity assisted localization and mapping for large scale environment," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1715–1721, 2021. DOI: 10.1109/LRA.2021.3059567.

[12] P. K. Yoon, S. Zihajehzadeh, B. S. Kang, and E. J. Park, "Adaptive Kalman filter for indoor localization using Bluetooth Low Energy and inertial measurement unit," *IEEE Engineering in Medicine and Biology Society (EMBS)*, 2015.

[13] C. Chen, Y. Chen, H.-Q. Lai, Y. Han, and K. R. Liu, "High accuracy indoor localization: A wifi-based approach," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, 2016.

[14] M. W. Mueller, M. Hamer, and R. D'Andrea, "Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation," in *International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA, 2015.

[15] P. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition*. Artech House Publishers, 2013.

[16] P. Lourenço, P. Batista, P. Oliveira, C. Silvestre, and C. L. Chen, "Sensor-based globally exponentially stable range-only simultaneous localization and mapping," *Robotics and Autonomous Systems*, vol. 68, pp. 72–85, 2015.

[17] V. Mai, M. Kamel, M. Krebs, A. Schaffner, D. Meier, L. Paull, and R. Siegwart, "Local Positioning System Using UWB Range Measurements for an Unmanned Blimp," *IEEE Robotics and Automation Letters (RA-L)*, vol. 3, no. 4, pp. 2971–2978, 2018.

[18] C. C. Cossette, M. Shalaby, D. Saussie, J. R. Forbes, and J. Le Ny, "Relative Position Estimation between Two UWB Devices with IMUs," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 3, pp. 4313–4320, 2021.

[19] B. Hepp, T. Nägeli, and O. Hilliges, "Omni-directional person tracking on a flying robot using occlusion-robust ultra-wideband signals," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, 2016.

[20] T. M. Nguyen, T. H. Nguyen, M. Cao, Z. Qiu, and L. Xie, "Integrated UWB-vision approach for autonomous docking of UAVS in GPS-denied environments," in *IEEE International Conference on Robotics and Automation (ICRA)*, Montreal, Canada, 2019.

[21] D. Neirynck, E. Luk, and M. McLaughlin, "An alternative double-sided two-way ranging method," in *13th Workshop on Positioning, Navigation and Communication (WPNC)*, Bremen, Germany, 2017.

[22] J. Cano, S. Chidami, and J. Le Ny, "A Kalman Filter-Based Algorithm for Simultaneous Time Synchronization and Localization in UWB Networks," in *International Conference on Robotics and Automation (ICRA)*, Montreal, Canada, 2019.

[23] Decawave, "APS014: DW1000 Antenna Delay Calibration Version 1.2," 2018.

[24] ——, "Sources of error in DW1000 based two-way ranging (TWR) schemes," 2014.

[25] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[26] H. Wymeersch, S. Maranò, W. M. Gifford, and M. Z. Win, "A machine learning approach to ranging error mitigation for UWB localization," *IEEE Transactions on Communications*, vol. 60, no. 6, pp. 1719–1728, 2012.

[27] W. Zhao, J. Panerati, and A. P. Schoellig, "Learning-based bias correction for time difference of arrival ultra-wideband localization of resource-constrained mobile robots," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 3639–3646, 2021.

[28] R. Piché, Simo Särkkä, and J. Hartikainen, "Recursive outlier-robust filtering and smoothing for nonlinear systems using the multivariate Student-t distribution," in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Santander, Spain, 2012.

[29] M. Kok, J. D. Hol, and T. B. Schon, "Indoor positioning using ultrawideband and inertial measurements," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1293–1303, 2015.

[30] J. Cano, G. Pages, E. Chaumette, and J. Le Ny, "Clock and Power-Induced Bias Correction for UWB Time-of-Flight Measurements," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 2, pp. 2431–2438, 2022.

[31] G. Miao, J. Zander, K. W. Sung, and S. Ben Slimane, *Fundamentals of Mobile Data Networks*. United Kingdom: Cambridge University Press, 2016.

[32] E. Allak, R. Jung, and S. Weiss, "Covariance Pre-Integration for Delayed Measurements in Multi-Sensor Fusion," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, 2019.

[33] E. Allak, A. Barrau, R. Jung, J. Steinbrener, and S. Weiss, "Centralized-equivalent pairwise estimation with asynchronous communication constraints for two robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, 2022.

[34] H. Mokhtarzadeh and D. Gebre-Egziabher, "Cooperative Inertial Navigation," *NAVIGATION: Journal of The Institute of Navigation*, vol. 61, no. 2, pp. 77–94, 2014.

[35] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[36] S. J. Julier and J. K. Uhlmann, "A Non-divergent Estimation Algorithm in the Presence of Unknown Correlations," in *American Control Conference (ACC)*, vol. 4, Albuquerque, NM, USA, 1997.

[37] L. C. Carrillo-Arce, E. D. Nerurkar, J. L. Gordillo, and S. I. Roumeliotis, "Decentralized multi-robot cooperative localization using covariance intersection," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Tokyo, Japan, 2013.

[38] E. D. Nerurkar, S. I. Roumeliotis, and A. Martinelli, "Distributed maximum a posteriori estimation for multi-robot cooperative localization," in *IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, 2009.

[39] D. Viegas, P. Batista, P. Oliveira, and C. Silvestre, "Decentralized state observers for range-based position and velocity estimation in acyclic formations with fixed topologies," *International Journal of Robust and Nonlinear Control*, vol. 26, pp. 963–994, 2016.

[40] J. Zhu and S. S. Kia, "Bias Compensation for UWB Ranging for Pedestrian Geolocation Applications," *IEEE Sensors Letters*, vol. 3, no. 9, pp. 7–10, 2019.

[41] C. C. Cossette, M. A. Shalaby, D. Saussie, and J. R. Forbes, "On-manifold Decentralized State Estimation using Pseudomeasurements and Preintegration," *arXiv: 2107.05842v1 [cs.ro]*, 2023.

[42] G. Casella and R. L. Berger, *Statistical Inference, Second Edition*. Belmont, CA: Duxbury Press, 2002.

[43] S. Sarkka, *Bayesian Filtering and Smoothing*. New York, NY: Cambridge University Press, 2013.

[44] M. Hamer and R. D'andrea, "Self-Calibrating Ultra-Wideband Network Supporting Multi-Robot Localization," *IEEE Access*, vol. 6, pp. 22 292–22 304, 2018.

[45] J. Solà, J. Deray, and D. Atchuthan, "A micro Lie theory for state estimation in robotics," *arXiv: 1812.01537 [cs.Ro]*, 2018.

[46] J. Arsenault, "Practical considerations and extensions of the invariant extended kalman filtering framework," M.A.Sc. Thesis, McGill University, 2019.

[47] A. Barrau and S. Bonnabel, "The invariant extended kalman filter as a stable observer," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1797–1812, 2017.

[48] E. Lefferts, F. Markley, and M. Shuster, "Kalman filtering for spacecraft attitude estimation," in *AIAA 20th Aerospace Sciences Meeting*, Orlando, FL, 1982.

[49] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice with MATLAB, 4th Edition*. Wiley-IEEE Press, 2014.

[50] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. USA: John Wiley & Sons, Inc., 2002.

[51] W. G. Cochran, "The χ^2 Test of Goodness of Fit," *The Annals of Mathematical Statistics*, vol. 23, no. 3, pp. 315–345, 1952.

[52] A. V. Rao, *Dynamics of Particles and Rigid Bodies: A Systematic Approach*. Cambridge University Press, 2006.

[53] Y. Cao, C. Yang, R. Li, A. Knoll, and G. Beltrame, "Accurate position tracking with a single UWB anchor," in *International Conference on Robotics and Automation (ICRA)*, Paris, France, 2020.

[54] Y. Cao and G. Beltrame, "VIR-SLAM: visual, inertial, and ranging SLAM for single and multi-robot systems," *Autonomous Robots*, no. 45, pp. 905–917, 2021.

[55] T. M. Nguyen, Z. Qiu, T. H. Nguyen, M. Cao, and L. Xie, "Distance-Based Cooperative Relative Localization for Leader-Following Control of MAVs," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, no. 4, pp. 3641–3648, 2019.

[56] S. van der Helm, M. Coppola, K. N. McGuire, and G. de Croon, "On-board range-based relative localization for micro air vehicles in indoor leader-follower flight," *Autonomous Robots*, vol. 44, pp. 415–441, 2020.

[57] P. Batista, C. Silvestre, and P. Oliveira, "Single range aided navigation and source localization: Observability and filter design," *Systems and Control Letters*, vol. 60, no. 8, pp. 665–673, 2011.

[58] R. K. Williams and G. S. Sukhatme, "Observability in topology-constrained multi-robot target tracking," in *IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA, 2015.

[59] L. Heintzman and R. K. Williams, "Nonlinear observability of unicycle multi-robot teams subject to nonuniform environmental disturbances," *Autonomous Robots*, vol. 44, no. 7, pp. 1149–1166, 2020.

[60] R. Liu, C. Yuen, T. N. Do, D. Jiao, X. Liu, and U. X. Tan, "Cooperative relative positioning of mobile users by fusing IMU inertial and UWB ranging information," in *International Conference on Robotics and Automation (ICRA)*, Singapore, 2017.

[61] P. Richardson, W. Xiang, and D. Shan, "An outdoor UWB tracking system to improve safety of semi-autonomous vehicle operations," *International Journal of Ultra Wideband Communications and Systems*, vol. 1, no. 3, pp. 209–221, 2010.

[62] S. Güler, M. Abdelkader, and J. S. Shamma, "Infrastructure-free multi-robot localization with ultrawideband sensors," in *American Control Conference (ACC)*, Philadelphia, PA, USA, 2019.

[63] T. M. Nguyen, A. H. Zaini, C. Wang, K. Guo, and L. Xie, "Robust Target-Relative Localization with Ultra-Wideband Ranging and Communication," in *IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, Australia, 2018.

[64] L. Asimow and B. Roth, "The rigidity of graphs, ii," *Journal of Mathematical Analysis and Applications*, vol. 68, no. 1, pp. 171–190, 1979.

[65] E. B. Lee and L. Markus, *Foundations of Optimal Control Theory*. John Wiley & Sons, Inc., 1967.

[66] B. Hendrickson, "Conditions for Unique Graph Realizations," *SIAM Journal on Computing*, vol. 21, pp. 65–84, 1992.

[67] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *2nd International Conference on Embedded Networked Sensor Systems*, Baltimore, MD, USA, 2004.

[68] Z. Sahinoglu, S. Gezici, and I. Güvenc, *Ultra-wideband Positioning Systems: Theoretical Limits, Ranging Algorithms, and Protocols*. Cambridge University Press, 2008.

[69] J. González, J. L. Blanco, C. Galindo, A. Ortiz-de-Galisteo, J. A. Fernández-Madrigal, F. A. Moreno, and J. L. Martínez, "Mobile robot localization based on Ultra-Wide-Band ranging: A particle filter approach," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 496–507, 2009.

[70] A. Ledergerber and R. D'Andrea, "Calibrating Away Inaccuracies in Ultra Wideband Range Measurements: A Maximum Likelihood Approach," *IEEE Access*, vol. 6, pp. 78 719–78 730, 2018.

[71] T. D. Barfoot, *State Estimation for Robotics, Second Edition*. Cambridge University Press, 2022.

[72] J. Sidorenko, V. Schatz, N. Scherer-Negenborn, M. Arens, and U. Hugentobler, "DecaWave Ultra-Wideband Warm-Up Error Correction," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 1, pp. 751–760, 2021.

[73] D. B. Jourdan, D. Dardari, and M. Z. Win, "Position error bound for UWB localization in dense cluttered environments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 2, pp. 613–628, 2008.

[74] V. Navrátil and F. Vejražka, "Bias and variance of asymmetric double-sided two-way ranging," *Navigation, Journal of the Institute of Navigation*, vol. 66, no. 3, pp. 593–602, 2019.

[75] I. Guvenc, S. Gezici, and Z. Sahinoglu, "Ultra-wideband range estimation: Theoretical limits and practical algorithms," in *IEEE International Conference on Ultra-Wideband*, Hannover, Germany, 2008.

[76] C. L. Sang, M. Adams, T. Hörmann, M. Hesse, M. Porrmann, and U. Rückert, "Numerical and experimental evaluation of error estimation for two-way ranging methods," *Sensors*, vol. 19, no. 3, 2019.

[77] IEEE Computer Society, *IEEE Standard for Low-Rate Wireless Networks. Amendment 1: Add Alternate PHYs (IEEE Std 802.15.4a)*, 2. 2018, pp. 51–51.

[78] S. Kay, *Fundamentals of Statistical Signal Processing, Volume I: Estimation Theory*. Prentice Hall PTR, 1993.

[79] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-Manifold Preintegration for Real-Time Visual-Inertial Odometry," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, 2017.

[80] LAN/MAN Standards Committee, "Part 15.4: Wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (LR-WPANs)," *IEEE Computer Society*, 2003.

[81] R. Jung and S. Weiss, "Scalable and modular ultra-wideband aided inertial navigation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, 2022.

[82] K. A. Horvath, G. Ill, and A. Milankovich, "Passive extended double-sided two-way ranging algorithm for UWB positioning," in *International Conference on Ubiquitous and Future Networks*, Milan, Italy, 2017.

[83] S. Shah and T. Demeechai, "Multiple simultaneous ranging in IR-UWB networks," *Sensors*, vol. 19, no. 24, pp. 1–14, 2019.

[84] T. Laadung, S. Ulp, M. M. Alam, and Y. L. Moullec, "Novel Active-Passive Two-Way Ranging Protocols for UWB Positioning Systems," *IEEE Sensors Journal*, vol. 22, no. 6, pp. 5223–5237, 2022.

[85] A. Ledergerber, M. Hamer, and R. D'Andrea, "A robot self-localization system using one-way ultra-wideband communication," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015.

[86] R. Zandian and U. Witkowski, "Robot Self-Localization in Ultra-Wideband Large Scale Multi-Node Setups," in *Workshop on Positioning, Navigation and Communications (WPNC)*, Bremen, Germany, 2017.

[87] D. E. Badawy, V. Larsson, M. Pollefeys, and I. Dokmanić, "Localizing unsynchronized sensors with unknown sources," *IEEE Transactions on Signal Processing*, vol. 71, pp. 641–654, 2023.

[88] A. Alanwar, H. Ferraz, K. Hsieh, R. Thazhath, P. Martin, J. Hespanha, and M. Srivastava, "D-SLATS: Distributed simultaneous localization and time synchronization," in *International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, Chennai, India, 2017.

[89] H. Xu, L. Wang, Y. Zhang, K. Qiu, and S. Shen, "Decentralized Visual-Inertial-UWB Fusion for Relative State Estimation of Aerial Swarm," in *IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, 2020.

[90] R. Jung and S. Weiss, "Scalable Recursive Distributed Collaborative State Estimation for Aided Inertial Navigation," in *IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, 2021.

[91] Q. Shi, X. Cui, S. Zhao, and M. Lu, "Sequential TOA-Based Moving Target Localization in Multi-Agent Networks," *IEEE Communications Letters*, vol. 24, no. 8, pp. 1719–1723, 2020.

[92] Q. Shi, X. Cui, S. Zhao, S. Xu, and M. Lu, "BLAS: Broadcast relative localization and clock synchronization for dynamic dense multiagent systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 5, pp. 3822–3839, 2020.

[93] Z. Dou, Z. Yao, and M. Lu, "Asynchronous Collaborative Localization System for Large-Capacity Sensor Networks," *IEEE Internet of Things Journal*, vol. 9, no. 16, pp. 15 349–15 361, 2022.

[94] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 61–76, 2012.

[95] A. Barrau and S. Bonnabel, "Linear observed systems on groups," *Systems and Control Letters*, no. 129, pp. 36–42, 2019.

[96] M. Brossard, A. Barrau, P. Chauchat, and S. Bonnabel, "Associating Uncertainty to Extended Poses for on Lie Group IMU Preintegration With Rotating Earth," *IEEE Transactions on Robotics*, pp. 1–16, 2021.

[97] A. W. Long, C. K. Wolfe, M. J. Mashner, and G. S. Chirikjian, "The banana distribution is Gaussian: A localization study with exponential coordinates," *Robotics: Science and Systems*, vol. 8, pp. 265–272, 2013.

[98] M. Behr, P. Benner, and J. Heiland, "Solution formulas for differential Sylvester and Lyapunov equations," *Calcolo*, vol. 56, no. 4, pp. 1–33, 2019.

[99] C. C. Cossette, M. A. Shalaby, D. Saussié, J. L. Ny, and J. R. Forbes, "Optimal multi-robot formations for relative pose estimation using range measurements," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, 2022.

[100] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based rules for designing consistent EKF SLAM estimators," *International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–528, 2010.

[101] G. P. Huang, N. Trawny, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based consistent EKF estimators for multi-robot cooperative localization," *Autonomous Robots*, vol. 30, no. 1, pp. 99–122, 2011.

[102] D. M. Bevly and B. Parkinson, "Cascaded Kalman Filters for Accurate Estimation of Multiple Biases, Dead-Reckoning Navigation, and Full State Feedback Control of Ground Vehicles," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 2, pp. 199–208, 2007.

[103] S. Zihajehzadeh, D. Loh, T. J. Lee, R. Hoskinson, and E. J. Park, "A Cascaded Kalman Filter-Based GPS/MEMS-IMU Integration for Sports Applications," *Measurement*, vol. 73, pp. 200–210, 2015.

[104] J. W. Song and C. G. Park, "Enhanced Pedestrian Navigation Based on Course Angle Error Estimation Using Cascaded Kalman Filters," *Sensors (Switzerland)*, vol. 18, no. 4, 2018.

[105] P. O. Arambel, C. Rago, and R. K. Mehra, "Covariance Intersection Algorithm for Distributed Spacecraft State Estimation," in *American Control Conference (ACC)*, Arlington, VA, USA, 2001.

[106] N. A. Carlson, "Federated Square Root Filter for Decentralized Parallel Processes," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 3, pp. 517–525, 1990.

[107] N. A. Carlson and M. P. Berarducci, "Federated Kalman Filter Simulation Results," *Navigation*, vol. 41, no. 3, pp. 297–322, 1994.

[108] G. Gobbini, "Relative navigation by means of passive ranging," PhD thesis, Massachusetts Institute of Technology, Dept. of Aeronautics and Astronautics, Aug. 2005.

[109] Z. Lendek, R. Babuška, and B. De Schutter, "Distributed Kalman Filtering for Cascaded Systems," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 3, pp. 457–469, 2008.

[110] J. Ali and J. Fang, "Multisensor Data Synthesis Using Federated Form of Unscented Kalman Filtering," in *IEEE International Conference on Industrial Technology*, Hong Kong, China, 2005.

[111] J. Steinbring, B. Noack, M. Reinhardt, and U. D. Hanebeck, "Optimal sample-based fusion for distributed state estimation," in *19th International Conference on Information Fusion (FUSION)*, Heidelberg, Germany, 2016.

[112] M. Reinhardt, B. Noack, and U. D. Hanebeck, "Covariance Intersection in State Estimation of Dynamical Systems," in *17th International Conference on Information Fusion (FUSION)*, Salamanca, Spain, 2014.

[113] Y. Dai, J. Trumpf, H. Li, N. Barnes, and R. Hartley, "Rotation Averaging With Application to Camera-Rig Calibration," *Lecture Notes in Comp. Sci.*, pp. 335–346, 2010.

[114] S. Kullback, *Information Theory and Statistics*. Dover Punlications, Inc., 1968, p. 409.

[115] J. Engel, J. Sturm, and D. Cremers, "LSD-SLAM: Large-scale sirect monocular SLAM," in *IEEE International Conference on Computer Vision (ICCV)*, Sydney, Australia, 2013.

[116] J. Engel, J. Stuckler, and D. Cremers, "Large-scale direct SLAM with stereo cameras," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015.

[117] D. Droeschel and S. Behnke, "Efficient continuous-time SLAM for 3D LIDAR-based online mapping," in *IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, Australia, 2018.

[118] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[119] N. Funabiki, B. Morrell, J. Nash, and A. A. Agha-Mohammadi, "Range-aided pose-graph-based SLAM: Applications of deployable ranging beacons for unknown environment exploration," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 1, pp. 48–55, 2021.

[120] P. Furgale and T. D. Barfoot, "Visual Teach and Repeat for long-range rover autonomy," *Journal of Field Robotics*, vol. 27, no. 5, pp. 534–560, 2010.

[121] T. Krajnik, F. Majer, L. Halodova, and T. Vintr, "Navigation without localisation: Reliable teach and repeat based on the convergence theorem," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, 2018.

[122] C. McManus, P. Furgale, and T. D. Barfoot, "Towards appearance-based methods for LIDAR sensors," in *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 2011.

[123] M. Warren, M. Paton, K. MacTavish, A. P. Schoellig, and T. D. Barfoot, "Towards Visual Teach and Repeat for GPS-denied flight of a fixed-wing UAV," *Springer Proceedings in Advanced Robotics*, vol. 5, pp. 481–498, 2018.

[124] M. Nitsche, F. Pessacg, and J. Civera, "Visual-inertial teach repeat for aerial robot navigation," in *European Conference on Mobile Robots*, Prague, Czech Republic, 2019.

[125] M. Paton, K. MacTavish, C. J. Ostafew, and T. D. Barfoot, "It's not easy seeing green: Lighting-resistant stereo Visual Teach & Repeat using color-constant images," in *IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA, 2015.

[126] M. Paton, K. Mactavish, M. Warren, and T. D. Barfoot, "Bridging the appearance gap: Multi-experience localization for long-term visual teach and repeat," in *International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, 2016.

[127] L. G. Camara, T. Pivonka, M. Jilek, C. Gabert, K. Kosnar, and L. Preucil, "Accurate and robust Teach and Repeat navigation by visual place recognition: A CNN approach," in *International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA, 2020.

[128] M. Gridseth and T. D. Barfoot, "DeepMEL: Compiling visual multi-experience localization into a deep neural network," in *IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, 2020.

[129] M. R. Cohen, K. Abdulrahim, and J. R. Forbes, "Finite-horizon LQR control of quadrotors on $SE_2(3)$," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 4, pp. 5748–5755, 2020.

[130] Pozyx, https://www.pozyx.io/.

[131] Uvify, *Ifo-s*, https://www.uvify.com/ifo-s/.

[132] S. Hong, M. H. Lee, H. H. Chun, S. H. Kwon, and J. L. Speyer, "Observability of error states in GPS/INS integration," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 2, pp. 731–742, 2005.

[133] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," *IEEE International Conference on Robotics and Automation (ICRA)*, 2007. DOI: 10.1109/ROBOT.2007.364024.

[134] M. A. Shalaby, C. C. Cossette, J. R. Forbes, and J. Le Ny, "IMU Preintegration for Multi-Robot Systems in the Presence of Bias and Communication Constraints," *arXiv: 2310.08686v2 [cs.ro]*, 2023.

[135] J. K. Uhlmann, "Covariance consistency methods for fault-tolerant distributed data fusion," *Information Fusion*, vol. 4, no. 3, pp. 201–215, 2003.